



# IBM Watson Content Analytics

## Discovering Actionable Insight from Your Content

Learn how to perform effective content analytics and search

Learn how to gain insights from your data and detect problems early

Ultimately, improve your products, services, and offerings



Wei-Dong (Jackie) Zhu  
Bob Foyle  
Daniel Gagné  
Vijay Gupta  
Josemina Magdalen  
Amarjeet S Mundi  
Tetsuya Nasukawa  
Mark Paulis  
Jane Singer  
Martin Triska





International Technical Support Organization

**IBM Watson Content Analytics: Discovering  
Actionable Insight from Your Content**

July 2014

**Note:** Before using this information and the product it supports, read the information in “Notices” on page xiii.

**Third Edition (July 2014)**

This edition applies to Version 3, Release 0, of IBM Watson Content Analytics (program number 5724-Z21).

**© Copyright International Business Machines Corporation 2010, 2011, 2014. All rights reserved.**  
Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>Notices</b> .....	xiii
Trademarks .....	xiv
<b>Preface</b> .....	xv
Authors .....	xvi
Now you can become a published author, too! .....	xix
Comments welcome .....	xix
Stay connected to IBM Redbooks .....	xx
<b>Summary of changes</b> .....	xxi
July 2014, Third Edition .....	xxi
New information .....	xxi
Changed information .....	xxi
<b>Chapter 1. Overview of IBM Watson Content Analytics</b> .....	1
1.1 Business need and the content analytics and search solutions .....	2
1.1.1 Business need and problem statement .....	2
1.1.2 The content analytics solution .....	3
1.1.3 The analytics-driven search solution .....	4
1.2 History, changes, and what is new in version 3.0 .....	5
1.2.1 Product history .....	5
1.2.2 Product changes .....	6
1.2.3 What is new in IBM Watson Content Analytics .....	7
1.3 Important concepts and terminology .....	12
1.3.1 Unstructured and structured content .....	12
1.3.2 Text analytics .....	13
1.3.3 Search, discovery, and data mining .....	13
1.3.4 Collections .....	14
1.3.5 Facets .....	14
1.3.6 Frequency .....	16
1.3.7 Correlation .....	16
1.3.8 Deviation .....	18
1.4 Content Analytics architecture .....	18
1.4.1 Main components .....	18
1.4.2 Data flow .....	24
1.4.3 Scalability .....	27
1.4.4 Security .....	28
<b>Chapter 2. Use case scenarios</b> .....	29

2.1	Customer insights	30
2.1.1	Call center	30
2.1.2	Quality assurance	31
2.2	Law enforcement and public safety	31
2.3	Investigation management	32
2.3.1	Insurance fraud	34
2.4	Healthcare	35
2.5	Case management	38
2.6	Data warehouse	39
<b>Chapter 3. Designing content analytics solutions</b>		<b>41</b>
3.1	Data considerations	42
3.1.1	Content analytics data model	42
3.1.2	Structured and unstructured sources	43
3.1.3	Multiple data sources	44
3.1.4	Date-sensitive data	46
3.1.5	Extracting information from textual data	46
3.1.6	The number of collections to use	48
3.2	Guide for building a content analytics collection	48
3.2.1	Building a content analytics collection	48
3.2.2	A walk through the building process	49
3.2.3	Planning for iteration	52
3.3	Programming interfaces	53
3.3.1	REST API	54
3.3.2	Search and Index API	55
3.3.3	Real time natural language processing API	56
<b>Chapter 4. Understanding content analysis</b>		<b>59</b>
4.1	Basic concepts of content analytics	60
4.1.1	Manual versus automated analysis	60
4.1.2	Frequency versus deviation	63
4.1.3	Precision versus recall	66
4.2	Typical cycle of analysis with Content Analytics	67
4.2.1	Setting the objectives of the analysis	68
4.2.2	Gathering data	68
4.2.3	Analyzing data	69
4.2.4	Taking action based on the analysis	74
4.2.5	Validating the effect	75
4.3	Successful use cases	75
4.3.1	Voice of the customer	76
4.3.2	Analysis of other data	83
4.4	Summary	84
<b>Chapter 5. Content analytics miner: Basic features</b>		<b>85</b>

5.1 Overview of the content analytics miner . . . . .	86
5.1.1 Accessing the content analytics miner . . . . .	86
5.1.2 Application window layout and functional overview . . . . .	87
5.1.3 Selecting a collection for analysis . . . . .	92
5.1.4 Changing the default behavior by using preferences . . . . .	92
5.2 Search and discovery features . . . . .	96
5.2.1 Limiting the scope of your analysis using facets . . . . .	97
5.2.2 Limiting the scope of your analysis using search operators . . . . .	98
5.2.3 Limiting the scope of your analysis using dates . . . . .	99
5.2.4 Query syntax . . . . .	99
5.2.5 Type ahead . . . . .	103
5.2.6 Saved searches . . . . .	105
5.2.7 Advanced search . . . . .	107
5.3 Query Tree . . . . .	107
5.3.1 Accessing the Query Tree . . . . .	108
5.3.2 Understanding the Query Tree . . . . .	108
5.3.3 Query Tree examples . . . . .	110
5.3.4 Editing the Query Tree . . . . .	120
5.4 Query builder . . . . .	123
5.4.1 Accessing the Query Builder . . . . .	123
5.4.2 Features of the Query Builder window . . . . .	125
5.4.3 Using the Query Builder . . . . .	128
5.4.4 Preferred practice for using the Query Builder and Query Tree . . . . .	135
5.5 Rule-based categories with a query . . . . .	135
5.5.1 Enabling the rule-based categories feature . . . . .	136
5.5.2 Configuring rules for rule-based categories . . . . .	136
5.5.3 Adding the current query as a category rule . . . . .	140
5.6 Common view features . . . . .	146
5.7 Document flagging . . . . .	148
5.7.1 Configuring document flags . . . . .	149
5.7.2 Setting document flags . . . . .	152
5.7.3 Viewing the document values of a flag facet . . . . .	155
<b>Chapter 6. Content analytics miner: Views . . . . .</b>	<b>159</b>
6.1 Views . . . . .	160
6.2 Documents view . . . . .	161
6.2.1 Understanding the Documents view . . . . .	162
6.2.2 Viewing the document contents and facets . . . . .	163
6.2.3 When to use the Documents view . . . . .	165
6.3 Facets view . . . . .	165
6.3.1 Understanding the Facets view . . . . .	168
6.3.2 When to use the Facets view . . . . .	168
6.4 Time Series view . . . . .	169

6.4.1	Features in the Time Series view . . . . .	170
6.4.2	Understanding the Time Series view . . . . .	174
6.4.3	When to use the Time Series view . . . . .	175
6.5	Trends view . . . . .	175
6.5.1	Features in the Trends view . . . . .	177
6.5.2	Sort criteria . . . . .	179
6.5.3	Understanding the Trends view . . . . .	180
6.5.4	When to use the Trends view . . . . .	181
6.6	Deviations view . . . . .	182
6.6.1	Features in the Deviations view . . . . .	183
6.6.2	Understanding the Deviations view . . . . .	184
6.6.3	When to use the Deviations view . . . . .	187
6.7	Facet Pairs view . . . . .	187
6.7.1	Table view . . . . .	188
6.7.2	Grid view . . . . .	189
6.7.3	Bird's eye view . . . . .	190
6.7.4	Understanding the Facet Pairs view with correlation values . . . . .	192
6.7.5	When to use the Facet Pairs view . . . . .	193
6.8	Connections view . . . . .	194
6.8.1	Features in the Connections view . . . . .	196
6.8.2	Understanding the Connections view . . . . .	201
6.8.3	When to use the Connections view . . . . .	205
6.9	Dashboard view . . . . .	206
6.9.1	Configuring the Dashboard layout . . . . .	207
6.9.2	Viewing the Dashboard . . . . .	218
6.9.3	Working with the Dashboard . . . . .	219
6.9.4	Saving Dashboard charts as images . . . . .	221
6.10	Sentiment view . . . . .	222
6.10.1	Document view with Sentiment Analysis enabled . . . . .	222
6.10.2	Understanding the Sentiment view . . . . .	226
	<b>Chapter 7. Performing content analysis . . . . .</b>	<b>231</b>
7.1	Working with the sample collection . . . . .	232
7.1.1	The sample data . . . . .	232
7.1.2	Getting insights from the sample collection . . . . .	236
7.1.3	Considerations about what you want to discover from the data . . . . .	239
7.2	Content analysis scenarios for the sample collection . . . . .	240
7.2.1	Scenario 1: Using a custom dictionary to discover package-related calls . . . . .	240
7.2.2	Scenario 2: Using custom text analysis rules to discover trouble-related calls . . . . .	244
7.2.3	Scenario 3: Discovering the cause of increasing calls . . . . .	249
7.2.4	Conclusion . . . . .	251



7.3 Overview of techniques to create facets for analysis . . . . .	252
7.3.1 Named Entity Extraction component. . . . .	252
7.3.2 Sentiment . . . . .	254
7.3.3 Terms of interest. . . . .	255
7.3.4 Custom dictionaries. . . . .	255
7.3.5 Facet ranges . . . . .	256
7.3.6 Field Filters . . . . .	257
7.3.7 Rule-based categories . . . . .	258
7.3.8 Syntax Pattern Rules . . . . .	259
7.3.9 Document clustering and classification. . . . .	260
7.4 Preferred practices . . . . .	262
<b>Chapter 8. Performing content analysis with built-in annotators . . . . .</b>	<b>265</b>
8.1 Terms of interest . . . . .	266
8.1.1 Basic algorithm for identifying terms of interest . . . . .	269
8.1.2 Limitations in using automatic identification of terms of interest . . . . .	273
8.1.3 Preferred use of terms of interest identified automatically . . . . .	273
8.2 Configuring dictionary-driven analytics . . . . .	277
8.2.1 Multiple viewpoints for analyzing the same data. . . . .	278
8.2.2 Configuring the Dictionary Lookup annotator . . . . .	282
8.2.3 When to use the Dictionary Lookup annotator . . . . .	282
8.2.4 Configuring custom user dictionaries . . . . .	283
8.2.5 Validation and maintenance . . . . .	291
8.3 Configuring the Pattern Matcher annotator . . . . .	291
8.3.1 When to use the Pattern Matcher annotator . . . . .	292
8.3.2 Configuring custom text analysis rules . . . . .	293
8.3.3 Designing the custom text analysis rules . . . . .	296
8.3.4 Validation and maintenance . . . . .	303
<b>Chapter 9. Content analysis with</b>	
<b>IBM Content Classification and document clustering . . . . .</b>	<b>305</b>
9.1 The Content Classification annotator . . . . .	306
9.1.1 When to use the Content Classification annotator . . . . .	306
9.1.2 The Content Classification technology . . . . .	307
9.2 Fine-tuning your analysis with the Content Classification annotator. . . . .	309
9.2.1 Building your collection . . . . .	309
9.2.2 Refining the analysis. . . . .	311
9.2.3 Using a conceptual search for advanced content discovery . . . . .	312
9.3 Creating and deploying the Content Classification resource. . . . .	313
9.3.1 Starting the Content Classification server . . . . .	313
9.3.2 Creating and training the knowledge bases . . . . .	314
9.3.3 Creating a decision plan . . . . .	318
9.3.4 Deploying the knowledge base and decision plan . . . . .	322

9.3.5	Configuring the Content Classification annotator . . . . .	325
9.4	Validation and maintenance of the Content Classification annotator . . .	327
9.4.1	Using the Content Classification sample programs . . . . .	328
9.4.2	Content Classification annotator validation techniques. . . . .	329
9.5	Preferred practices for Content Classification annotator usage . . . . .	329
9.6	Document clustering . . . . .	330
9.6.1	Setting up document cluster . . . . .	330
9.6.2	Creating a cluster proposal . . . . .	333
9.6.3	Refining the cluster results . . . . .	334
9.6.4	Deploying clusters to a category . . . . .	338
9.6.5	Working with the cluster results . . . . .	340
9.6.6	Creating and deploying the clustering resource . . . . .	341
9.6.7	Preferred practices . . . . .	343
<b>Chapter 10. Importing CSV files, exporting data, and performing deep inspection . . . . .</b>		<b>345</b>
10.1	Importing CSV files . . . . .	346
10.2	Overview of exporting documents and data . . . . .	353
10.2.1	Crawled documents . . . . .	355
10.2.2	Analyzed documents. . . . .	356
10.2.3	Search result documents . . . . .	356
10.2.4	Exported data manifest . . . . .	357
10.3	Location and format of the exported data . . . . .	358
10.3.1	Location of the exported data . . . . .	358
10.3.2	Metadata format . . . . .	360
10.3.3	Binary content format . . . . .	360
10.3.4	Common Analysis Structure format . . . . .	361
10.3.5	Extracted text format . . . . .	362
10.4	Common configuration of the export feature . . . . .	362
10.4.1	Document URI pattern . . . . .	362
10.4.2	Exporting XML attributes and preserving file extensions . . . . .	362
10.4.3	Adding exported documents to the index . . . . .	363
10.4.4	Exporting information about deleted documents. . . . .	363
10.4.5	Scheduling . . . . .	363
10.5	Monitoring export requests . . . . .	364
10.6	Enabling export and sample configurations . . . . .	365
10.6.1	Exporting crawled documents to a file system for IBM Content Collector . . . . .	366
10.6.2	Exporting analyzed documents to a relational database. . . . .	371
10.6.3	Exporting search result documents to the file system for IBM Content Classification. . . . .	376
10.6.4	Exporting search result documents to CSV files. . . . .	383
10.7	Deep inspection. . . . .	387

10.7.1	Location and format of the exported data . . . . .	389
10.7.2	Common configuration . . . . .	390
10.7.3	Enabling deep inspection . . . . .	393
10.7.4	Generating deep inspection reports . . . . .	394
10.7.5	Optional: Scheduling a deep inspection run . . . . .	397
10.7.6	Monitoring the deep inspection requests . . . . .	400
10.7.7	Validating the deep inspection reports generation . . . . .	401
10.8	Creating and deploying a custom plug-in . . . . .	403
<b>Chapter 11. Customizing content analytics with IBM Content Analytics Studio . . . . . 405</b>		
11.1	ICA Studio overview . . . . .	406
11.2	The building process of UIMA pipeline . . . . .	408
11.3	Use case: Building a UIMA pipeline for analyzing customers complaints. . . . .	410
11.3.1	Creating the ICA Studio project. . . . .	411
11.3.2	Creating the UIMA pipeline . . . . .	412
11.3.3	Configuring the basic UIMA pipeline. . . . .	413
11.3.4	Testing the UIMA pipeline and reviewing output. . . . .	416
11.3.5	Creating custom dictionaries for Lexical Analysis. . . . .	419
11.3.6	Creating parsing rules . . . . .	424
11.4	Exporting annotators. . . . .	437
11.4.1	Creating the IBM Brands annotator. . . . .	438
11.4.2	Exporting the annotator. . . . .	440
11.5	Conclusion. . . . .	442
<b>Chapter 12. Enterprise search . . . . . 445</b>		
12.1	Overview of enterprise search capability in Content Analytics . . . . .	446
12.1.1	Adding values with Content Analytics features . . . . .	447
12.1.2	Enterprise search application user interface . . . . .	448
12.1.3	Components supporting Content Analytics enterprise search capability . . . . .	449
12.1.4	REST Search API . . . . .	451
12.2	Use case overview . . . . .	452
12.3	Customizing crawling, parsing, and indexing . . . . .	453
12.3.1	Setting up multiple content sources for crawling. . . . .	453
12.3.2	Mapping multiple content sources to the index. . . . .	454
12.3.3	Adding additional facets and fields with a custom annotator . . . . .	457
12.3.4	Adding a category tree . . . . .	458
12.3.5	Altering field values to conform to uniform standard. . . . .	461
12.4	Customizing runtime search . . . . .	462
12.4.1	Tuning queries and results . . . . .	462
12.4.2	Enhancing free text search to use field search . . . . .	466

12.4.3	Expanding the search query . . . . .	466
12.4.4	Grouping results . . . . .	473
12.4.5	Ranking the search results . . . . .	475
12.5	Search Customizer . . . . .	476
12.5.1	Adding the Person facet . . . . .	476
12.5.2	Adding the timeline . . . . .	477
12.5.3	Adding the Category Tree . . . . .	477
12.6	Performing search . . . . .	479
12.6.1	Search strategies . . . . .	479
12.6.2	Search example . . . . .	479
12.7	Security . . . . .	482
12.7.1	Authentication . . . . .	482
12.7.2	Authorization (access control) . . . . .	483
12.8	Summary . . . . .	486
<b>Chapter 13. Adding value to Cognos Business Intelligence . . . . .</b>		<b>487</b>
13.1	Integration overview . . . . .	488
13.2	Integration architecture . . . . .	488
13.2.1	Data model integration . . . . .	490
13.2.2	Cognos report generation . . . . .	492
13.3	Initial setup . . . . .	493
13.3.1	Verifying IBM Cognos BI . . . . .	493
13.3.2	Creating a data source connection by using Cognos BI Administration . . . . .	497
13.3.3	Configuring default application user roles . . . . .	501
13.3.4	Configuring an export to a relational database using Content Analytics . . . . .	503
13.3.5	Configuring the Cognos BI server for reporting by using Content Analytics . . . . .	510
13.4	Generating Cognos BI reports . . . . .	515
13.5	Creating custom Cognos BI reports . . . . .	521
13.5.1	Exporting search results . . . . .	522
13.5.2	Loading the exported data model into Cognos . . . . .	524
<b>Chapter 14. Customizing and extending the content analytics miner . . . . .</b>		<b>535</b>
14.1	Reasons for custom development . . . . .	536
14.2	Analytics Customizer . . . . .	538
14.3	Creating the sample plug-in: Spatial Analysis . . . . .	539
14.3.1	Preparation . . . . .	539
14.3.2	Plug-in structure . . . . .	542
14.3.3	Adding a map to the plug-in . . . . .	544
14.3.4	Displaying documents on the map . . . . .	546
14.3.5	The entire code for the Spatial Analysis plug-in so far . . . . .	547

14.3.6 Adding selection mode . . . . .	550
<b>Appendix A. Spatial Analysis plug-in code.</b> . . . . .	557
Spatial Analysis plug-in overview . . . . .	559
plugin.js . . . . .	559
plugin.html . . . . .	565
style.css . . . . .	565
<b>Appendix B. Additional material</b> . . . . .	567
Locating the Web material . . . . .	567
Using the Web material . . . . .	568
System requirements for downloading the Web material . . . . .	568
Downloading and extracting the Web material . . . . .	568
<b>Related publications</b> . . . . .	569
IBM Redbooks . . . . .	569
Other publications . . . . .	569
Online resources . . . . .	569
Help from IBM . . . . .	570



# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.


## **COPYRIGHT LICENSE:**

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

# Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

Cognos®	IBM Watson™	Redbooks®
DataStage®	IBM®	Redbooks (logo)  ®
DB2®	LanguageWare®	SPSS®
developerWorks®	Lotus Notes®	WebSphere®
Domino®	Lotus®	Worklight®
FileNet®	Notes®	
i2®	OmniFind®	

The following terms are trademarks of other companies:

Netezza, and N logo are trademarks or registered trademarks of IBM International Group B.V., an IBM Company.

Worklight is trademark or registered trademark of Worklight, an IBM Company.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.



# Preface

IBM® Watson™ Content Analytics (Content Analytics) Version 3.0 (formerly known as IBM Content Analytics with Enterprise Search (ICAwES)) helps you to unlock the value of unstructured content to gain new actionable business insight and provides the enterprise search capability all in one product. Content Analytics comes with a set of tools and a robust user interface to empower you to better identify new revenue opportunities, improve customer satisfaction, detect problems early, and improve products, services, and offerings.

To help you gain the most benefits from your unstructured content, this IBM Redbooks® publication provides in-depth information about the features and capabilities of Content Analytics, how the content analytics works, and how to perform effective and efficient content analytics on your content to discover actionable business insights.

This book covers key concepts in content analytics, such as facets, frequency, deviation, correlation, trend, and sentimental analysis. It describes the content analytics miner, and guides you on performing content analytics using views, dictionary lookup, and customization. The book also covers using IBM Content Analytics Studio for domain-specific content analytics, integrating with IBM Content Classification to get categories and new metadata, and interfacing with IBM Cognos® Business Intelligence (BI) to add values in BI reporting and analysis, and customizing the content analytics miner with APIs. In addition, the book describes how to use the enterprise search capability for the discovery and retrieval of documents using various query and visual navigation techniques, and customization of crawling, parsing, indexing, and runtime search to improve search results.

The target audience of this book is decision makers, business users, and IT architects and specialists who want to understand and analyze their enterprise content to improve and enhance their business operations. It is also intended as a technical how-to guide for use with the online IBM Knowledge Center for configuring and performing content analytics and enterprise search with Content Analytics.

**Name change:** This book covers IBM Watson Content Analytics (formerly known as IBM Content Analytics with Enterprise Search, or simply ICAwES).

# Authors

This book was produced by a team of specialists from around the world working at the IBM International Technical Support Organization (ITSO).

**Wei-Dong (Jackie) Zhu** is an Enterprise Content Management Project Leader with ITSO. Jackie joined IBM in 1996 and has more than 10 years of software development experience in accounting, image workflow processing, and digital media distribution. She is a Certified Solution Designer for IBM Content Manager and has managed and lead the production of many Enterprise Content Management Redbooks publications. Jackie holds a Master of Science degree in Computer Science from the University of the Southern California.

**Bob Foyle** is the Senior Product Manager in charge of IBM Watson Content Analytics and IBM Content and Predictive Analytics for Healthcare. Coming to IBM through the FileNet® acquisition, Bob brings a strong background in Enterprise Content Management (ECM) and adds the ability to securely search, analyze, and extract intelligence and insight from unstructured data (both in and out of ECM repositories) using Natural Language Processing and Content Analytics. Bob manages the overall product direction for IBM Watson Content Analytics and its capabilities, and helps drive the overall strategy and success of the product.

**Daniel Gagné** is a Client Solution Professional and a Senior Certified Enterprise Content Management Architect in Montreal, Quebec, Canada. Daniel joined IBM in 1996 and currently has 30 years of experience in the Information Technology and Software industry. He specializes in collaboration, information management, content management, and content analytics technologies. For the past eight years, Daniel has been the Canadian domain specialist in enterprise search and content analytics solutions from IBM. He is a Certified Architect from the OpenGroup and a certified AIIM Professional member.

**Vijay Gupta** is Senior Managing Consultant and Senior Solutions Architect in Enterprise Content Management Lab Services Organization in IBM US. He is in the Professional Services and Enablement team and works directly with clients to provide consulting and implementation services for complex Content Analytics, Discovery, and ECM solutions. Vijay has over 17 years of experience in enterprise solutions architecture, consulting, design and development, education, training, quality verification, and client services. Vijay holds a Master's degree in Business Administration from Babson College, MA, US, and a Master's degree in Computer Science from University of Pune.

**Josemina Magdalen** is a Senior Team Leader and Architect for the IBM Israel Software Group (ILSL). She has a background in Natural Language Processing (including text classification and search) and in text mining and filtering technologies. Josemina joined IBM in 2005 and has worked in the Content Discovery Engineering Group doing software development projects in text categorization, filtering, and search, as well as text analytics. Before joining IBM, Josemina worked in Natural Languages Processing research and development (machine translation, text classification and search, data mining), for over 10 years. Josemina co-authored the Redbooks publication *IBM Classification Module: Make It Work for You*, SG24-7707.

**Amarjeet S Mundi** is a Client Solution Professional in the IBM Industry Solutions Enterprise Content Management Technical Team. He joined IBM in 2005 and has more than 13 years of experience in ECM. He is a Certified IBM ECM Solution Architect. Amarjeet works with clients and IBM Business Partners in adopting IBM ECM Solutions. He is a regular speaker on ECM at customer events, conferences, and seminars. Amarjeet holds a Degree in Electronics Engineering and Master's in Business Administration from Symbiosis, Pune.

**Tetsuya Nasukawa** is a Research Staff Member with IBM Research in Tokyo, Japan. He joined IBM in 1989 and has been leading text mining projects since 1997. Tetsuya is the primary inventor of the Text Analysis and Knowledge Mining (TAKMI) system that has been integrated into Content Analytics. He has 25 years of experience in the Natural Language Processing field. His areas of expertise include text mining, machine translation, sentiment analysis, and conversation mining. Tetsuya has authored and co-authored more than 50 academic papers and received eight academic awards. He has also written multiple books in Japanese and wrote the text mining section of Encyclopedia of Natural Language Processing (Japanese). Tetsuya holds a PhD. degree in engineering from Waseda University, Japan.

**Mark Paulis** is a Solutions Consultant for IBM Watson Content Analytics and Enterprise Content Management with IBM Software Group in the United States. He has been with IBM since 2006, architecting various Analytics, ECM, and BPM solutions for the Financial Services, Government, Energy and Utilities, and Healthcare industries. Before joining IBM, he designed Portal and Identity Management solutions for various industries, and has worked in a various roles ranging from technical support to product management.

**Jane Singer** is the Level 3 Support Team Lead for IBM Content Classification with IBM Software Labs in Jerusalem Israel. Jane leads and provides support for presales and services. She has worked on the IBM Omnifind Enterprise Search L2 team, the Quality Assurance teams for IBM Case Manager, Case Manager mobile, and Content Navigator mobile. She has written multiple IBM developerWorks® articles on these IBM products and solutions.

**Martin Triska** was a Business Analytics and Optimization Solutions Consultant at GBS, Czech Republic. He specialized in the field of text analytics, social media, and big data technologies, together with IBM Worklight® (hybrid mobile application development) products. He also spent part of his time working for IBM Research lab in Israel, further improving MobileFirst platform.

A special thank you to Teruo Koyanagi for allowing us to base our performance chapter on his performance white paper.

A very special thank you to the following key people for guiding us, helping us to write, and revise the IBM Content Analytics Studio chapter and many other chapters in the book:

Akihiro Nakayama  
Kei Sugano

Special thanks to the following people for making this book production possible:

Seiji Hamada  
Takeshi Inagaki

We also thank the following people for their contributions to this project. We could not produce the book without their help and assistance:

Qifeng Cao  
Daisuke Hayashi  
Shunsuke Ishikawa  
Hiroaki Kikuchi  
Masaki Komedani  
Hiroshi Kurokawa  
Sayaka Kuroki  
Yuki Makino  
Yutaka Moriya  
Takuma Murakami  
Hirofumi Nishikawa  
Keisuke Nitta  
Yuichi Suzuki  
Takuya Tejima  
Fumihiko Terui  
Makoto Yamamoto  
Hirokazu Yasumuro  
Tokyo Software Development Laboratory, SWG, IBM Japan

Amine Akrouf  
Barton Emanuel  
Maurizio Gallotti  
Richard Salac  
Julie Vaccaro  
Software Group, IBM Czech Republic, Ireland, Italy, and US

## Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- ▶ Use the online **Contact us** review Redbooks form found at:

[ibm.com/redbooks](http://ibm.com/redbooks)

- ▶ Send your comments in an email to:

[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400

## Stay connected to IBM Redbooks

- ▶ Find us on Facebook:  
<http://www.facebook.com/IBMRedbooks>
- ▶ Follow us on Twitter:  
<http://twitter.com/ibmredbooks>
- ▶ Look for us on LinkedIn:  
<http://www.linkedin.com/groups?home=&gid=2130806>
- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:  
<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>
- ▶ Stay current on recent Redbooks publications with RSS Feeds:  
<http://www.redbooks.ibm.com/rss.html>

# Summary of changes

This section describes the technical changes made in this edition of the book. This edition might also include minor corrections and editorial changes that are not identified.

Summary of Changes  
for SG24-7877-02  
for IBM Watson Content Analytics: Discovering Actionable Insight from Your Content  
as created or updated on July 3, 2014.

## July 2014, Third Edition

### New information

- ▶ Chapter 2, “Use case scenarios” on page 29 - New chapter
- ▶ Chapter 11, “Customizing content analytics with IBM Content Analytics Studio” on page 405 - New chapter
- ▶ Chapter 12, “Enterprise search” on page 445 - New chapter
- ▶ Chapter 13, “Adding value to Cognos Business Intelligence” on page 487 - Update of existing chapter but with mostly new content and procedures
- ▶ Chapter 14, “Customizing and extending the content analytics miner” on page 535 - Update of an existing chapter but with new example and content

### Changed information

All chapters in the book have been revised to reflect the latest version and features of IBM Watson Content Analytics Version 3.0 (formerly known as IBM Content Analytics with Enterprise Search, or simply ICAwES). In addition, several chapters have been expanded and restructured to include more descriptions on various use cases, and more explanation about how to perform content analysis.

Specifically, the following chapters have been expanded, revised, and restructured:

- ▶ Chapter 7, “Performing content analysis” on page 231
- ▶ Chapter 8, “Performing content analysis with built-in annotators” on page 265
- ▶ Chapter 9, “Content analysis with IBM Content Classification and document clustering” on page 305

This version covers IBM Watson Content Analytics (Content Analytics) Version 3.0. More information was added to better explain how to perform content analysis to gain insights to improve business operations or products and services. Updates have also been done throughout the book to reflect the new features and functions and user interface of Content Analytics.

More detailed look of what is new in Content Analytics Version 3.0 is provided in 1.2.3, “What is new in IBM Watson Content Analytics” on page 7.

The previous version of the book covers Version 2.2.





# Overview of IBM Watson Content Analytics

With major advances in linguistic analysis of the written word, combined with the increased computational power of today's hardware, comes the field of *content analytics*. Content analytics enables businesses to gain insight and understanding from their structured and unstructured content (also referred to as *textual data*). Combining this technology with more traditional secure, enterprise search technology, IBM Watson Content Analytics (Content Analytics) Version 3.0 merges content analytics and the enterprise search capability in a secure and scalable manner. This process allows businesses to make the best use of all of their content and the insights found therein.

This chapter includes the following sections:

- ▶ Business need and the content analytics and search solutions
- ▶ History, changes, and what is new in version 3.0
- ▶ Important concepts and terminology
- ▶ Content Analytics architecture

**Name change:** This book covers IBM Watson Content Analytics (formerly known as IBM Content Analytics with Enterprise Search, or simply ICAwES).

# 1.1 Business need and the content analytics and search solutions

This section introduces you to the Content Analytics product. First, it explains the business need and problem statements. Then, it provides the content analytics and search solutions to the problems followed by a brief history of the product. Lastly, this section describes what is new in release 3.0 of Content Analytics.

## 1.1.1 Business need and problem statement

A large percentage (estimated at 80% or more) of the information in a company is maintained as unstructured content, which includes valuable assets such as emails, customer correspondence, free-form fields on applications, wikis, blobs of text in a database, content in enterprise content repositories, social media posts, and messages of all kinds. Because this content lacks structure, it is difficult to search and analyze it without extensive effort.

Searching is particularly difficult because traditional methods used on the web for search ranking algorithms, such as link analysis, often do not apply to content inside a company. Content can be found by basic keyword searching but often times the result volume is too high and the corresponding ranking of documents is not sufficient to prioritize the most pertinent content to a given user to the top of the result list.

Analyzing unstructured content is even more difficult and error prone without automation. To understand and analyze the content, generally a human must read and understand what is being communicated. As a consequence, human involvement can be an expensive and time-consuming part of your overall business process. Further complicating matters is the fact that humans have a limited concentration time on any given task and over the long periods, accuracy of human based classification and annotation can become lower and lower.

Applying content analytics to your unstructured (textual) content through the help of software can result in many benefits. In a search solution example, the ability to locate the content of interest will increase dramatically when you have attached additional metadata to the documents that will provide the user feedback as to the nature of the document. If, for example, a customer service representative is searching for information that is related to a defective product, analytics can help rank the results by extracting additional details about the product, lot numbers, customers related, and defect and even solution to allow the user to quickly find the information for which they are searching.

When applying content analytics technology in other business intelligence solution scenarios, this provides for a broad range of examples, such as:

- ▶ Improvements to product development and product quality analysis
- ▶ Enhancements to crime intelligence and public safety, finding patterns in text that indicate criminal behavior
- ▶ Improvements to fraud analytics to better identify fraudulent trends and patterns
- ▶ Improvement in health care outcomes and improvement in patient health
- ▶ Analysis of customer feedback and sentiment from emails and online postings

Content analytics technology provides insight to drive customer-oriented decision making, boosting loyalty, and creating new opportunities. All of which translate directly to more informed, better decision-making, which drives improved results.

## 1.1.2 The content analytics solution

The biggest benefit that a content analytics solution provides is the ability to use a computer to analyze massive amounts of unstructured content (textual data) in such a way as to discover the “why” to business scenarios. Traditional business intelligence, analyzing structured data such as volume of calls, average length of call, amount of sales increase or decrease, is very good at describing “what” is happening. For example, sales are up, customer satisfaction ratings are up, product defect reports are increasing. What content analytics can do is analyze the additional unstructured or textual data associated with those events and help you find out “why”. For example, customer satisfaction ratings are up because the top trending phrases for the period are “better data plan”, “wireless data is faster”, and “user interface is easier”.

Content Analytics provides the content analytics solution that processes your textual data in ways that help you to search, discover, and perform the same analytics on textual data that is currently performed on structured data in a business intelligence style of application. With Content Analytics, you can now use your text in ways that were only previously attainable from your structured data.

Content Analytics delivers new business understanding and visibility from the content and context of textual information. For example, you can identify patterns, view trends and deviations over time, and reveal unusual correlations or anomalies. You can explain why events are occurring and find new opportunities by aggregating the voices of customers, suppliers, and the market. You can track

and drive improvement in non-quantitative business metrics through content dashboards, reports, and scorecards.

With Content Analytics, you can also analyze your textual data when it is not practical to analyze it manually. For example, if you conduct a survey with 1,000,000 people on what they do over the weekend, Content Analytics helps you to analyze all 1,000,000 survey forms fast enough that the results may still be pertinent to determine how best to influence the events of the weekend.

With Content Analytics, you can define many *facets* (or aspects) of your data, with each facet potentially leading to valuable insights for various users. For example, you might define a weekend destination facet that consists of major places where people travel over the weekend. You might also define an activity facet that consists of typical activities people do during their weekend travel. With such facets, a tourist industry analyst can analyze which types of people (based on their age, profession, gender, and other aspects) tend to travel to which specific locations. You can further identify the types of activities they engage in over the weekend.

In another example, you might define a shopping place facet that consists of major places for shopping and a purchase facet that consists of items being purchased by people. With these facets, retailers can analyze the type of people that tend to buy particular items at a given location.

Content Analytics is tool for reporting statistics and for obtaining *actionable insights*. Actionable insights is a key concept that refers to insight into data that leads to action. Content Analytics provides far more value than just being a tool to reduce the workload of manual analysis. Content Analytics brings the power of business intelligence to all of your enterprise information, not just your structured information. The result helps you achieve the most value from all your data, regardless of its structure.

### **1.1.3 The analytics-driven search solution**

Applying these same text analytics driven methodologies of natural language process and text annotation to a search scenario further drives value into the organization by vastly improving findability of content in a business scenario.

Traditional enterprise search solutions are heavily reliant on structured metadata about content being searched to drive faceted navigation, allowing the users to drill down and supplement their query terms with additional filters based on this additional information. However, many times, that metadata is insufficient or completely missing to allow for accurate and timely searching.

Content Analytics provides the analytics-driven search solution that uses natural language processing (NLP) and text analytics to extract and normalize that additional information from the unstructured content. For example, an insurance claims adjuster searching for information by claim number might find what they are looking for in the claims system where the number is always a piece of metadata, but finding that same information in an email would be problematic because it is not there to search on. Even if the user entered it into the body of the email, they might have it in various unexpected formats. Similarly, things like claim date, account number, location, or reason for claim are buried in the unstructured data. Content Analytics extracts and normalizes that data from the unstructured content and allows for searching.

Furthermore, it analyzes the text regarding clustering or classification techniques and correlation or relative co-occurrence, giving users visual clues as to which search elements are important to focus on and possibly drill in on.

## 1.2 History, changes, and what is new in version 3.0

Content Analytics has gone through several revisions and changes. This section provides a brief history of the product, summarizes the changes in the product, and explains what is new in Version 3.0.

### 1.2.1 Product history

In 1997, IBM started a text mining project in IBM Research, Tokyo, Japan. It combined NLP technology that was inherited from machine translation and digital library projects.

By 1998, the Text Analysis and Knowledge Mining (TAKMI) system was developed from the text mining project. TAKMI was used to analyze 500,000 customer contact records at a PC help center in Japan. A corresponding help center in the United States started using the English version of the TAKMI system in 1999. The help center reported a significant cost reduction based on identification of product failures in their early stages with the TAKMI system.

From 2000, large enterprises started using the TAKMI system both in Japan and the US. To protect their competitive advantage, most of these companies kept their use of the TAKMI system confidential. As a direct result of using the TAKMI system, the PC help center in Japan achieved number one in problem-solving ratio of web support among PC companies operated in Japan in 2003, as ranked by *Nikkei Personal Computing*, the premiere PC magazine in Japan.

The use of the TAKMI system has expanded greatly since 2003, primarily because of On Demand Innovation Services (ODIS), in which IBM Research directly supports client use of text mining by sending researchers as consultants.

IBM LanguageWare® was at the middle of the TAKMI project. Developed by IBM, LanguageWare was a powerful, flexible Unstructured Information Management Architecture (UIMA)-based tooling environment to build underlying models that improve an organization's contextual and semantic understanding of content. As the crown jewel of the TAKMI strategy, LanguageWare Resource Workbench (now IBM Content Analytics Studio), provided a complete development environment for the building and customization of dictionaries, rules, ontologies, and associated UIMA annotators.

In 2007, IBM Software Group released the TAKMI system as a Price Requested Quote (PRQ) service offering, named *IBM OmniFind® Analytics Edition*. OmniFind Analytics Edition received Spring 2008 SSPA Recognized Innovator Awards from the Service and Support Professionals Association (SSPA).

In 2008, IBM renamed OmniFind Analytics Edition to *IBM Content Analyzer*. In 2009, IBM developed IBM Content Analytics by integrating technology from Content Analyzer.

In October 2009, version 2.1 (the first release) of Content Analytics debuted, but it was originally branded *IBM Cognos Content Analytics*. Then, in October 2010, version 2.2 of IBM Content Analytics was released.

Meanwhile, Omnifind Enterprise Edition version 8.2 was released in 2004 to address the demand for secured enterprise search solutions. It was on an annual release cycle for a few years with Omnifind Enterprise Edition version 8.3 in 2005 and Omnifind Enterprise Edition version 8.4 in 2006. Then, the product transitioned to a slightly longer 18-month release cycle and version 8.5 came out in 2008 followed by the final release of the independent Omnifind Enterprise Edition product, version 9.1, in 2010.

With the release of IBM Watson Content Analytics version 3.0, in early 2012, the two product lines were merged to provide a combined enterprise search capability and content analytics platform using a common technology infrastructure.

## 1.2.2 Product changes

The integration of the OmniFind Enterprise Edition product into the Content Analytics product resulted in an entirely new type of product in the marketplace, one that uses the power of analytics to drive analysis and derive insight that carries with it the capabilities of secure enterprise search capability.

## Major enhancements

Content Analytics provides the following enhancements since the earlier product:

- ▶ Merging of enterprise search capability and content analytics allows for the centralized management of all the enterprise search and content analytics collections in an implementation.
- ▶ Administration console has been completely reworked for ease of use and enhanced administrative function.
- ▶ Integration to Hadoop via IBM Big Insights for massive scale processing.
- ▶ Introduction of Sentiment Analytics views and administrative functions.
- ▶ Introduction of Contextual Views, allowing for documents to be broken into multiple sections for distinct analysis.
- ▶ Improvement of integration with IBM Content Analytics Studio.
- ▶ Improvement in the result ranking and result aggregation capabilities in the product.

### 1.2.3 What is new in IBM Watson Content Analytics

Content Analytics Version 3.0 has greatly extended the features and functions of content analytics as well as the functionality of enterprise search. These new capabilities benefit the general search user, system administrator, and business analyst users of Content Analytics. The new features for each are briefly described in the following sections.

#### New features for the enterprise search user

Enterprise search users can take advantage of the following new features:

- ▶ New enterprise search functions:
  - The Named Entity Extraction (NER) facets are now available for enterprise search collections. Similar to how NER functioned in prior versions with an analytics collection, administrators can now go into an enterprise search collection and enable NER with a simple check box or enable specific Named Entities (Person, Location, or Organization) individually via the administrative interface. Once enabled and the collection is reindexed, users will have additional facets for the enabled NER values (Person, Location, or Organization) in their facet tree.
  - The Document Clustering facets are also now available for enterprise search collections. Similar to how Document Clustering operates in prior versions of Content Analytics for analytics collections, administrators can now obtain a cluster proposal, edit the cluster names and proposed cluster words, and then deploy the edited cluster proposal to a specific collection.

When deployed and the document clustering global process is complete, users will have the additional document cluster facet in their facet tree.

- Document Flagging is also now enabled for enterprise search collections just as it was for analytics collections in prior versions. Administrators can enable up to 64 flags and their labels and colors and users will be able to flag individual documents or entire document sets resulting from a given query with a couple of simple clicks. These flags will then be available as a facet for additional faceted navigation and query drill-down.
- Export of Flagged Documents is now available for enterprise search users when the administrator has enabled it in the administration console. There are several options for export of flagged content including export directly to an analytics collection. This very powerful option enables enterprise search and business analyst users to work together to search through potentially millions of documents and then export only those that are pertinent to a specific analysis directly to an analytics collection for further analysis using the full power of content analytics miner.

► Analytics Driven Search view

This is a new primary search view that is enabled via the search customizer in the administration console. When it is enabled, the administrator has the ability to turn on additional panes in the search view that include the following capabilities:

- The *Query Builder* is a quick pop-up user interface that allows the user to quickly add to and modify the query without being required to know the full query syntax. Users can select an individual document in the results pane and then add and remove words from the query to further search and discover results.
- The *Query Tree* panel is an expansion of the standard query pane that allows users to modify and build complex queries without having to know the specific syntax of the Content Analytics query language.
- The *Duplicate Document Detection* capability allows the search results to be completely filtered to hide documents that are exact duplicates of each other. Users can enable a button in their preferences panel to show and hide duplicates in the search results.
- The *Find Similar* documents function is now available for enterprise search collections. When this feature is activated, you can see and group documents that Content Analytics has determined to be near duplicates of each other. The elimination of duplicates can greatly improve the accuracy of the calculations made by the system and your subsequent analysis.



- The *Time Series* pane is a new view in the enterprise search collection that mimics the time series view in a content analytics collection available in prior versions of the product. This allows the user to quickly query and drill down by date metadata as well as normalized date values extracted from the unstructured content. This really powerful capability enabled via custom annotators in IBM Content Analytics Studio (ICA Studio) extends traditional metadata-based date searching, such as file system created or modified date, to include anything that can be extracted using NLP. For example, emails containing claim dates such as “01/20/2014” and “January 20, 2014” and “Jan 20th 2014” can all be queried with a simple date range query for January of 2014.
- The *Correlation* indicator on facet entries can now be viewed as color underlines in the Facet Tree or as a color indicator bar in the Dynamic Facet Chart (see Correlation description below). This allows users to quickly identify facet values that warrant further attention in their discovery process.
- The *Custom* pane is a new pane in the enterprise search application. With this new feature, you can integrate other applications or custom interfaces that search users need to access frequently into the enterprise search application for their convenience. For example, if a user is searching documents for documents that are related to a specific claim, a custom pane could be enabled to search the claim system at the same time to quickly find the claim details without having to leave the enterprise search application.

## **New features for the Content Analytics administrator**

Content Analytics administrators can use the new administration interface to enable or disable the following new features:

► The administration console interface

The interface has been significantly refactored to allow for a vastly improved ease-of-use and user experience:

- The new dashboard style of interface allows for a quick view of all the collections, which servers for which collections are running, and which ones are stopped. Any errors are presented to the administrator at the top level of the dashboard for rapid response.
- Crawlers, document processing servers and search/analytics applications can be quickly started or stopped with the click of a button.
- The redesigned System tab allows for a graphical representation of which servers are running in a multi-node environment. Again, you can stop and start processes on different servers from a single interface with the click of a button.

- The redesigned Security tab allows for much simplified Lightweight Directory Access Protocol (LDAP) configuration with additional functions for rapid testing to ensure that the connections are configured correctly.
- ▶ Analytics Driven Search

This feature can be enabled by a user with administration privileges by running the Search customizer and then scrolling to the right to the far right tab and checking the box to enable Analytics Driven Search. When this feature is enabled and the application is restarted, the above options will be available to the end users.
- ▶ Query Rewriting

This is an advanced feature used by the enterprise search administrator to help tune and refine common user queries or important corporate concepts. There are several options for essentially a systematized way for a user's query to be modified in such a way as to return a more relevant set of results. For example, if many queries are being executed for "expense report", "expenses", and "travel expenses", the administrator might want to create a series of rules and dictionaries that allow the system to supplement that query with "Expense Accounting Submission", or whatever appropriate query will result in the appropriate internal website for expense submission being ranked the highest. Query rewriting can be done as a supplement to the original query or in replacement of the original query. In the former case, the results from the original or the rewritten supplemental query can be prioritized on top of the other results.
- ▶ Result Aggregation

This is another mechanism available to the Search administrator that can be used to ensure that users see the most relevant search results. Oftentimes, when searching a collection that has multiple data sources, at least one of those sources will be more prolific than the others. For example, if you are searching a collection that is drawing from enterprise email with hundreds of thousands of messages and an internal wiki with a few hundred pages. In most instances, the results from email will significantly outnumber the results from the wiki pages, causing the first several pages of search results being dominated by the email results. But if your users would like to ensure that they see the top results from all the data sources, you could design a result aggregation scheme that shows the top five from each source with the option to drill down further into the details from that source. This will ensure that the most significant results from the wiki pages will be seen by the users.
- ▶ Big Insights Integration configuration

This is a new option that is available when installed on supported Linux operating systems. This option creates processes on the BI Hadoop nodes, which take over the document processing and parsing stage of Content

Analytics, allowing for massive scale processing capabilities. The administrator simply needs to specify the master node name and port information and then check the “use IBM Big Insights” box when creating a large collection, and Content Analytics takes care of the rest.

- ▶ Sentiment analysis

This is an optional text analytic feature which can be enabled when creating a new collection or via the edit collection settings option in the Content Analytics administrative application. Once enabled, a new tab and view appears in the content analytics miner, which allows users to analyze sentiment in the text in a collection. Additionally, the administrator can modify the polarity of given words such that words that would normally be considered to be positive should be negative or neutral in this business context, and vice versa. For example, if you were analyzing sentiment associated with a store called “Excellent Bodega”, you would want to modify the standard sentiment associated with the word excellent to be neutral to allow for all occurrences of the word “excellent” to not be counted as false positives.

## **New features for the business analyst**

Business analysts can take advantage of the following new features:

- ▶ *Sentiment view*

This is a new view and a tab in the content analytics miner that allows business analysts to understand the sentiment associated with various concepts in the text of the documents and content that is being analyzed. For example, if you had a custom annotator that identified products for a specific company, you would then be able to understand the customer sentiment associated with each of those product names: Positive, negative, ambivalent, or neutral. Once you get a general idea of the sentiment, you can view specific details of positive and negative words and phrases, trends associated with each, and preview the sentiment words in conjunction with surrounding document snippets in preview. Adding the sentimentality to the query allows for further drill-down in other views of the Analytics Miner interface.

- ▶ *Contextual Views*

This allows for a document or piece of content to be broken into multiple sections for individual analysis. For example, a medical report analyzing “Medications” annotations could be divided into a “Family History” context/section and a “Known Allergies” context/section. An analysis of medications can vary depending on which section the business analyst has selected for evaluation.

## 1.3 Important concepts and terminology

To understand the Content Analytics product, you must first understand the key concepts and terminology that are associated with the product and technology. This section explains the following concepts and terminology:

- ▶ Unstructured and structured content
- ▶ Text analytics
- ▶ Search, discovery, and data mining
- ▶ Collections
- ▶ Facets
- ▶ Frequency
- ▶ Correlation
- ▶ Deviation

You *must* read this section before proceeding to the rest of the book.

**Fast path to content analysis:** If you are familiar with the concept and mechanical operation of Content Analytics and you are interested in content analysis immediately, use the following fast path:

1. Locate a working Content Analytics system, which can be a VMware image that has the product installed.
2. Configure the sample collection from the First Steps tutorial.
3. Read Chapter 4, “Understanding content analysis” on page 59 in its entirety.
4. Read Chapter 7, “Performing content analysis” on page 231 and Chapter 8, “Performing content analysis with built-in annotators” on page 265.

### 1.3.1 Unstructured and structured content

*Unstructured content* is information that is generally recorded in a natural language as free text. The text contains all of the complexities and ambiguities of the language that is being used. It is easily understood by a human reader but difficult to process by a computer program. In contrast, *structured information* is data that has unambiguous values and is easily processed by a computer program.

For example, in a typical email, the from, to, and date fields contain structured data with implied rules about what they mean and how they are to be processed. The from and to fields are email addresses. The date field has a data value and conforms to one of a limited set of date formats. Conversely, the body of the

email (a field itself) has no implied structure, only to the extent that the text conforms to the grammatical rules of a particular natural language. Even then, the variance in each user's style of writing cannot guarantee that all grammatical rules are followed precisely.

In this book, unstructured content is referred to as *textual data*.

### 1.3.2 Text analytics

*Text analytics* is a general term that refers to the automated techniques of converting textual data into structured data. A program that reads text and extracts person names is considered a text analytic. A program that classifies the content into one or more categories based on the text that was read is also a text analytic. After the information in the text is converted to a structured form, the data can then be processed by conventional business intelligence and data processing tools that are available.

### 1.3.3 Search, discovery, and data mining

Search and discovery are oftentimes mistaken as having the same meaning or at least as being interrelated in some way. However, search and discovery are different concepts. One way to contrast the two is to classify them by what you know and do not know. You search for what you know and discover what you do not know.

When you *search*, you already have a target in mind, such as a document, product, or piece of information. Your task is to formulate your query in a way that improves the chances for an exact or partial match of the target document. *Keywords* in your query tend to be more descriptive to qualify exactly what you are looking for. For example, the query “replacement air filter for a car model <x> year <y>” leaves little room for ambiguity.

**Keywords:** As the term implies, keywords are usually words and phrases that are extracted from textual content. However, they can also be obtained from structured fields such as date or numeric fields.

*Discovery* is exploratory in nature and is generally goal driven. A search engine becomes a discovery engine when the query is used as a starting point from which to learn more about a particular topic.

*Data mining* is the process of identifying patterns in your data that might be used to answer a business problem, question, or concern. Data mining is a natural part

of discovery. Many techniques can be used in the data mining process with the patterns revealed in many different ways.

Content Analytics embodies each of the concepts described previously. The product is both an enterprise search and content discovery tool as well as a data mining tool for textual content. It combines the results with those of your structured data. By using the content analytics miner, you can explore and mine your data regardless of where it comes from and whether it is structured or unstructured. Search is also integrated into the product, so that you can limit your analysis to only those documents that match your search query.

### 1.3.4 Collections

A single content analytics or enterprise search *collection* represents the entire group of documents that are available to an application for search and analysis. An application can access multiple collections. The entire group of documents within a collection is sometimes referred to as a *document corpus*.

You can set up your collection as an *enterprise search collection* or a *content analytics collection*. An enterprise search collection is set up for use in a case of the enterprise search application. A content analytics collection is set up for use when discovery and data mining are required.

Content Analytics supports the creation of multiple collections. Each collection has its own set of configuration files and processes, such as crawlers, document processors, an indexer, and search run times. A collection is empty until content is added to it through the definition and scheduling of one or more crawlers and subsequent parsing and indexing of the crawled content. Documents can also be pushed into a collection by using the Representational State Transfer (REST) application programming interfaces (APIs).

Additional configuration options are available when defining crawlers and configuring the parser, indexer, and the enterprise search application. For more information, see 1.4, “Content Analytics architecture” on page 18.

### 1.3.5 Facets

*Facets* represent the different aspects or dimensions of your document corpus. They are a crucial mechanism for navigating and analyzing your content with the content analytics miner.

Facets can be populated with values that are obtained directly from the structured data fields in your collection. For example, you might have a set of traffic accident reports. Each report records the time of day that the accident

occurred, the road condition (dry, wet, or icy), the number of vehicles involved, and so on. Each report also contains a free-form text field where a detailed description of the accident is recorded.

You want to investigate whether the road condition is a factor in causing accidents. Therefore, you create a road condition facet in Content Analytics. You populate the facet with the values from the road condition field of the traffic accident report.

Facets can also be populated with information from your text. For example, you want to know what type of cars were involved in the accidents. Therefore, you create a car model facet. The traffic accident report does not contain a structured field for car model. However, this information might exist in the detailed description field. To identify and extract the car model value for the facet, you employ text analytics to the description field and obtain the value from it for further analysis.

A set of default facets is automatically defined and populated by Content Analytics. The default facets represent the parts of speech (such as nouns and verbs) discovered in your text. Phrase constituent is another default facet. These facets provide important dimensions to your data and are a useful tool in your analysis.

When used alone or combined with a search query, facets provide a powerful way to navigate and filter your corpus of documents so that you focus on only those documents that are relevant to your analysis. In the previous example of traffic accidents, consider that you only want to focus on accidents that occurred when the roads were wet. You can easily obtain this information by selecting the wet value in the road condition facet and adding it as a constraint to your query.

Let us further assume that, for this smaller set of accident reports, you also want to focus on just those accidents that involved a brake failure. A structured field that identifies brake failure does not exist in the accident report. However, you can still narrow your document set by adding a search query term, such as “brake failure,” to dynamically filter your documents.

Upon defining your facets and building a content analytics collection, all of your facets are displayed in the content analytics miner. Where the data came from is irrelevant after it is examined by the content analytics miner. Most important is that you now have a way to navigate through your documents by the various dimensions represented by the facets.

You can now pose queries regarding the following information:

- ▶ Show me the distribution and frequency of accidents across the different types of road conditions (using the road conditions facet).

- ▶ Show me the frequency of accidents across the different models of cars involved (using the model car facet).

More importantly you can compare facets to each other to determine if they are correlated. See 1.3.7, “Correlation” on page 16, for more information about correlations.

### 1.3.6 Frequency

*Frequency counts* in Content Analytics represent the total number of documents that contribute to a particular keyword. The frequency can change as your query constraints changes.

For example, by selecting the road condition facet described in 1.3.5, “Facets” on page 14, the content analytics miner shows the total number of documents (frequency) for each keyword: dry, wet, and icy. You can add terms, such as “brake failure,” to the search query to further constrain the document set. In this case, the frequency counts are automatically updated to reflect only those document totals within the number of accidents reports that contain the words “brake failure” in their description field.

Frequency counts can be useful in identifying trends in your data. The content analytics miner, for example, shows whether the number of car accident reports are increasing or decreasing over time.

### 1.3.7 Correlation

Although frequency counts are useful, relying on them alone can sometimes be misleading. Because a particular keyword has a high frequency count does not mean that it is relevant to your analysis. Content Analytics also calculates a correlation statistic for facets.

*Correlation* is a measure of how strongly a facet value is related (correlated) to the current query or selection criteria. In a facet pair, it indicates how two facets are correlated to each other. It is used to better gauge the relevance of a particular keyword as it compares to other data in your document corpus.

Let us consider how correlation values can be used in the traffic accident analysis. In this case, you want to focus on traffic accidents that occurred when the roads were wet. You have the car model facet and the road condition facet. You set the road condition facet to the “wet” value. Content Analytics automatically updates the correlation values of all facets as they compare to wet road conditions. The higher the correlation value is, the more correlated they are



to each other. In this example, the higher the correlation value is, the more relevant the traffic accident is linked to the wet road condition.

Figure 1-1 shows all the traffic accidents reported for car model X and car model Y. It also shows the number of car accidents when the road condition is wet for both car models. In this diagram, the letters X and Y represent different models of cars involved in accidents. Each instance of a letter is a unique accident report for the car model. Thus you have the following statistics:

- ▶ A total of 20 traffic accidents involved car model X, 10 of which occurred when the roads were wet.
- ▶ A total of nine traffic accidents involved car model Y, seven of which occurred when the roads were wet.

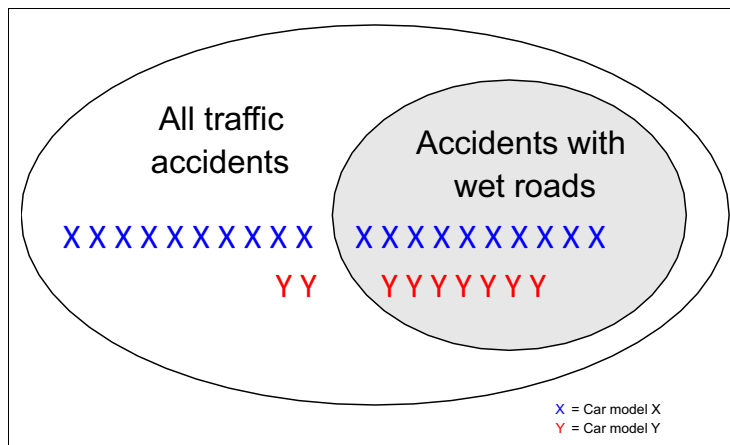


Figure 1-1 Car models correlated with a wet road condition

At first glance, based on the frequency of accidents alone, you might think that car model X has a problem when the roads are wet because traffic accidents occurred 10 times when the roads were wet for car model X. However, upon further examination, you notice that car model X has the same number of accidents regardless of whether the roads were wet. This means that a wet road condition might not be statistically relevant for the traffic accidents for car model X. The high number of accident reports for car model X (when the road was wet) is probably because car model X is a popular selling car and more of them are on the road.

Using the correlation value calculated by Content Analytics, car model Y has a bigger problem when the roads are wet. People who drive car model Y tend to have higher traffic accidents when roads are wet (7 related to a wet condition in a total of 9 traffic accidents). The correlation of wet road conditions and car accidents for car model Y is much higher than for car model X.

A high correlation value is important and is worth further investigation. In the example, the high correlation value indicates that a problem might exist with car model Y when the roads are wet. The problem can contribute to brake failure or an electrical problem caused by wet road conditions. Therefore, further investigation is recommended.

### 1.3.8 Deviation

In addition to frequency and correlation statistics, Content Analytics can identify trends and patterns that occur over time. The time is based on one or more date fields that you identify in your data set. In the traffic accident example, the date on which the accident occurred can serve this purpose.

*Deviation* measures the average change in a facet over *time*. It is a weighted, moving average. Content Analytics first establishes the norm for the facet. In the traffic accident example, the norm is the average number of daily traffic accidents. When severe weather occurs, such as an ice storm, the roads might be icy, and an unusually high number of accidents might occur in a particular region during that time. Deviation calculated by Content Analytics shows a higher frequency in that timeline relative to data for other timelines and highlights the data during that time. Other data is not highlighted in response to over-time changes. The deviation aims to measure how facets deviate from the average frequency over a specific time period.

## 1.4 Content Analytics architecture

This section provides details about the overall architecture of Content Analytics including a description of each component, the flow between components, scalability, and security.

### 1.4.1 Main components

Content Analytics consists of the following major components as shown in Figure 1-2 on page 19:

- ▶ Crawlers: Extract content from enterprise data sources.
- ▶ Document processors: Process crawled documents in preparation for indexing.
- ▶ Indexer: Builds a document index for high-speed text mining and analysis.
- ▶ Search run time: Services user search and analytic requests.

- ▶ Content analytics miner: Used to perform text analysis.
- ▶ Administration console: Used to configure and administer Content Analytics.

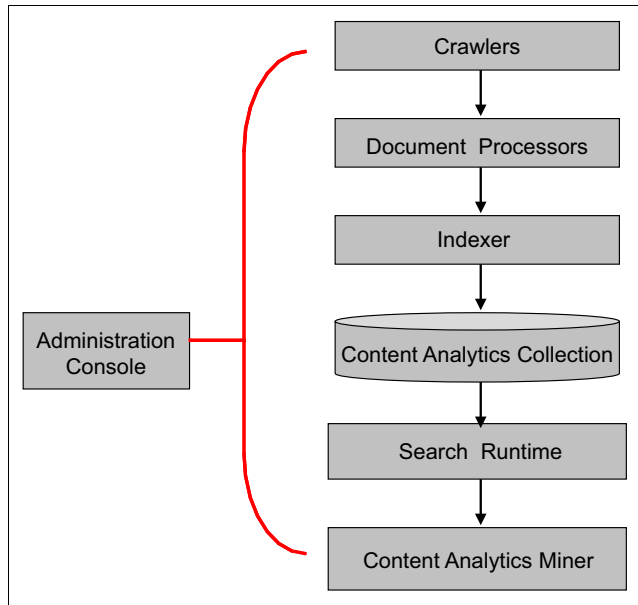


Figure 1-2 Component architecture of Content Analytics

## Crawlers

*Crawlers* extract content from the various enterprise data sources at intervals configured by the administrator. Crawlers are available for many different types of enterprise data sources. Content Analytics supports the following categories of crawlers:

- ▶ Web-based crawlers that support the HTTP/HTTPS and Network News Transfer Protocol (NNTP)
- ▶ Enterprise data source crawlers that support IBM Content Manager, IBM FileNet Content Manager, Microsoft SharePoint, and IBM Lotus® Web Content Management
- ▶ IBM Case Manager
- ▶ IBM Connections
- ▶ Collaboration crawlers that support IBM Domino®
- ▶ File system crawlers
- ▶ IBM WebSphere® portal crawlers

- ▶ Relational database crawlers such as IBM DB2®, Oracle, Microsoft SQL Server, and Java Database Connectivity (JDBC)-supported database access
- ▶ Email crawlers that support IBM Lotus Notes® and Microsoft Exchange

Most crawlers can crawl multiple data sources of the same type and do so with multiple threads, the number of which are configurable.

You can set up rules for crawlers to govern their behavior. For example, you can specify rules to control how a crawler uses system resources. The set of data sources that is eligible to be crawled constitutes the *crawl space*. You can edit the properties of a crawler to alter how it collects data. You can also edit the crawl space to change the crawler schedule, add new sources, or remove sources that are not to be searched any longer.

Crawlers can be started and stopped manually, or schedules can be set up. When scheduling a crawler, you specify when it must run initially and how often it must visit the data sources to crawl new and changed documents.

Some crawlers, such as those for web and NNTP sources, run continuously. After specifying the URLs or NNTP news groups to be crawled, the crawler returns periodically to check for data that is new and changed. The frequency of crawling the web and NNTP data sources is determined automatically based on configuration guidelines that specify the lower and upper limit of crawl frequencies. For example, you might specify a lower limit of weekly crawl frequency and an upper limit of daily crawl frequency. These limits guide the determination of the actual crawl frequency with statistics accumulated about the percentage of changes detected over past crawl frequencies.

Each data source type is associated with a different crawler type. For example, all IBM DB2 data sources have a DB2 crawler, and file system data sources have a file system crawler. However, multiple crawlers can be defined for different data sources of the same data source type. For example, one crawler can be defined for a set of DB2 tables in the human resources system, while another crawler can be defined for a set of DB2 tables in a data warehousing system.

One or more security tokens can be associated with crawled documents. A security token plug-in can be written to generate relevant security tokens for a document by using appropriate lookups of access control lists (ACLs).

## Document processors

The *document processor* component is responsible for processing crawled documents and preparing them for indexing. In this component, the various text analytics are applied to the crawled documents. Each text analytic is referred to as an *annotator* because its job (in most cases) is to annotate a document with additional information that it extracts or deduces from the document content.

Annotators are built according to the Unstructured Information Management Architecture (UIMA) standard. UIMA is an open source standard sponsored by the Apache organization.<sup>1</sup> Content Analytics is UIMA-compliant and has a UIMA document processing pipeline built in. With this pipeline, you can plug in a prescribed number of text analytics annotators to process and extract what you need from your text (as illustrated in Figure 1-3).

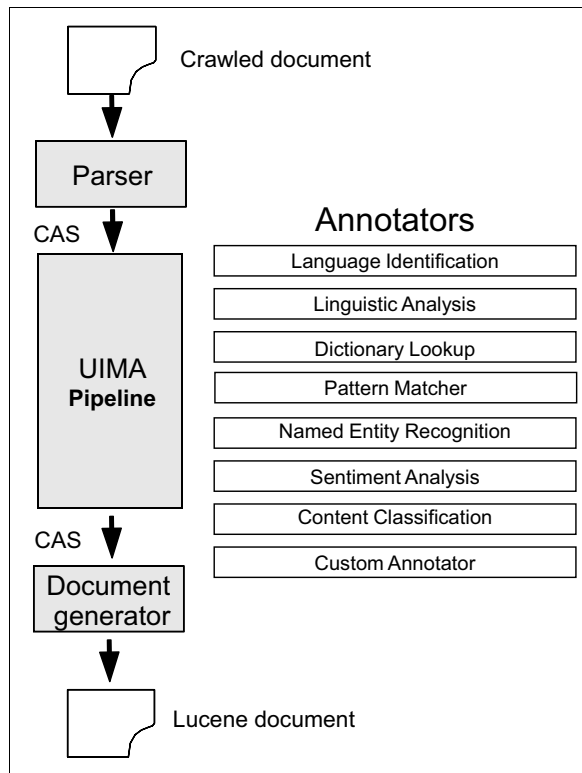


Figure 1-3 Document processor architecture in Content Analytics

<sup>1</sup> See <http://uima.apache.org>

In the UIMA pipeline, Content Analytics provides a set of required annotators that are necessary to begin the text analytics processing:

- ▶ Language Identification annotator: Identifies the language of each document.
- ▶ Linguistic Analysis annotator: Applies linguistic analysis for each document.

These annotators cannot be changed. Content Analytics also comes with the following annotators that you can configure or enable next in the UIMA pipeline, each adding their own annotations to the incoming documents:

- ▶ Dictionary Lookup annotator: Matches words and synonyms from a dictionary with words in your text. The annotator also associates the keywords with user-defined facets.
- ▶ Pattern Matcher annotator: Identifies patterns in your text by using the rules that you defined. The annotator also associates the patterns with user-defined facets.
- ▶ Named Entity Recognition annotator: Extracts person names, locations, and organization names. This annotator can only be enabled or disabled. It cannot be modified.
- ▶ Sentiment Analysis annotator: Extracts sentiment expressions in documents to realize the new Sentiment view by integrating a sentiment extractor engine as a UIMA annotator.
- ▶ Content Classification annotator: Classifies content into categories.

You can also add your own custom annotator to the pipeline, which is run at the end.

For more information about annotators, see the following chapters:

- ▶ For the Dictionary Lookup and Pattern Matcher annotators, see Chapter 8, “Performing content analysis with built-in annotators” on page 265.
- ▶ For the Content Classification annotator, see Chapter 9, “Content analysis with IBM Content Classification and document clustering” on page 305.
- ▶ For all other annotators, see Chapter 11, “Customizing content analytics with IBM Content Analytics Studio” on page 405.

UIMA and its constituent annotators use a Common Analysis Structure (CAS) to represent each document as it is processed. The CAS allows for the independent development of text analytic annotators. Each annotator adds its own annotations to the CAS. The CAS in its entirety is then made available to the next annotator in the UIMA pipeline.

The parser subcomponent is responsible for converting the crawled document in its native format into a CAS. The output of the UIMA pipeline is a CAS that

contains the original document content plus any annotations added by the annotators. The document generator reads the CAS information and prepares the content for indexing by converting the document to a Lucene document.

## **Indexer**

The *indexer* component is responsible for building a highly optimized index of document content that is suitable for high-speed text mining and analysis. The index is based on the open source *Apache Lucene indexer* with IBM extensions. IBM is an authorized committer of the Lucene open source project and frequently contributes selected features and functions back to the Lucene community.

When started, the indexer automatically indexes documents after they are processed by the document processors. You can manually perform a full rebuild of an index at any time. This option is useful if you add a text analytic to the UIMA pipeline and you want to include its results in your content analytics collection. If you enabled the document cache option for your collection, it is not necessary to recrawl your documents again. In this situation, the document content is obtained from the document cache, which is the temporary repository of crawled content.

## **Search run time**

The search runtime component is a server-based component that is responsible for servicing user search and analytic requests. Client service requests are made by using the REST API. The REST APIs are a programming interface based on Java that operates remotely by using the HTTP/HTTPS protocol. The enterprise search application and content analytics miner are example REST APIs client applications that make service requests to a search runtime component.

A single content analytics collection is associated with at least one search run time or multiple search run times to support large-scale multiuser environments. The search runtime components are not dependent on the indexer component and are designed for continuous operation. To maintain this independence, they operate on copies of their associated content analytics collection. For more information about this topic, see 1.4.2, “Data flow” on page 24.

## **Content analytics miner**

The content analytics miner is what you use to perform content analysis. Its user interface is browser-based and communicates with the content analytics miner that runs under either Jetty or WebSphere Application Server. Jetty is the default installation. The Content Analytics Miner web application issues REST client requests to the search run time associated with a given content analytics collection. The search runtime server component can either be installed locally or remotely from the analytics miner application.

With the content analytics miner, you can easily switch between multiple content analytics collections within a given browser session but can operate on only one content analytics collection at a time. Concurrent but independent analysis of multiple content analytics collections can be achieved with multiple browser sessions, one for each content analytics collection.

For more information about the content analytics miner, see Chapter 5, “Content analytics miner: Basic features” on page 85, and Chapter 7, “Performing content analysis” on page 231.

### **Administration console**

Content Analytics comes with a robust administrative component. With this component, you can create and administer collections, start and stop components, and monitor system activity and log files. You can also configure administrative users, the enterprise search application, and content analytics miner, the collections, and specify information to enforce security. Similar to the content analytics miner, the administration console is browser-based.

For more information about the administration console, see section 4.2, “Administering Content Analytics” from the previous version of this IBM Redbooks publication.

## **1.4.2 Data flow**

The primary entity in an Content Analytics system is the content analytics collection. A *content analytics collection* is an optimized index of your document content that is designed for high-speed text mining and content analysis.

The administrator’s job is to create, configure, and manage content analytics collections (see Figure 1-2 on page 19). Business and research analysts use the content analytics miner to analyze their data in the content analytics collection.

After a content analytics collection is initially created, the administrator configures one or more crawlers for the collection. Crawlers are responsible for extracting the data from the enterprise and storing the data as documents in a cache on disk. Crawlers can be scheduled on a recurring basis or started on demand and can operate independently of the rest of the system components.

The documents are read from the cache by the document processors. Document processors run all configured text analysis annotators against the content and prepare the content for indexing by formatting the content into a Lucene document (see Figure 1-3 on page 21). Because document processing is a prerequisite step before indexing, the document processor component is started when the indexing subsystem is started.



The indexing component stores the Lucene documents into the content analytics collection. The flow of documents just described is typically a continuous operation when the indexing component is active. As soon as a crawler extracts documents from the enterprise and places them into the cache, the document processors pick them up and prepare them for indexing. Likewise, after a Lucene document is available to be indexed, the indexing component picks it up, indexes the document, and stores the information in the index. This series of steps continues until all of the designated documents are fetched by the crawlers and stored into the index. The next time the crawlers run, the steps are repeated.

Let us assume that the crawlers have done their work and a content analytics collection has been successfully built. Let us also assume that you have added another text analysis annotator to the document processing pipeline to identify and extract person names. In this case, it is logical to assume that a full recrawl of your documents is necessary so that they enter the document processing pipeline and are eventually reindexed with person names added. But if you have enabled the document cache option for your collection, it is not necessary to recrawl your documents again. In this situation, the document content is obtained from the document cache, which is the temporary repository of the crawled content.

After a content analytics collection is built, the collection is copied to its corresponding search runtime components. With their own copy of the content analytics collection, a search runtime component can operate independently of the rest of the system components. When all of these components are installed on the same server, a copy of the generated index is not made with the search run time accessing the indexer generated collection.

## **Export points in the data flow**

As documents flow through the system, they go through three major transformations. After each transformation point, the documents can be optionally exported for consumption by other external applications. In this way, Content Analytics is used as a text analytics platform by an application using only those parts that are useful to the application.

Figure 1-4 on page 26 illustrates the three export points within the Content Analytics data flow. For all three export points, the exported documents can be exported to either the file system in XML format, directly into a relational database, or to a CSV formatted file. You can write a custom Java export plug-in to change the format and destination of the exported content. For example, you can develop a custom plug-in to feed the documents directly into a case management system.

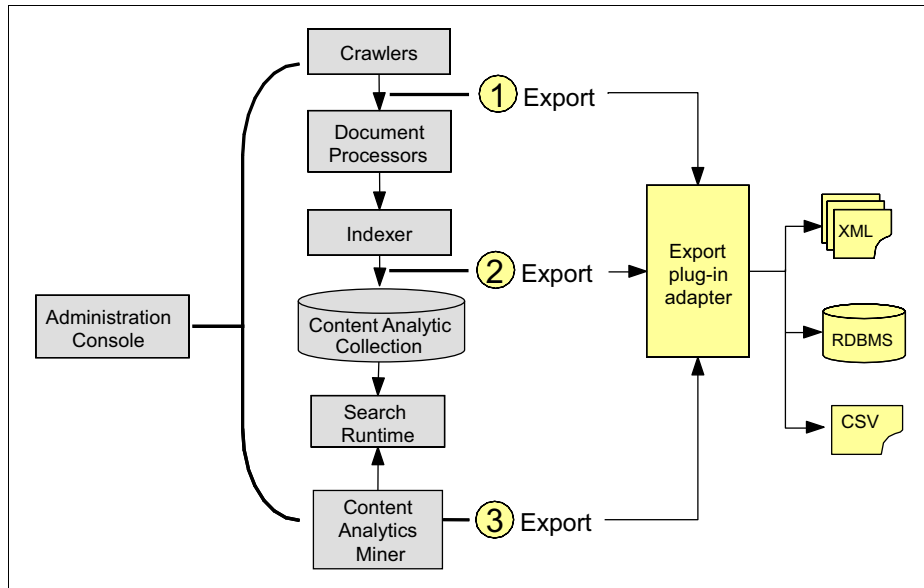


Figure 1-4 Export points in Content Analytics

The first export point is after documents are crawled and stored in the document cache. At this point, an application can intercept the original document in its binary form and any additional metadata provided by the crawler.

The second export point is after text analysis by the document processors and indexing. Here you get the same content available after the first export point plus any annotations added by the text analysis annotators in the UIMA pipeline.

The third and last export point is after a search has been performed. At any time during data analysis in the content analytics miner, you can export the current search results set. Here, the same content is available from second export point but filtered down to only those documents that match your current query.

For more information about export, see Chapter 10, “Importing CSV files, exporting data, and performing deep inspection” on page 345.

## Deep inspection

*Deep inspection* is a feature in Content Analytics that is a variation of the third export option mentioned in “Export points in the data flow” on page 25. After deep inspection has been enabled for your collection using the administration console, a deep inspection icon is displayed in each content analytics miner view (except for the documents view). When you click the deep inspection icon, a batch submission is made to deep inspection for subsequent background

processing of your current query. The results of the deep inspection are saved into an XML or CSV file on the file system. The exported results from deep inspection are not the individual search result document but rather the currently selected keywords along with their frequency counts and correlation values. In this way, deep inspection provides the same results as viewed from the content analytics miner.

The primary reason for using the deep inspection function is when the number of keywords to be analyzed is quite large (for example, greater than 500) and it is possibly causing a noticeable delay in the normal operation of the content analytics miner. The content analytics miner is designed as an on-demand text mining tool that supports rapid calculations in response to changes in a query. As the number of keywords increases, the time it takes to perform the calculations also increases. The default is 100 keywords. When increasing the value, be aware of the implication on the performance.

Deep inspection runs in the background and saves its results as an XML or CSV file in the file system. You cannot view the deep inspection results in the content analytics miner user interface.

For more information about deep inspection, see 10.7, “Deep inspection” on page 387.

### **1.4.3 Scalability**

Content Analytics is an advanced text analytics product that can grow with your needs. As you start to realize the benefit of text analytics, you might find new applications for the product resulting in more content analytics collections and business analysts using the content analytics miner.

Adding more users to the system can consume the resources of the search runtime component, eventually compromising the performance of the content analytics miner. To improve performance, you can use Content Analytics to scale the search runtime component across multiple machines. After each content analytics collection is built on the indexing server, it is copied to each of the search runtime servers. A single copy can be shared between search run times if using a shared mass storage device.

The document processing component, with its applied text analytics, can also be a compute-intensive and time-consuming task. Content Analytics supports scaling of the document processing component across multiple machines. Documents are distributed across the pool of available document processor machines on a round-robin basis.

Additional document processor and search runtime machines can be added to the system in real time on an as-needed basis. You are not required to shut down the system and restart it to use the added servers. Also any given document processor server can be switched to serve as a search runtime machine and vice versa.

For more information about scalability, see IBM Watson Content Analytics IBM Knowledge Center or section 15.6 “Scalability” from the previous version of this book, which can be downloaded as additional material with this book.

#### **1.4.4 Security**

A content analytics collection contains the content extracted from the enterprise. Therefore, a system must provide stringent safeguards to protect content from unauthorized access. Content Analytics addresses this need in several ways:

- ▶ Administrative access control
- ▶ Collection level access control
- ▶ User application role-based access control
- ▶ Data encryption

Security is an advanced topic that is beyond the scope of this chapter. For more information about Content Analytics and security, see Content Analytics IBM Knowledge Center or Appendix A “Security in IBM Content Analytics” from the previous version of this book, which can be downloaded as the additional material along with this book.



## Use case scenarios

IBM Watson Content Analytics (Content Analytics) is a powerful tool that can be used to provide valuable insight from your textual data (also referenced as *unstructured content*) sources. The insight depends on the questions you ask of your data and that you must consider in the context of your particular use-case scenario. This chapter presents common real-life scenarios in which Content Analytics has been used to provide actionable results and insights. From these scenarios, you can better understand where Content Analytics can be of help to your organization, and thus help you to design and prepare for your Content Analytics solution.

This chapter includes the following sections:

- ▶ Customer insights
  - Call center
  - Quality assurance
- ▶ Law enforcement and public safety
- ▶ Investigation management
  - 2.3.1, “Insurance fraud” on page 34
- ▶ Healthcare
- ▶ Case management
- ▶ Data warehouse

## 2.1 Customer insights

Many industries capture information about their customers in order to differentiate their products and services. Studying the unstructured information collected from customers either through interaction with them or from their online comments and posting regarding company's products and services might provide insights that differentiate one company over another one.

Content Analytics helps companies to gain insights about the “why” customers interact with your company, “where” and “what” there might be defective or unsatisfied products and services from the unstructured content, in addition to what the traditional structured data provides. These new insights can be used to optimize the type of products and services you deliver to your customers. Examples of where Content Analytics can be used to obtain this type of insight include call center data and quality assurance-related records and data.

### 2.1.1 Call center

Call centers are a common and often a necessary component of a customer service department for a company. Generally, through the call center, a customer contacts a company representative to ask a question or resolve an issue with a product or service offered by the organization. Call center agents log conversations between them and their customers. These logs are free-format text records (textual data). Content Analytics can be used to mine this data to find insight into the types of questions that customers ask and discover unexpected correlations between multiple products or services mentioned in the call center records.

Valuable information can be locked inside these call center records. This information can indicate the overall sentiment of the customer and their interaction with the organization. For example, the text can reveal if the customers are generally satisfied with their purchase or service, or more importantly, if they are dissatisfied. Equipped with this information, you can make better decisions about what is working and what is not. If something is not working, the transcript might reveal the root cause of the problem, such as poor services. See 4.3.1, “Voice of the customer” on page 76, for more detailed information about this use-case scenario.

Customer service departments leverage Content Analytics to support *next best actions* initiatives, by deriving insights to help trigger business function actions associated with marketing, customer retention, service, billing, and customer satisfaction. By crawling call center logs and other content sources, Content Analytics surfaces specific business insights from customer-related information

sources and uses to take future actions or interactions that make sense to the customer, driving long-term customer loyalty and value.

For more information, review the latest report from Aberdeen Group, which tells how Content Analytics is improving customer service:

- ▶ Content Analytics: Helping the Best-in-Class Drive Superior Customer Service

[ftp://ftp.software.ibm.com/software/data/sw-library/ecm-programs/Aberdeen\\_ContentAnalytics.pdf](ftp://ftp.software.ibm.com/software/data/sw-library/ecm-programs/Aberdeen_ContentAnalytics.pdf)

## 2.1.2 Quality assurance

Good quality is a measure of the commitment of a company to its customers to deliver products that they can depend on. A reduction in quality can result in a loss of customers to your competition. Therefore, it is important to maintain a high level of quality assurance by monitoring the continued quality of your products. Manufacturers have long recognized this important aspect of their business and put in place policies and procedures that track certain quality markers in their products.

For example, auto manufacturers use the maintenance and repair records from their dealerships as one source of information to track the quality of their automobiles. Maintenance records typically contain structured information that is coded by the technician that identifies the particular repair that was made (for example, a transmission repair). Usually a comment field accompanies the report in which the technician enters a more detailed description of the problem and the repair solution.

In this textual information, Content Analytics can provide an early warning of a potential quality issue that is emerging for a particular model of car. For example, Content Analytics can reveal that several technicians from different dealerships cited faulty wiring harnesses due to premature corrosion and that this particular problem only occurred for a particular model of car. With this information, you can detect the problem earlier and take corrective action before it becomes a more expensive problem.

## 2.2 Law enforcement and public safety

Law enforcement and public safety agencies use bleeding edge technologies to speed up the process of discovering, analyzing, and linking vast sources of information, including Internet sites, social media sources, investigation reports, forensic reports, personal records, and witness statements. These agencies

were in fact among the first users leveraging the strong capabilities of Content Analytics, to extract, analyze, and search crucial information and related insights from disparate sources of information and improves the speed and quality of intelligence gathering.

Using Content Analytics, agencies cannot only solve cases more quickly but also identify non-obvious relationships within data that could possibly prevent a crime from happening in the first place.

As law enforcement and public safety agencies continuously strive to sharpen crime intelligence gathering and response time, it is vital to embrace technological advances that can make a difference. Content Analytics provides that difference and benefits to the agencies, by:

- ▶ Reducing investigation time from weeks and months to hours or days.
- ▶ Analyzing unstructured information to derive new trends and patterns in a shorter time.
- ▶ Increasing forensic analysis capability by accurately extracting key entities like persons or objects of interest when investigating mass amounts of textual case information.
- ▶ Delivering rapid insight by connecting structured data with unstructured information to provide a 360-degree view of suspects and relationships.
- ▶ Easing strain on law enforcement budgets by improving operational efficiency.
- ▶ Freeing up personnel to pursue faster case resolutions with historical and real-time analysis of trends and patterns in unstructured information and related structured content.
- ▶ Analyzing web pages and social media sites to identify potential threats before they occur.

For an example of how Content Analytics helped fight crime, refer to the following IBM Solution Brief:

IBM Content Analytics: Rapid insight for crime investigation

<http://public.dhe.ibm.com/common/ssi/ecm/en/zs03073usen/ZS03073USEN.PDF>

## 2.3 Investigation management

Today, effective investigations are a critical part of different industries. Whether it is the investigation of potential fraud or suspicious activity, enhanced due diligence when it comes to a customer request, or background checks as part of



an application process, an effective and efficient investigative approach gives you the ability to move forward with confidence, taking action on activity that is potentially harmful to the business while quickly processing legitimate requests and transactions from loyal and high-value customers.

IBM delivers an offering called *IBM Intelligent Investigation Manager (IIM)*, which delivers a powerful, integrated set of capabilities for managing different types of investigations. Intelligent Investigation Manager optimizes investigation and analysis for insurance, banking, healthcare, public safety programs for government and law enforcement organizations, and for other industries. It dynamically coordinates and reports on cases, provides analysis and visualization, and enables more efficient and effective investigations.

This solution helps organizations in the following ways:

- ▶ Improves investigation effectiveness with a flexible case investigation environment
- ▶ Expedites investigations by analyzing and visualizing fraud within structured and unstructured data across silos
- ▶ Augments existing enterprise fraud management solutions with advanced visualization, analytics, and case management

Intelligent Investigation Manager integrates best-of-breed case management, forensics, analysis, and content analytics to optimize investigations that are based on IBM Case Manager, Content Analytics, and IBM i2® solution components.

The content analytics component makes the content of the investigation case actionable using its textual analytics approach. New entities are discovered that contribute to the investigation. Analysts and investigators can conduct searches and analytics across cases and repositories to uncover new intelligence and new supporting evidence.

For more information about this topic, refer to the following sites:

- ▶ IBM Intelligent Investigation Manager offering web page:  
<http://www.ibm.com/software/ecm/investigation-manager>
- ▶ Demos, white papers, analyst reports, brochures, and data sheets, including:
  - Aite Group Paper: Fraud Investigation Management for Finance
  - Aite Group Paper: Fraud Investigation for Insurance
  - Pursue fraud with IBM Intelligent Investigation Manager: an IBM Solution brief

<http://www.ibm.com/software/ecm/investigation-manager/downloads.html>

For an example of how Content Analytics helped fight crime, refer to the following IBM Solution Brief:

IBM Content Analytics: Rapid insight for crime investigation:

<http://public.dhe.ibm.com/common/ssi/ecm/en/zzs03073usen/ZZS03073USEN.PDF>

### 2.3.1 Insurance fraud

Insurance fraud is a targeted application of investigation management and a serious crime that affects all of us in the form of higher insurance premiums. Insurance fraud is the act of requesting reimbursement for expenses incurred because of an insured accident, procedure, or service that did not actually happen. Insurance companies spend enormous amounts of time and effort to detect insurance fraud. By reducing the amount of fraud incurred, insurance companies can increase their profit and be more competitive through lower premiums.

Insurance fraud analysis has traditionally been performed only with structured information about the insurance claim forms and other supporting documents. Such information includes the name of the claimant, their date of birth, and the date of the last insurance claim. With Content Analytics, the information in the text, such as notes made by the insurance adjuster or comments made by eye witnesses, can now be used. With Content Analytics, new patterns can be exposed from the data extracted from this text. For example, the text might reveal, in many fraudulent cases, a lack of trauma in the claim report. The reason is that it is unlikely that people might harm themselves to satisfy a claim. Lack of trauma can then become one factor among many others that indicates potential fraud.

Content Analytics analyzes vast and dispersed amount of data and provides valuable insights, guiding recommended next best steps across multiple operations to accelerate and improve fraud detection. Content Analytics is also used within a larger scope of an intelligent investigation solution to go beyond just transaction-level and account-level views to analyze all related activities and relationships in a holistic network dimension. See 2.3, “Investigation management” on page 32 for more detailed information about this extended use-case scenario.

Fraud challenges not only apply to the insurance industry, fraud costs private and public sector enterprises hundreds of billions in revenues each year. Today, fraud affects the following types of enterprises:

- ▶ Banking
- ▶ Energy and utilities
- ▶ Taxation
- ▶ Healthcare
- ▶ Insurance
- ▶ Warranties
- ▶ Worker's Compensation
- ▶ Travel
- ▶ and more

## 2.4 Healthcare

One industry where Content Analytics demonstrates strong value is in healthcare. Today, the global healthcare industry is redefining itself to reduce costs and more efficiently manage its resources while improving patient care. To deliver that strategy, healthcare organizations are becoming more data driven, defining their data as a very strategic asset, and putting their processes and systems to access and analyze the right data and derive actionable results.

The ability to analyze a wide volume and variety of content and data is key to that strategy, not only from traditional sources such as electronic medical records (EMRs), doctors notes, healthcare operational systems, but also from nontraditional sources such as social media and public health records.

IBM delivers a series of offerings in which Content Analytics plays a major role:

- ▶ IBM Content Analytics for Healthcare  
The native Content Analytics product, complemented with an IBM asset entitled "Healthcare Annotator". A deeper description is provided later in this section.
- ▶ IBM Content and Predictive Analytics for Healthcare  
Provides the above capabilities from IBM Content Analytics for Healthcare, complemented with IBM predictive analytics (IBM SPSS®), predictive scoring, predictive modeling, probability, and outcome analysis, complementing traditional healthcare predictive analysis with healthcare-specific insights extracted from textual content by IBM Content Analytics for Healthcare.

- ▶ IBM Advanced Care Insights  
Provides the above capabilities from Content and Predictive Analytics, complemented by IBM specific assets delivering similarity analytics, utilization pattern analysis, patient similarity algorithms, treatment efficacy, outflow visualization, and so on. Content Analytics focuses on extracting the medical insights from the healthcare information and feeds data to the predictive models.
- ▶ IBM Patient Care and Insights  
A comprehensive healthcare solution providing data driven population analysis to support patient-centered care processes. The solution enables interactive, secure collaboration between patients, care givers, healthcare providers, and payers. The solution is uniquely capable of analyzing structured and unstructured data, from various types of documents, patient records, and claims information, then using this data to predict health risks such as onset of disease or readmission. IBM Patient Care and Insights includes the above capabilities from IBM Advanced Care Insights, complemented by IBM Care Manager, an IBM Case Manager custom solution, providing a patient-centered care management platform bringing together the patient care team (patient, family, nurses, doctors, and so on) to ensure the best follow-up to the patient situation.

This section specifically focuses on IBM Content Analytics for Healthcare, as it represents the base for all of the preceding solution offerings. IBM Content Analytics for Healthcare leverages the base Content Analytics product, combined with a custom annotation asset identified as the *Healthcare Annotator*. Built in ICA Studio, the Healthcare Annotator is a series of content analytics UIMA custom annotators developed to identify valuable facts in medical and healthcare unstructured documents and convert those facts into a structured form, as healthcare insights. In the healthcare and medical fields, textual content is usually found in the following places:

- ▶ Physician notes and discharge summaries
- ▶ Patient history, symptoms, and non-symptoms
- ▶ Pathology reports
- ▶ Satisfaction surveys
- ▶ Claims and case management data
- ▶ Forms-based data and comments
- ▶ Emails and correspondence
- ▶ Trusted reference journals
- ▶ Paper-based records and documents
- ▶ Other medical and healthcare content sources

After crawling one or more of those sources, Content Analytics processes each document within its document processor pipeline, configured to leverage the custom annotator, which would be able to identify the following indicators:

- ▶ Problems indicators
  - Result of a series of interim annotations that identify diseases, symptoms, and disorders
  - Normalize to standard terms and standard *Unified Medical Language System* coding systems, including SNOMED CT, ICD-9, Hierarchical Condition Categories (HCC)
  - Capture problem time frames, determining if past or current problem
  - Determine confidence, like positive, negative, rule out, and so on
- ▶ Procedures indicators
  - Support for compound medical procedures
- ▶ Medications indicators
  - Result of a series of interim annotations that identify drugs, administrations, and measurements
  - Normalize to standard terms, including RxNorm
- ▶ Demographic and social indicators
  - Patient age, living arrangement, employment status, smoking status, alcohol use, and so on
- ▶ Compliance and noncompliance indicators
  - From patient history of compliance or noncompliance with medical instructions
- ▶ Laboratory results indicators
  - Type of laboratory test, unit of measure, result values, and so on
  - Coding systems
  - Identify various coding system values from Current Procedural Terminology (CPT), CCS, HCC, and National Drug Codes (NDC)
- ▶ Ejection fraction indicator
  - Specific measure for Congestive Heart Failure (CHF) cases

As you can see, IBM leverages Content Analytics at the center of a major investment supporting the healthcare industry, by tapping deeper into unexplored unstructured content sources, representing a larger volume of data for any industries.

For more information about this topic, refer to the following web pages:

- ▶ IBM Content and Predictive Analytics for Healthcare web page:  
<http://www.ibm.com/software/ecm/content-analytics/predictive/healthcare.html>
- ▶ IBM Patient Care and Insights web page:  
<http://www.ibm.com/software/ecm/patient-care>

## 2.5 Case management

Many organizations struggle with the changing nature of casework, often having to do more with less: Less time, less budget, and less staff. The IBM Case Manager solution is an advanced case management offering that unites information, people, and processes to provide a 360-degree view of case information and to optimize outcomes through integrated business rules, collaboration, and analytics. The result: organizations can work cases more efficiently and with better business results.

IBM Case Manager leverages Content Analytics to analyze and visually explore large volumes of case-specific content to unlock new business insights, by:

- ▶ Using its natural language processing (NLP) capabilities to enable analysis of unstructured content to extract facts and concepts that facilitate making fact-based decisions
- ▶ Providing content classification capabilities that allow knowledge workers to automatically categorize information for improved accessibility, greater usability, better compliance, and enhanced analysis
- ▶ Leveraging a specific Case Manager crawler to return results of structured and unstructured data, regardless of where they are located. The crawler can be configured with filters to include or exclude specific case objects, such as case type, document type, file extension, or document name pattern. The case crawler allows to index cases for specific properties, for example, name, state, creation date, or comments. You can specify which properties to index in the Content Analytics administration console, when configuring the crawler properties.

For example, content analysis is performed on a credit card dispute case management solution, where specific dispute documents and data are investigated. A bank dispute agent noticed that a merchant has gone out of business and used the case analytics tool to search for cases against the specific merchant to determine whether other customers are affected. He also analyzes recent transactions involving the merchant to determine the bank's level of

exposure. He then can navigate case content by specific facets (Vendor, Industry, Dispute Type, and so on) and specific trends, deviations, or correlations can be investigated.

In conclusion, Case Manager can use the natural language processing techniques of Content Analytics to discover patterns, trends, and correlations between various cases. These discoveries help the caseworkers and executives with insights into their business and an opportunity to respond with appropriate actions.

## 2.6 Data warehouse

Success in any enterprise depends on having the best available information in time to make sound decisions. Anything less wastes opportunities, costs time and resources, and can even put the organization at risk. But finding crucial information to guide the best possible actions can mean analyzing billions of data points and petabytes of data, whether to predict an outcome, identify a trend, or chart the best course through a sea of ambiguity. Companies with this type of intelligence on demand react faster and make better decisions than their competitors.

Content Analytics is an extensible platform that had been used in specific data warehouse and business intelligence use cases. This section presents a specific integration pattern with IBM Netezza®, a leading data warehouse appliance solution. Integration of Content Analytics can and has also been achieved with non-IBM data warehouse solutions, such as, but not limited to, Teradata, Oracle, SAP/Sybase, Microsoft, and HP/Vertica.

IBM Netezza transforms the data warehouse and analytics landscape with a platform built to deliver extreme, industry-leading price-performance with appliance simplicity. It is a new frontier in advanced analytics, with the ability to carry out monumental processing challenges with blazing speed, without barriers or compromises. For users and their organizations, it means the best intelligence for all who need it, even as demands for information escalate.

Content Analytics provides three different integration patterns with Netezza:

- ▶ **Netezza data warehouse toward Content Analytics**

The most common pattern is where the Netezza data warehouse appliance provides additional informational context to Content Analytics by sending the Netezza column data, such as customer comments, to Content Analytics. Native and custom annotators can then produce insights into various meanings, like customer satisfaction issues or sentiment analytics, using native text analytics capabilities. As needed, the Content Analytics resulting

insights can be injected back into the Netezza appliance, as defined in the next integration pattern.

► Content Analytics insights towards Netezza

Content Analytics provides analytical strength to extract insights from textual information. Data warehouses and business intelligence solutions can greatly benefit from that additional information, and those insight results can be loaded into Netezza data warehouse appliance for querying. In short, any unstructured content becomes available to users of Netezza data warehouse appliances for extracting, integrating, and reporting.

You can integrate Content Analytics results to Netezza in the following ways:

- By producing analyzed documents such as XML objects on disk, then using tools such as IBM DataStage®, or other ETL tools, to load the XML information into Netezza
- By producing analyzed documents as CSV objects on disk, then using either the Netezza NZLoad tool, IBM DataStage, or other ETL tools to inject them into Netezza
- By directly injecting analyzed documents into Netezza, using Content Analytics native JDBC export capability

► Cross-system queries via UDF

Applications or systems queries can execute against Netezza, which can then broker to Content Analytics via UDF for specific information residing in Content Analytics. Results are then aggregated by the Netezza system as part of the SQL execution plan.

For more details about how Content Analytics integrates with Netezza, refer to the following IBM Solution Brief:

*Discover deep insight with IBM Content Analytics and IBM Netezza:*

<http://public.dhe.ibm.com/common/ssi/ecm/en/ims14378usen/IMS14378USEN.PDF>





## Designing content analytics solutions

The main objective of IBM Watson Content Analytics (Content Analytics) is to help your organization derive new business understanding, insights, and visibility from the content and context of your textual information, either kept in structured or unstructured sources. When starting a content analytics project, it is important to spend time thinking about the overall design of your content analytics solution and your expectations of the resulting analysis. You must do this phase before attempting to work with Content Analytics. A carefully designed content analytics collection and corresponding content analytics miner can unlock the value in your textual sources and become an indispensable component of your overall corporate business analytics assets.

This chapter addresses content analytics miner design and data preparation. It includes the following sections:

- ▶ Data considerations
- ▶ Guide for building a content analytics collection
- ▶ Programming interfaces

For information about how to use the content analytics miner, and how to perform content analytics, refer to the chapters that follow this one.

## 3.1 Data considerations

This section presents several aspects of data that you must consider when designing and configuring your content analytics application. Understanding these data characteristics and how they interact with the Content Analytics data model is important.

### 3.1.1 Content analytics data model

Figure 3-1 provides an overview of the Content Analytics logic data model.

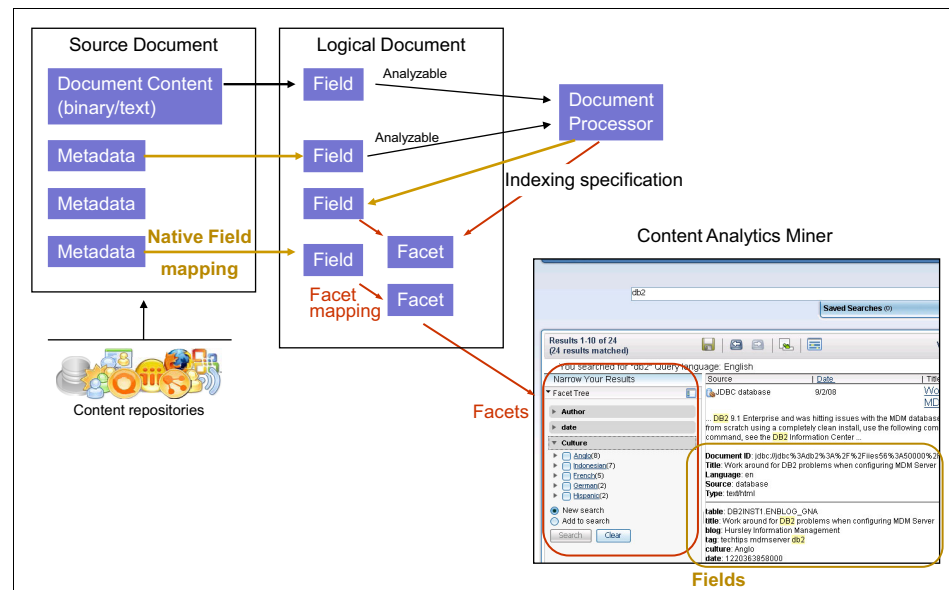


Figure 3-1 Content Analytics data model

A single content analytics collection consists of one or more *documents* crawled from one or more content source repositories illustrated on the left in Figure 3-1.

**Document:** Depending on a specific customer's terminology, a *document* could also be called a record, a verbatim, a row, a call, a log entry, and so on.

A crawled document consists of its main content (often referred to as the *document body* stored in binary or text form) and one or more metadata fields. These fields are often referred to as *native fields* because they originate from the source document.

A *logical document* shown in the middle of Figure 3-1 on page 42 is created from the original document and from any information added by text analytic annotators during the document processing phase. Only one logical document can be created for a given collection. A logical document consists of fields and facets.

Fields (also known as *index fields*) are populated with data obtained either from the original source of the document or from information added by annotators in the Unstructured Information Management Architecture (UIMA) pipeline. Fields identified as analyzable have their content passed to the document processor for text analysis by the UIMA annotators. In turn, the annotators can populate new fields and facets in the logical document with information they derived from the text.

A set of default fields is already defined for your content analytics collection. Default fields are common across all crawler source types, such as the document identifier or document title. Crawlers are aware of these fields and populate them with data from the document automatically. Besides the document identifier, title, and date field, it is not mandatory that you use the default fields. You can ignore them or delete them from your list of defined index fields.

Facets are similar to fields but are used to represent the different dimensions (or views) across all documents in your collection. Similar to fields, facets are populated with data from the original documents or from annotators in the UIMA document processing pipeline. In the content analytics miner, facets are displayed in the Facet Navigation pane on the left side of the content analytics miner window (at right in Figure 3-1 on page 42). However, fields are shown with each document search result in the middle of the content analytics miner window. Default fields are differentiated from any custom fields by a visible horizontal divider line in the search result. Default fields are listed first above the divider line.

**index field:** An index field refers to a field that exists within your collection and can be acted upon. An index field can be used for search, returned in a search result, used to sort a search result, and much more. An index field in a content analytics collection is semantically similar to a column in a database table.

### 3.1.2 Structured and unstructured sources

Content Analytics supports a broad spectrum of content sources, with more than 25 different types of enterprise sources, including everything from plain text files to relational databases.

For a list of detailed information about each specific data source that is supported by Content Analytics, refer to the following web page:

<http://www.ibm.com/support/docview.wss?uid=swg27023680>

The amount of structure in the data from these supported sources varies widely and must be considered when designing your content analytics application.

Highly unstructured data sources, such as text files and web pages, consist mostly of text and have few structured fields or metadata associated with them. In this case, you must rely more heavily on the various text analytics capabilities in Content Analytics to extract meaningful information from the text portion of the content.

Some data sources can have an almost equal amount of structured and unstructured data. Documents crawled from document management systems fall into this category. For example, documents maintained in a Domino database can have a rich set of structured fields and textual fields. The contents of a facet can be populated by the values in your structured fields and the values extracted from your unstructured data using content analytics.

Data sources, such as relational databases, have a high degree of structure. Relational databases support a near infinite set of relationships that can be defined for the data. Relational databases can also contain text in VCHAR, CLOB, or BLOB fields. You normally have these kinds of fields analyzed by the text analytic annotators.

The data model for Content Analytics is that of a simple document where a document consists of a set of fields. Your job is to map more complex relationships (defined in your relational database for example) to the document model used in Content Analytics. This task might require preprocessing of your data outside the Content Analytics product.

### **3.1.3 Multiple data sources**

For a given content analytics collection, you can define one or more crawlers to acquire the documents to be analyzed. If your documents are from a single source, you configure a single crawler for that particular source type. Crawler configuration includes mapping the native fields in the source document to the index fields you defined for your content analytics collection. If you have a single document source and hence one crawler, this mapping step is trivial to perform. Most, if not all, of your index fields are derived from the native fields in the crawled documents.

If your content analytics collection obtains data from multiple sources, you must be aware of the ramifications. A single common logical document is created and

indexed in the content analytics collection for each document retrieved by a crawler. Documents retrieved by different crawlers *are not* merged into one logical document that is then indexed into your content analytics collection. Consequently, it is possible to have a logical document with sparsely populated index fields depending on the diversity between crawled data sources.

This functionality can work to your advantage depending on your data. For example, suppose a regulatory agency wants to monitor the performance of medical devices in the marketplace. The agency developed a form that captures all of the pertinent information when a medical device experiences a failure. The forms can be submitted by three different sources: hospitals, patients, and the manufacturer of the device. In addition, suppose that the three sources use different formats and destinations for submitting the form data. In this situation, you can create and configure three different crawlers, one for each source. Fortunately the data is consistent across each crawler. Each form retrieved by each crawler extracts the same set of fields and maps them to their corresponding index field.

Now consider a case where the data (fields) retrieved by multiple crawlers are different. For example, suppose that you have insurance claim forms that you want to analyze. You define and configure a crawler to retrieve these forms. The majority of the information that you need is in the claim form except for the age of the claimant, which you deemed important in your analysis. However, the age of the claimant is in a relational database. Therefore, you configure a relational database crawler mapping the age column in the relational database to the age index field.

The result is two separate documents in the collection for each claim:

- ▶ One for the insurance claim form itself where most of the index fields are populated with values from the form.
- ▶ One that represents the age data extracted from the relational database. The document contains only one populated field value (age). All of the remaining index fields in this document are empty.

This behavior might not be what you want. When you require a joining of information from multiple data sources, you must perform your own extract, transform, and load (ETL) processing outside of Content Analytics to merge the data into one logical document. After the data is merged, Content Analytics can then use a crawler to crawl these merged documents into a content analytics collection.

### 3.1.4 Date-sensitive data

An important feature of Content Analytics is its ability to identify trends and patterns in your data that occur over time. In order for Content Analytics to perform this task, it must base its calculations on one or more date values that are consistently contained in each logical document of the content analytics collection. Without these date fields, Content Analytics cannot perform any of the date-sensitive calculations, which in turn are used to calculate the time series, deviations, and trends views of the content analytics miner.

At a minimum, you must identify one or more date fields for a given collection in order for the time series, deviations, and trends views to be operable. When dealing with a single data source, the task can be as trivial as identifying which date fields to use. You can also have the date fields in your logical document populated by multiple crawlers of different types. Use care to ensure that the date fields from these sources are semantically the same and represent the same information.

Working with different date fields and different date formats can be delicate: You might have experienced date index fields displaying in a different format than what you expected. Do not panic: This effect can usually be resolved by simple configuration changes.

Remember that the Content Analytics parser can automatically detect different date and time formats from your source data. In addition to these formats, you can configure the parser to recognize custom date formats for the content that you include in a collection. When the parser detects a date value, it converts the value into epoch time (the number of milliseconds since 1 January 1970, 00:00:00 GMT), such as 1235487600000.

Review carefully the “Date fields and custom date formats” section of the Content Analytics documentation, in the chapter “Parse and index administration” found either in the *IBM Content Analytics with Enterprise Search Version 3.0 - Administration Guide* (SC19-3349-00), or in the Content Analytics v3.0 IBM Knowledge Center.

### 3.1.5 Extracting information from textual data

For the textual data in documents, you can use the annotators that are provided by Content Analytics to extract useful information.

You can configure the following key annotators for text analysis:

- ▶ Dictionary Lookup annotator: Match words and synonyms from a dictionary with words in your text. The annotator also associates the keywords with

user-defined facets. For configuration information, see 8.2.2, “Configuring the Dictionary Lookup annotator” on page 282.

- ▶ Pattern Matcher annotator: Identify patterns in your text by using the rules that you defined. The annotator also associates the patterns with user-defined facets. For configuration information, see 8.3, “Configuring the Pattern Matcher annotator” on page 291.

In addition to these annotators, you can categorize your documents by using IBM Content Classification (a separate product). Content Classification is a powerful classifier that you can train by presenting it with sample text examples for each category that you want to recognize. After the Content Classification is trained, you can activate its corresponding annotator in Content Analytics to automatically categorize your documents as they pass through the UIMA document processing pipeline. To use document classification, you must purchase a separate IBM Content Classification license.

IBM Content Classification can also be used to group your documents into self-organizing clusters based on the frequency of words and their collocation to others words used in your documents. This type of text analytic provides insight as to the main topics being discussed in your documents, with each cluster representing a potential main topic. At any time, you can activate document clustering for a content analytics collection by using the administration console. The use of document clustering does not require the purchase of IBM Content Classification.

IBM Content Classification can also be used to support concept-based searching. Concept-based searching can return relevant search results without necessarily containing any of the keywords used in your search expression. For example, the search expression “home damage caused by bug infestations” might return documents about termite damage even though the term “termite” was not used in your search expression. This a powerful way to search. To use concept-based searching, it is assumed that you have already obtained the IBM Content Classification product and have trained a knowledge base using a sample set of clustered results for your collection.

For configuration and usage information, see Chapter 9, “Content analysis with IBM Content Classification and document clustering” on page 305. For other annotators available in Content Analytics, defined below, refer to the “Annotators” section of the Content Analytics documentation, in the chapter “Text analytics” found either in *IBM Content Analytics, Version 3.0 - Text Analysis Integration* (SC19-3350-00) or in the Content Analytics v3.0 IBM Knowledge Center:

- ▶ CAS2JDBC annotator
- ▶ Language Identification annotator
- ▶ Linguistic Analysis annotator
- ▶ Named Entity Recognition annotator

- ▶ UIMA Regular Expression annotator
- ▶ Multimedia annotators

### 3.1.6 The number of collections to use

A single installation of Content Analytics supports the creation and maintenance of multiple content analytics collections. Each content analytics collection consists of its own defined crawlers, content analytics, and indexing options. Each collection operates independently of the other content analytics collections. Multiple content analytics collections are useful when you want to use Content Analytics for different and independent research efforts.

One or more content analytics collections can be organized into a grouping that is assigned an *application ID* by the content analytics administrator. By using application server roles, users can be associated with a particular *application ID*. In this fashion, users can be restricted to which content analytics collections they can access through the content analytics miner.

Even though multiple content analytics collections can be created, keep all required documents for a specific analysis effort in a single content analytics collection. The reason for this approach is because the statistics calculated in the content analytics miner are designed to apply to only one collection. They are not designed to be calculated across multiple content analytics collections.

## 3.2 Guide for building a content analytics collection

This section explains the steps to build a content analytics collection. The content analytics collection is the central component to your research. The collection supports the rapid search and discovery features provided in the content analytics miner.

### 3.2.1 Building a content analytics collection

The Content Analytics administration documentation defines the different steps to create a new content analytics collection. Review the “Create a collection” section of the Content Analytics documentation, in the chapter “Collection administration” found either in the *IBM Content Analytics with Enterprise Search Version 3.0 - Administration Guide* (SC19-3349-00) or in the Content Analytics v3.0 IBM Knowledge Center.



As an overview, the following steps are recommended:

1. Design your index fields for the content analytics collection following the guidelines in 3.1, “Data considerations” on page 42.
2. Create your content analytics collection.
3. Create the index fields for your content analytics collection.
4. Create your facets and map your index fields to the facets.
5. Define your crawlers and map the source native fields to your index fields.
6. Run your crawlers to extract the documents from the different content sources.
7. Build your content analytics collection.
8. Verify your content analytics collection.
9. Repeat steps 2 through 8 to make adjustments to your content analytics collection as explained in 3.2.3, “Planning for iteration” on page 52.

A content analytics collection is compiled from data extracted from one or more content sources in your organization. These sources could include the file system, relational databases, collaboration systems, the web, and document management systems.

### 3.2.2 A walk through the building process

When designing your content analytics collection (step 1), you first determine which native fields in the source repositories to extract and index into your content analytics collection. You also identify the index fields that will be populated with data extracted by the content analytics in the UIMA document processing pipeline. The resulting list of index fields represents the superset of fields that are available to your content analytics miner.

For each index field, you define how the application must use it:

► Returnable

The content of the field is returned with the search results. Returnable is the default setting. There can be times when you might want a field to be searchable but not returned for display such as an employee salary field. Also, many returned fields in the search results (for example, a hundred or more) can impede performance. Choose these fields sparingly.

► Free text search

This type of field can be searched with a keyword portion of a search expression. Normally the body of a document (for example an email) is searched in this manner. You can also specify fields, such as the title field, to

be examined (that is free text searched). If you select the free text index field attribute for a particular field, optionally you can specify whether to use the contents of the field in the makeup of the dynamic summary that is displayed for the search result. The dynamic summary highlights the words that match the keywords that are specified in your query.

► **Fielded search**

This type of field can be explicitly referenced in a search expression. For example, you might want to find all documents where the author field contains “John Doe” expressed as `author:“John Doe”` in the search query entry box. If you select the fielded search attribute, you also can specify whether an exact match is required (that is all the words in the field must match), whether the match is case-sensitive or not, whether to boost ranking of documents that match this field by a specific boost factor (0 - 100), and whether to return spelling suggestions from this field in the search result.

► **Fielded search expansion**

If the index field is either Fielded search or Free text search, you can enable the fielded search expansion option to expand the search query to this specific field. For example, if a user submits a free text query to find documents that contain “IBM” and expansion is enabled for an index field named “company”, the query is automatically expanded to `(IBM OR company:IBM)`. If you also specify a boost factor of 5.0 for the “company” field, the query that gets processed is `(IBM OR company:IBM^5.0)`. If IBM occurs in the content or in the “company” field, those documents are then boosted higher in the search results.

► **Parametric search**

A parametric search is a type of fielded search that enables you to do comparative or evaluative queries. This type of field is either of type numeric or date and allows you to use algebraic expressions such as less than, equal, greater than, in-between, and combinations thereof during searches.

For example, you can search for documents that are of a certain size or that were written after a certain date. To sort results numerically according to a field’s value, you must enable the field for parametric search.

► **Text sortable**

This type of field indicates whether it can be sorted in the search results. By default, your search results are sorted by relevance and date.

To enable users to sort search results alphabetically by the values in a field, select the *Text sortable* check box. If the field contains numeric values, select the *Parametric search* check box to specify that the field values can be used to sort the search results numerically.

- ▶ Analyzable

The content of this type of field can be analyzed by the Content Analytics document processing UIMA pipeline, in addition of the default analyzable *body* field. Content Analytics applies advanced text analytics to extract parts of speech, phrase constituents, named entities, and any other annotators that you have configured for the content of this field.

This additional information, which is stored with the documents in the index, can help improve the relevance of search results and enable statistical analysis to be performed for documents in a content analytics collection.

- ▶ Faceted search

This type of field is displayed in the Facet Navigation pane in the left pane in both the content analytics miner and the enterprise search application. A faceted field typically has a finite set of discrete values that can be enumerated and is *not* normally a free text field.

If you select this attribute, a facet with the same name as the index field is created. If you do not want the facet to have the same name, do not select this attribute. You can create a facet with a different name by configuring the facet tree, and map one or more index fields to the specific facet.

After you design your application fields and facets, you are ready to create the content analytics collection by using the administration console (step 2 on page 49). Be sure to indicate that collection is a content analytics collection rather than an enterprise search collection.

After the collection is created, you configure the index fields (step 3 on page 49). You perform most of the work for defining your index fields in step 1 on page 49. In step 3 on page 49, you manually enter them into the system. If you have a large number of index fields, you might find it more convenient to import the definitions by using a single XML file.

In step 4 on page 49, you define the facet hierarchy to be used in the content analytics miner. The facet names that you assign to the hierarchy are for display purposes only. The index fields that you map to the display facet names are used to populate the tree.

In step 5 on page 49, you define the crawlers for the target sources. Crawlers extract documents from the different content sources and make them available for indexing. For each crawler, you map the native fields in the source documents to the indexed fields in the content analytics collection. The field names do not need to match. It is common to encounter fields from different back-end sources that are semantically the same but have different field names. For example, the *from* field in an email document can be mapped to the *author* field in the index.

However, in an office document, you map the creator field of the document to the same author index field.

**Order of the steps:** The order of these steps can vary slightly depending on your preference and experience. For example, after you create the content analytics collection in step 2 on page 49, you can define your crawlers next for the collection (step 5 on page 49). However, you cannot map your native crawled fields to the index fields of the collection until you define the index fields in step 3 on page 49. The system does not prevent this sequence of steps. At any time, you can go back to previously defined crawlers and add or change the mapping of its native fields to the index fields of the collection.

In step 6 on page 49, after you create the crawlers, you run them to extract the needed documents from your enterprise data source. If this is the first time for running your crawlers, limit the number of documents crawled to a smaller, more manageable set. After you are confident with the configuration and building of your content analytics collection, you can run the crawlers to retrieve the entire corpus of documents.

Content Analytics offers continuous operation. If all components are running (for example, crawlers, indexer, and search run time), as soon as crawlers retrieve documents, they are processed and indexed. After being indexed, the documents are made available to the content analytics miner for analysis. Again, if this is the first time you use the product, step through each component independently. Start and stop them with the completion of each task. Remember the number of documents that has been crawled up to this point.

In step 7 on page 49, build the content analytics collection. With each click of the **Refresh** button on the administration console, you see the progress of the index build and the number of documents currently in the index. Depending on how many documents were crawled and how many content analytics capabilities you have configured, building and completing your content analytics collection can take some time. You can get a rough idea of the remaining time by monitoring how long it takes to process a certain number of documents per minute. Then apply that factor to the remaining documents that will be crawled. You are then informed by an appropriate message when the collection has been completed.

By step 8 on page 49, you are ready to start the search runtime function for collection. Use the content analytics miner to verify your configuration.

### 3.2.3 Planning for iteration

It is a best practice in content analytics solution design to iterate through the steps in the design guide in 3.2.1, “Building a content analytics collection” on

page 48. Start with a small set of sample data to rapidly validate your design assumptions. Depending on the application, this data set can range anywhere from a few hundred documents to tens of thousands of documents. The more documents that you have, the longer it takes to crawl, parse, analyze, and build the content analytics collection. Select a number that is reasonable for your needs.

Based on our experience, we always started with a few hundred documents to verify our field mappings, the content analytics applied, and facets. With fewer documents, we did not get meaningful statistics such as correlation values. With each iteration, we increased the number of documents by a factor of ten to double check that the content analytics are doing what we expect. With a larger data set, we were more likely to encounter occurrences of specific entities and patterns in the text that our content analytics were looking for.

After you are certain that everything is working as expected, you can use Content Analytics on your entire corpus of documents. The length of time to process your entire corpus depends on the hardware being used and the number of documents in your corpus. You can get a rough estimate of the time to complete this task based on your previous iterations through the smaller sets of data.

One best practice is to design iteratively with a *small* reference collection, with minimal number of documents (hundreds to few thousands), and either use the *clone collection* or *export* then *import* functions to create a *large* collection from your current state design in the small collection. By doing so, you can continue the iterative solution design in the smaller collection, and let the larger collection process and index a larger set of documents to validate different results from your content analytics collection (Terms of interest, Time Series, Frequency, Correlation, Deviations, Trends, Connections, Facets Pairs, and so on).

Keep in mind the following considerations when iterating through the building of your final content analytics collection:

- ▶ If you discover an error in the mapping of native fields to index fields (for example, you miss a field), rebuilding the index can fix the field mapping.
- ▶ If you change the index field attributes or change the facet tree, it is not necessary to recrawl all of your data. In this case, you simply redeploy the index and then rebuild the index only.

### 3.3 Programming interfaces

Content Analytics provides several sets of application programming interfaces (APIs) from which to create search, analytics and administration applications, modify crawled documents, filter search results, export documents, set up an

identity management component to enforce document-level security, and perform ad hoc text analysis on documents.

This section provides a high-level overview of the main programming interfaces. For detailed information about the different programming interfaces, refer to the general programming guide, *IBM Content Analytics with Enterprise Search Version 3.0, Programming Guide* (SC19-3348-00). Content Analytics provides the following APIs:

- ▶ REST APIs
- ▶ IBM search and index APIs
- ▶ Plug-in APIs
- ▶ Identity management component APIs
- ▶ Real-time natural language processing (NLP) APIs

Specific application programming interfaces documentation and samples are respectively provided under the following directories:

- ▶ *<installation directory>/docs/api/xxxx* directory
- ▶ *<installation directory>/samples/xxxx* directory

Chapter 14, “Customizing and extending the content analytics miner” on page 535 provides more details about customizing and extending the content analytics miner.

### 3.3.1 REST API

Content Analytics provides a Representation State Transfer (REST)-based API for the development of search, text analytic, and associated administration applications. In particular, you can use the REST API to perform the following tasks:

- ▶ Manage collections
- ▶ Control and monitor components
- ▶ Add documents to a collection
- ▶ Search a collection and federated collections
- ▶ Search and browse facets

The REST API offers the following benefits:

- ▶ A language independent and pure remote call. Any client modules are not required to use the API.
- ▶ Easy to understand. Almost all communications between client and server are in text format, and you can use your web browser to try the API.
- ▶ Standards-based: Runs on top of HTTP.

- ▶ Platform-independent: Any clients that support HTTP can use the API. You can build client applications on various platforms.
- ▶ Security friendly: Able to be used in the presence of firewalls.

The REST API has two categories:

- ▶ Search REST API
- ▶ Admin REST API

The Search REST API is available on machines that have an Content Analytics search server component deployed. It is available on any machine that has a search role. The Admin REST API is available on machines that have the master server role.

You can use both the HTTP GET and HTTP POST methods to call most of the REST APIs. The HTTP POST method is recommended for reasons of greater security.

### More information

For more information about specifically using the REST API, see the general programming guide, *IBM Content Analytics with Enterprise Search Version 3.0, Programming Guide* (SC19-3348-00). Also, look in the following directories:

- ▶ Sample programs in the `<installation directory>\samples\rest` directory
- ▶ Javadoc information in the `<installation directory>\docs\api\rest` directory

## 3.3.2 Search and Index API

Content Analytics provides a Search and Index API (SIAPI). In the Content Analytics implementation of SIAPI, the search server can be accessed remotely. The SIAPI supports the following types of enterprise search and content mining tasks:

- ▶ Searching collections
- ▶ Customizing specific information that is returned in search results
- ▶ Searching and browsing facets
- ▶ Querying multiple enterprise search collections as though they were a single enterprise search collection (also known as *search federation*)
- ▶ Viewing results with clickable URLs and viewing ranking score

- ▶ Searching and retrieving documents from multiple data sources, such as IBM Content Integrator repositories and Lotus Domino databases
- ▶ Performing real-time text analytics on documents without adding the analyzed documents to the index

## SIAPI implementation restrictions

**Warning:** As of Content Analytics v3.0, the SIAPI administration APIs are deprecated and are no longer supported. The SIAPI search APIs are being deprecated and will not be supported in future releases.

We recommend using the REST APIs instead of the SIAPI APIs to create custom applications. Refer to section 3.3.1, “REST API” on page 54 for additional information.

Not all SIAPI classes and methods are supported by Content Analytics. Refer to the documentation for the list of specific unsupported methods. Starting with Content Analytics v3.0, the following packages are deprecated:

- ▶ The `com.ibm.siapi.index` and `com.ibm.siapi.admin` packages, also known as the SIAPI Administration APIs, are deprecated and are no longer supported.
- ▶ The `com.ibm.siapi.search` and `com.ibm.siapi.browse` packages, also known as the SIAPI Search APIs, are still supported, but they are being deprecated in this release.

### More information

For more information about specifically using the SIAPI, see the general programming guide, *IBM Content Analytics with Enterprise Search Version 3.0, Programming Guide (SC19-3348-00)*. Also, look in the following directories on the Content Analytics server:

- ▶ SIAPI sample programs in the `installation directory\samples\siapi` directory
- ▶ The SIAPI Javadoc information in the `installation directory\docs\api\siapi` directory

## 3.3.3 Real time natural language processing API

One of the programming interfaces that is used more and more is the real-time NLP API, which allows users to perform ad hoc text analytics on documents.



The following list provides some of the typical usages:

- ▶ A dictionary developer creates a content analytics collection with dictionaries for testing results, and uses the real-time NLP API to examine how the dictionaries attach facets for various input documents.
- ▶ A workflow system uses real-time NLP to determine how to process documents that are based on the facets that are extracted in real time from a specific document.
- ▶ An alert system constantly processes input documents, such as chat logs, news feeds, or electronic manifests, and sends email to managers immediately if a particular insight is detected in the input document.

Real-time text analysis uses the existing text analytics resources that are defined for a collection, but analyzes documents without adding them to the index. Users can immediately check the analysis results without waiting for the index to be built or updated.

Both SI-API and REST API versions of the real-time NLP API are provided. The NLP REST API accepts both text and binary content; however, the SI-API version only accepts content in text format.

**Restriction:** The SI-API version of the real-time NLP API is being deprecated and will not be supported in future releases. Use the REST API version instead of the SI-API version to create custom applications.

## More information

For more information about specifically using the NLP REST API, see the general programming guide, *IBM Content Analytics with Enterprise Search Version 3.0, Programming Guide* (SC19-3348-00). Also, look in the following directory:

Javadoc information in the *installation directory*\docs\api\rest directory

For more information about specifically using the NLP SI-API, see the general programming guide, *IBM Content Analytics with Enterprise Search Version 3.0, Programming Guide* (SC19-3348-00). Also, look in the following directory:

NLP SI-API sample program in the *installation directory*\samples\siapi\RealtimeNLPExample.java





# Understanding content analysis

This chapter provides details about the process of content analysis and how IBM Watson Content Analytics (Content Analytics) can be used as a tool to help you analyze large amounts of textual data. From this content analysis, you can gain actionable insight from your data. To be successful, you do more than use the product and perform a series of mechanical operations. You must have a deep understanding of what Content Analytics does and how it works. To take full advantage of Content Analytics, you must understand what you can analyze, what you can expect, and how you can interpret and use its output.

This chapter includes the following sections:

- ▶ Basic concepts of content analytics
- ▶ Typical cycle of analysis with Content Analytics
- ▶ Successful use cases

## 4.1 Basic concepts of content analytics

Textual data can be complex and ambiguous. Because of the ambiguities of natural language, textual data often obscures factual information and insight that you can otherwise use and act on to make better business decisions. This type of information is difficult to understand and process by using automated methods. Consequently, businesses are handicapped without considering this large body of information.

Content analytics specifically unleashes the value trapped in your textual data. It is a tool for reporting statistics and to obtain *actionable insights*, which are business insights that lead to actions for better business operations. Content analytics is used to reduce your manual workload of text analysis and to enable a higher level of analysis that provides insights not previously attainable.

For example, by reading customer contact records one by one, you can understand what happened to each customer, but you cannot understand if they are unique cases or common cases or if such cases are increasing or decreasing. Such insights can be acquired only by analyzing the data set as a whole. However, an entire data set will generally contain enormous varieties of information, making it important to focus on only the valuable information.

This is the challenge that content analytics excels at: Allowing you to analyze entire data sets to find patterns and trends, and providing tools to help you find and focus on those particular patterns and trends that are important to you. In our experience, content analytics users reach greater levels of awareness and achievement with their data. By using content analytics, users can take actions based on the insights they obtain from their data and make their business operations more efficient and better managed.

To gain the insights of this textual data, new users must understand the basics of content analysis.

### 4.1.1 Manual versus automated analysis

When manually analyzing textual data, an analyst typically classifies the data according to a predefined set of classifications, tallies the number of documents that conform to each classification, and then reports on their distributions. The problem comes when the amount of data to be analyzed is large. An analyst or a group of analysts can find it increasingly difficult to process and discover anything out of the ordinary that is not predefined.

For example, assume that you have over 1,000,000 survey forms from people indicating their plans for the upcoming weekend. Each survey entry consists of

information such as the person's age, profession, and gender, and description of their weekend plans. The information in this survey can be valuable to businesses because it enables the businesses to predict demands for goods and services from the respondents and to subsequently prepare for them. However, because the weekend activities are described as free-form text, it is impossible to manually read through and analyze each of the 1,000,000 survey forms.

Consequently, a human analyst might randomly select a much more manageable subset of survey forms (for example, 1000 forms or fewer). After reading a portion of these forms, the analyst might then define an arbitrary set of classes for the weekend plans (for example, shopping and traveling). Finally, the analyst might tally each of the survey forms according to the defined classes and produce the following result:

- ▶ 250 shopping
- ▶ 200 visiting friends
- ▶ 150 playing sports
- ▶ 100 traveling
- ▶ 300 others

Figure 4-1 shows the results in a graphical form.

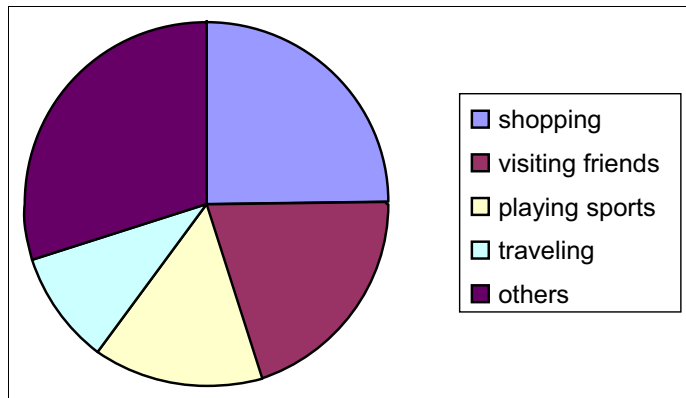


Figure 4-1 Distribution of weekend activities

Manual text analysis that follows this pattern can be a difficult and time-consuming task to achieve. Defining a proper set of classes for the data is not always trivial. For example, the statement “Traveling to Florida to play golf with my friends” might belong to multiple classes, such as traveling and sports. The statement “I’m planning to go hiking unless it rains. Otherwise, I’ll be shopping.” is ambiguous as to whether to classify it as sports or shopping depending on the weather condition.

The composition of the classes can be difficult. What must a classification be and to what level of detail? In our example, for the sports category, we want to further understand the kinds of sports (for example, tennis and hiking) that people are planning for the weekend. Originally, the analyst only classified the entries to the single high-level sports category. Drilling down into each sports activity after the first round of analysis is done requires additional workload to tally more survey forms that correspond to each specific sporting activity.

With Content Analytics, you can now use an automated means to analyze your textual data. In our example, Content Analytics can easily process 1,000,000 survey forms. By defining appropriate facets for various viewpoints, the survey data set might lead to valuable insights for businesses.

For example, the tourist industry might define a travel destination facet that consists of major places for travel and an activity facet that consists of typical activities performed during travel. With these facets, the tourist industry can now analyze the type (such as age, profession, and gender) of people who tend to travel to specific areas of the country for certain kinds of activities. By defining a shopping place facet and a purchase item facet, retailers can analyze the type of people who tend to buy what at where. This information can now be extracted from your textual data.

The distribution of high-level classifications tends to be similar over time and is not useful. The role of Content Analytics is to change the process of generating distribution reports into recommendations for action. The essential goal of the analyst must be changed from document classifier and chart maker to interpreter of the analytical results, identification of actionable insights, and planning for action.

For more information about defining facets and performing analysis, see Chapter 5, “Content analytics miner: Basic features” on page 85.

## **Challenges in text analytics**

Natural language contains many ambiguities that are difficult for automated methods to resolve. For example, the phrase “Time flies like an arrow” can be interpreted in several ways. The word *like* might be a verb meaning “to be fond of” or an adjective meaning “similar to.” Likewise, the word *flies* might be interpreted as the verb “to fly” or a plural noun as in “multiple fire flies.” Polysemous words (words with different meanings) are difficult for automation. For example, the name *Arizona* can be a place or the name of a person. The word *saw* in the sentence “They saw a girl with a telescope” can be interpreted as the past tense of the word “see” or an object that you use to cut something. Furthermore, “with a telescope” might modify either “saw” or “a girl.”

In addition to words being ambiguous, the concepts conveyed by the text can also be ambiguous. This ambiguity is demonstrated in our example while trying to classify the surveys on weekend plans. It is difficult to classify the hiking activity as either a sports activity or a travel activity. The result is often subjective and can differ from person to person. Even simple errors, such as misspellings in the original text, can cause problems in its proper interpretation.

Because of these challenges, the analytical results might not always match your expectations. These difficulties can affect the overall distribution of items with associated frequencies being suspect. Even the order might not be reliable because modifications made to the dictionaries can inadvertently change the order. For example, if we define “CD” as a synonym of “compact disc,” the number of records that might seem to be related to the concept of “compact disc” might be miscounted if some of the instances of “CD” were intended to refer to “certificate of deposit” or “check digit.”

Having accurate frequency counts as much as possible is preferred. Content Analytics provides several solutions for improving the accuracy in its text analytics. It uses tools such as the dictionary and pattern matching rules components (Dictionary Lookup and Pattern Matcher annotators). However, improving the overall accuracy in your text analytics can be an elusive task. It is much more productive instead to spend more time in the analysis phase and action taking phase of your work. With Content Analytics, you can examine *all* of your data interactively and the large amount of data in itself leads to the power of the analysis and value of the result.

Using human intuition and understanding during manual analysis is the best way to improve accuracy but quickly becomes impractical as the amount of textual data increases. As the amount of data increases, human involvement needs to shift from the manual reading and understanding of text to the automated analysis of the corpus as a whole. You cannot add more people and expect the overall accuracy to improve because different analysts produce different results. Even the same analyst can produce different results over time. However, Content Analytics can display distributions from various viewpoints over the whole of the data. By using Content Analytics, the criterion remains the same for the whole corpus of documents.

### 4.1.2 Frequency versus deviation

The terms *frequency* and *deviation*, as defined in 1.3, “Important concepts and terminology” on page 12, are important to understand in the context of content analysis.

*Frequency* is the number of documents that contain keywords identified by the text analytics. Frequency might not be reliable because of the difficulty

encountered by the text analytics as mentioned in “Challenges in text analytics” on page 62. Even if you can reach 100% perfection in text analytics, the numbers produced might still not reflect reality.

For example, suppose that you want to analyze customer contact records that involve a specific product failure. In this case, you want to know the total number of units sold that are experiencing the failure or the total number of people who have encountered the product failure. It is not practical to expect that all of the customers might report the failure. Only some of them might call the customer contact center. If the failure is serious, many people might report the problem. If the failure is not serious, only a few people might call in. In the customer contact center, the agents might inadvertently leave out critical information when recording the problem as described by the customers.

Because of all of these factors, even after you make a significant effort to improve the accuracy of your text analytics, you cannot determine the true number of customers who experienced the problem nor the number of products that caused the problem. Figure 4-2 illustrates this point.

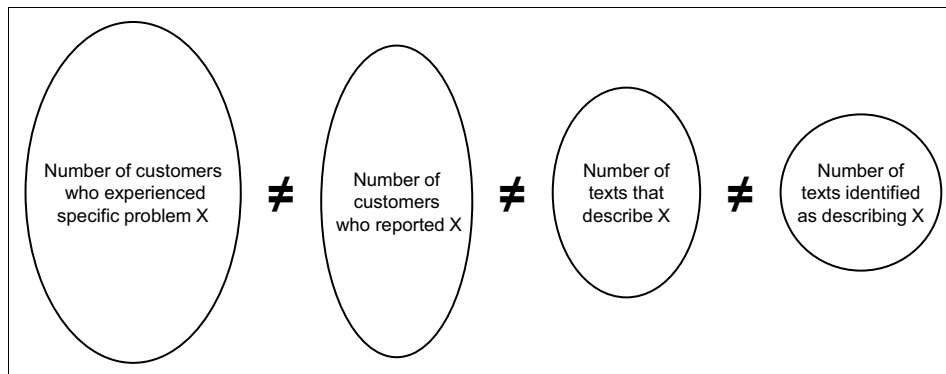


Figure 4-2 Frequency of textual items not matching reality

Even though the frequency counts might not be as reliable as you want, their deviations often lead to valuable insights. For example, 10% of the calls this month on product A were associated with problem X, and last month only 3% of the calls on product A were associated with problem X. In this case, the change or *deviation* is an indicator of a potential problem that is worth further investigation. That is assuming that no surge in selling product A in recent months nor more demand in using product A in recent months has occurred. Making the example more concrete, let us use the paint peeling complaints on toy series. If complaints on paint peeling for toy X series have been increasing from June 2013, the company should investigate to see why there is a change in the complaints frequencies.



**Changes in frequency:** Changes in the frequency often lead to useful results rather than the frequency alone.

Likewise, consider an example where 10% of the calls this month on product A are associated with problem X, where only 3% of calls this month on product B are associated with problem X. In this case, it is better to take action and discover why product A is having a higher incidence rate than product B (assuming that functionality and quality of product A is similar to product B). Making the example more concrete, let us again use the paint peeling complaints on toy series. If for example, paint peeling makes up 20% of the complaints on toy X series whereas less than 5% of the similar complaints are on other product series, the company needs to check out immediately why toy X series has the high deviation from the norm.

**Deviation:** The deviations often lead to useful results rather than the frequencies alone.

These examples illustrate how the use of Content Analytics can you help obtain actionable insights.

Often, legitimate reasons can explain the changes and deviations. Changes in frequencies and deviations are typically based on a real-life phenomenon related to your products or customers. This situation often provides a great opportunity to improve the business by dealing with product failures early or coping with customer behaviors before customers become disenchanted and leave. Although frequency numbers provide some insight, it is more important to focus on the deviations and changes in those numbers to gain greater actionable insight.

In the example of the weekend activities survey, say that you identify about 10% of the survey forms that mention sports as an activity. Consider that the text contains a description of a sport, such as tennis. It might not indicate that the respondent actually plans to play tennis over the weekend but rather plans to watch a tennis game or buy tennis shoes over the weekend. The Content Analytics-based analysis can capture non-sports activities as sports activities if Content Analytics is configured naively. The real power of Content Analytics is to analyze the entire corpus of data and treat the mistakes as *noise* when you focus on changes and deviations.

With Content Analytics and its broader analysis of all your data, you can gain insight and discover new relationships in your data. For example, you might discover that teenagers are more active in sports than older people. Within sports activities, golf might be strongly correlated with people in Florida, and hockey might be strongly correlated with people in Minnesota. Although some of

these correlations might seem obvious, the nonobvious correlations are important to watch for and are the ones that Content Analytics highlights.

### 4.1.3 Precision versus recall

When dealing with a large amount of textual data, you must consider *precision* (accuracy) and *recall* (coverage) in the analytical results. Precision is the ratio of the correctly returned results as compared to the total returned results from Content Analytics. You can think of the precision of Content Analytics results as the number of true positives. Recall refers to the ratio of the correctly returned results as compared to the total number of correct results in the data set.

For example, from the 1,000,000 survey forms about people's weekend activities, you might look for data that describes dining at a French restaurant. One thousand documents are returned from Content Analytics that contain the words "French restaurant" and "eat." However, such results usually contain documents that do not describe dining at a French restaurant, but contain such phrases as "We will make our anniversary plan to eat at a French restaurant," "We will eat at a hamburger shop next to the French restaurant," and so on. If the total number of documents that correctly describe dining at a French restaurant is 700 out of the 1,000 returned documents, the precision (of the results) is 70% ( $700/1000$ ). That is, 70% of the returned results from Content Analytics are correct.

The approach of getting results with the query "French restaurant" and "eat" might not capture all the data in which people describe dining at a French restaurant. The activity of dining at a French restaurant might be described as "we are planning to have dinner at a French restaurant," and the name of a specific French restaurant might be used instead of the literal word "French restaurant."

Thus, in the overall survey, data might exist that describes the activity of dining at a French restaurant without using either the words "eat" or "French restaurant." The correct number of documents that describe the activity of dining at a French restaurant probably exists outside the 1,000 returned results that you received from the system. Consider that the total number of correct results (from the entire data set) is 2000, and you received 700 (correctly returned documents from the 1000 returned results) that describe the activity of dining at a French restaurant. The recall (of the result) is 35% ( $700/2000$ ). That is, you captured 35% of all the correct results in the entire data set.

Precision and recall are often incompatible. If you aim for a higher recall because you do not want to miss any relevant documents, you might be faced with too many irrelevant documents (noise) because of lower precision. If you aim for higher precision because you do not want to be faced with irrelevant documents, you can miss relevant documents because of a lower recall.

In general, aiming for higher recall is often more difficult and time consuming than aiming for higher precision. To achieve higher precision, the basic process entails adding constraints in your query to eliminate the noise. To aim for a higher recall, you must craft increasingly complex query expressions to search and capture all relevant documents.

To analyze trends and characteristics of the data with Content Analytics, focus on high precision data. When the data set is large enough, you can obtain enough samples for meaningful statistical analysis even when the recall is low. On the contrary, if the precision is low, the data might have too much noise, and the trends and characteristics identified with such noisy data might not be reliable.

Special applications, such as those that are aimed to identify critical problems related to safety or legal compliance, might require a higher recall. For such applications, Content Analytics provides powerful functionality to gain clues for achieving higher recall.

To achieve higher recall value, you must first capture high precision data by adding constraints for patterns and eliminate noise. After you achieve a target precision value, look for relevant expressions in the data. Then you must use the relevant expressions that are identified from the second step to capture more correct data in the entire data set.

For example, to capture data with the activity of dining at a French restaurant, you might be able to achieve a higher precision value with the pattern of “eat,” without “plan” or negation in its front. It might immediately be followed by “at” and “French restaurant,” except for the determiners between “at” and “French.” If the precision is reasonable, you can use the Facets view to identify expressions that are relevant to the activity of dining at a French restaurant such as wine, cheese, sommelier, baguette bread, and foie gras. By analyzing expressions relevant to such expressions with the Facets view of Content Analytics, you can identify expressions to capture more recall of your data. In this case, you can capture the activity of dining at a French restaurant and include alternative expressions for French restaurant such as “place for French cuisine” and names of French restaurants.

## 4.2 Typical cycle of analysis with Content Analytics

In 3.2, “Guide for building a content analytics collection” on page 48, you go through the administrative steps required to build a content analytics collection. In this section, you revisit those steps in the context of the overall content analysis process described in this chapter. The result is content analysis of your data that meets your expectations and allows you to gain actionable insight to make better business decisions.

Using Content Analytics consists of iterating through the following steps:

1. Set the objectives for the analysis.
2. Gather the data.
3. Analyze the data.
4. Take actions based on the analysis.
5. Validate the effect.

The data analysis step (step 3) consists of iterations of multiple steps as follows:

1. Apply text analytics and generate the index.
2. Perform analysis using the content analytics miner.
3. Generate or modify dictionaries, patterns, or both.
4. Reapply the text analytics and regenerate the index.
5. Repeat data analysis by using the content analytics miner while verifying changes in each facet.
6. Regenerate and modify dictionaries and patterns as necessary.

### **4.2.1 Setting the objectives of the analysis**

Your content analysis objectives depend on what you want to do and what the data can reveal. For example, if you want to decrease product failures, early identification of product failures is a reasonable objective. However, if none of the data describes product failures, it is not feasible to set early identification of product failures as the objective.

You cannot always determine if your objectives are reasonable until you perform the actual content analysis. However, it is important to consider several objectives and list them up front as potential objectives. Even though some objectives are not feasible because of a lack of measurable data, you can improve the data for such objectives in the next cycle.

### **4.2.2 Gathering data**

Data is an essential part of your analysis. For example, if you want to understand how your customer perceives your product or services (positively, negatively, or neutral), you need to gather data that contains opinions expressed by your customers.

As you go through each iteration of your analysis, think about how to improve your data for better results. Sometimes this task requires you to gather or integrate additional data, and sometimes it means you must cleanse the data for input.

For example, you might analyze the sentiment of customers recorded in a call center. In this case, data improvement can consist of integrating email messages from customers for a broader coverage of data. You can also ask the call center agents to get additional information from customers such as the reason they chose the product. This process might also involve modifying crawlers or data input systems.

### 4.2.3 Analyzing data

After you set your objectives and gather the data, you can start analyzing the data. The content analytics miner is your primary tool for content analysis. The process of analysis is an iterative one. With each cycle, you refine your dictionary, matching rules, and input data so that you can discover new insights, perform necessary actions, and achieve the objectives you set.

#### **Analysis with the content analytics miner**

At the beginning of the content analysis cycle, Content Analytics runs text analytics with a set of default dictionaries and patterns that capture grammatical information such as nouns, verbs, adjectives, adverbs, and other parts of speech. This information can be useful for the initial analysis of your data. A review of the Part of Speech facets gives you a synopsis as to what subjects are involved and an overview of the major concepts in the data set.

You must also browse through other facets in the Facets view to understand the information that is available for analysis. Facets populated from structured data are important at this phase of analysis because the data is generally unambiguous and is much more informative compared to your textual information at this time. Always explore other views for facets that might be interesting. Use your curiosity and imagination.

#### **Content analytics miner views summary**

You can browse through different views of the content analytics miner to discover insight. See Chapter 6, “Content analytics miner: Views” on page 159, for a detailed description of the views. For your convenience, a brief summary of each view is provided here:

- |                       |   |
|-----------------------|---|
| <b>Documents view</b> | Shows a list of documents that match your query. The view shows the actual content of individual documents and their metadata. By reading the original text, you can quickly verify that the text is supporting the reported results. |
| <b>Facets view</b>    | Shows a list of keywords for a selected facet. Each keyword has a corresponding frequency count and correlation value. This view is useful for seeing the keywords that make up a given facet in your data.                           |

<b>Time Series view</b>	Shows the frequency change over time. This view is used to analyze frequency and to select a range of documents for analysis for a given time period.
<b>Trends view</b>	Shows sharp and unexpected increases in frequency over time. Usually a sharp increase warrants further investigation. If the highlighted trend is associated with an undesirable trait, for example product failures, you must investigate how to reverse the trend.
<b>Deviations view</b>	Shows deviation of keywords for a given time period. This view is focused on how much the frequency of a facet deviates from the expected average for a given time period (not from its past history as in the Trends view). You use this view to observe cyclic patterns in your data by selecting a cyclic item in “Time scale” such as “Days of the month” or “Days of the week” showing patterns that occur on a monthly or weekly basis.
<b>Facet Pairs view</b>	Shows the correlation of keywords from two selected facets in either one of Table, Bird’s eye, or Grid view. This view enables you to find correlations between keywords in the selected facets. Usually a high correlation warrants further investigation.
<b>Connections view</b>	Shows the correlation of keywords from two selected facets in a graphical way. This view enables you to visually see the connections between selected facets.
<b>Sentiment view</b>	Shows the positive, negative, and mixed sentiment of text. This view enables you to quickly understand how types of positive and negative correspond to facet values.
<b>Dashboard view</b>	Shows a configured dashboard layout with one or more graphs and tables in a single view. With this view, you can see multiple views (that are of your interest) at the same time.

Figure 4-3 shows a collection of all the content analytics miner views.

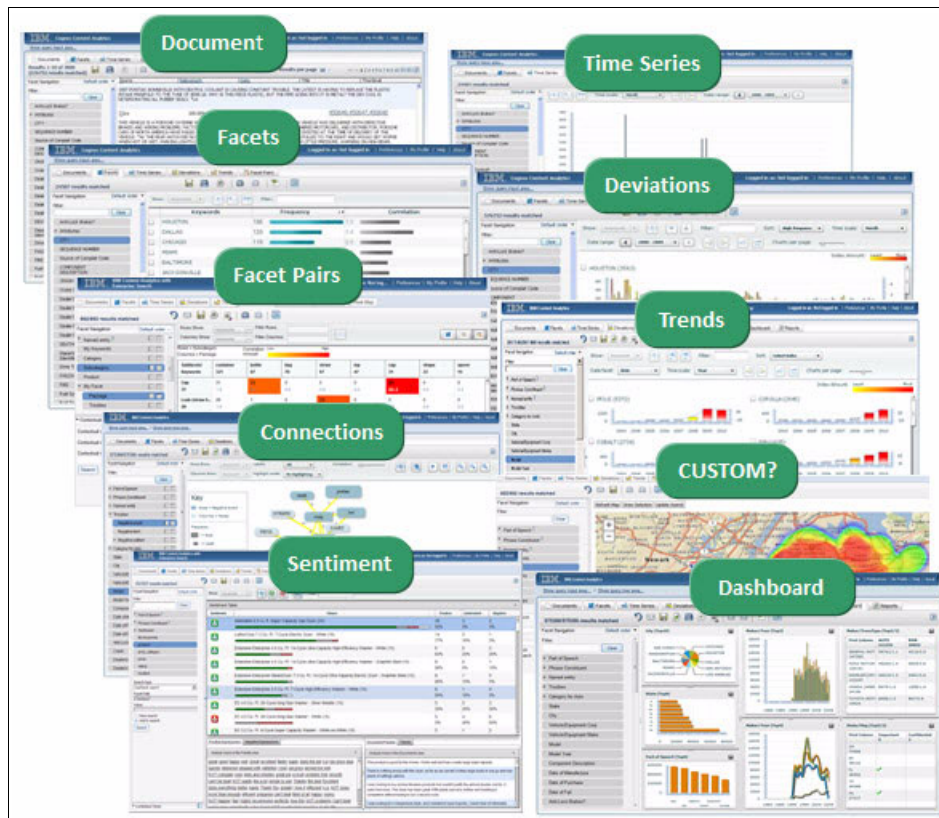


Figure 4-3 Content analytics miner views

Each view in the content analytics miner provides its own unique kind of insight from your content analysis. Analysis of distributions based on day of week and month of year in the *Deviations* view often leads to interesting insights. You might be able to find unexpected patterns of weekend activities and seasonal differences. With the *Trends* view, you can also obtain actionable insights. If something undesirable is increasing, you might need to reverse the trend. The *Facet Pairs* view unveils remarkable associations if you select the appropriate pairs of facets. Repeated selections of different facets to compare is crucial to your success. Content Analytics is designed to be interactive and scalable for repeated iterations through your data.

To understand the content of the data, it is essential to read the original text with the *Documents* view. Some people misunderstand *text mining* as something that allows you to avoid reading the actual documents. Even though text mining allows you to avoid reading the entire amount of textual data, it still requires you

to read selected portions of text for better understanding. After you identify interesting keywords or patterns, make sure to focus on the data with the specific keyword or pattern to verify the fact. What is thought to be an interesting pattern might be caused by noise in the data, errors in the text analytics, or duplication of data. Content Analytics is designed to make it easy for you to verify your results with highlighted keywords in your original text. You can usually get to the original text with a couple of clicks.

By using the content analytics miner during your content analysis, you might identify and select new objectives. It is important to be flexible so that you can set some initial objectives suitable to your data.

### **Generation and modification of dictionaries and patterns**

After you set the objectives of your content analysis, the next step is to tune your text analytics to capture the right information that matches your objectives. You can usually extract most of the information you need by using the Dictionary Lookup and Pattern Matcher annotators.

A dictionary consists of a list of words and phrases supplied by you. In the dictionary, you create facets that describe different aspects of your data that you want to investigate and analyze. For example, facets can be colors and shapes. You associate each facet with a list of words (also called *keywords*) that you create. For example, you can create the keywords *yellow*, *blue*, and *red* and associate them with the Color facets. You can also create the keywords *square*, *circle*, *triangle*, and *rectangle* as keywords and associate them with the Shape facet. When documents are processed in Content Analytics, the text in each document is broken into individual words (or phrases) and then checked against the dictionaries that you built. If a match occurs, that document is associated with the facet, and the frequency count of the facet is incremented by one.

Refining your dictionary with synonyms is one way to fine-tune your text analytics. For example, by registering multiple terms for “International Business Machines” as synonyms of each other (in this example, IBM) you can treat them as a single token. With the pattern matcher, you can identify relationships with parts of speech patterns. For example, you can add a pattern to extract a noun immediately followed by the verb “love” for extracting the relationship of something being loved.

The essential purpose of the dictionary and pattern matching definition is to build different viewpoints for your analysis that will meet your objectives. It consists of defining the appropriate facets and their corresponding expressions.

For example, if your objective is to identify early warnings of product failures in customer complaint records, you need to focus on defining facets relevant to



product failures. Definitions of facets that are relevant to product failures optimize your analysis. Otherwise, the trends or patterns you want to analyze will be hidden in various irrelevant facets.

You must define facets in such a way that their expressions are semantically or grammatically correct. For example, when analyzing computer product failures, a logical facet to define is one that might identify specific components that failed in the product. You can label this facet the Failed Component facet. The values of this facet might consist of nouns such as “keyboard,” “mouse,” “display,” and “battery.” You can define another facet that identifies the cause or type of failures. You can label this facet the Failure Type facet. The facet can consist of verb patterns with diverse parts of speech, such as “crack” and “break,” and be used as a noun or a verb.

In general, start with the dictionaries first, rather than the patterns, especially when the facet consists of simple nouns. It is much easier to define entries in the dictionary than to define grammatical patterns. However, using grammatical patterns is much more flexible in extracting ambiguous terms. Most often you need to use the pattern matcher to handle the conjugation of verbs.

The best resource for words and synonyms to be defined in your dictionary is your textual data. The list of nouns in the Facets view (with the Noun facet underneath the Part of Speech facet) is generally the best candidate list. The list of compound nouns (in the Noun Sequence facet under the Noun Phrase facet, which is under the Phrase Constituent facet) also provides good candidates.

These facets represent how words, which are potentially important to your analysis, are expressed in your textual data. The official names of products are rarely expressed by default because they are not part of a normal set of generic nouns. In addition, abbreviations or short hand might be used in the textual data. Therefore, it is better to use the words as they occur in the text. The top values in these Parts of Speech facets are the most frequently occurring in the textual data. Starting with these words is efficient and effective because words with lower frequencies do not contribute as much to your deviation analysis. For creating your dictionaries, you can list a large number (such as hundreds of thousands) of nouns or compound nouns by setting the “Number of results per page:” through the “Preference” setting and output the results to a CSV file as a dictionary resource file to work with.

After you identify initial expressions for a specific facet, it is often useful to apply correlation analysis for extracting more expressions for the facet. For example, in the PC help center example, we identify the word “screen” to be associated with the Component facet. One verb that highly correlated with the word “screen” in the data was the word “turn.” Among the nouns that highly correlated with the word “turn,” we found that the words “monitor” and “display” can be added as

synonyms of the word “screen.” Also the words “power” and “battery” can be added to the Component facet.

For detailed procedures on generating and modifying the dictionary and patterns, see 8.2.2, “Configuring the Dictionary Lookup annotator” on page 282, and 8.3, “Configuring the Pattern Matcher annotator” on page 291.

The amount of effort it takes to develop your dictionary and patterns depends on your data and your objectives. Some users spend significant amounts of effort developing their dictionary and patterns and end up with a comprehensive and robust lexicon. Also, many users have generated an initial dictionary within an hour or so of work and then modify it as they encounter more terms during their analysis. In either case, the use of dictionaries and patterns is essential for analysis and often has a significant impact on your analysis. It is important that you take the time to create an initial dictionary in support of your objectives for analysis.

### **Repeating analysis with the content analytics miner**

After you create or modify your dictionary and patterns, you must apply the text analytics again and regenerate the collection. After you reindex, based on the latest dictionary and patterns, you are ready to analyze the results with the content analytics miner.

First, you must verify the changes in each facet. Some words and phrases might not have matched as you intended. Mistakes, such as misspellings and format errors, can cause inaccurate statistics for the facet. Expressions in the data might not always represent the concept you expected. The facets can represent a different concept altogether, and the concept you expected might be expressed differently. Thus, it is important to validate your dictionary and patterns after you generate or modify them.

In addition to the Facets view, use the Deviations, Trends, and Facet Pairs views while verifying the results of your dictionary and patterns. Even if the dictionary and patterns are in their early stages of development, analysis with the various views often inspires you to try better approaches for modifying the dictionary and patterns.

## **4.2.4 Taking action based on the analysis**

The goal of your analysis is to take positive actions based on the insights acquired from your data. Because a large amount of textual data usually contains complex relationships full of various insights, it is ideal for action takers to be able to use Content Analytics during their decision-making process. This way, they can consider various aspects of potential actions for better decisions.

The use of Content Analytics often leads to significant results when the analysts themselves can control the actions. Based on our experience, some companies have recognized the value of this new insight and have made the analyst team position a career path for executives.

#### **4.2.5 Validating the effect**

Evaluating the results of your actions is important for determining the validity and quality of your analysis and for planning of the next cycle of analysis.

Because the analysis of your data provides clues of insights unveiled from your data, you usually need to verify the discovered insights in the real world. For example, you might be analyzing complaint data about cars. By using Content Analytics, you can identify a rapid increase in the number of complaints that apply to a specific car model with a specific problem. With Content Analytics, you can also identify different facets that are strongly correlated to this problem, and such facets can indicate certain solutions and actions to take.

The increase in complaints might be caused by changes in the data input operation or a rumor about a potential problem (of which people might call in simply to inquire about the potential rumored problem). Or it might be caused by a problem that exists in the database (for example, causing duplicates of the data). In any case, there must be a reason for the changes and deviations in the data, and it is usually worth further investigation.

Validation of the effect is important because it leads to improved analysis in the next cycle and setting of new objectives for the new cycle.

### **4.3 Successful use cases**

Content Analytics provides infinite possibilities for your company by enabling you to take advantage of your data that is accessible to you. You inspire the outcome, which can be great insight or actions that enhance and improve your business operations, products, or services.

This section introduces successful use-case scenarios in which data is analyzed with Content Analytics. The intent is that these scenarios might inspire you to learn and use Content Analytics in innovative ways to enhance and improve your business. Each use-case scenario is based on real-life experience.

### 4.3.1 Voice of the customer

Analysis of the voice of the customer (VOC) has been a major application of Content Analytics. Analysis of VOC is critical for a business because it provides crucial information about customers and products. Regardless of how hard you test your products before shipment, unexpected use of, or a defect in, your products is unavoidable.

For example, a customer called into a PC help center and claimed that the cup holder on the PC was broken. Obviously, PCs do not have cup holders. After further conversation, the call center agent determined that the customer was actually using the CD tray as a cup holder.

The same VOC records can be used for other purposes. For example, one of the customer contact records in a PC help center contained the sentence, “CX'S DOG ATE HER POWER SUPPLY.” The VOC record further indicated that the agent looked up the part number for the specific power adapter and transferred the customer to the parts division. The development division can use this information to change the design or material of the power adapter because it might lead to a safety issue. Also, they might need to analyze similar cases with other animals and even young children. The PC help center can use this information to coach the agent to be more sensitive and comment on the condition of the dog in addition to helping to order a new power adapter. Such an attitude can impress customers and improve their satisfaction. The Sales division noted that this customer owns a dog, and that this person might be a potential customer for dog-related products.

The analysis of VOC can lead to cost reduction, improvement of customer satisfaction, and an increase in the hit ratio of target marketing.

#### **Early identification of product failures**

Early warning of product failures is one of the most promising applications of VOC analysis with Content Analytics. We have observed significant results across industries including manufacturing, catalog retailers, and service providers.

The PC help center estimated that, in the US alone, the savings from one of their early identifications of a specific product failure using Content Analytics was worth several million dollars. The PC help center received, on average, more than 10,000 calls per week from customers. The call center agents typed in a brief overview of each customer contact that recorded what each customer talked about and how the agent answered. Each call center record also contained structured information such as the machine type and a time stamp. In addition, the call center agents selected predefined classification codes to define the problem type, component type, and action type. This approach tends to fall

short because there might be inconsistency in how agents assign codes, and the codes themselves may not be granular and informative enough for effective business analytics.

Before the introduction of Content Analytics, analysts in the PC help center manually analyzed approximately 300 call center records for a weekly report. With less than 3% of the call center records being analyzed, it was unlikely for the report to have a significant impact for the business. Also the task was expensive and required much effort to read through 300 records to prepare the report.

With Content Analytics, the PC help center used the entire data set, not just the 10,000 records for each week, but the millions of records recorded for the past couple of years. The PC help center also compared the data of the current week with the data from the past several weeks and compared the data of the same week with the data from the past couple of years.

The PC help center defined facets such as Product Name, Hardware, Software, Subcomponent, and Problem. A keywords distribution for product X that is significantly different from the keywords distribution in the same facet for comparable products can indicate a potential problem with product X. You can easily capture such differences (frequency and correlation) in the Facet Pairs view of Content Analytics. Consider the example where 5% of the records for a brand new computer contain the words “LCD pane1” in the Hardware facet. In this example, only 1% of records for comparable computers contain the same words “LCD pane1” in the Hardware facet. In this case, the Facet Pairs view in combination with the Hardware facet and the Product Name facet helps to identify the strong correlation of the new computer to the words “LCD pane1”.

Analysis using the Trend view is also effective for early warning of product failures because such failures generally result in a rapid increase of customer calls. To do such analysis, you monitor the Trends view by setting the sort criteria to the Latest Index. This sorting shows the products with the highest frequencies first in descending order. With this view option, you look at the top of the list in the view. The appropriate facet to be selected for this analysis is the Product Name facet to identify the product name with the most increasing numbers of calls. It is also useful for analyzing the subcomponents of a product or other attributes of a specific product after identifying a specific product for further investigation.

For example, Figure 4-4 shows that the number of records for product X increased the most for the latest month, April 2009, as compared to the other products.

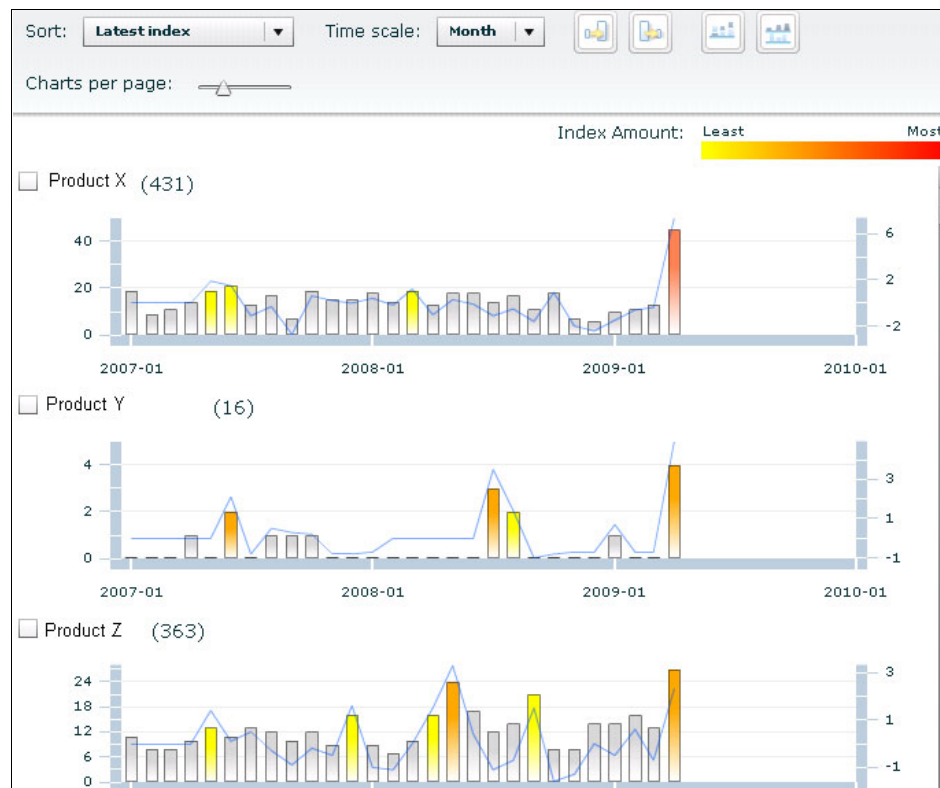


Figure 4-4 Trends view: Identifying the highest frequency product in the latest month

By focusing on the data of product X for April 2009 and identifying significant words found in the data (Figure 4-5), you can conclude that product X might have product failures due to a frame rusting problem.

Keywords	Frequency	Correlation
<input type="checkbox"/> rust	38	46.8
<input type="checkbox"/> frame	38	43.3

Figure 4-5 Facets view: Identifying problem verbs with a high correlation

Early warning of product failures is effective and rewarding because it generally leads to actions that can be taken. Also, it is often easy to evaluate the value of

your early warning detection based on the expected number of product units sold and the cost of fixing the defect in each product with the failure.

### **Timely update of FAQs based on voice of the customer**

Content Analytics was also used at a PC help center in Japan for updating frequently asked questions (FAQ) based on customer contact records. The use of Content Analytics by the PC help center led them to achieve number one status in the problem solving ratio of web support among computer companies operating in Japan in 2003. Their web support also became number one in personal computer support ranking in 2004 according to a premiere PC magazine in Japan.

VOC is invariably the best resource for FAQ because it enables you to count the frequency of questions asked. However, relying on the frequency of questions asked is not always a reliable measure as discussed in “Challenges in text analytics” on page 62. In this scenario, the PC help center staff took a smart approach by focusing on the correlation values.

The staff identified high correlations for a specific type of product and its operation in comparison to similar types of products and their operations (using the Facets view and Facet Pairs view). For example, product X had a high correlation with the printer setting because the default setting of product X was not appropriate for major users. Immediately, they added or modified entries of the FAQ database. In this way, they enriched the FAQ database with specific questions for specific products and problems, instead of simply general questions and answers that were not necessarily helping many users. Because the solutions were targeted to specific questions, this activity significantly improved the problem-solving effectiveness of their web support site, and improved overall customer satisfaction.

As a result, access to web support increased while calls to the PC help center remained the same even though the number of products shipped were increasing. It allowed the staff to reduce the number of PC help center agents, which resulted in a cost reduction by several millions of dollars, with an increase in overall customer satisfaction for the PC.

The reason for the success of this operation was that the PC help center staff who made the analysis with Content Analytics took immediate action and updated the FAQ database.

### **Customer profiling for target marketing**

Customer contact records usually contain valuable information about customers. Such information can be a valuable resource for product planning and marketing. For example, if dogs and cats are frequently mentioned in customer contact records, many customers might be interested in pet-related products.

One of the successful use cases of Content Analytics is to extract customer information from customer contact records for target marketing. Many companies keep track of the people and organizations who bought their products or who have expressed a strong interest in their products. The information is maintained in a relational database for customer relationship management and target marketing.

It is still uncommon to keep track of information about competitors' products even though such information might be important to know. Many customer contact records in the sales department indicate the reason why they were unable to sell their products. For example, people or organizations they contacted might have bought competitive products or had different requirements. Such information is often described in customer contact records to justify their purchase decisions. Although this information can be regarded as useless if the sale failed, it can be valuable later as a customer conversion opportunity or for another division that sells different products or services. However, because there is no plan to use such information, agents usually describe them briefly in free-form textual data. This brief description makes it difficult for anyone to take advantage of such information.

By using Content Analytics, you can extract information about who is using what product and who is considering which solution. You can define product names and solution names in a dictionary or define patterns. For example, you can define a pattern that extracts nouns followed by the verb use.

A financial company in Japan took this approach to generate a target marketing list for various products. To sell government bonds, they looked for customers who mentioned (in their customer contact records) a preference for low risk. In order to sell foreign-currency-based products, the company looked for customers who mentioned international activities such as a business trip abroad. As a result, the company achieved a better hit ratio compared to their traditional approach of nearly random sampling. Selling agents were positive to this approach because most of the target customers showed interest in the product being sold, and the agents felt more comfortable when talking with these customers.

## **Human resource development**

Customer contact records are also a good resource for improving agent skills. A customer contact center in Japan had a problem of maintaining highly skilled agents because the retention rate of the call center was low. Changes in senior agents affected the overall skill levels of the center and customer satisfaction. Because it is not easy to hire new agents with a strong information technology background, it was important to educate new agents effectively. After months of education that was based on FAQs, the new agents started taking calls and were educated through on-the-job training.



When a customer calls in and the first contact agent has trouble answering the customer's question, the call center transfers the call to a second-level senior agent. Because the number of calls and the ratio of new agents were increasing, the number of transfers to the second-level agents was also rising too much.

The call center used Content Analytics to analyze customer contact records specifically for the following information:

- ▶ Calls transferred to second-level agents
- ▶ Calls taking a long time
- ▶ Calls based on product failures

The analysis of calls transferred to second-level agents allowed the call center to identify skill areas that required education. Through analysis of the calls that take a long time, the call center identified specific behaviors of each agent that led to long calls such as full verbal guidance of step-by-step operations instead of guidance to the web FAQ page.

Based on the insights gained through the Content Analytics analysis, the call center developed a more focused education course to improve the skill level of their agents. As a result, the skill level of their agents improved significantly in a short period. The number of calls transferred to second-level agents dropped to less than 20% from the number before the change in education. In addition, the call center forwarded their insights through the analysis of calls based on product failures to the product development team. This activity led to a remarkable reduction in the number of product failure calls.

People in this call center hold monthly meetings to report and share the new analytic approach used in Content Analytics. They also share any insights obtained through the new analytics. Thus, this call center has been running many cycles of analysis with Content Analytics over time, which has led to continuous improvement of this call center.

### **Best practices of sales agents**

By analyzing customer contact records at an outbound call center for telemarketing, Content Analytics was used to identify the characteristics that separated good agents from mediocre ones. The agent enters a brief overview of each customer contact as a memo to prepare for the next call and a report to their managers. These customer contact records can be considered as sales activity reports.

As a part of this analysis, the managers of the agents were asked to select high performing agents. Then the differences in the words used between the reports of high performing agents and low performing agents were analyzed. However, such analysis did not lead to any valuable insights because the differences in the words used did not lead to meaningful actions. Therefore, the focused shifted to

the best performing agent. Content Analytics revealed that this agent tended to initiate contact by expressing appreciation for some action the customer made such as visiting a workshop, submitting a survey result, or for using their product. This unique characteristic was confirmed in the Facets view where expressions for appreciation were highly correlated to this particular agent.

Another characteristic of this agent was that the person kept frequent contact with all of their customers, generally a couple of times each month, as was discovered in the Trends view.

Based on these findings, all agents were sorted by correlation to expressions for appreciation in the Facets view. As a result, agents were found to be divided into two groups, each at opposite ends. One group of high performing agents was highly correlated to expressions for appreciation. All members of this group showed similar patterns of frequent contacts with their customer according to the Trends view with customer names selected as the facet.

The other group of high performing agents was negatively correlated to expressions of appreciation. The contact pattern for this group, according to the Trends view, was different from the frequent contacts made by the other group of high performing agents. For almost all of their customers, they contacted them intensively during a relatively short time, spread across a couple of months to several times a month, with no contact before and after the intensive contacts.

Interestingly, other agents that the managers did not classify as high performers showed another contact pattern. In a sense, their contact pattern was not consistent. They did not contact all of their customers frequently nor intensively. They often contacted customers frequently for a while, and then after a couple of months of no contact, they reconnected.

The difference between the frequent contact group of high performing agents and the short but intensive contact group of high performing agents was analyzed, indicating a clear difference. The high performing agents in the intensive contact pattern have extremely high skill levels as revealed in the Facets view. The skill levels of the high performing agents in the frequent contact group were not necessarily high except for some communication-related skills.

This insight was used to motivate agents to acquire higher skills and to keep a frequent relationship with customers until they gained enough skills.

## 4.3.2 Analysis of other data

Content Analytics can be applied to various types of textual data other than customer contact records. However, the basic cycle of analysis remains the same for most types of textual data. You form your analysis by acquiring insights based on your objectives, perform actions based on the insights, and then validate your actions, which lead to better analysis in the next cycle.

For analysis of surveys, reviews, and bulletin board data, you can often apply approaches similar to the analysis of customer contact records because they also contain the sentiments of the customer. Through the analysis of such data, you can often identify product failures in their early stages. You can also analyze the positive and negative aspects of each product from the consumers' viewpoints, how customers chose each product, and how they are using them. If the data contains demographic information of the submitters, such as age and profession, you can analyze characteristics of opinions and behaviors based on generations and professions.

Project reports are another type of textual data that you can analyze for best practices and identify potential problems by using Content Analytics. You might be able to identify projects in trouble or that are beginning to show signs of trouble by analyzing its textual data with Content Analytics.

### Technical documents

By analyzing technical documents, such as patents, you can identify technical trends, the technical strength of companies, and so on. For example, by adding a technical term as a search condition, you can make a list of companies that filed patent documents that contain the term. With the Trends view, you can identify which companies filed, the number of related patents, and the time they filed them.

For example, if you type “text mining” as a query term, you can list the company names whose patents contain the phrase “text mining” in the Facets view. Then, by selecting the Trends view, you can see the competitive landscape of the companies working in text mining. By using the Facets view, you can identify terms relevant to text mining such as classification, query, and knowledge.

After you select a specific company from the Facets view, you can analyze technical terms relevant to the patents from that company. By analyzing technical terms of a specific company with the Trends view, you might be able to identify technical trends within that particular company.

## 4.4 Summary

Text mining is an interactive procedure with discovery throughout multiple cycles of analysis. The duration of a single cycle of analysis can vary greatly from days to months. In our experience, users often become accustomed to text mining operations for each new cycle, and the results lead to bigger impacts to the business as they become more experienced.

Large amounts of textual data often contain a great wealth of knowledge. With Content Analytics, you can acquire valuable insights depending on the objectives and viewpoints that you set.

**Performing content analysis:** If you already have a working environment that uses Content Analytics, have configured sample data, and are familiar with the user interface of the content analytics miner, go to Chapter 7, “Performing content analysis” on page 231.



## Content analytics miner: Basic features

Chapter 3, “Designing content analytics solutions” on page 41 provides the details about the process of content analysis and how IBM Watson Content Analytics (Content Analytics) is used as a tool to help you analyze textual content. This chapter focuses on the basic features of the content analytics miner, a web-based application that helps you to discover actionable insight from your textual data. It provides details about the application, focusing on the search and discovery features. For information about the content analytics miner views, which are also part of the basic features, see Chapter 6, “Content analytics miner: Views” on page 159.

This chapter includes the following sections:

- ▶ Overview of the content analytics miner
- ▶ Search and discovery features
- ▶ Query Tree and Query builder
- ▶ Rule-based categories with a query
- ▶ Common view features
- ▶ Document flagging

If you are familiar with the user interface of the content analytics miner, including the search and discovery features, and all its views, proceed to Chapter 7, “Performing content analysis” on page 231.

## 5.1 Overview of the content analytics miner

This section provides an overview of the content analytics miner by covering the basic application window layout and functionality. The sections that follow go into greater detail about the various forms of analysis that you can perform by using the sample data set packaged with Content Analytics.

**Sample Text Analytics Collection:** Most of the examples in this chapter refer to the “Sample Text Analytics Collection” that is created when you select **Text Analytics Tutorial** in the First Steps program.

In the previous version of the IBM Redbooks publication, we covered extensively how you can build a content analytics collection with the same data and configuration. Refer to Chapter 4, “Installing and configuring IBM Content Analytics” of the book. You can download it from the additional material associated with this book. See Appendix B, “Additional material” on page 567 for detail.

If you have not build one already, you must build this collection so that you can follow along using your installed version of Content Analytics.

### 5.1.1 Accessing the content analytics miner

The content analytics miner is a Java Platform, Enterprise Edition (Java EE), web-based application that is automatically deployed onto the search server when you install Content Analytics. Before you access the content analytics miner, ensure that at least one content analytics collection is available for search by using the administration console.

You can access the content analytics miner at the following address:

`http://<Content Analytics Server Host Name:Port Number>/analytics/`

*Content Analytics Server Host name* is the server on which you install Content Analytics. *Port Number* is the number that you specified during installation. By default, the port number is 8393 for the embedded web application server, or 80 if using WebSphere Application Server.

If you install Content Analytics with a multiserver installation, you can access the content analytics miner that is installed on each search server node.

If the search run time is not started or if no content analytics collection search process is running, you see the following Alert message, which is also shown in Figure 5-1:

"The Collection is not available. Confirm that the search server is running and that collections are available."

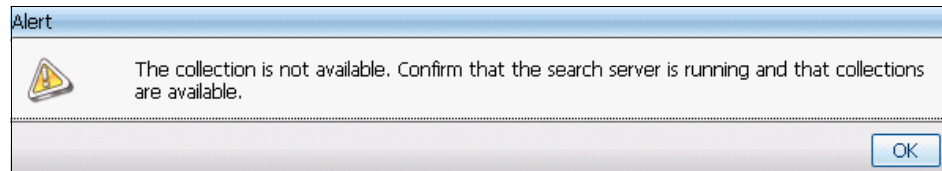


Figure 5-1 Alert message that is displayed if no content analytics collection is available

When you install Content Analytics on a Windows platform, you can also access the content analytics miner from the Start menu on the Content Analytics server by selecting **Start** → **All Programs** → **IBM Content Analytic with Enterprise Search** → **Content Analytics Miner**.

## 5.1.2 Application window layout and functional overview

When you open the content analytics miner with a browser, you see a window similar to the one in Figure 5-2 on page 88. The content analytics miner has the following areas:

- ▶ Application toolbar

This toolbar is at the top of the window (not shown in Figure 5-2 on page 88, but see Figure 5-3 on page 89). You use it to select the content analytics collection to work with, set preferences, modify your profile, obtain help at any time, or access the About information.

- ▶ Query search text field and controls

The search box and controls are located beneath the application toolbar. You use them to build and add to your search expression. Content Analytics shows only those documents that match your current query conditions. This area is hidden by default. To reveal or hide the search text field and controls, you either click the **Show/Hide query input area** icon or click the **Restore/Collapse query input area** icon (circled in Figure 5-2 on page 88).

- ▶ Facet Navigation pane

The Facet Navigation pane is on the left side of the application window. You use this area to filter the results based on selected facets and keywords. This pane is shown by default. To collapse it, you click the **Collapse this area** button between the Facet Navigation and Results view panes (Figure 5-2 on page 88).

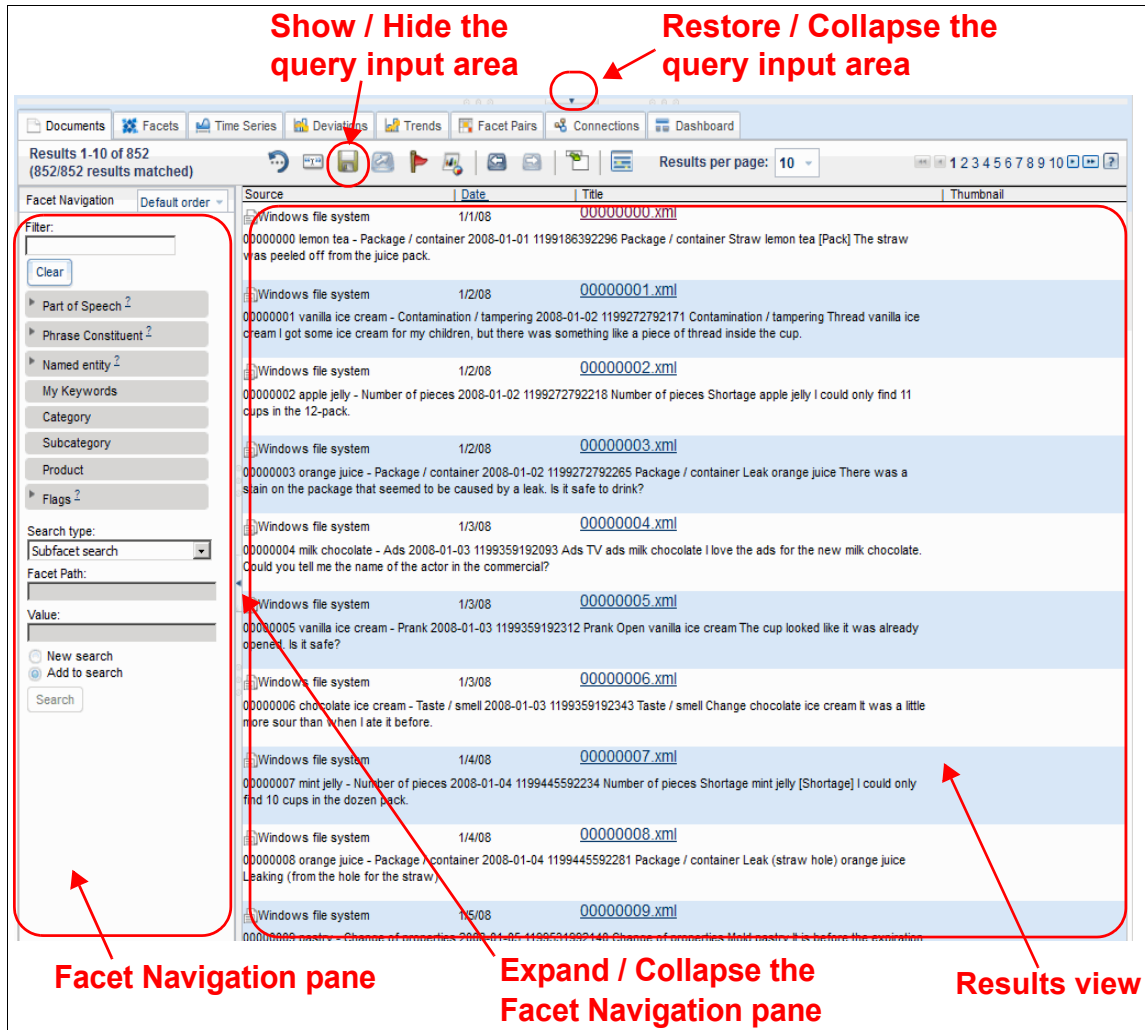


Figure 5-2 Content analytics miner window layout

► Results view

The Results view is located in the center and to the right of the Facet Navigation pane. This area shows the results that match your current search and facet selections. The view changes depending on which of the following view tabs you select:

- Documents view
- Facets view
- Time Series view
- Trends view



- Deviations view
- Facet Pairs view
- Connections view
- Dashboard view

See Chapter 6, “Content analytics miner: Views” on page 159, for more details about each of these views.

**Dashboard view:** The Dashboard view requires additional configuration. See 6.9, “Dashboard view” on page 206 for more details.

## Application toolbar

The application toolbar is at the top of the content analytics miner window. This toolbar shows the name of the current content analytics collection that is under analysis. You can easily switch to another content analytics collection by clicking the **change** link (Figure 5-3).



Collection: Sample Text Analy... (change)    Logged in as: Not logged in | Preferences | My Profile | Help

Figure 5-3 Application toolbar

The *Logged in as* field shows the name of the currently logged in user. If security is not enabled, the “Not logged in” message is then displayed.

**More information:** See the Content Analytics IBM Knowledge Center for details about security for content analytics collections.

When you click the **Preferences** link, a new dialog is displayed in which you can set various parameter options for Search, Results, Result Columns, and each of the specific Content Analytics views. When you click the **My Profile** link, you see a table of security credentials to use when accessing secured data sources. If security is enabled for the application server and your content analytics collection was created with security enabled, you need to specify access credentials (user IDs and passwords) for each data source that requires secured access.

At any time during the operation of the content analytics miner, you can click the **Help** link for assistance, which opens a new browser window. When accessing Help, the IBM Content Analytics Information Center must be open on the Content Analytics server that is running.

**Assumption:** This chapter was written with the understanding that security is not enabled on the content analytics collection.

**Knowledge Center:** By default, the IBM Content Analytics Information Center on the Content Analytics server starts when you start the Content Analytics server by using the `esadmin system startall` command.

## Search box and controls

The search box under the application toolbar is used to find, mine, and filter documents based on your queries. Content Analytics contains a fully functioned search engine that is both scalable and fast so that you can explore your data by using conventional search methods.

By default, the search box is hidden. After you either click the **Show query input area** button or the **Expand query input area** twisty, you will see the search box dialog (as in Figure 5-4). You can modify the query that is used for your analysis from this field.

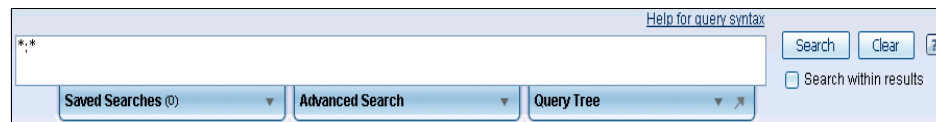


Figure 5-4 Search fields and controls

When expanded, the current query is displayed in the search box dialog. Each addition to your search condition is appended to the query. At any time, you can save your current query state by clicking the **Saved Searches** tab. After giving your query a name and saving it, you can return to that state by clicking the **Saved Searches** tab and selecting any previously saved queries.

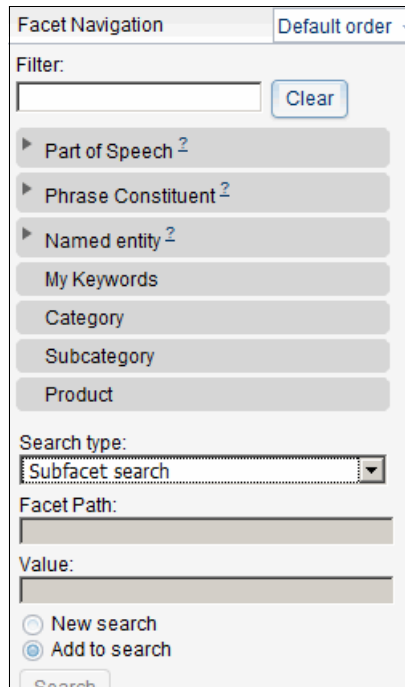
You can use the **Advanced Search** tab to assist in the formulation of your search expression. By using the **Advanced Search** tab, you can specify complex search options, such as an exact phrase search or a search on date ranges, without knowing the Content Analytics query syntax.

A graphical layout of your query is displayed when you click the **Query Tree** tab.

For more information about these tabs, see the section 5.2.6, “Saved searches” on page 105, section 5.2.7, “Advanced search” on page 107, and section 5.3, “Query Tree” on page 107.

## Facet Navigation and search filter

The Facet Navigation pane, described in the Figure 5-2 on page 88, is always present and displayed on the left when you access the content analytics miner. In this pane, you see the facet tree that you defined for the specific content analytics collection in the administration console. The facet tree includes a list of predefined facets, such as Part of Speech and Phrase Constituent, as shown in Figure 5-5.



The screenshot shows the 'Facet Navigation' pane with a 'Default order' dropdown menu. Below the title is a 'Filter:' section with an empty text input field and a 'Clear' button. A list of facets follows: 'Part of Speech' (with a twisty icon and a superscript 2), 'Phrase Constituent' (with a twisty icon and a superscript 2), 'Named entity' (with a twisty icon and a superscript 2), 'My Keywords', 'Category', 'Subcategory', and 'Product'. Below the facets is a 'Search type:' dropdown menu currently set to 'Subfacet search'. Underneath is a 'Facet Path:' text input field, followed by a 'Value:' text input field. At the bottom, there are two radio buttons: 'New search' (unselected) and 'Add to search' (selected), and a 'Search' button.

Figure 5-5 Facet Navigation pane and search filter

You can click a facet in the Facet Navigation pane to select it, at which point the selected facet is highlighted in blue. For facets that are hierarchical, you can expand and collapse elements of the facet tree by clicking the twisty on the left of the top-level facet in the facet tree hierarchy. Also, several controls are available to assist you in locating a specific facet. These controls are useful when the number of facets is large. You can filter which facets to display by the name of the facet or by the actual value of a particular facet.

## Content analytics miner views

You can work with eight default views (Figure 5-6) in the content analytics miner, depending on your analysis goals, and interact with each view to mine further into the collection data. In addition to the six analytics views, the Document view is always present for you to work with documents responding to your selection criteria. The Dashboard view presents analysis statistics in different formats, such as pie charts, bar charts, and timelines, in a single view. For a detailed description of each view, see Chapter 6, “Content analytics miner: Views” on page 159.



Figure 5-6 Tabs for each view, with the Documents view selected (white background)

### 5.1.3 Selecting a collection for analysis

To start analyzing your data with the content analytics miner, you must select a collection from the application toolbar. If the system manages multiple content analytics collections where the enterprise search capability of those collections is active, a default collection will usually start when launching the content analytics miner. If you want to select a different collection that you have access, click the **change** link (see Figure 5-3 on page 89). You can analyze only one collection at a time with the content analytics miner.

### 5.1.4 Changing the default behavior by using preferences

You can change the default behavior of the content analytics miner by using the Preferences window that is accessible from the application toolbar. The changes made by using the Preferences window remain in effect during your browser session. Otherwise, if global security is enabled, they are persistently saved in your profile. After you click the **Preferences** link in the application toolbar, the Search and Result Preferences window (Figure 5-7 on page 93) opens.

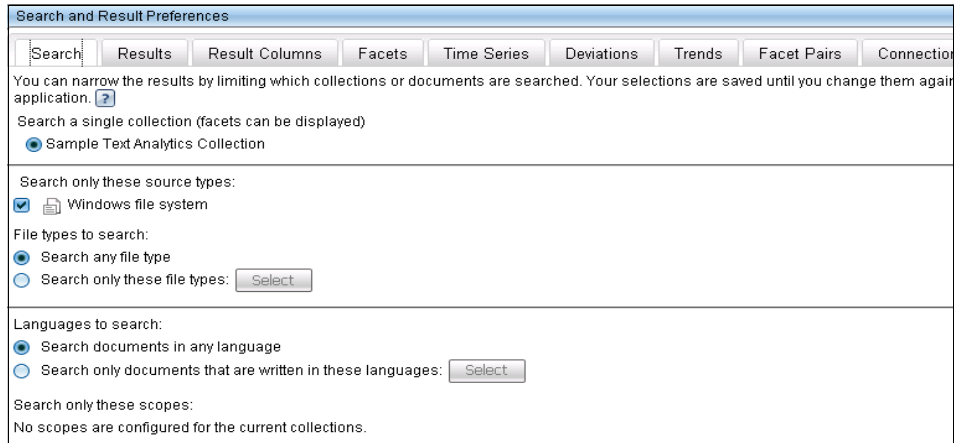


Figure 5-7 Content analytics miner: Preferences window - Search tab

This section highlights the options that are available in the Search and Result Preferences window, in particular, the Search, Results, and Result Columns tabs. These three tabs affect the results that are displayed in the Documents view and help you to review the details of the documents that are selected.

You can also set the preferences for other views, such as the Facets view, Time Series view, Deviations view, Trends view, Facet Pairs view, Connections view, or Dashboard view from the Preferences window.

## Search tab

From the **Search** tab, you must select a content analytics collection to analyze. You also select the following properties on this tab:

- ▶ The source types that you want to search. More than one source could be available if the collection is configured to crawl multiple data sources.
- ▶ The file types to search. By default, **Search any file type** is selected. You can also select one or more specific available file types in the collection.
- ▶ The languages to search. By default, **Search documents in any language** is selected. You can specify the language to search if you focus on documents written in one or more specific languages.
- ▶ Search only the specific scopes. This option is available only when you define a search scope. A scope is a group of related URLs in an index. When you configure a scope, you limit the documents that users can see in the collection. When users search the collection, they search only the documents in the scope, not the entire index. Refer to the Content Analytics documentation for more details.

## Results tab

In the **Results** tab, you control how the results are displayed in the Documents view. The change takes effect right after it is saved. The properties are useful when you perform the search from the Search box field and are important when you filter your data with a query. Figure 5-8 presents the Results tab of the Preferences window.

The screenshot shows the 'Search and Result Preferences' window with the 'Results' tab selected. The window has a title bar and a tabbed interface with tabs for Search, Results, Result Columns, Facets, Time Series, Deviations, Trends, Facet Pairs, Connections, and Dashboards. Below the tabs, there is a descriptive text: 'Specify how documents that match your query are organized and displayed. Your selections take effect with the next query that you submit. Y saved until you change them again or until the current session ends.'

The main configuration area includes several sections:

- Number of results per page:** A spinner box set to 10.
- Sort by:** A dropdown menu set to [Relevance].
- Sort by (2nd):** A dropdown menu set to None.
- Sort by (3rd):** A dropdown menu set to None.
- Query language:** A dropdown menu set to English.
- Sort order:** A dropdown menu set to Descending.
- Sort order (2nd):** A dropdown menu set to Descending.
- Sort order (3rd):** A dropdown menu set to Descending.
- Query mode (how to match query terms):** A dropdown menu set to No preference.

Below these are a **Summary length:** slider ranging from Minimum to Maximum, and **Number of type ahead suggestions:** a spinner box set to 10. The **Type ahead mode:** dropdown is set to 'Suggest matches from queries, then matches fro'.

At the bottom, there are four radio button options:

- Include quick links:** Yes (unselected), No (selected).
- Collapse results from same source:** Yes (unselected), No (selected).
- Suggest spelling corrections:** Yes (selected), No (unselected).
- Search for synonyms:** Yes (selected), No (unselected).

Figure 5-8 Content analytics miner: Preferences window - Results tab

You can select the following properties from this tab:

- ▶ *Number of results per page.* By default, 10 results per page are displayed. If you click the **count up** or **count down** button, the value increases or decreases by 5. You can also specify a discrete number in the box. Valid values are 1 - 100.
- ▶ *Query language.* By default, the same value sets in the **Language to search** field on the **Search** tab, or a selection of the available languages if more than one.
- ▶ *Query mode.* You can select the linguistic parsing method for the query that you input. By default, this option is set to **No preference**. The following values are available:
  - Base form matching

- Exact form matching
- Base and exact form matching

For the difference of each mode, refer to the Content Analytics documentation.

- ▶ *Sort by.* You can configure up to three different sorting fields. By default, in the first field, **Relevance** is selected. All index fields that are specified as sortable in the administration console are available in each selectable list.
- ▶ *Sort order.* After you select an index field in any of the **Sort by** fields, you can select the sort order as either **Descending** or **Ascending**.
- ▶ *Summary length.* By default, the length of the summary is set as a medium value, but you can adjust the length of the summary from minimal to maximal lengths.
- ▶ *Number of type-ahead suggestions.* By default, 10 suggestions are specified from the type-ahead feature. Valid values are 1 - 100.
- ▶ *Type-ahead mode.* By default, **Suggest matches from queries, then matches from the index** is selected. Following are the methods that are available:
  - Disable type-ahead suggestions
  - Suggest matches from previous queries
  - Suggest matches from previous index terms
  - Suggest matches from queries, then matches from the index
  - Suggest matches from index, then matches from the queries
- ▶ *Include quick links.* By default, **No** is selected.
- ▶ *Suggest spelling corrections.* By default, **Yes** is selected.
- ▶ *Collapse results from the same source.* By default, **No** is selected.
- ▶ *Search for synonyms.* By default, **Yes** is selected.
- ▶ *Show a file type filter.* By default, **No** is selected.
- ▶ *Show icons to list documents similar to a selected document.* By default, **No** is selected.
- ▶ *Similarity.* If you enable document similarity, set the Similarity field value to define how similar documents must be in order to be identified as being similar to one another. Valid values are Low (0.1) - High (1.0).
- ▶ *Show categories when you preview documents.* By default, **Yes** is selected.
- ▶ *Show category rules when you preview documents.* By default, **No** is selected.

## Result Columns tab

On the **Result Columns** tab, you select which columns to display in the Documents view and the order of the columns. You can select or clear the column name and change the column order.

When you configure the system to use either the *document flags* or *Query Builder* capabilities, customizing the column order is helpful. The document flags functionality adds a *Flags* column to the Documents view, and the Query Builder adds an *Actions* column. The order of these columns can be modified. For the *Flags* column to appear in the Documents view, one or more flags need to be defined by the collection administrator with the Content Analytics administration console.

## 5.2 Search and discovery features

The content analytics miner is a working application that enables you to mine content, similar to a business intelligence tool, over content analytics collections used to gain valuable insights from one or more of your data sources. Content mining itself is an exploratory task with basic goals set by you. The goals that you define serve as a guide for the strategies that you employ to explore your data.

At the core of the Content Analytics product is a search engine that is scalable, fast, and helps you meet your analytics and business intelligence goals. The search engine is based on the open source Lucene indexer that is enhanced with extensions made by IBM. It is uniquely engineered to support rapid search and discovery of your data.

The exploration of your data is one of the iterative steps taken through the many dimensions of your data. Content Analytics guides you through the exploration of these various dimensions, alerting you to any interesting patterns or trends that warrant further investigation.

When you initially start the content analytics miner and choose a content analytics collection for analysis, your search query (which is displayed in the search box) is set to the wildcard expression “\*:\*”. This search expression results in all documents in the collection to be matched and thus equates to all documents in the collection being analyzed. From this starting point, you can navigate through the documents by using the Documents view, or you can see how all of the documents are distributed over time by using the Time Series view. You can also start to explore the various facets of your data by using the Facets view.



Eventually you can narrow the scope of your analysis to a more focused subset of documents. Narrowing down the documents is achieved by using facet selection, search expressions, or a combination of both.

This section provides details about the following search and discovery features:

- ▶ Limiting the scope of your analysis using facets
- ▶ Limiting the scope of your analysis using search operators
- ▶ Limiting the scope of your analysis using dates
- ▶ Query syntax
- ▶ Type ahead
- ▶ Saved searches
- ▶ Advanced search

For other search and discovery topics, see the following sections:

- ▶ 5.3, “Query Tree” on page 107
- ▶ 5.4, “Query builder” on page 123
- ▶ 5.5, “Rule-based categories with a query” on page 135




## 5.2.1 Limiting the scope of your analysis using facets

In the Facet Navigation pane, you can select a particular facet for viewing specific aspects or dimensions of your document corpus in the analytic collection. Typically, you use the Facets view to see the most frequently occurring values found in your data for a particular facet. For example, in the sample collection, you have a facet labeled “Subcategory”. When you select the Facets view, the system lists the top 40+ most frequently occurring product subcategories of the problem reports mentioned in your textual data. The view shows 500 values by default, which you can change by using the Preferences window as previously mentioned.

At any time, you can select one or more keywords and add them to your query to limit the scope of your analysis. In our subcategory facet example, if we select the keywords Straw and Allergy and add them to our query, only a subset of our collection documents, mentioning Straw or Allergy, are then used in analysis by the content analytics miner.

You can change the way that the keywords are added to your current query expression by using the Boolean logic: AND, OR, and AND NOT. In each of the content analytics miner result views (except for the Documents view), you see the search Boolean operators icons: AND, OR, and AND NOT. Each icon gives a visual representation of a Venn diagram depicting the type of Boolean operation to be performed, as described in Table 5-1 on page 98.

Table 5-1 Boolean operators and their description

Icon	Description
AND 	This operator generates the appropriate search syntax for the keywords that you select and appends it to your query with an AND condition. The result is documents that match your query condition <i>and</i> contain any of the selected keywords.
OR 	This operator generates the appropriate search syntax for the keywords that you select and appends it to your query with an OR condition. The result is documents that match your query condition <i>or</i> the presence of any of the selected keywords.
AND NOT 	This operator generates the appropriate search syntax for the keywords that you select and appends it to your query with an AND NOT condition. This method is a convenient way to exclude certain documents from your analysis.

For additional information and examples of using these search operators, see section 5.3.2, “Understanding the Query Tree” on page 108.

## 5.2.2 Limiting the scope of your analysis using search operators

Working with facets is one natural way to navigate the documents set and constrain the scope of your analysis to only those documents that match specific facets values. However, facets must be contrived and defined in advance when you are building your content analytics collection.

What if during your analysis with the content analytics miner, a facet is not defined for the type of constraint that you want to filter on? In this case, you can use the powerful search capabilities of Content Analytics to limit your analysis to only those documents that match your query. The filtering of documents based on your search expression can be used in addition to the facets filtering that you have already selected.

Content Analytics provides a comprehensive query syntax with a robust set of search operators. You can use the **Advanced Search** function to assist you in your search expression. You can also use the Query Tree function to view the logical structure of your query as it grows more complex. When analytics collection security is turned on, you can save at any time the current state of your queries for future use during a browser session or persistently across a browser session. Saving queries is a convenient way for you to go back to the analysis that you previously did.

For more information about query syntax and Query Tree, see 5.2.4, “Query syntax” on page 99.

## 5.2.3 Limiting the scope of your analysis using dates

If your documents contain one or more *date* fields, you can limit the scope of your analysis to documents that match specific dates or that fall within a given date range. The date range facet can be used to analyze the data within a given range.

You can specify date constraints by using the **Advanced Search** tab or by manually entering a parametric (date) search expression. An easier way to select specific dates or ranges is to use the Time Series view as explained in 6.4, “Time Series view” on page 169. In addition, when you configure a date range facet, you can use the data range facet to analyze the data within a given range.

A similar and important feature of Content Analytics is its ability to identify *trends* and *patterns* in your data that occur over time. In order for Content Analytics to identify trends and patterns, it must base its calculations on a reference date value that consistently occurs in each document of the content analytics collection.

With the date field, Content Analytics performs time-sensitive calculations and renders the various views including Time Series, Deviations, and Trends views. For more information about the various views, see Chapter 6, “Content analytics miner: Views” on page 159.

When your documents contain multiple date fields, you can use any of the various date fields for your analysis after you configure the date facet. By changing the specific date facet to use for analysis, you can then analyze the same documents set from another aspect based on a different date field.

**Configuring the date facet:** To configure the date facet to contain more than one date field, to understand which date formats are detected by default and in which order, to set up custom date formats, to control the dates display in the query results, or to configure which fields to use as a date facet in the content analytics miner, refer to the following sections in the Content Analytics IBM Knowledge Center:

- Date fields and custom date formats
- Configuring date range facets for a content analytics collection
- Analytic resources and content analytics

## 5.2.4 Query syntax

As your analysis progresses, you notice in the search box the precise query expression translated for the current set of documents being analyzed. Usually

you do not need to consider the detailed query syntax itself. Other times you might find it useful to modify or add to the search expression manually as you become more comfortable with the query syntax. The query syntax supports multiple search operators. For example, you might want to use the following query syntax to help you narrow down the target documents set and discover the data:

- ▶ **Faceted Path search**

You can use the query with facet name and its path. For example, the query [keyword::/"Product"/"apple juice"] returns the documents in the Product facet with the keywords “apple juice.”

**Facet path used in query:** Because Content Analytics uses an internal representation for the facet path, it might be difficult to construct the faceted path search yourself when you use the content analytics miner. However, you can confirm how the Facet Path search is constructed when you add the facets with search operators in other views. You can see how the actual query keyword is displayed from the index field when you expand the Query Tree.

- ▶ **Proximity search**

You can search a keyword within a specified number of keywords or within a sentence. For example, the query (cream dirty) WITHIN 8 returns the documents that have those keywords within eight words of each other (and in any order).

If you need to consider the word order, you can add INORDER at the end of the query, such as (cream dirty) WITHIN 8 INORDER. This way the query returns the documents that include the keyword within specified word gap in order.

Alternatively, the query (purchase ice cream) WITHIN SENTENCE returns the documents that include all query keywords in the same sentence.

- ▶ **Fuzzy search**

You can set the ambiguity with the ~ operator. For example, the query apple juice~0.5 returns the documents that include apple juice, apple juicer, and so on.

- ▶ **Wildcard search**

You can replace some part of the query keyword or phrase with wildcard characters, such as a question mark (?) or an asterisk (\*). For example, the query \* juice returns documents that include apple juice, pine juice, and so on. Also, the query bot??? returns the documents that include bottle, bottom, and so on.

► Conceptual search

If you configured either IBM Content Classification integration or document clustering, you can perform the conceptual search. For more details, see 9.2.3, “Using a conceptual search for advanced content discovery” on page 312.

Those are only a few examples of the extensive query syntax available to allow you to mine your documents set. Find below some additional query syntax that is available:

► Simple query syntax characters

~ (prefix)  
~ (postfix)  
+  
-  
=  
\  
\*.\*  
\*  
?  
“ “  
  
/facet\_name/value\_level\_1/.../value\_level\_n  
^boost  
~ambiguity  
( )

► Query syntax for query keywords

IN contextual view  
(terms) WITHIN context IN ORDER  
(terms) ANY number  
site:text  
url:text  
link:text  
field:text  
docid:documentid  
samegroupas:URI  
facetName::/facet\_name\_1/.../facet\_name\_n  
facetValue::/facet\_name\_1/.../facet\_name\_n/value  
date::/facet\_name/time\_scale/value  
facet::/facet\_name/value\_level\_1/.../value\_level\_n  
flag::/flag\_name  
scope::/scope\_name  
rulebased::category\_ID  
\$source::source\_type  
\$language::language\_id

```

$doctype::document_type
$similar::document_id~similarity
#field::=value
#field::>value
#field::<value
#field::>=value
#field::<=value
#field::>value1<value2
#field::>=value1<=value2
#field::>value1<=value2
#field::>=value1<value2
#field::>"Date"
ACL constraints: (security_tokens)

```

► Query syntax characters for opaque terms

```

@xmlf2::'<tag1> text1 </tag1>'
@xmlf2::<tag1><.depth value="$number"><tag2> ...
</tag2></.depth></tag1>
@xmlf2::<tag1><.depth value='$number'><tag2> ...
</tag2></.depth></tag1>
@xmlf2::'<tag1> ... </tag1>'
@xmlf2::'+text1 ... +text2 -text3 ... -text4 text5'
@xmlf2::'<tag1> <.or> ... </or> <.and> ... </and> </tag1>'
@xmlf2::'<annotation1+annotation2> ... </annotation1+annotation2>'
@xmlf2::'<annotation1*annotation2> ... </annotation1*annotation2>'
@xmlxp::'/tag1/@tag1'
@xmlxp::'/tag1[tag2 or tag3 and tag4]'
@xmlxp::'tag1//tag2/tag3'
@xmlxp::'tag1/.../tagn'

```

For details about the query syntax concept or a specific query syntax, review the IBM Content Analytics Information Center at the following address, and search on *query syntax*:

<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/index.jsp>

**Help for the query syntax:** If the IBM Knowledge Center is running on your Content Analytics server, you can access the query syntax help from the **Help for query syntax** link in the content analytics miner (see Figure 5-9).

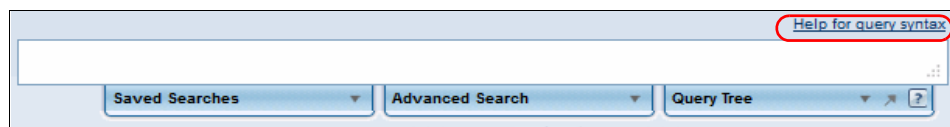


Figure 5-9 Help for query syntax

## 5.2.5 Type ahead

The type-ahead feature helps you find the query keyword that you can use in your analysis. With the type-ahead feature, query suggestions based on the indexed terms or previous queries can be shown as you type the query keywords in the content analytics miner. You can use the type-ahead feature when the content analytics collection is configured to enable the type-ahead feature.

### Configuring the type-ahead feature

The type-ahead feature is automatically enabled when you create a content analytics collection. You can specify where the type-ahead suggestions come from. By default, the query suggestions from both indexed terms and previous queries are built in the query index. You configure the type-ahead options from the administration console as shown in Figure 5-10.

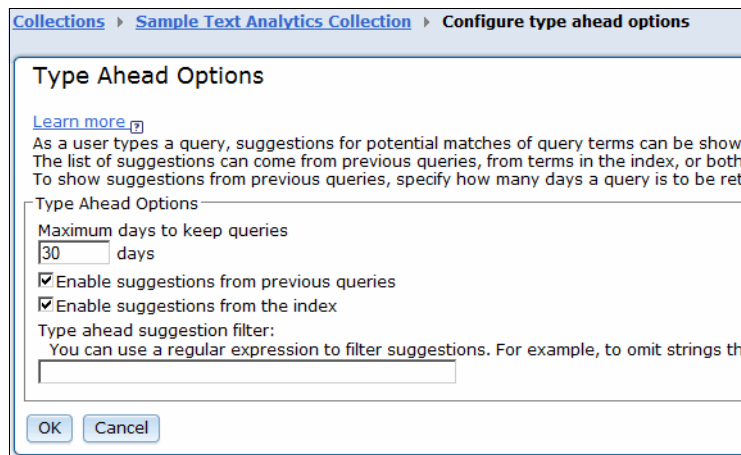


Figure 5-10 shows a screenshot of the 'Configure type ahead options' dialog box in the administration console. The dialog is titled 'Type Ahead Options' and includes a 'Learn more' link. Below the link, there is explanatory text: 'As a user types a query, suggestions for potential matches of query terms can be shown. The list of suggestions can come from previous queries, from terms in the index, or both. To show suggestions from previous queries, specify how many days a query is to be retained.' The dialog contains several configuration options: 'Maximum days to keep queries' with a text input field set to '30' and the label 'days'; two checked checkboxes: 'Enable suggestions from previous queries' and 'Enable suggestions from the index'; and a 'Type ahead suggestion filter:' section with a text input field and the instruction: 'You can use a regular expression to filter suggestions. For example, to omit strings that'. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

Figure 5-10 Configuring the type-ahead feature on the administration console

For details about the type-ahead feature configuration, review the IBM Content Analytics Information Center at the following address, and search on *type ahead support for queries*:

<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/index.jsp>

### Using the type-ahead feature in the search box

When the type-ahead feature is configured, you see the query suggestion, as shown in Figure 5-11 on page 104. For example, when you start typing the letter “p” in the query input area, the specific query suggestions are displayed based on your Preferences settings. You can configure how many suggestions are displayed in the window and in which order the suggestions are displayed (for

example, below, the preference was set to “Suggest matches from previous queries, then matches from the index”).

p			
pine juice yesterday	Previous queries	(Estimated results)	3
pine			40
Index terms (Estimated results)			
Package			310
pack			200
pieces			100
pastry			60
Price			60
properties			50
prices			40
pine juice			40

Figure 5-11 Query suggestions when you type a character

When you type more than one character, such as “pi” in the search query field, the query suggestions are narrowed to those words that match the query that you typed, as shown in Figure 5-12.

pi			
pine juice yesterday	Previous queries	(Estimated results)	3
pine			40
Index terms (Estimated results)			
pieces			100
pine juice			40
piece			40
pieces Shortage chocolate ice cream			10
pieces Shortage chocolate			10
pieces Shortage vanilla ice cream			10
pieces Shortage pastry			10
pieces Shortage milk chocolate			7

Figure 5-12 Query suggestions when you type more than one character

## Using the type-ahead feature in the Facet Navigation pane

In addition to using the type-ahead feature in the search query field, you can also use it in the Facet Navigation pane when it is enabled. In the Facet Navigation pane, you can use the type-ahead feature by selecting a facet and typing a keyword.

For example, Figure 5-13 on page 105 shows the result of clicking the **Product** facet, selecting **Value search** in the Search type field, and typing “cho” in the Value field. The possible keywords (up to 10 keywords) for the Product facet that begins with cho are displayed in alphabetical order.



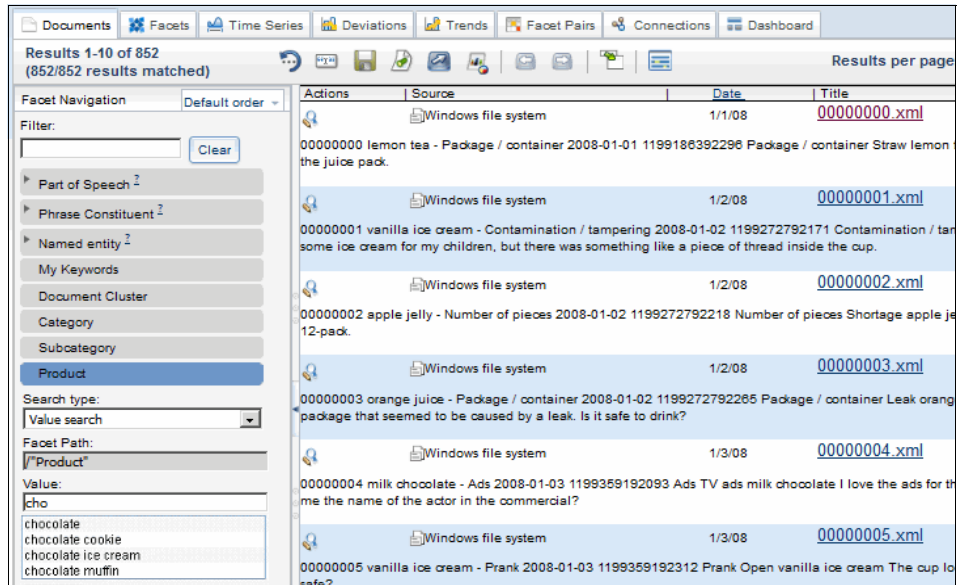


Figure 5-13 Type-ahead feature in the Value field of the Facet Navigation pane

To use the type-ahead feature in the Facet Navigation pane, you must select **Value search** for the *Search type* field. Otherwise, if the Search type field is set to Subfacet search, the Value field is not editable.

**Type-ahead feature in the Facet Navigation pane:** The type-ahead feature in the search box differs from the type-ahead feature in the Facet Navigation pane. The Facet Navigation pane has the following functionality differences:

- ▶ Up to 10 suggestions are shown by default, and the number of suggestions to display cannot be changed to another number.
- ▶ The suggestions are displayed in alphabetical order, not in frequency order.
- ▶ A facet must be selected because the suggestions are based on the facet value.

## 5.2.6 Saved searches

During your analysis, you can save the current state of your query at anytime by clicking the **Save** icon. When you click the right arrow of the **Saved Searches** tab in the search field area, you see the number of the saved queries and a list of saved queries with the names that you assigned, as shown in Figure 5-14 on page 106.

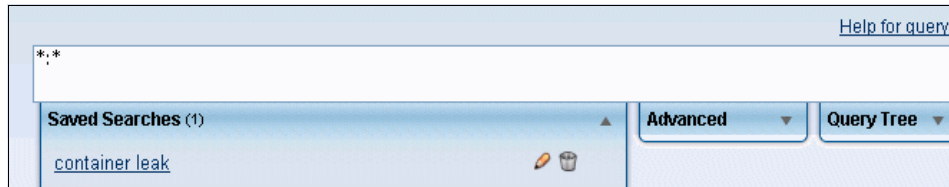


Figure 5-14 Saved Searches window

When you click the name of the saved search, the query starts. If you need to edit the saved query, click the **Pencil** icon on the right side.

In the Edit Saved Search window (Figure 5-15), you can edit the name, query, and description similar to when you saved the query condition before.

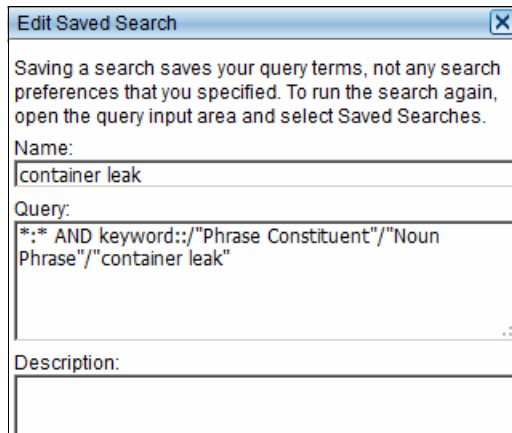


Figure 5-15 Edit Saved Search window

You can also click the **Trash** icon to discard the saved query condition.

**Global security and saved search:** If the login security *is* enabled and you log in to the content analytics miner, you can save search queries persistently between login sessions. However, you cannot share the query condition with other users.

If the login security is *not* enabled and you are not requested to log in to the content analytics miner, a saved query is only saved during the current login session. If you need your query condition to be saved longer than your session, you must enable global security as in the Content Analytics IBM Knowledge Center.

## 5.2.7 Advanced search

On the **Advanced Search** tab, you must first select whether you want to perform a new search or add to a search. The default is to create a search.

Next, you can set the query keywords in each field based on your requirement. For example, you can complete the *All of these words* field, *The exact phrase* field, *Any of these words* field, or *None of these words* field. You can also specify the *Start date* or *End date* to search documents within a specified time period.

Figure 5-16 shows the expanded Advanced Search page.

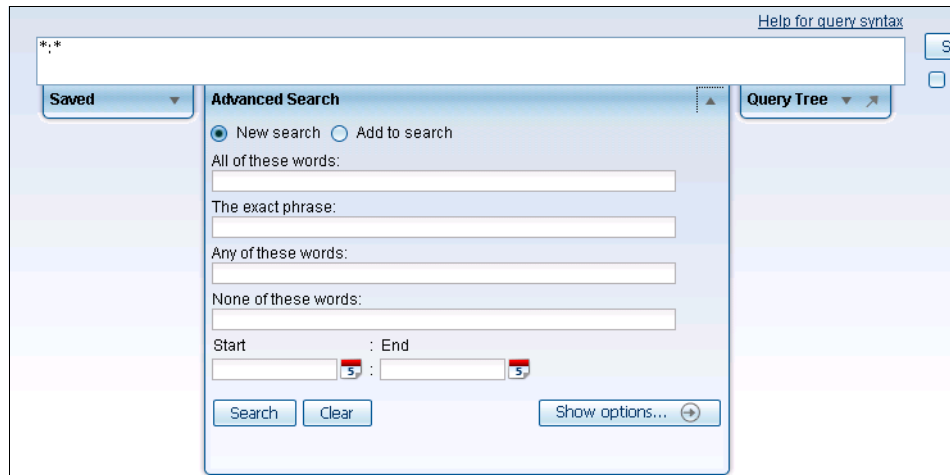
The screenshot shows the 'Advanced Search' window. At the top, there is a search input field containing '\*.\*' and a 'Help for query syntax' link. Below the input field is a 'Saved' dropdown menu. The main area is titled 'Advanced Search' and contains two radio buttons: 'New search' (selected) and 'Add to search'. There are four text input fields: 'All of these words:', 'The exact phrase:', 'Any of these words:', and 'None of these words:'. Below these is a date range selector with 'Start' and 'End' labels and two date input fields. At the bottom, there are 'Search' and 'Clear' buttons, and a 'Show options...' button with a right-pointing arrow.

Figure 5-16 Advanced Search window

**Show Options button:** You can change the Search preference when you click **Show Options** on the Advanced Search page. This option is less frequently used in the content analytics miner.

## 5.3 Query Tree

The *Query Tree* is a visual representation of the logical query structure that you enter into the search query field. It helps you to view the logical hierarchical structure of the current query. The Query Tree is most useful when the entire query becomes large. You can see how the various sections of the query contribute to the overall result set. For each query section, the Query Tree shows the number of documents that match the specified criteria within the collection.

You can change the operator or keywords or tentatively remove each selected node. To remove the selected node, select the **Not And** icon to see the search results without the node included in the query. You can also delete sections of the query by selecting the node that you want to remove and clicking the **Trash** icon next to it.

This section provides information about how to interpret the Query Tree and use the search operators contained therein.

### 5.3.1 Accessing the Query Tree

To view the Query Tree, select the **Query Tree** tab in the search field area. Each query keyword and search operator is presented as a node.

Figure 5-17 shows the Query Tree when you narrow your search to “leak” using the faceted search with the AND operator to the default query \*:\*.

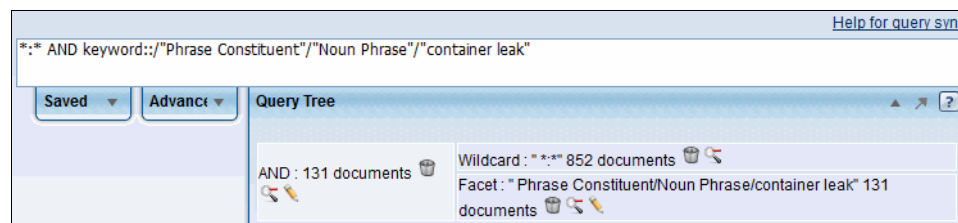


Figure 5-17 Query Tree tab

You can minimize the Query Tree by clicking the triangle icon in the upper-right corner of the Query Tree view. You can maximize the Query Tree area by clicking the arrow in the upper-right corner of the Query Tree view.

### 5.3.2 Understanding the Query Tree

To help you understand the Query Tree, consider the example that is shown in Figure 5-18 on page 109, which shows the results of searching all documents in the collection. By default, the query keyword is set as \*:\*, which means to search for all documents in the collection.

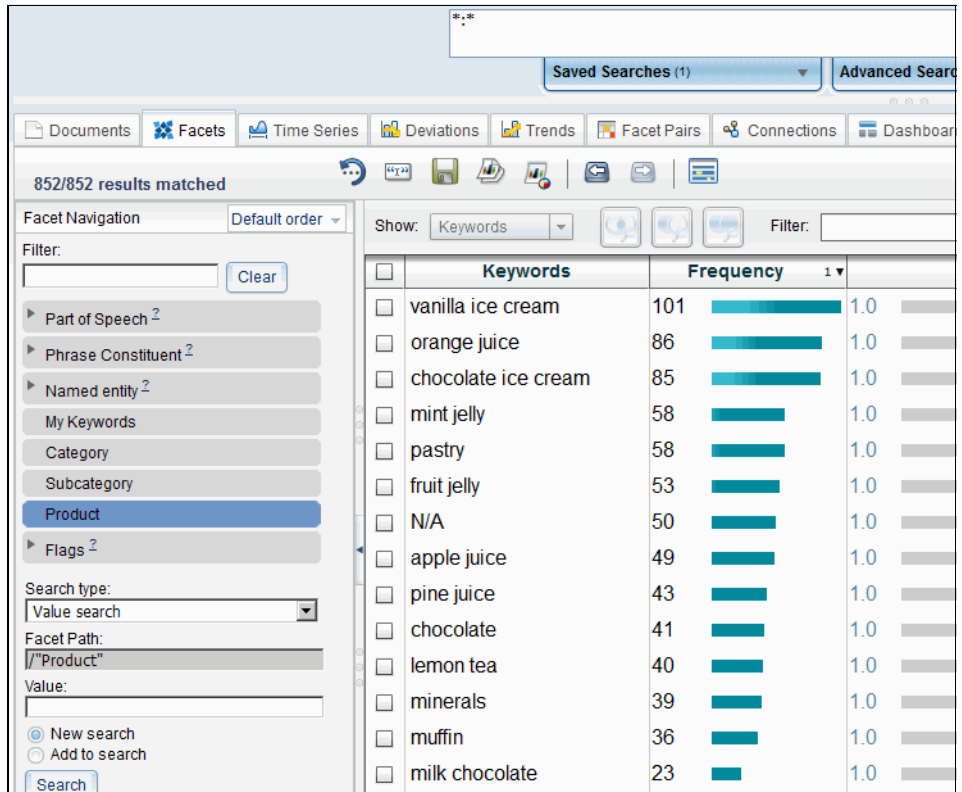


Figure 5-18 The result with the default query \*.\* in the Facets view

### 5.3.3 Query Tree examples

This section shows examples of using the search operators.

#### The AND operator

From the Facet Navigation pane (Figure 5-19), we select the **Product** facet and the keyword **pine juice**, and then we click the **AND** operator. The results are limited to show only the pine juice-related documents.

The screenshot shows the IBM Watson Content Analytics interface. On the left, the Facet Navigation pane is visible, with the 'Product' facet selected. In the main search results area, a table lists various keywords and their frequencies. The 'pine juice' keyword is checked, and its row is highlighted. The 'AND' operator icon is circled in red, with a red arrow pointing to it from the text 'AND operator'.

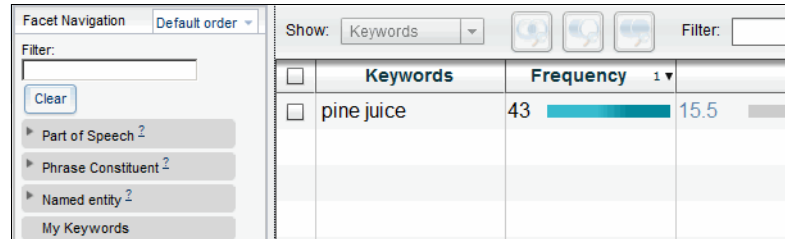
Keywords	Frequency	1
<input type="checkbox"/> vanilla ice cream	101	1.0
<input type="checkbox"/> orange juice	86	1.0
<input type="checkbox"/> chocolate ice cream	85	1.0
<input type="checkbox"/> mint jelly	58	1.0
<input type="checkbox"/> pastry	58	1.0
<input type="checkbox"/> fruit jelly	53	1.0
<input type="checkbox"/> N/A	50	1.0
<input type="checkbox"/> apple juice	49	1.0
<input checked="" type="checkbox"/> pine juice	43	1.0
<input type="checkbox"/> chocolate	41	1.0
<input type="checkbox"/> lemon tea	40	1.0
<input type="checkbox"/> minerals	39	1.0
<input type="checkbox"/> muffin	36	1.0
<input type="checkbox"/> milk chocolate	23	1.0

Figure 5-19 Selecting the Product facet and pine juice and clicking the AND operator

Figure 5-20 shows the query changes as follows:

```
*:* AND keyword::/"Product"/"pine juice"
```

Only pine juice-related documents are shown in the Facets view.



Keywords	Frequency
pine juice	43

Figure 5-20 The query and result changes in the Facets view with the AND operator

Figure 5-21 shows the associated Query Tree in this example.

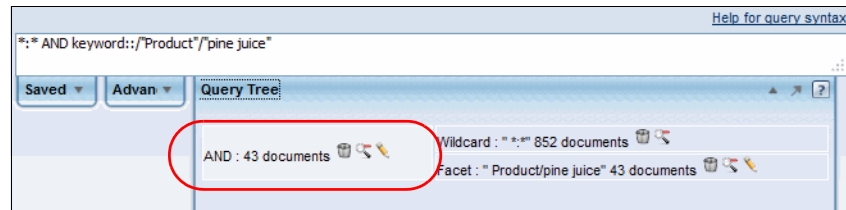


Figure 5-21 Query Tree: The Product facet, pine juice, and \*.\*

The left node of the Query Tree shows an AND operator and the number of the documents found (43 documents) as a result of the query following query:

```
*:* AND keyword::/"Product"/"pine juice"
```

In this case, the query \*.\* and the query keyword::/"Product"/"pine juice" are logically aggregated (ANDed). Also Content Analytics finds 43 documents that satisfy the query. From each node on the right side of the Query Tree, the query \*.\* returns 852 documents, and the facet search returns 43 documents.

## The AND NOT operator

From the Facet Navigation pane (Figure 5-22), we select the **Product** facet and **pine juice**, and then we click the **AND NOT** operator.

The screenshot shows the IBM Watson Content Analytics interface. The Facet Navigation pane on the left has 'Product' selected. The main table displays search results with columns for Keywords, Frequency, and a score. The 'pine juice' row is highlighted and has its checkbox checked. A red circle highlights the 'AND NOT' operator icon in the toolbar, with a red arrow pointing to it and the text 'AND NOT operator'.

Keywords	Frequency	Score
<input type="checkbox"/> vanilla ice cream	101	1.0
<input type="checkbox"/> orange juice	86	1.0
<input type="checkbox"/> chocolate ice cream	85	1.0
<input type="checkbox"/> mint jelly	58	1.0
<input type="checkbox"/> pastry	58	1.0
<input type="checkbox"/> fruit jelly	53	1.0
<input type="checkbox"/> N/A	50	1.0
<input type="checkbox"/> apple juice	49	1.0
<input checked="" type="checkbox"/> pine juice	43	1.0
<input type="checkbox"/> chocolate	41	1.0
<input type="checkbox"/> lemon tea	40	1.0
<input type="checkbox"/> minerals	39	1.0
<input type="checkbox"/> muffin	36	1.0

Figure 5-22 Selecting all products, except pine juice, and clicking the AND NOT operator

As a result, the query changes as follows:

```
*:* AND -keyword::"Product"/"pine juice"
```



You see the various keywords other than “pine juice” in the Facets view, as shown in Figure 5-23. The result between Figure 5-20 on page 111 and Figure 5-23 is different, because this time, we select all products except “pine juice”.

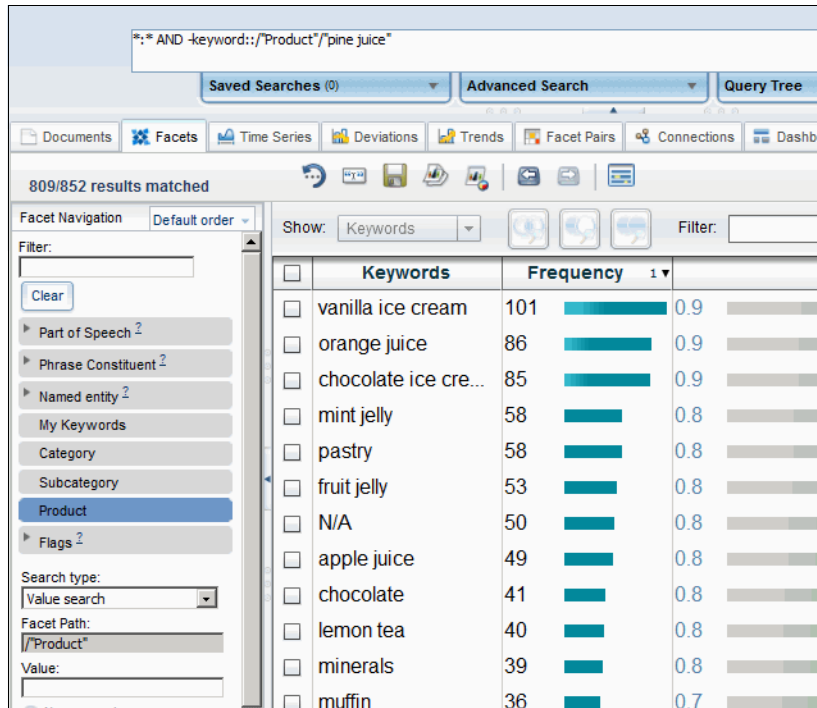


Figure 5-23 All products selected, except pine juice in the Facets view

Figure 5-24 shows the associated Query Tree in this example.

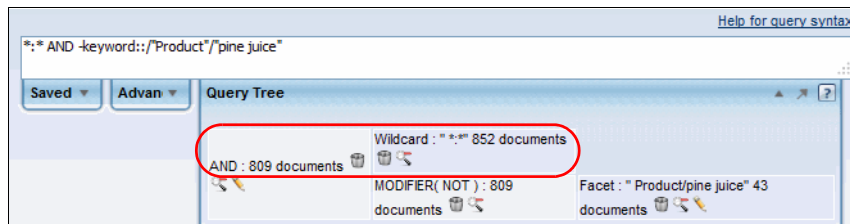


Figure 5-24 Query Tree: All products selected, except pine juice

In the left node of the Query Tree, you see an AND operator and the number of the documents found (809 documents) as a result of the following query:

```
*:* AND -keyword::/"Product"/"pine juice"
```

The NOT operator applied to the search query `keyword::/"Product"/"pine juice"` returns 809 documents, and the query `*.*` returns 852 documents. These results are aggregated with the AND operator, and Content Analytics returns 809 documents.

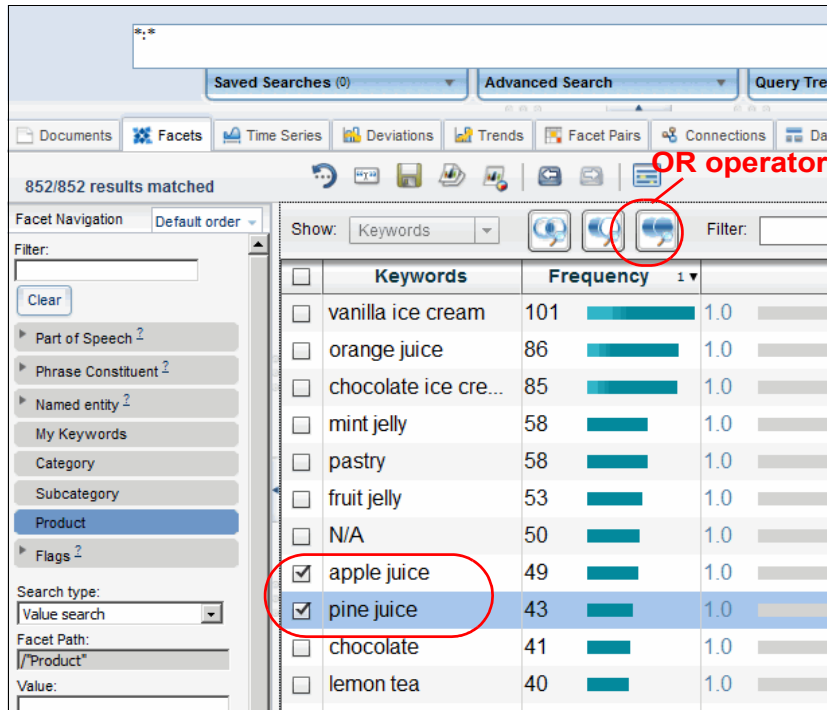
In summary, the following results occurred:

- ▶ The default query `*.*` returned 852 documents, which are the total number of documents in the collection.
- ▶ The query that limits the results to pine juice only (with `AND pine juice`) returned 43 documents.
- ▶ The query that excluded pine juice (with `AND NOT pine juice`) returned 809 documents.

The last two results reflect the total number of documents that are found in the collection.

## The OR operator

The example in this section shows how many documents are returned, including the apple juice or pine juice product-related information. From the Facet Navigation window (Figure 5-25), we select the **Product** facet and both **apple juice** and **pine juice**. Then, we click the **OR** operator.



The screenshot shows a search interface with a Facet Navigation window on the left and a table of results on the right. The Facet Navigation window has 'Product' selected under the 'Facets' section. The table of results has 'apple juice' and 'pine juice' checked. The 'OR operator' button is circled in red.

Keywords	Frequency	1
<input type="checkbox"/> vanilla ice cream	101	1.0
<input type="checkbox"/> orange juice	86	1.0
<input type="checkbox"/> chocolate ice cre...	85	1.0
<input type="checkbox"/> mint jelly	58	1.0
<input type="checkbox"/> pastry	58	1.0
<input type="checkbox"/> fruit jelly	53	1.0
<input type="checkbox"/> N/A	50	1.0
<input checked="" type="checkbox"/> apple juice	49	1.0
<input checked="" type="checkbox"/> pine juice	43	1.0
<input type="checkbox"/> chocolate	41	1.0
<input type="checkbox"/> lemon tea	40	1.0

Figure 5-25 Selecting apple juice or pine juice and clicking the OR operator

This operation changes the query as follows:

```
*:* OR keyword::/"Product"/"apple juice" OR keyword::/"Product"/"pine juice"
```

The query returns 852 documents, which is the same result if you search with the query \*:\* , as shown in the Figure 5-26.

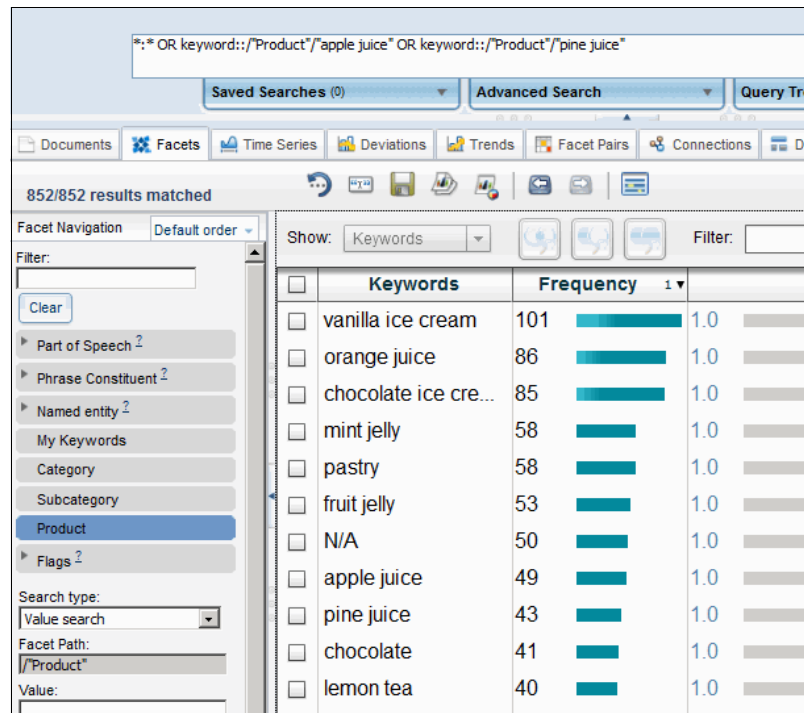


Figure 5-26 Apple juice or pine juice, using the OR operator, but all documents returned

Figure 5-27 shows the associated Query Tree in this example.

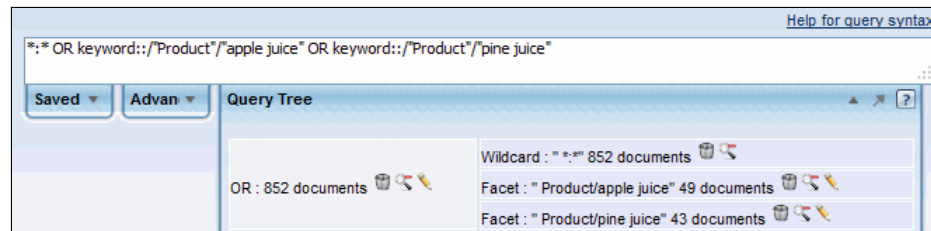


Figure 5-27 Query Tree with the OR operator

As you can see, the OR operator is applied for the selected facet search keywords `apple juice` and `pine juice`. It is also applied to the default query `*:*`. When the default query is included with the OR operator, Content Analytics returns the same result as the default query.

To get the documents that are only apple juice or pine juice-related, we must remove the default query `*:*` from the entire query. The easiest way is to remove the node that contains the default query `*:*` from the Query Tree. After we click the **Trash** icon on the right side of the default query `*:*` node (in Figure 5-27 on page 116), the node is removed from the query. Figure 5-28 shows the new Query Tree.

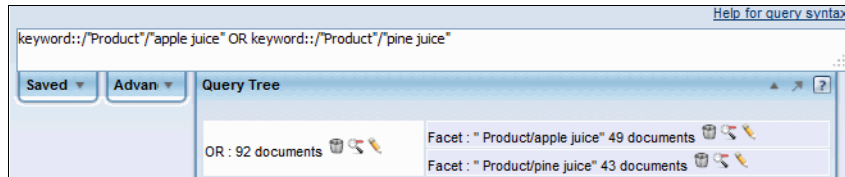


Figure 5-28 Query Tree changes after one condition is removed

Now the query changes to `apple juice` or `pine juice` only. You no longer see the default query node in the Query Tree. When we go back to the Facets view, a different result is displayed (Figure 5-29), as compared to the result shown in Figure 5-26 on page 116.

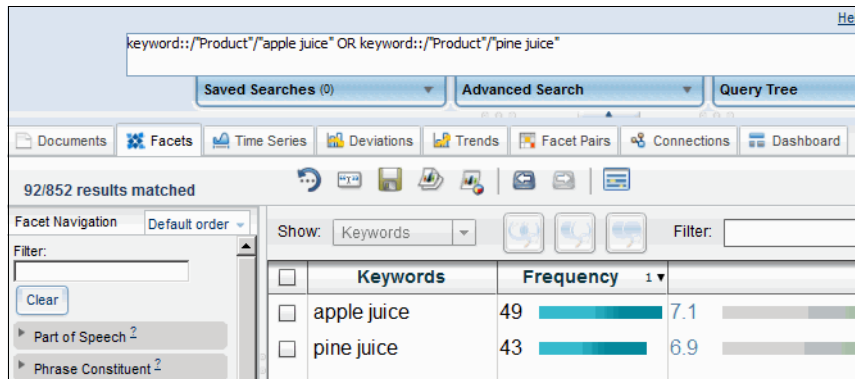


Figure 5-29 Apple juice or pine juice products using only the OR operator

## Applying search operators several times

In this example, we apply the search operators several times and see how the Query Tree changes. We want to drill down in the search results that we saw in “The OR operator” on page 115 from a different aspect. From the Facet Navigation window (Figure 5-30), we select the **Verb** facet and the keyword **leak**. Then, we click the **AND** operator.

The screenshot shows the IBM Watson Content Analytics interface. At the top, the search query is displayed as `keyword::/"Product"/"apple juice" OR keyword::/"Product"/"pine juice"`. Below the search bar, there are tabs for 'Saved Searches (0)', 'Advanced Search', and 'Query Tree'. The main navigation bar includes 'Documents', 'Facets', 'Time Series', 'Deviations', 'Trends', 'Facet Pairs', 'Connections', and 'Dashboard'. The search results show '92/852 results matched'. On the left, the 'Facet Navigation' window is open, showing a tree structure with 'Part of Speech' expanded to 'Verb', which is selected. Below the facet navigation is a table of search results. The table has columns for 'Keywords', 'Frequency', and 'Score'. The row for 'leak' is highlighted in blue, and its checkbox is checked. A red circle highlights the 'AND operator' button (a blue circle with a white 'A') in the toolbar above the table. A red arrow points from the text 'AND operator' to this button.

Keywords	Frequency	Score
<input type="checkbox"/> save	9	3.4
<input type="checkbox"/> drink	14	2.8
<input checked="" type="checkbox"/> leak	45	2.6
<input type="checkbox"/> hold	7	1.2
<input type="checkbox"/> be	71	0.9
<input type="checkbox"/> have	13	0.9
<input type="checkbox"/> like	9	0.8
<input type="checkbox"/> leave	5	0.5
<input type="checkbox"/> buy	19	0.5
<input type="checkbox"/> cause	3	0.5
<input type="checkbox"/> make	6	0.4

Figure 5-30 Selecting a keyword and clicking the AND operator button

This operation changes the query to find all documents that have either “apple juice” or “pine juice,” and that have the word “leak”:

```
(keyword::/"Product"/"apple juice" OR keyword::/"Product"/"pine juice")  
AND keyword::/"Part of Speech"/"Verb"/"leak"
```

As shown in Figure 5-31, the query returns 45 documents.

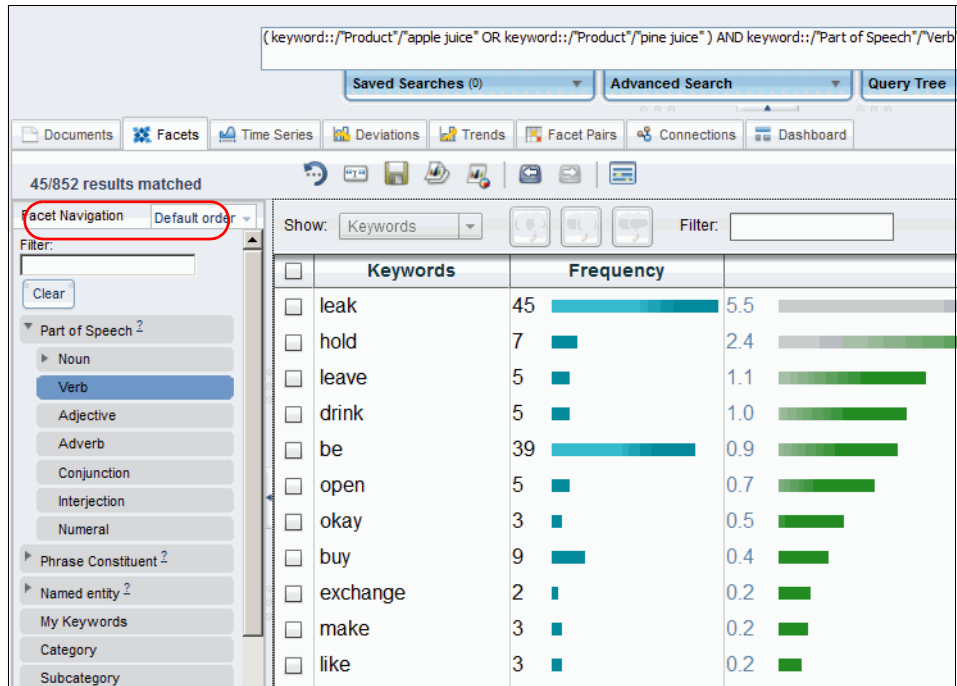


Figure 5-31 Combination operators: (apple juice OR pine juice) AND leak

Notice that the second AND operator is applied at the end of the existing apple juice or pine juice query:

keyword::/"Product"/"apple juice" OR keyword::/"Product"/"pine juice"

The existing query is set as one group with parentheses. Figure 5-32 shows the associated Query Tree in this example.

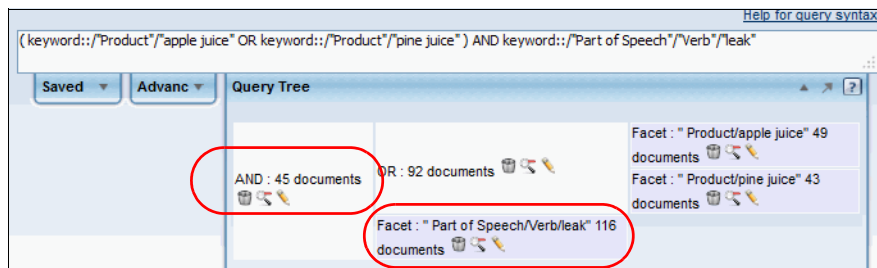


Figure 5-32 Query Tree: (Apple juice OR pine juice) AND leak

The AND operator is applied to the node, which returns the selected facet values “apple juice” or “pine juice.”

From the Query Tree, in summary, the following results occur:




- ▶ 49 documents for the apple juice product
- ▶ 43 documents for the pine juice product
- ▶ 92 documents for either apple juice or pine juice products
- ▶ 116 documents with a leak problem
- ▶ 45 documents with a leak problem that applies to apple juice or pine juice

You can apply the search operators for the selected nodes. In this example, we do not explicitly select the specific node in the Query Tree. If you do not select the specific node in the Query Tree, the search operator is applied to the “root” node on the left side. However, if you select a specific node and apply the search operator, the search operator is applied to the selected node. Thus, you can build the complex query further by iterating the process, such as selecting a specific node and applying the search operator several times.

### 5.3.4 Editing the Query Tree

With the Query Tree, you can also edit the query to find the useful query for your analysis. Table 5-2 defines the icons that are displayed in the Query Tree.

Table 5-2 The Query Tree icons and description

Icon	Description
	You click this icon to exclude a selected term from the query. When you click this icon, the AND NOT operator is added to the Query Tree. This icon is useful when you quickly examine a query that does not have the term.
	You click this icon to edit the keyword or the BOOLEAN operator. This icon is useful when you change the operator type from AND to OR, and vice versa. You can also edit the keyword in the node.
	You click this icon to delete the keyword or node from the Query Tree.

#### Excluding a node

When you build the Query Tree, each term is represented as a node. When you want to exclude one of the query terms, click the **Exclude** icon. As a result, the selected node is added with the AND NOT operator. For example, if you click the **Exclude** icon associated with the keyword: `:"Part of Speech"/"Verb"/"leak"`



node, as shown in Figure 5-32 on page 119, the Query Tree changes as shown in Figure 5-33.

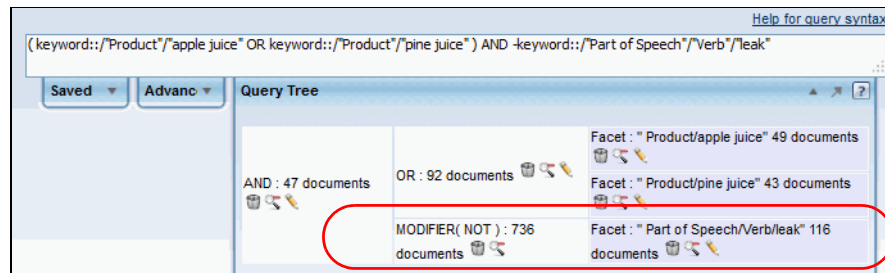


Figure 5-33 Excluding a keyword node in the Query Tree

The search results are immediately displayed in your view.

## Editing a node

You can edit the operator type or the keyword without building the query from the beginning. When you click the **Edit** icon for a keyword node, a selection list is displayed, as shown in Figure 5-34.

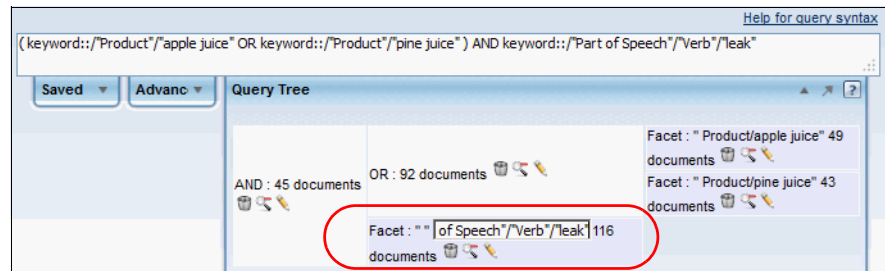


Figure 5-34 Editing a keyword node in the Query Tree

For example, you can modify the keyword “leak” to be a different keyword found in the Verb facet.

You can change the search operator by using a drop-down field, as shown in Figure 5-35 on page 122.

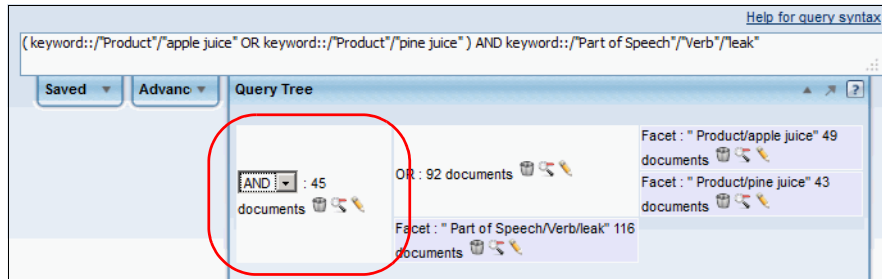


Figure 5-35 Editing search operator node in the Query Tree

The query change is reflected immediately and the search results are updated in your view. You can review the query results for various queries without building the query from the scratch.

### Deleting a node

When you want to delete a keyword, click the **Delete** icon next to the particular keyword. This operation deletes the selected node and its child nodes (if they exist). Make sure that you do not use the keyword anymore before you delete it.

For example, consider when you delete the OR node as shown in Figure 5-36.

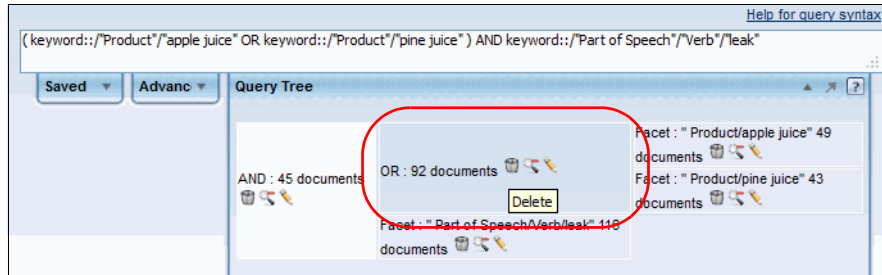


Figure 5-36 Deleting the search operator node in the Query Tree

In this case, the child nodes (`keyword::/"Product"/"apple juice"` and `keyword::/"Product"/"pine juice"`) of the selected OR node are deleted, as shown in Figure 5-37.

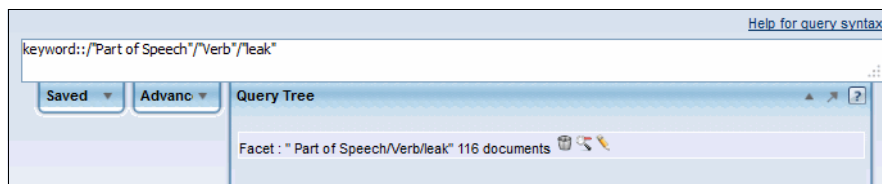


Figure 5-37 After the search operator node is deleted in the Query Tree

If you want to exclude a keyword from the query, click the **Exclude** icon next to that particular keyword. Excluding a node in the Query Tree can be used instead of deleting the node.

The Query Tree is helpful when you analyze the data with queries, especially when you examine the data from different aspects. You can modify the query by clicking icons on the right side to correct the query. The change is reflected immediately to your query statement, and you can continue editing the query until you find the useful query for your analysis.

For more information, review the IBM Content Analytics Information Center at the following address, and search on *searching collections and using the Query Tree*:

<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/index.jsp>

## 5.4 Query builder

By using the two different methods provided, the Query Builder and the Query Tree, you can build many types of analytics and mining queries from the content of documents in a collection and quickly evaluate the usefulness of the query results by seeing the number of documents that match. The Query Builder helps you to build the query by selecting keywords or facets for the selected documents and adding the built query to an existing query or selected query node in the Query Tree.

In addition, a preview of the selected documents is displayed with the Query Builder. You can quickly select keywords from the preview, add the condition to the existing query, and verify the results immediately. After the query is built with the Query Tree, you can interact with the Query Builder to confirm the structure of the query, and you can modify the query to find further insight.

This section explains how to use the Query Builder and how to interact with the Query Tree.

### 5.4.1 Accessing the Query Builder

To use the Query Builder, the “Build queries with the query builder” application user role must be enabled. By default, this user privilege is not enabled. You must enable the feature from the administration console explicitly.

After you enable the Query Builder feature, the Query Builder icon is displayed in the Actions column within the Documents view, as shown in Figure 5-38.

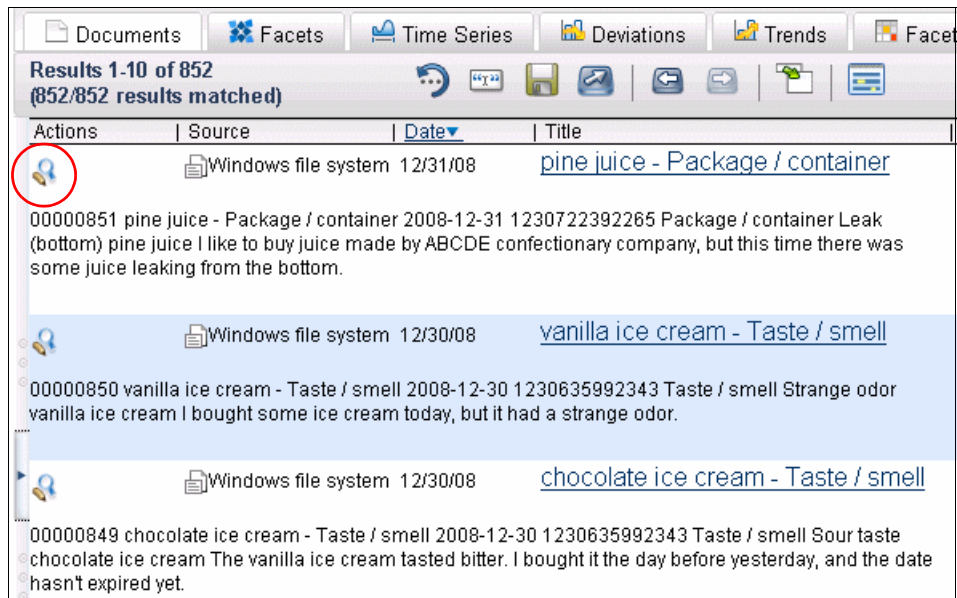


Figure 5-38 Query Builder icon appears in Actions column

When you click the **Actions** icon, a Query Builder window (Figure 5-39 on page 125) opens.

## 5.4.2 Features of the Query Builder window

The Query Builder window has the following main areas (Figure 5-39):

- ▶ Query building area
- ▶ Document preview area
- ▶ Facet list area

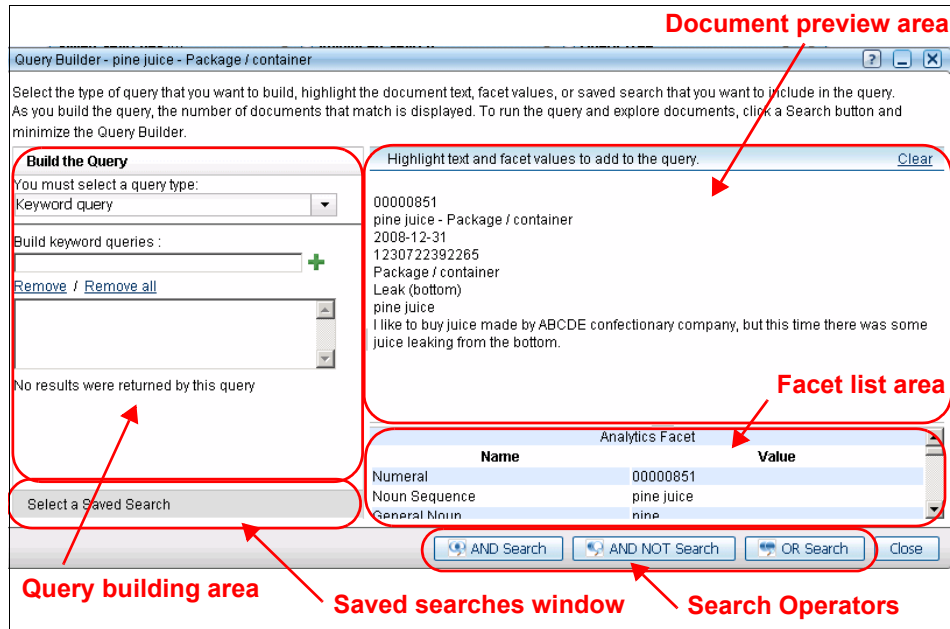


Figure 5-39 Query Builder main window

You can build the following query types from the Query Builder:

- ▶ Keyword query
- ▶ Phrase query
- ▶ Facet query
- ▶ Field query
- ▶ Proximity query
- ▶ Exact match query
- ▶ Base form match query
- ▶ Fuzzy query
- ▶ Boost query
- ▶ Parametric range query
- ▶ Date query

You select the query type and drag the keyword or facet in the document from the right area of the Query Builder (either from the document preview area or the facet list area). Then, as shown in Figure 5-40, the query keyword or facet is added immediately, and you see the result count.

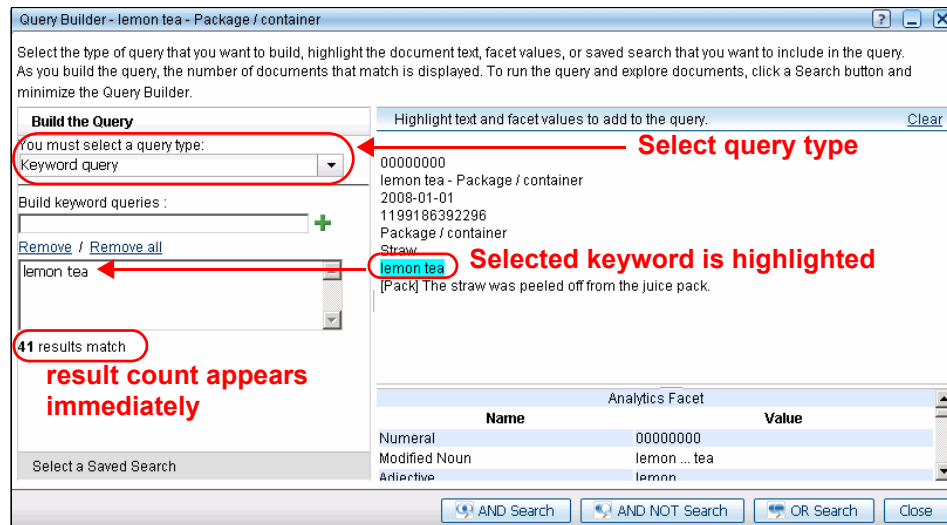


Figure 5-40 Selecting keywords in the document on the Query Builder

**Result count in the query building area:** The result count in the Query Builder is the result of the current query that you built using the Query Builder. The query that you already issued in the content analytics miner (such as in the Documents view) is not considered at this point.

You can use *Saved Searches* to build the query. In this case, you click the **Select a Saved Search** area below the query building area, as shown in Figure 5-39 on page 125, to expand the Saved Searches window and choose a saved search to build the query from. Figure 5-41 on page 127 shows how the Query Builder presents the list of saved searches. The saved query is displayed in the Query area. You can use the saved query or add the saved query to the selected node in the Query Tree with the search operator.

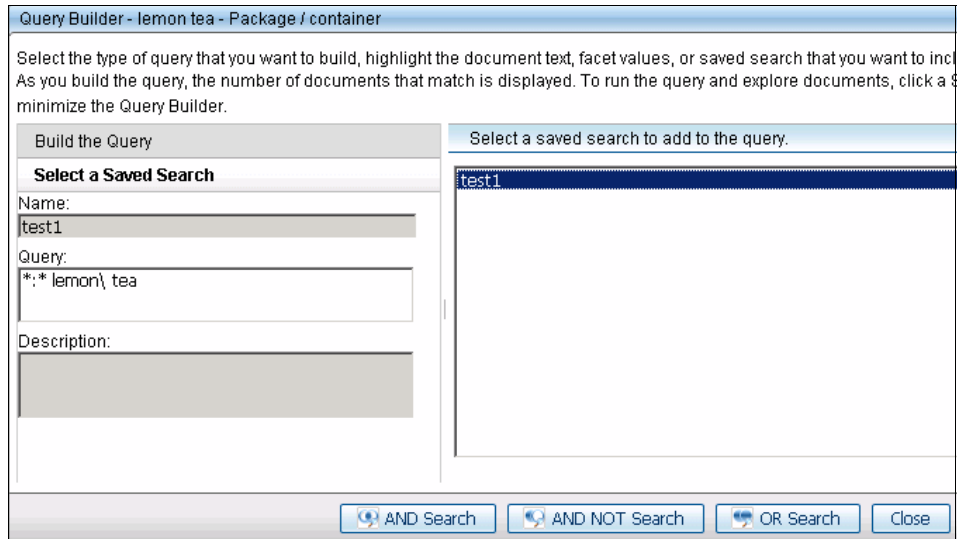


Figure 5-41 Selecting a saved search in the Query Builder

When you decide on the query to be examined, select the search operators (AND, AND NOT, or OR) that you want to apply, at the bottom of the Query Builder window. Then, your search results are automatically updated. After you select the Boolean search operator, the Query Builder window is minimized, as shown in Figure 5-42. You can open the Query Builder window again by selecting the **Expand this area** icon.

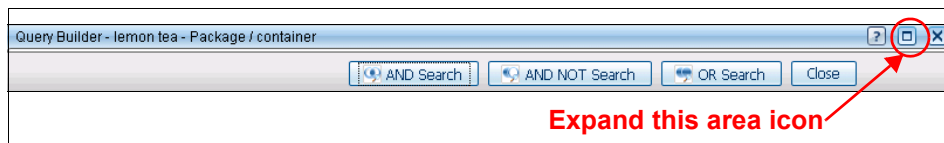


Figure 5-42 After the query is issued from the Query Builder

In the search box area, the entire query and the Query Tree structure on the Query Tree tab are displayed, as shown in Figure 5-43. The selected keyword is highlighted in the summary of each document in the Documents view.

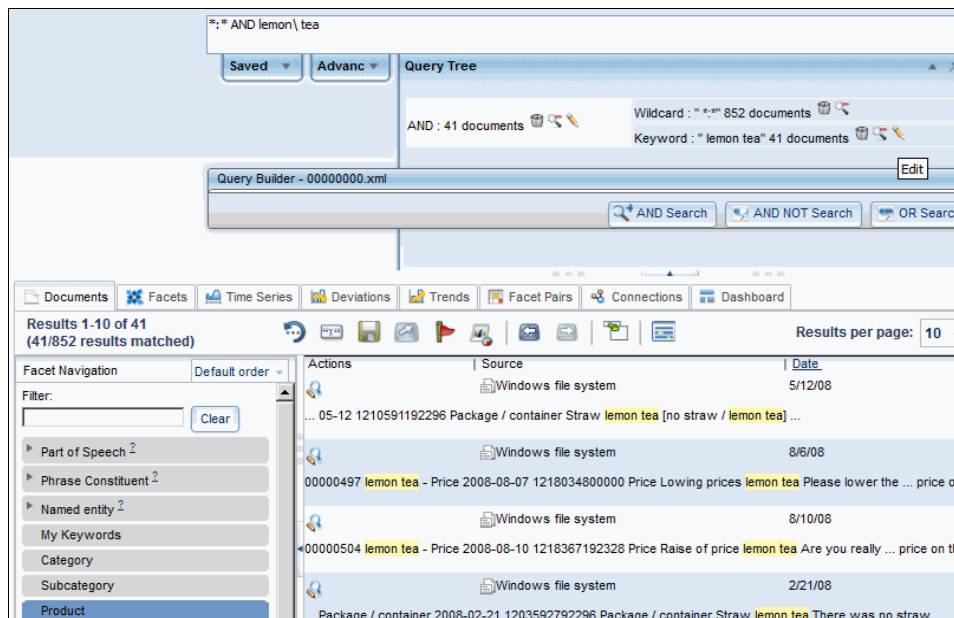


Figure 5-43 The query with the Query Tree and selected keyword highlighted in results

After you issue the query, you can go back and forth to the Query Builder to build more complex queries. You can add the query to the selected node in the Query Tree, and you can operate the query from the Query Tree.

### 5.4.3 Using the Query Builder

This section shows examples of how to use the Query Builder. You can build a query with the Query Builder by using the following steps. Then, you repeat the steps until you decide whether the query provides the necessary information for your analysis:

1. Select a document from the Documents view and open the Query Builder window.
2. Build a query in the Query Builder. Select the query type from the selection box, and select the keyword that you are interested in from the Document preview area.
3. Add the query with one of the search operators and decide whether it is what you want:



- If you need to modify the query, you can modify the query either from the Query Builder or from the Query Tree.
  - When you want to add the built query to a certain node in the Query Tree, select the query node in the Query Tree and issue the query with the search operator.
4. Repeat steps 1 – 3 to build the query that you want. You can use the Query Tree to help you build the query.
  5. After you complete the query building, save the query for reuse in the future, such as for document export or to build another query.

### **Selecting a document and opening the Query Builder**

Consider an example where a specific document contains keywords that you want to use to find similar information. The document contains information about the “lemon tea” product. Select the document in the Documents view, and open the Query Builder window, as shown in Figure 5-40 on page 126.

### **Building and issuing a query**

After you open the Query Builder, a preview of the selected document is displayed in the Document preview area or the Facet list area. Select a query type from the drop-down list, and drag the keyword in the document.

In this example, we select **Facet query** as the query type, and select **General Noun: package** from the Facet list. We select another **General Noun: pack** from the Facet list, as shown in Figure 5-44 on page 130.

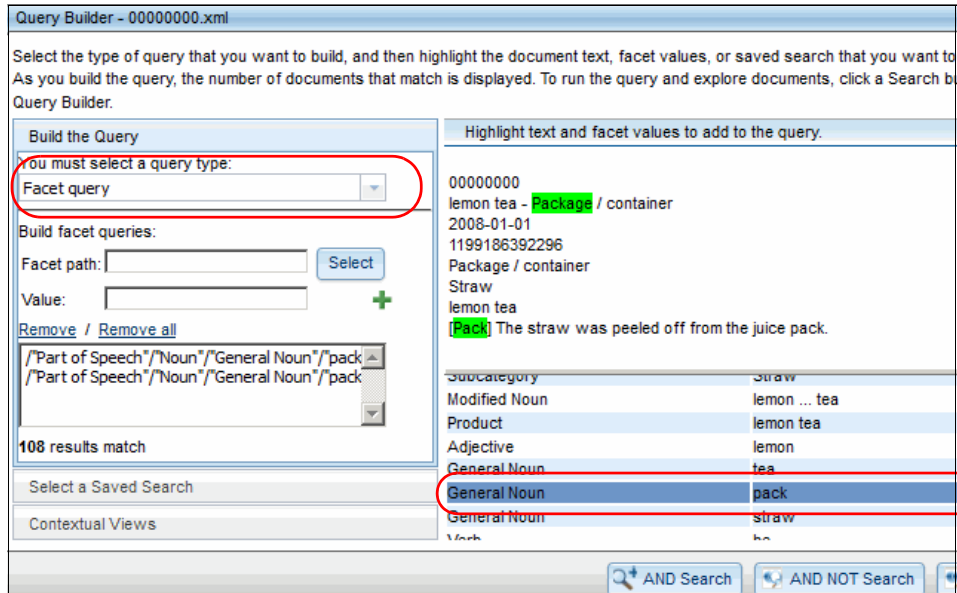


Figure 5-44 Selecting the Facet query with two keywords from the Facet list

After selecting the keywords, issue the query with the AND search operator. The query results are updated, and 108 documents match this node in the query, as shown in Figure 5-45.

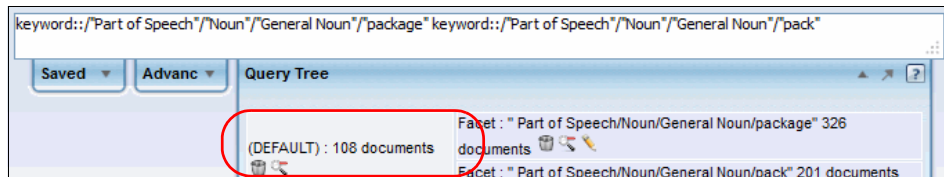


Figure 5-45 The Query Tree for the query with the AND operator

## Adding a query to a selected node in the Query Tree

You can also add a query to the selected node in the Query Tree to narrow down the data further. When you do not explicitly select the node in the query, the query is added to the entire query.

For example, perform the following steps to add a query to the entire node:

1. Remove the previous condition by clicking the **Remove all** link in the Build the Query area.
2. In the query text area, type Product: lemon tea, as shown in Figure 5-46.
3. Click **AND NOT Search** to add the Product: lemon tea keyword with the AND NOT search operator. This option shows how many documents are not regarding the lemon tea product.

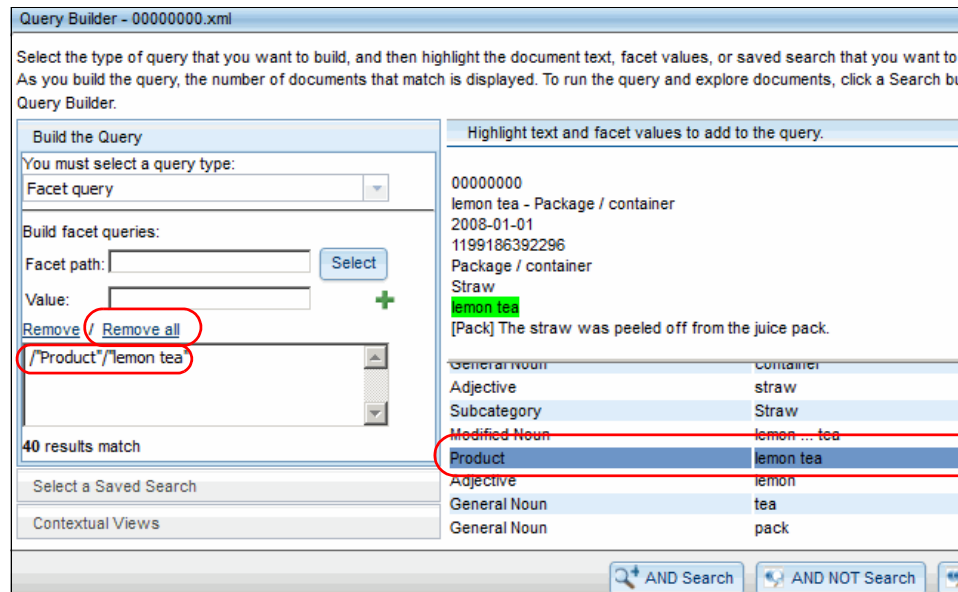


Figure 5-46 Selecting the facet query after flushing the existing keyword

In this case, we do not select any node in the Query Tree. The query is added to the entire query, and the new query returns 97 documents, as shown in Figure 5-47.

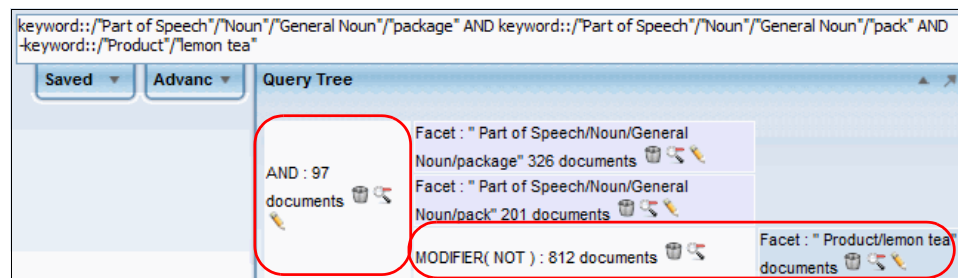


Figure 5-47 The Query Tree for the query with the AND NOT operator

The results indicate that the documents that mention the “package” and “pack” nouns might be related to products other than the lemon tea product. The new query produced a result set of 97 documents, which means that only 11 of 108 documents are related to the lemon tea product.

When we look at the search result in the Documents view, a document that contains the term “orange juice” is displayed at the top of the search result, as shown in Figure 5-48. We wonder if the documents are mostly related to the product “orange juice”, or if those documents are for other products.

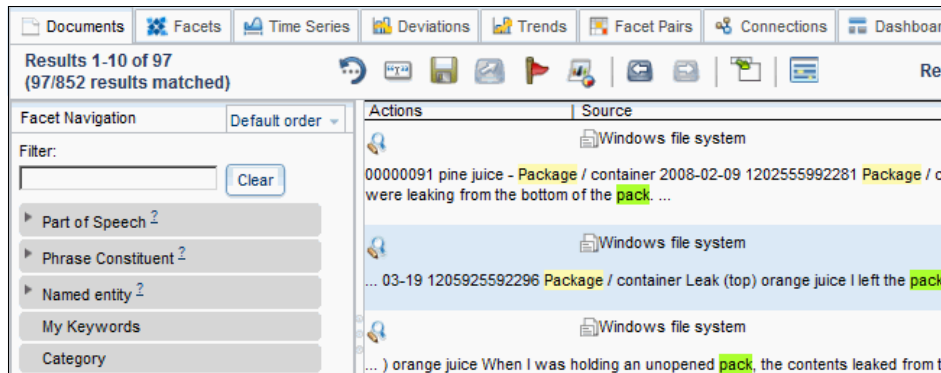


Figure 5-48 The query result in the Documents view

To confirm how many documents are returned if the product is neither “lemon tea” or “orange juice”, follow these steps:

1. Select the **Product: lemon tea** node in the Query Tree (Figure 5-49).

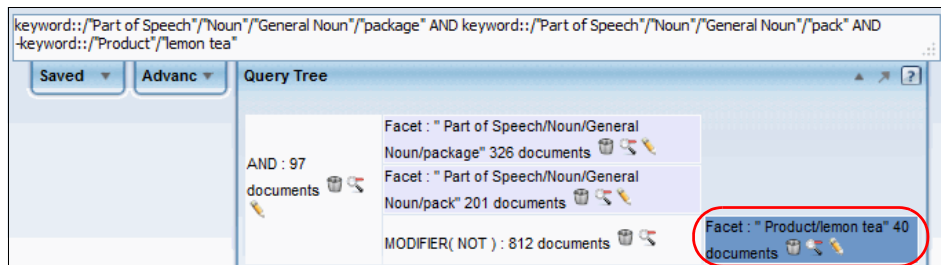


Figure 5-49 Selecting a specific node in the Query Tree

2. Close the Query Builder for the first document (the document related to the “lemon tea” product), and open the Query Builder for the first document in the search results. This document mentions the “orange juice” product.

- In the Query Builder window, select **Facet query** for the query type, and select the keyword **Product: orange juice** from the Facet list, as shown in Figure 5-50.

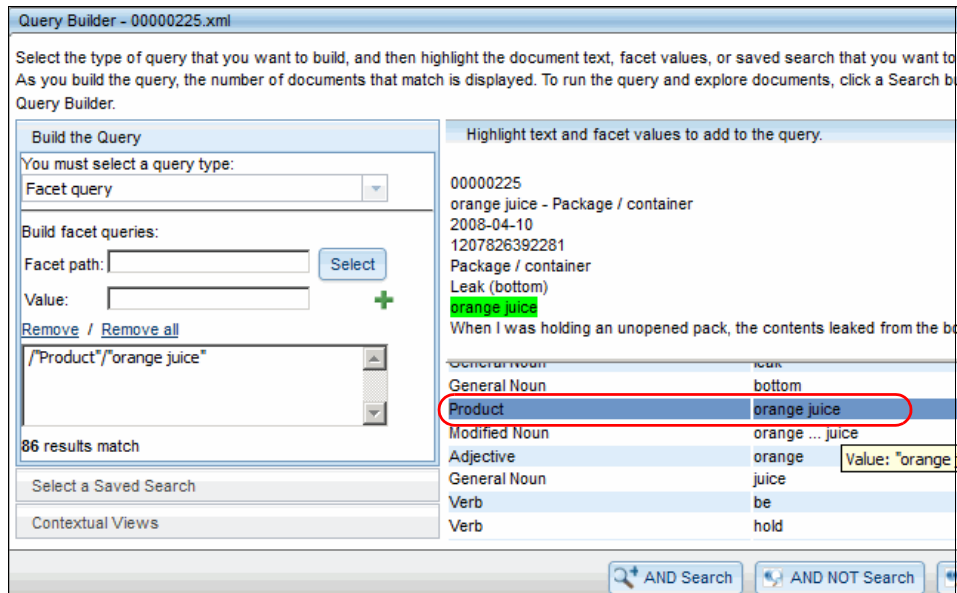


Figure 5-50 Selecting another keyword “orange juice”

- Issue the query with the OR search operator to add the query to the selected node, as shown in Figure 5-51.

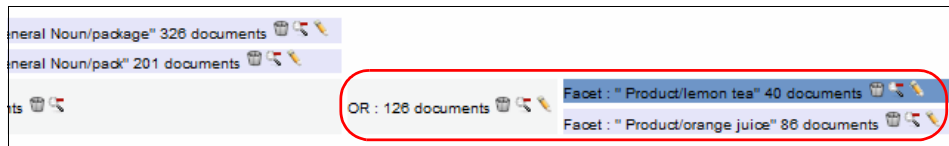


Figure 5-51 Adding the OR query to the selected node

Forty-three documents match the query and are returned in the query results. See Figure 5-52.

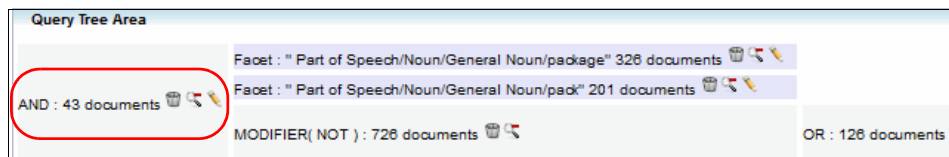


Figure 5-52 Returned query result with OR query added to the selected node

In our example, the query becomes “Product: lemon tea” keyword OR “Product: orange juice” keyword. However, we previously added the NOT operator before “Product: lemon tea.” Thus, the query returns 43 documents that represent documents that mention “package” and “pack,” but that are not for the products “lemon tea” nor “orange juice.”

Now that you have the result set, you can view the documents in the Documents view, or add other keywords to the query to gain insight into the content. When we look at the document in the Document view, we see the keyword “leak.”

To investigate further, follow these steps in the Query Builder window (Figure 5-53):

1. Select the **Product: pack** node in the Query Tree.
2. Change the query type to **Keyword query**.
3. In the keyword field, type Leak.
4. Click the **AND Search** button.

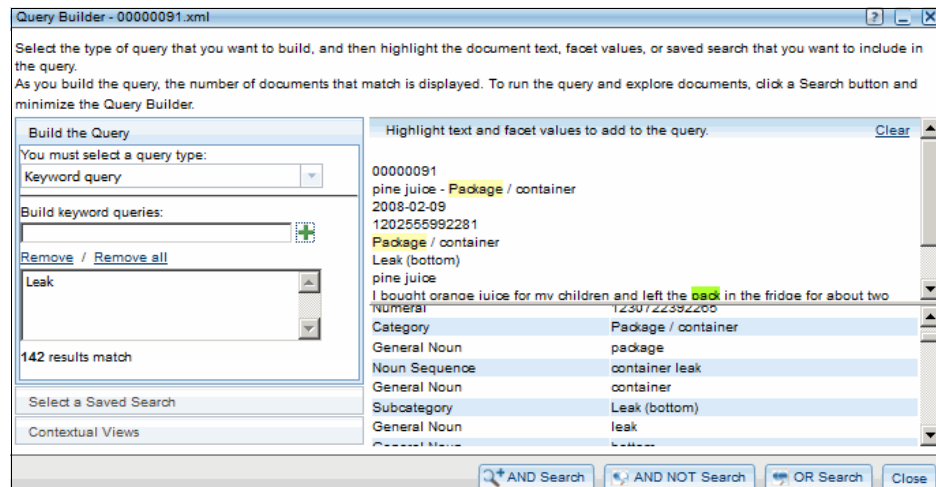


Figure 5-53 Selecting another keyword “Leak” as a keyword query

As a result, the query returns 40 documents, as shown in Figure 5-54.

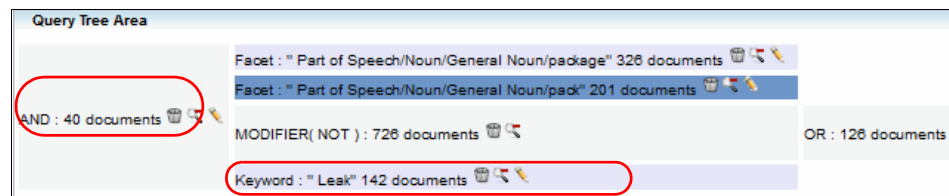


Figure 5-54 Query Tree with the addition of “Leak” with the AND operator in the query

Based on this insight, the results show that 40 of the 43 documents results are related to the term “leak.”

To view the only documents, based on the query, that do not mention the term leak, click the **Exclude** icon at the node “leak” in the Query Tree, as shown in Figure 5-55. As a result, the three documents that are displayed (Figure 5-55) mention “pack” and “package,” do not mention “leak,” and are not “Product: orange juice” nor “Product: lemon tea.”

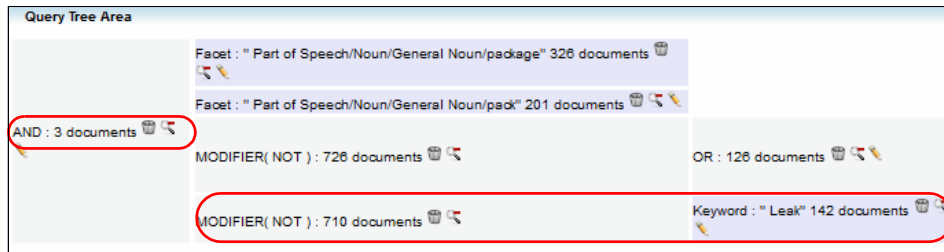


Figure 5-55 The Query Tree after clicking the Exclude icon of the “leak” node

To review the three resulting documents in detail, open the Documents view.

#### 5.4.4 Preferred practice for using the Query Builder and Query Tree

As shown in 5.4.3, “Using the Query Builder” on page 128, you can interactively select a keyword used in the document by using the Query Builder and validating the query by using the Query Tree. One reason to use the Query Builder is that you can add the keyword to the query easily from the document preview. You can use the Query Builder to select a keyword from a document in the result set of the current built query or from the initial selected document. In addition, you can add a query to a selected node in the Query Tree.

The Query Builder and Query Tree aid with building a complex query so that you can go back and forth between the Query Builder and the Query Tree.

### 5.5 Rule-based categories with a query

When you build a query to find insight, you might want to use the query that you built to categorize the documents consistently. With rule-based categories, you can define a category and view documents by using a facet. This section explains how to configure the rule-based categories and how to use the rule-based categories.

## 5.5.1 Enabling the rule-based categories feature

To use the rule-based categories feature, you must enable the rule-based categorization type. Rule-based categorization is enabled for the collection by default.

**Document clusters:** To use document clustering, select **Rule-based and Document clusters** as the categorization type when creating the collection. For further information about document clustering, see 9.6, “Document clustering” on page 330.

In addition, if you want to add the query that you built from the content analytics miner, you must enable both the “Add rules to categories” and “Rebuild the category index” application user roles. By default, these user privileges are not enabled. Therefore, you must explicitly enable the roles by using the administration console.

**Configuring the application user role:** See “Configuring application roles” in Appendix A of the previous version of the book for further details about configuring the application user roles. The previous version of the book can be downloaded from the additional material associated with this book. See Appendix B, “Additional material” on page 567 for details.

After you enable the “Add rules to categories” feature, an icon is displayed in the content analytics miner toolbar (Figure 5-56).

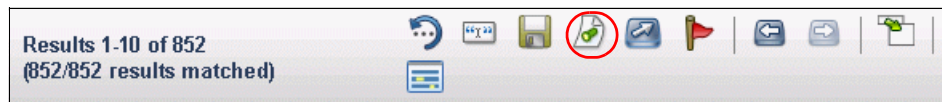


Figure 5-56 Adding the current rule as a new category rule icon

## 5.5.2 Configuring rules for rule-based categories

With Content Analytics, you can define a rule category that is either based on Uniform Resource Identifier (URI) patterns or Document content rules. You can use the query that you build to define the Document content rules and easily add a category rule from the content analytics miner.



You can configure the category tree from scratch in the administration console. You can also modify the existing rule-based categories or delete categories. To access and configure a category tree, follow these steps:

1. From the administration console, expand your collection. In the “Parse and Index” section, expand the **Edit - Configure** button and select the **Global processing** option.
2. Select the **Rule-based categories** option (Figure 5-57).

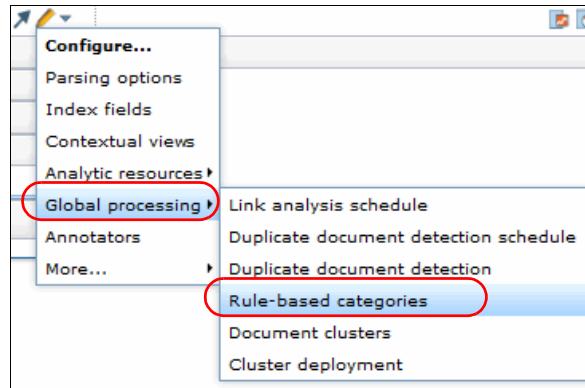


Figure 5-57 Rule-based categories option

3. The category tree is now displayed (Figure 5-58 on page 138).

Root

**Add a new category**

\*Category path:

Category name:

**Add**

**Edit a category**

Category path:

Category name:

Category order:

Rules:

**Apply**

**Remove a category**

**Remove**

**OK** **Cancel**

Figure 5-58 Category tree view in the administration console

- To add a category, in the “Add a new category” section (Figure 5-59), complete the category path and name, and click **Add**.

**Add a new category**

\*Category path:

Category name:

**Add**

Figure 5-59 Adding a category

After you add a category, you see the category in the category tree area (Figure 5-60).

5. To edit or delete the category, select the category in the category tree and perform the wanted action. In our example, we define the Juice and Tea subcategories under the Drink category. The rule for each category is not defined at this point.

The screenshot shows a category management interface. On the left is a category tree with 'Root', 'Drink', 'Juice', and 'Tea'. The main area has three sections: 'Add a new category' with fields for 'Category path' and 'Category name' and an 'Add' button; 'Edit a category' with fields for 'Category path' (set to 'Drink'), 'Category name' (set to 'Drink'), 'Category order' (a list with 'Juice' and 'Tea'), and 'Rules' (an empty list), with 'Apply' and 'Remove' buttons; and 'Remove a category' with a 'Remove' button. At the bottom are 'OK' and 'Cancel' buttons.

Figure 5-60 Adding, editing, or removing the category in the category tree

6. After you finish configuring the category tree, click **OK**.

**Applying the category changes:** For category changes to take effect, select the appropriate option:

1. If the document cache is enabled, you need to redeploy the analytic resources, then either run a full rebuild of the index or recrawl or reimport all documents so they can be indexed again.
2. If the document cache is not enabled, you must deploy the analytic resources, and then recrawl or reimport all documents so they can be indexed again.

After you define the new categories and rebuild the index, the defined category is shown as a new facet in the Facet Navigation pane (Figure 5-61). In our example, we added a Drink category with two subcategories, Juice and Tea. As a result, the Drink facet is shown with two child facets, Juice and Tea.

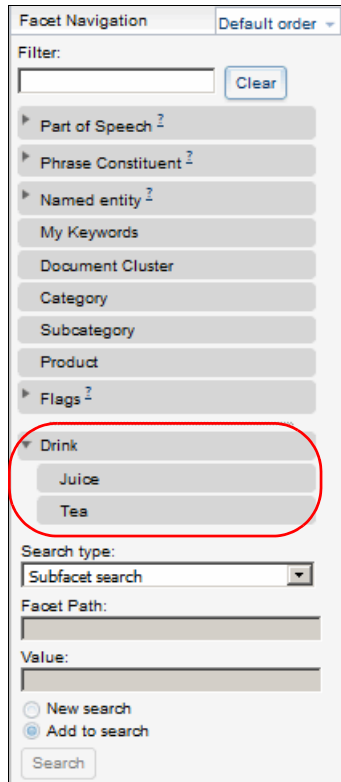


Figure 5-61 Facet Navigation after the rule-based category is added

### 5.5.3 Adding the current query as a category rule

As explained earlier, you can define a category from the administration console. To define a category rule, you must know the details of the rule to add it. You might not know your wanted category rule until you start to analyze your content by using the content analytics miner.

With Content Analytics, you can define the category rule based on a query that you want to use for document categorization within the content analytics miner. This feature of adding the query to the category rule within the content analytics miner is useful because it eliminates the need to go back to the administration console to define the rule.

For example, consider a case where you want to see the documents that are related to the term “juice.” in the Facets view, You filter the product by the term “juice” and add the filtered keywords to the query with the AND operator. Example 5-1 shows the resulting query for this example.

*Example 5-1 The query filtered by the term “juice” in the Product facet*

---

```
keyword::/"Product"/"orange juice" OR keyword::/"Product"/"apple juice"
OR keyword::/"Product"/"pine juice" OR keyword::/"Product"/"apple juice
(bottle)" OR keyword::/"Product"/"peach juice" OR
keyword::/"Product"/"orange juice (bottle)"
```

---

The query returns 202 documents, as shown in Figure 5-62.

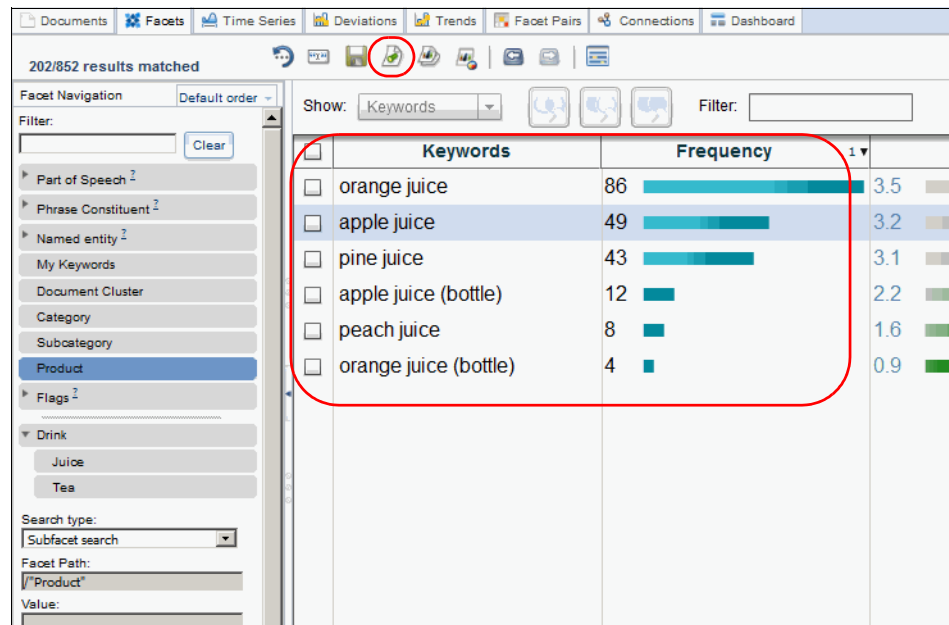


Figure 5-62 The 202 documents returned by the query

To add the query as a category rule, follow these steps:

1. Click the **Add the current query as a category rule** icon (circled in Figure 5-62).
2. In the Add the Current Query as a Category Rule window (Figure 5-63 on page 142), complete these steps:
  - a. Verify that the current query that is shown is correct.

- b. Select the category that you want to use. For our example, we click the **Juice** facet.
- c. Type the rule name. We type Juice in the Rule name field.
- d. Click **Add Rule** to add the current query as a category rule for the selected category.

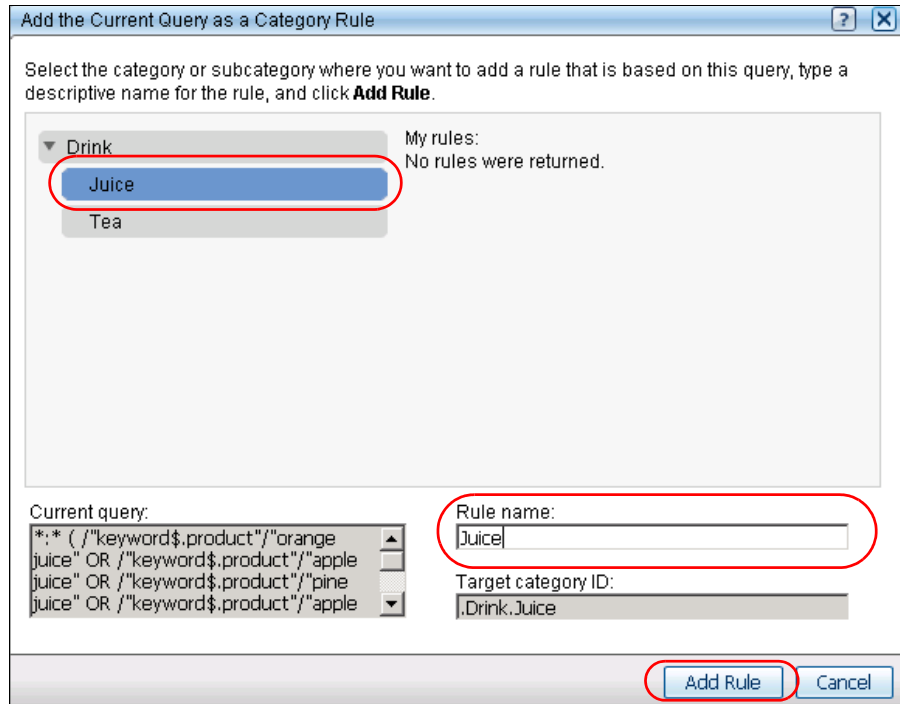


Figure 5-63 Adding the current query as a category rule

3. In the message window that opens (Figure 5-64), click **Rebuild Categories** to start the document categorizer to rebuild the categories.

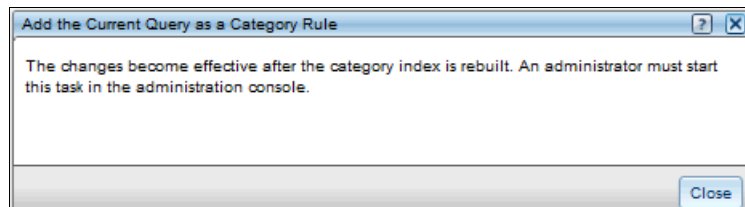


Figure 5-64 Starting to rebuild the categories after adding the query as a category rule

**Rebuilding the categories:** Make sure the parse and index process of the collection is up and running.

To monitor the progress of the Rule-based categories from the administration console, expand the **Global Processes** section under **Parse and Index** area for your collection, and review the status of the rebuild. Figure 5-65 shows the status of the progress.

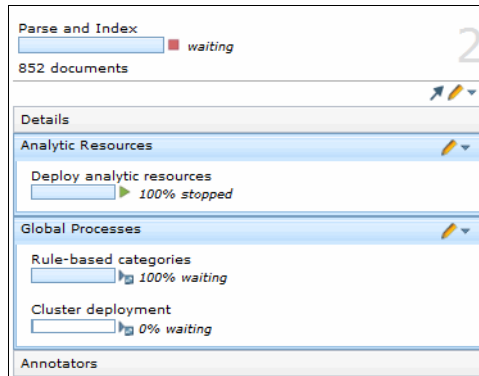


Figure 5-65 Checking the Rule-based categorization status on the administration console

After the document categorizer process is completed, the new category facets are displayed in the content analytics miner (Figure 5-66 on page 144).

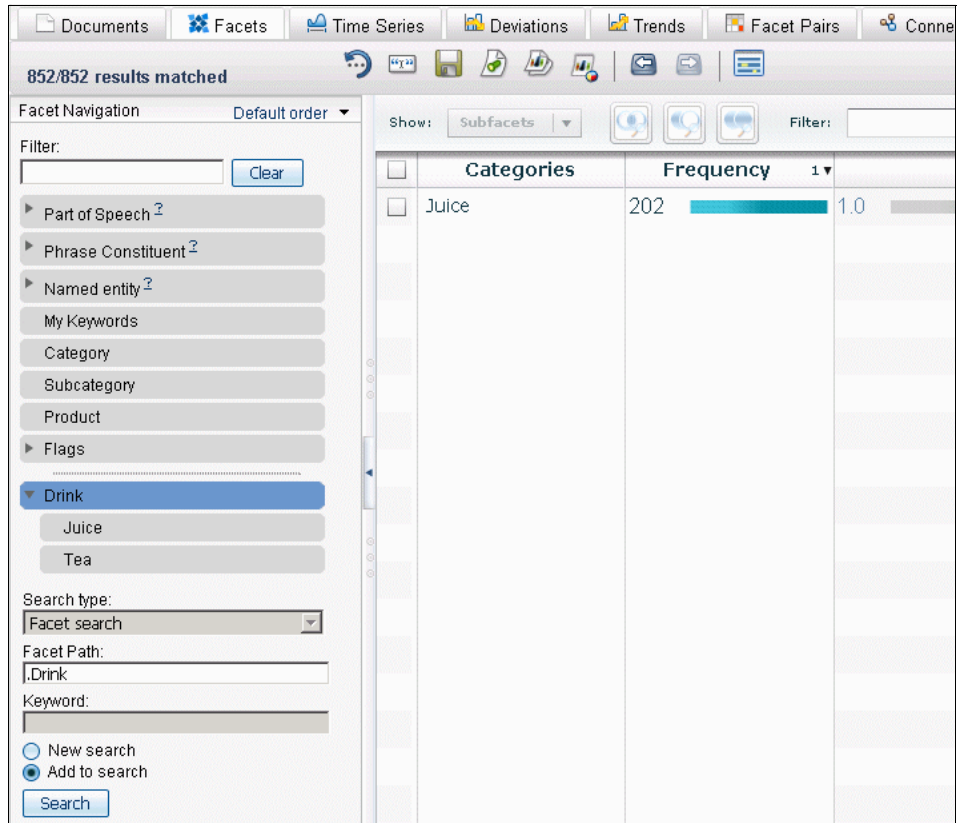


Figure 5-66 New categories returned

In our example, we added the query shown in Example 5-1 on page 141 as a category rule to the Juice category. The result is that 202 documents make up the Juice category. This result set is the same as searching for the query described in Example 5-1 on page 141. However, the column label is different for the two situations. The column label is “Keywords” for the direct query search (Figure 5-62 on page 141), and it is “Categories” when using rule-based categories (Figure 5-66).

To review or modify a rule-based category, follow these steps:

1. From the administration console, expand your specific collection. Then, expand the **Global Processes** section under the **Parse and Index** area.
2. Select the **Rule-based categories** option. You are presented with the rule-based categories windows.



- Expand the category tree until you see the specific category that you want to modify. In this example, we select the *Juice* category (Figure 5-67).

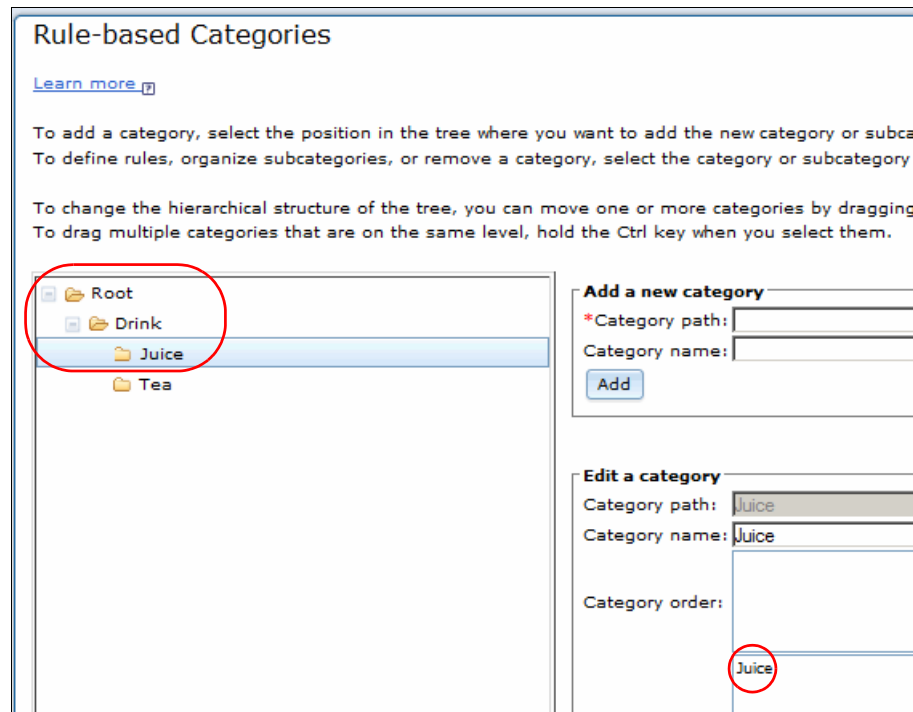


Figure 5-67 Editing the Rule-based Categories configuration

- Select the **Edit button** to modify the rules for that category (Figure 5-67). You are presented with the list of defined rules for this category (Figure 5-68).

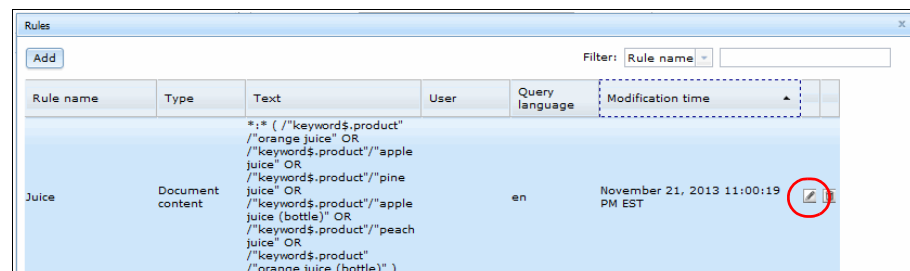


Figure 5-68 Editing the rules defined for a category

- Select the specific rule to modify by using the **Edit button** to the right of the rule. In our example, we edit the *Juice* rule (Figure 5-68).

- The detailed rule configuration can then be reviewed and modified (Figure 5-69).

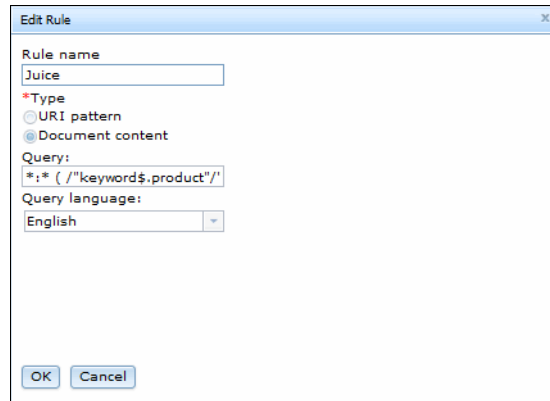


Figure 5-69 Editing the rule configuration

In conclusion, you can add the query as a category rule and use the category to find further insight into a set of documents. To build a query, you can use the Query Tree or Query Builder functions to interactively perform your analysis.

For more information about *rule-based categories*, review the IBM Content Analytics Information Center at the following address, and search on *rule-based categories*:

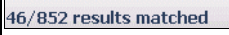
<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/index.jsp>








## 5.6 Common view features





Content Analytics provides several features that are common to all views in the content analytics miner. These features are on the menu bar under the view tabs. Except for the Search result counts label, each feature is identified and activated by its own icon.

Table 5-3 presents a description of each common view feature.

Table 5-3 Common label and icons in the analysis window

Label and icon	Description
Search result counts 	At all times, Content Analytics shows the number of documents that match the current query out of the total number of documents in the entire corpus of documents. These statistics are for your reference.

Label and icon	Description
Clear the current condition 	Click this icon to remove the current query condition from the search box, thus resetting the documents that are being analyzed back to the entire corpus of documents. Use this function when you want to start a new analysis with a different query condition. Because the current query condition is not saved automatically, you might want to click the <b>Save the search</b> button before you click the <b>Clear the current condition</b> button. This operation is the same operation when you click the <b>Clear</b> button in the search field.
Show and hide query input area 	Click this icon to show or hide the search field area. This icon works the same as when you click the <b>Show query input area</b> link. After you click this icon, the search box is either displayed or hidden at the top of the window.
Save the search 	Click this icon to save your query condition. A window opens that prompts you to enter the name of your query. You can also provide a description for the query. After you save the query, you can retrieve the query by its name from the <b>Saved Searches</b> tab.
Export the results 	Click this icon to export your search results. This button is displayed only if exporting searched documents is enabled.
Go Back/Forward to a query to rerun 	These icons behave similar to the back and forward buttons of a browser. Click these icons to move back or forward through each query progression when it is built during your analysis. Click the icon to execute the previous or next query and refresh the search results of all views. This capability applies to the current browser session. Information is not preserved permanently even if you log in to the application when the security is enabled. Also, when you click the <b>Clear the current condition</b> button, the history is cleared.
Show/Hide document properties 	Click these icons to show or hide the detailed document properties, such as the DocumentID and Title in the Documents view, for each document.
Specify preferences for viewing search results 	When you click this icon, the same window opens as when you click the <b>Preferences</b> link in the application toolbar. When you click this icon, the preference for the view that you are using or results tab is displayed. See 5.1.4, “Changing the default behavior by using preferences” on page 92, for a description of what you can set in this window. The <b>Results</b> tab is selected by default when you click this icon.

Label and icon	Description
Add the current query as a new category rule 	When you click this icon, a window opens in which you can add the current query as a new category rule for an existing category. This button is displayed only if the rule-based category is enabled. For further information about rule-based categories, see 5.5, “Rule-based categories with a query” on page 135.
Set and clear Document flags 	This button is displayed only in the Documents view. It is displayed if document flagging is enabled and the “Manage document flags” application user role is enabled. For further information, see 5.7, “Document flagging” on page 148.
Create deep inspection reports 	When you click this icon (except in the Documents view), you can create a deep inspection report. You must select a facet to create a deep inspection report. This button is displayed when the “Create deep inspection reports” application user role is enabled. For further information, see 10.7, “Deep inspection” on page 387.
Create a report for Cognos BI or download the report as CSV file 	When you click this icon (except in the Documents view), a window opens. You specify whether you want to save the output in the comma-separated values (CSV) format and where to save the output, or you can specify how to create the Cognos BI report. This button is displayed if “Create IBM Cognos BI reports” is enabled for the collection. For further information, see Chapter 13, “Adding value to Cognos Business Intelligence” on page 487.

**Icons for enabled features:** Some of the icons in Table 5-3 on page 146 are displayed only if you enabled that particular feature.

## 5.7 Document flagging

With document flagging, you can assign a custom flag to a single document or a group of documents for classification, export, or additional analysis purposes. This feature is convenient after you perform multiple searches to find the set of documents that you want to further examine to export them, or to classify them. The administrator defines the document flags that are selectable by users in the content analytics miner. After the documents are assigned with a flag, users can narrow down the documents by using the flag or view the flag count on the document result set for further analysis.

## 5.7.1 Configuring document flags

Configure document flags that will be selectable within the content analytics miner. In this scenario, you create two document flags named *Public Relations* and *Quality Assurance*.

**Flags in a collection:** A collection can contain a maximum of 64 flags.

To configure document flags, perform these steps:

1. From the administration console, expand your specific collection. Then, in the “Search and Content Analytics” administrative section, expand the **Edit - Configure** button and select the **Configure Document flags** (Figure 5-70) option.

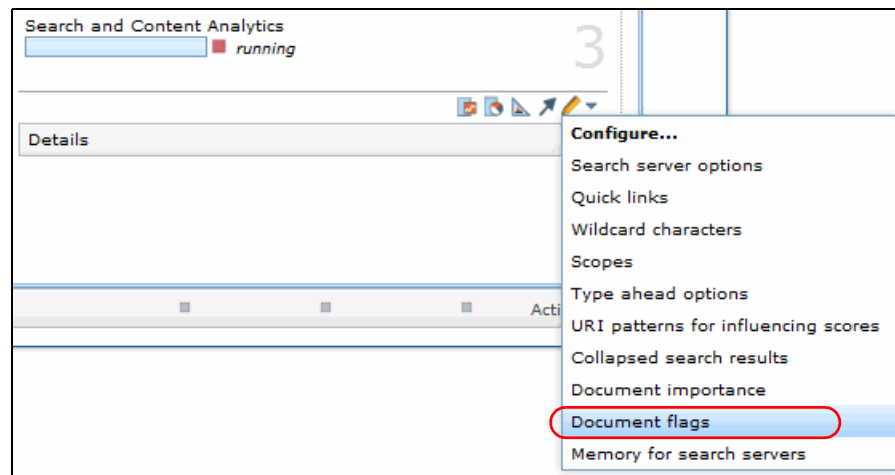


Figure 5-70 Configuring document flagging

2. In the Document Flags window (Figure 5-71 on page 150), complete these steps:
  - a. Enter **Flags** as the Root Label.
  - b. Click **Add New Document Flag**.

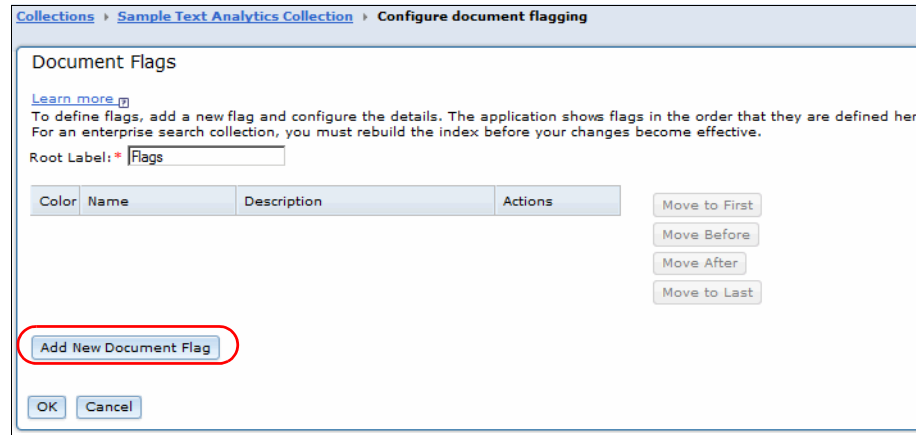


Figure 5-71 Configuring Document Flags

3. In the Document Flag edit window (Figure 5-72), complete these steps:
  - a. In the Name field, type Public Relations.
  - b. In the Description field, type Documents that might result in public relation exposure.
  - c. In the Color field, select a red color or type #8b0000.
  - d. Click **OK** to add the new document flag.

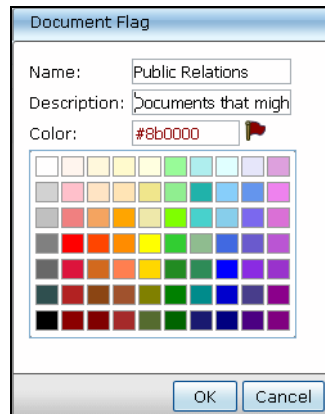


Figure 5-72 Adding a document flag

4. Repeat step 3 but use the field values indicated in Table 5-4 on page 151.

Table 5-4 Document flag properties

Field	Value
Name	Quality Assurance
Description	Documents that might indicate a quality assurance defect
Color	Blue or #000080

The document flag window now looks similar to the example shown in Figure 5-73. As a result, the Public Relations and Quality Assurance facets are displayed under the Flags facet in the content analytics miner.

5. Click **OK**.

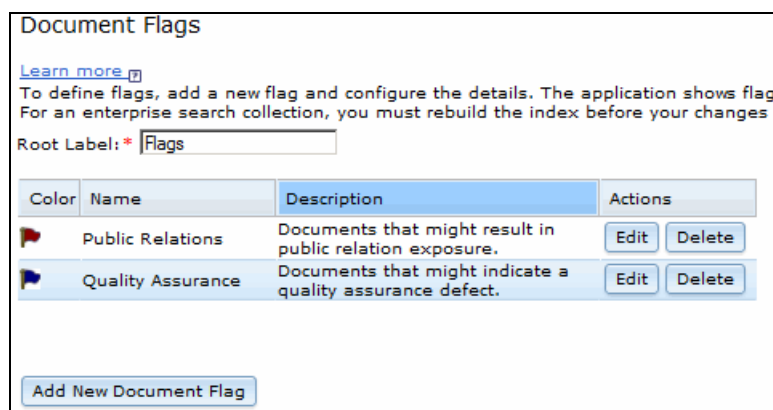


Figure 5-73 List of defined document flags

**Authority to manage document flags:** Users must have proper authority to manage document flags in order to add and remove flags associated with documents. If users are required to log in to the content analytics miner, select **Security** → **System-Level Security** → **Application user privileges** to provide the user with authority to manage document flags. If users are not required to log in to the content analytics miner, select **Security** → **System-Level Security** → **Actions** → **Specify default application user privilege** and ensure that the option **Manage document flags** is checked to provide the authority.

## 5.7.2 Setting document flags

Now that you configured the document flags, you can select and view them in the content analytics miner. For our scenario, a new root facet named *Flags*, with Public Relations and Quality Assurance facets as children, is displayed in the content analytics miner facet window. This section explains how to set these new document flags.

### Associating all search documents with a document flag

To associate all search documents with a document flag, follow these steps:

1. Open the content analytics miner.
2. Expand the query text area, and type `needle`. Click **Search** to search for the term `needle` (Figure 5-74). Now the document result set contains documents that are related to “needle”.

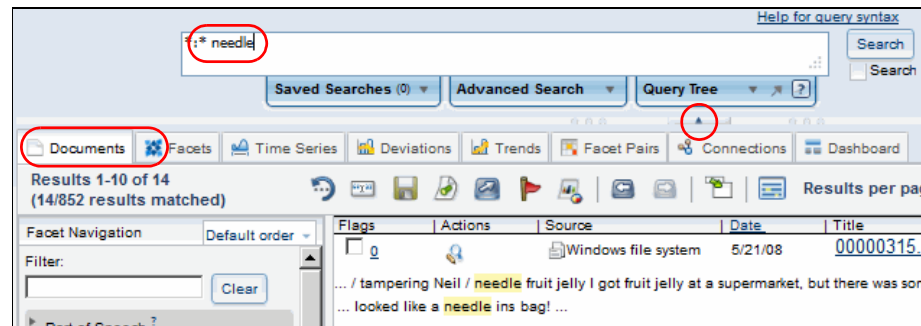


Figure 5-74 Documents containing the term “needle”

3. Click the **Documents** tab.
4. Click the **Document Flag** icon (highlighted in Figure 5-75).

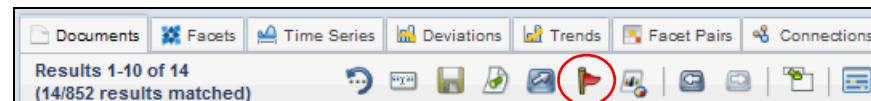


Figure 5-75 Document Flag icon in the content analytics miner



5. In the Manage Flags window (Figure 5-76), select **Public Relations**, and click **Apply changes to all results**. This action marks every document in the query result set with a *Public Relations* flag. Click **Save**.

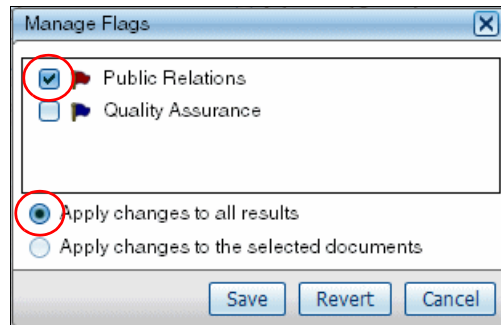


Figure 5-76 Selecting document flags for query results

Notice that the documents now have a red flag and the number 1 next to them in the Documents view table, as shown in Figure 5-77. The number indicates how many document flags are associated with this document. In this scenario, one document flag is associated so far.

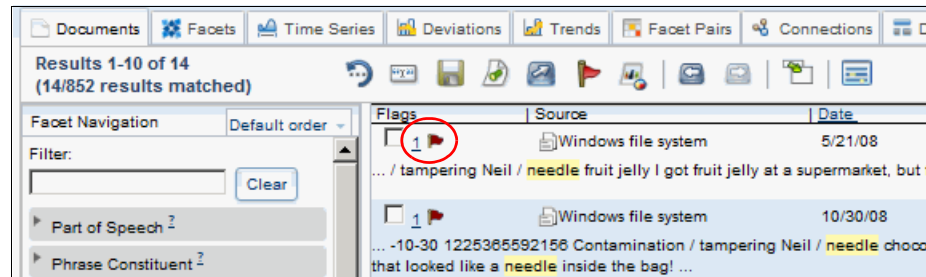


Figure 5-77 Document associated with the *Public Relations* flag

## Adding a document flag to a single document with quick flags

You can assign more than one flag to a single document after you have already assigned a document flag. With quick flags, you can quickly associate a flag to a document without selecting the document. To do so, follow these steps:

1. In the Documents view, click the **1** flag link for the first document listed in the table (Figure 5-77).
2. In the quick flag pop-up window, select the **Quality Assurance** flag (Figure 5-78 on page 154). This action automatically adds the Quality Assurance flag to the document that you selected.

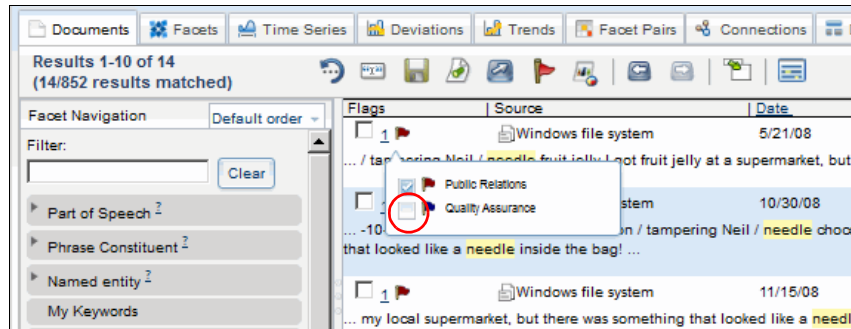


Figure 5-78 Adding a document flag to a single document using the quick flag link

Now the document shows two flag icons associated with it, and it contains a “2” link to indicate two document flags, as shown in Figure 5-79. One of the flag icons is the color of the *Public Relations* flag, and the other flag is the color of the *Quality Assurance* flag.

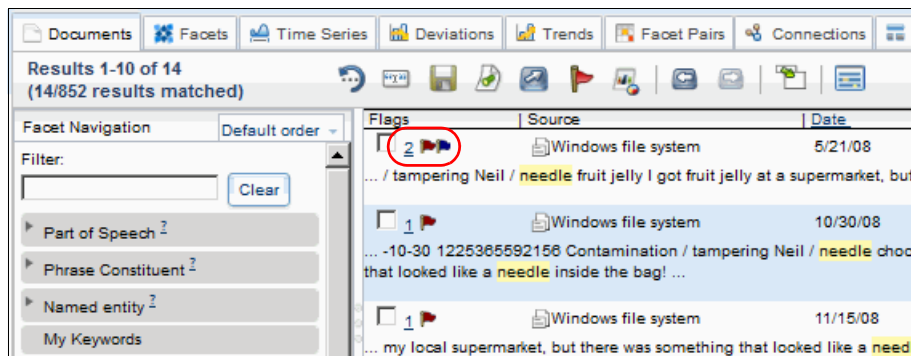


Figure 5-79 Document with two flags associated to it

## Associating selected documents to a document flag

You can associate a selected set of documents to a particular flag rather than associating the flag to the entire query result set. To do so, follow these steps:

1. Click **Clear** in the query text area to clear the search query to start a new search view.
2. Click the **Documents** tab (Figure 5-80 on page 155).
3. In the query search text area, type `leak`. This action shows all documents that contain the term “leak” so that you can analyze them further.
4. Select the first two documents in the result set by clicking the check box to the left of the document.

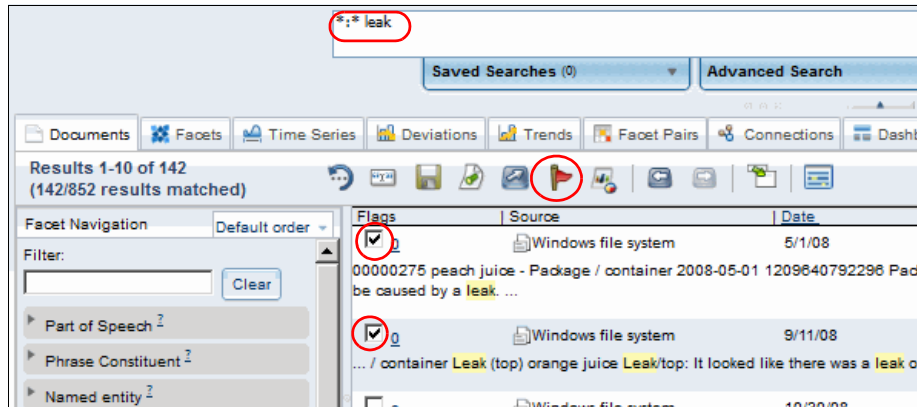


Figure 5-80 Selecting documents to assign the document flag

5. Click the **Flags** icon (highlighted in Figure 5-80).
6. In the Manage Flags window (Figure 5-81), select the **Quality Assurance** check box, and click **Apply changes to the selected documents**. Then, click **Save**.

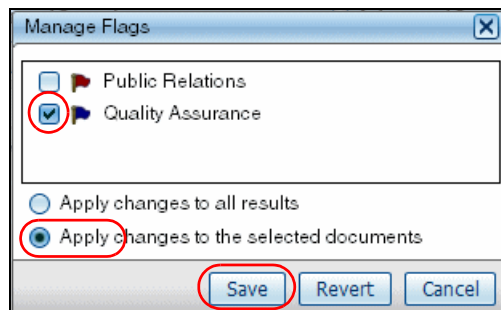


Figure 5-81 Applying the flag changes to selected documents

### 5.7.3 Viewing the document values of a flag facet

With the facet view, you can view all of the documents that are associated with one flag. You can narrow down the documents by flag or view the flag count on the returned documents. To view the documents that are associated with the Public Relations document flag, follow these steps:

1. Click **Clear** to clear the search query to start a new search view.
2. Click the **Facet** tab.

- Expand the **Flags** facet, and count the documents for each flag that is displayed (Figure 5-82).
- To view all of the documents marked with the Public Relations flag, select the **Public Relations** facet check box and click the **Add to search with Boolean AND** icon (highlighted in Figure 5-82).

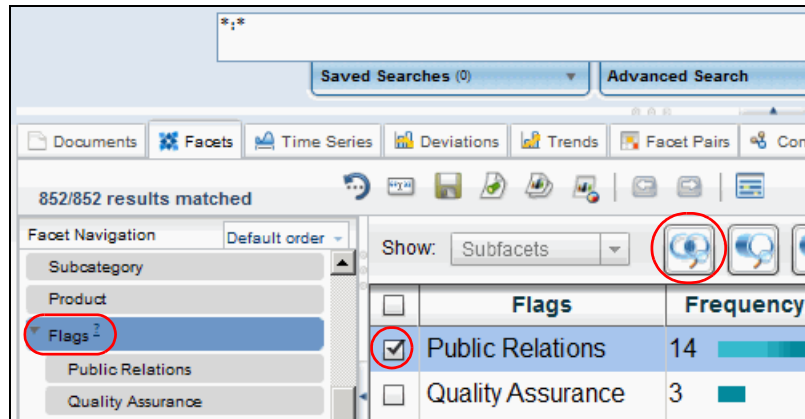


Figure 5-82 Display of document flag facets

- Click the **Documents** tab (Figure 5-83) to view and further analyze all of the documents that are associated with the *Public Relations* flag.

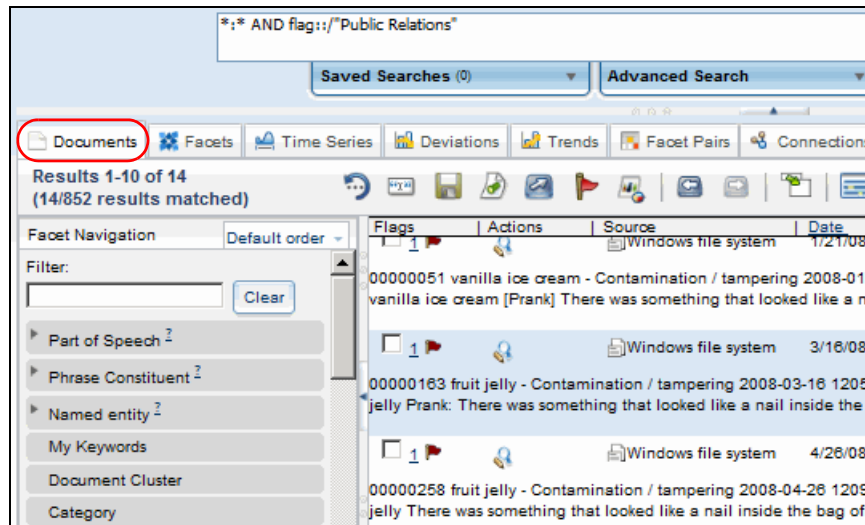


Figure 5-83 Documents associated with the Public Relations facet

The content analytics miner provides many views for content analysis. For information about the different content analytics miner views, see Chapter 6, “Content analytics miner: Views” on page 159.

If you are already familiar with all the views, proceed to Chapter 7, “Performing content analysis” on page 231.





## Content analytics miner: Views

IBM Watson Content Analytics (Content Analytics) provides a content analytics miner to help you to perform content analytics. Chapter 5, “Content analytics miner: Basic features” on page 85, provides information about the basic features of the application, with a specific focus on the search and discovery features. This chapter focuses on the application’s views, features, and functions.

Specifically, this chapter includes the following sections:

- ▶ Views
- ▶ Documents view
- ▶ Facets view
- ▶ Time Series view
- ▶ Trends view
- ▶ Deviations view
- ▶ Facet Pairs view
- ▶ Connections view
- ▶ Dashboard view
- ▶ Sentiment view

If you are already familiar with the user interface of the content analytics miner, including its views and search and discovery features, proceed to Chapter 7, “Performing content analysis” on page 231.

## 6.1 Views

The content analytics miner provides the following views to assist in text mining. These views (shown in Figure 6-1) are generated dynamically based on your query and facet selections:

- Documents view** Shows a list of documents that match your query.
- Facets view** Shows a list of keywords for a selected facet.
- Time Series view** Shows the frequency change over time.
- Trends view** Shows unexpected increases in frequency over time.
- Deviations view** Shows the deviation of keywords for a given time period.
- Facet Pairs view** Shows the correlation of keywords from two selected facets.
- Connections view** Shows the correlation of keywords from two selected facets.
- Dashboard view** Shows a configured dashboard layout with one or more graphs or tables in a single view.
- Sentiment view** Shows the sentiment (positive, negative, or neutral) for language within documents, for the selected facet.

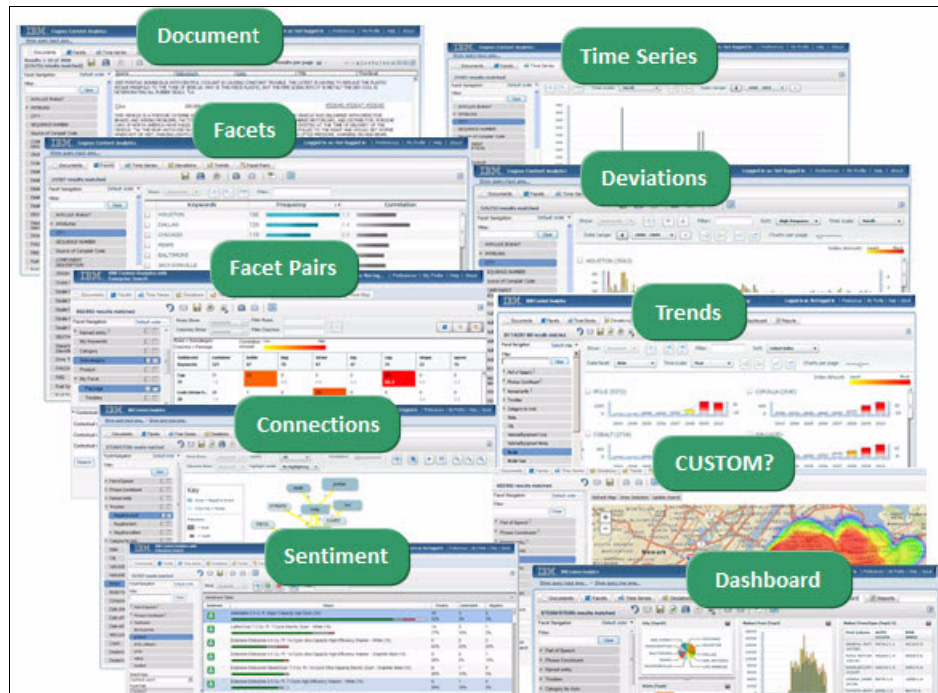


Figure 6-1 Views available in the content analytics miner



For information about the general user interface of the content analytics miner, see 5.1, “Overview of the content analytics miner” on page 86. For information about the search and discovery features of the content analytics miner, see 5.2, “Search and discovery features” on page 96.

## 6.2 Documents view

The Documents view shows a list of documents that match your query or facet selections, with details such as preview text and metadata. You use this view when you want to see the contents of an individual document. If you are analyzing a collection of thousands (or millions) of documents, the other views will provide a broader perspective on the patterns in your collection data. After using the other views to filter down to a pattern you are interested in, the Documents view allows you to quickly view the source content, to help confirm whether that pattern or insight is correct and meaningful.

By default, ten results are displayed per page in the Documents view, though up to 100 can be configured to display at one time. You can click the page buttons to navigate between results pages or to move to a specific page.

Each entry in the document list contains the following information:

- ▶ A dynamic summary of the document based on your query terms
- ▶ The source and date of the document
- ▶ A thumbnail of the document, if the type of document supports thumbnails and the thumbnail feature is configured
- ▶ A document link displayed as the title, which, when clicked, shows the original document from its crawled source

Figure 6-2 shows the Documents view with the sample documents. The default search condition is \*:\*.

Source	Date	Title
Windows file system	1/1/08	00000000.xml 00000000 lemon tea - Package / container 2008-01-01 1199186392296 Package / container Straw lemon tea was peeled off from the juice pack.
Windows file system	1/2/08	00000001.xml 00000001 vanilla ice cream - Contamination / tampering 2008-01-02 1199272792171 Contamination / tampering cream I got some ice cream for my children, but there was something like a piece of thread inside the cup.
Windows file system	1/2/08	00000002.xml 00000002 apple jelly - Number of pieces 2008-01-02 1199272792218 Number of pieces Shortage apple jelly I cups in the 12-pack.
Windows file system	1/2/08	00000003.xml 00000003 orange juice - Package / container 2008-01-02 1199272792265 Package / container Leak orange juice stain on the package that seemed to be caused by a leak. Is it safe to drink?
Windows file system	1/3/08	00000004.xml 00000004 milk chocolate - Ads 2008-01-03 1199359192093 Ads TV ads milk chocolate I love the ads for the Could you tell me the name of the actor in the commercial?
Windows file system	1/3/08	00000005.xml 00000005 vanilla ice cream - Prank 2008-01-03 1199359192312 Prank Open vanilla ice cream The cup looked opened. Is it safe?
Windows file system	1/3/08	00000006.xml 00000006 chocolate ice cream - Taste / smell 2008-01-03 1199359192343 Taste / smell Change chocolate ice more sour than when I ate it before.
Windows file system	1/4/08	00000007.xml 00000007 mint jelly - Number of pieces 2008-01-04 1199445592234 Number of pieces Shortage mint jelly [Sh find 10 cups in the dozen pack.

Figure 6-2 Documents view showing the results based on the default search condition

## 6.2.1 Understanding the Documents view

In the Documents view, you can see the following information and actions:

- ▶ The total number of documents that match your query. In the example shown in Figure 6-2, 852 documents are returned.
- ▶ The query that is used to produce the result. The query is hidden when the search box is hidden.
- ▶ If configured, a **Flags** option that lets users select individual documents that can be added to a dynamic facet.
- ▶ An action to launch the **Query Builder**, which helps a user create a new search query based on the attributes of the current document. The query builder is covered in 5.4, “Query builder” on page 123.

- ▶ Selected fields such as Source, Date, Title, and Thumbnails. The fields displayed as columns are based on your configuration settings on the Results Columns tab, under Preferences.
- ▶ Preview text, which is dynamically generated based on your query.
- ▶ The details of each document when you click the **Show detailed properties** icon.
- ▶ A link from the Title field, which enables retrieval of the full document content if the document content can be retrieved.
- ▶ When sentiment analysis is enabled for a collection, positive and negative expressions are highlighted and filterable. See 6.10, “Sentiment view” on page 222 for more information about these features.

## 6.2.2 Viewing the document contents and facets

When you click the source icon, the Document Analysis window opens on top of the Documents view. This window contains details about your document. It shows the individual field values for the document and all annotations made to the document during text analysis. In the Document Analysis window (Figure 6-3 on page 164), the Analytics Facet is listed in the left pane, and the Metadata Facet is shown in the right pane.

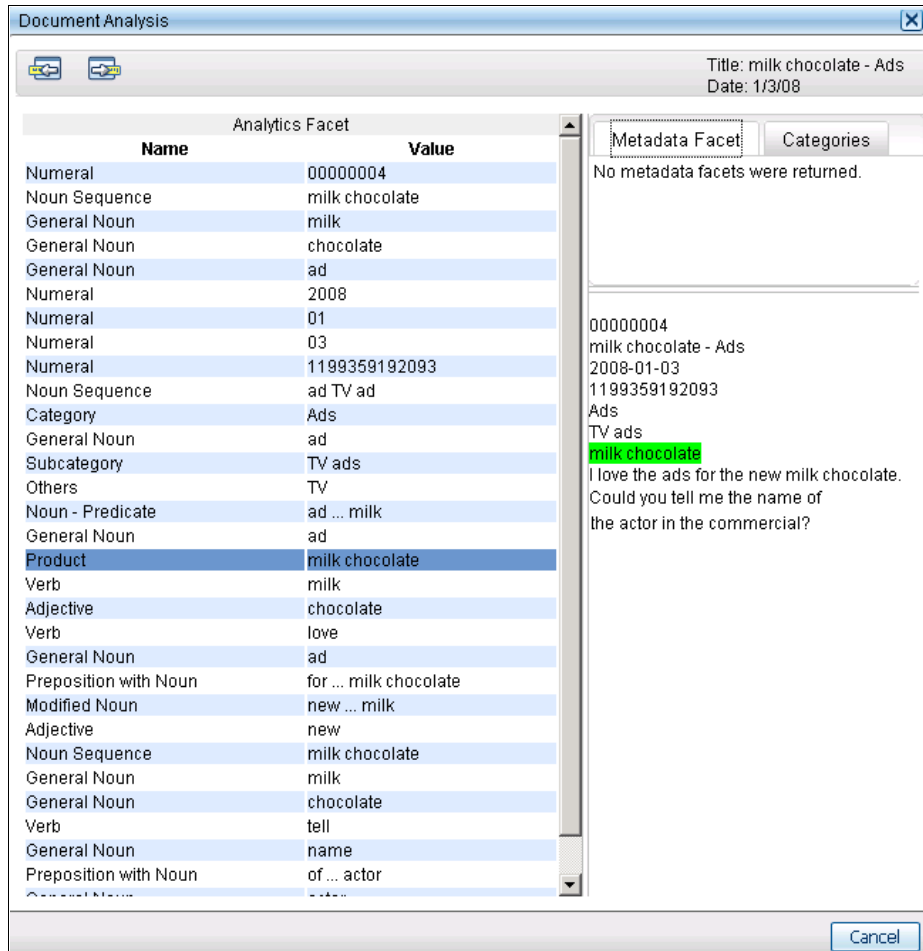


Figure 6-3 Document Analysis window

When you select the Analytics Facet, the corresponding keywords are highlighted in green in the structured and unstructured fields of your document where the values occur. For example, we select the Product analytics facet called “milk chocolate”, as shown in Figure 6-3. As a result, the keyword “milk chocolate” is highlighted in green within the Metadata Facet pane. This function helps you to understand the data that determined the facet.

## 6.2.3 When to use the Documents view

The Documents view is useful when you want to see the details of individual documents after filtering to a smaller results set using other views or a search expression. By using the Preferences configuration, you can control how the search results are displayed in the Documents view.

## 6.3 Facets view

The Facets view shows a list of keywords that are displayed for a selected facet. The frequency count and correlation value accompany each keyword. This view is useful for seeing the keywords that make up a given facet in your data.

In the Facets view, frequency and correlation are shown as a bar chart. Each column is described as follows:

<b>Keyword</b>	The values that are associated with the selected facet. These values can be words, phrases, text patterns, field values, date ranges, or numeric ranges.
<b>Frequency</b>	Indicates the number of documents found in the current result set that contain the given keyword.
<b>Correlation</b>	Indicates how the keyword is interrelated to the documents that are matched by your query.

You can sort the output by frequency or correlation.

**Default sort order in the Facets view:** By default, the documents in the Facets view are sorted by frequency in descending order. You can modify the default value by using the Preferences window.

**Important:** Even if you select correlation as the default sort order, the list of facet values that are displayed are chosen by frequency. Therefore, if you leave the default of 100 keywords, you get only the 100 most frequent keywords. This means that sorting by correlation will *not* include values that are more highly correlated but not in the top 100 by frequency. This approach to sorting is by design, with the rationale that keywords with a very low frequency count would typically not be of interest.

Figure 6-4 shows the view when you select the Product facet from the Facet Navigation pane. You can select any facet that you configured.

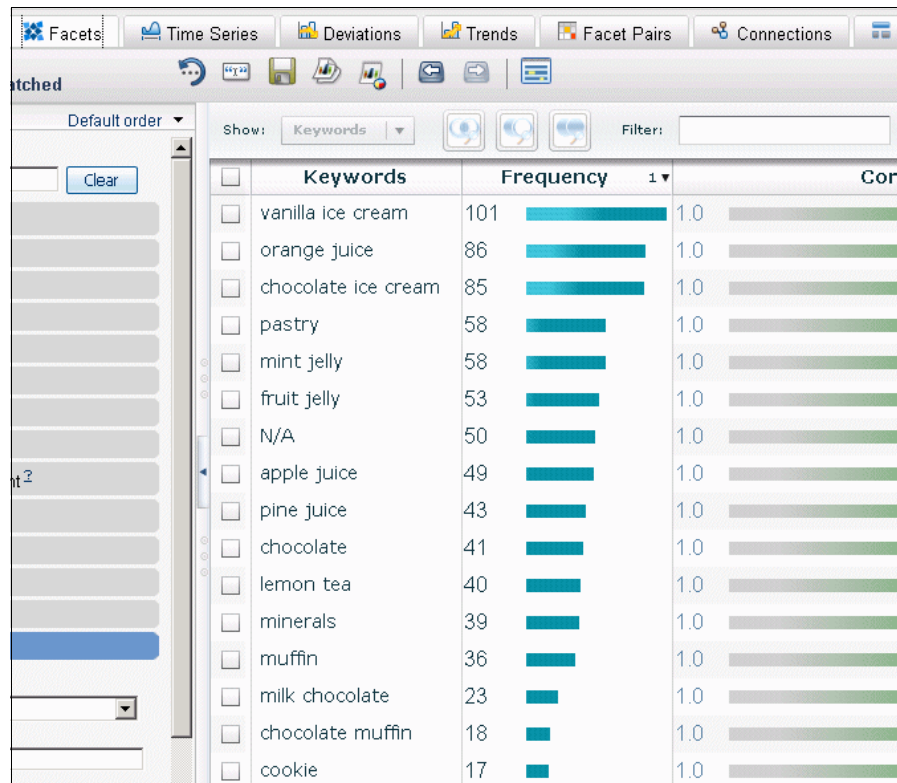


Figure 6-4 Facets view with the Product facet selected

You can limit the scope of your analysis to one or more keywords by selecting them using their corresponding check boxes and adding them to the query. When you select a keyword, the Boolean search operator (AND, AND NOT, OR) icons are highlighted and become active, as shown in Figure 6-5.

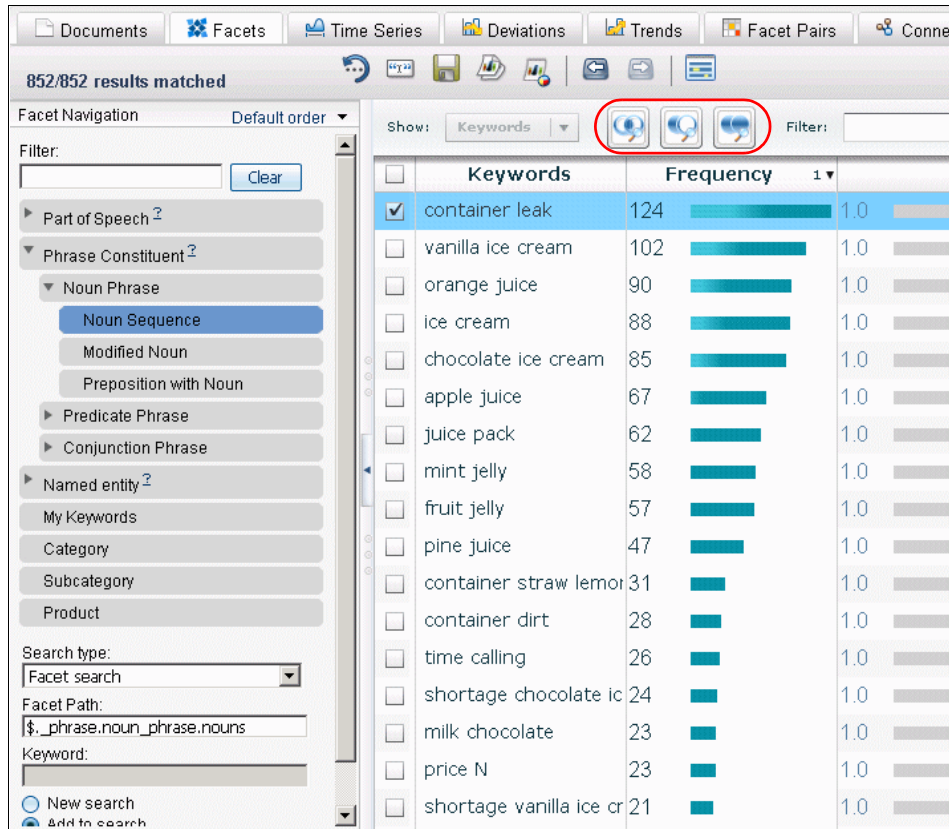


Figure 6-5 Facets view with search operators highlighted when a keyword is selected

**Search operators:** See 5.2.1, “Limiting the scope of your analysis using facets” on page 97, for details about the search operators.

After you click a specific Boolean search operator, the view is updated with the new search results and the result counts. At any time, you can go back to the Documents view to see the content of the documents that match the given query.

### 6.3.1 Understanding the Facets view

To effectively use the Facets view, you must understand the difference between frequency and correlation, which are described in 1.3, “Important concepts and terminology” on page 12. As a review, the *frequency* value counts the number of documents in the current filter. The *correlation* value measures the amount of uniqueness of the high frequency as compared to other documents that match your query.

Although the frequency value is useful, it might not always be as revealing as the correlation value. For example, high frequency counts for a particular model of car can be attributed to the overall popularity of the car: More cars of that model are sold than any other models. A high correlation value means that, regardless of that popularity, that model of car has a higher than expected frequency in the current query results.

The conclusion that the frequency is “higher than expected” is based on comparing the percentage of “hits” for this model in the current query results to the percentage of hits for the model in the entire set of data for the collection. If a query or other filter has not been applied yet and you are examining the entire data set, all correlation values will be 1.0. After you add a keyword to your query or enter a search expression, the correlation values are recalculated and reflect how interrelated each keyword is to the current filter.

### 6.3.2 When to use the Facets view

You use the Facets view when you want to see a list of keywords that are associated with a given facet, and how much each of these keyword values is related to the current set of filters you have applied. See 7.3, “Overview of techniques to create facets for analysis” on page 252 for an understanding of the breadth of options to create useful facets for your analysis.



## 6.4 Time Series view

The Time Series view shows document frequencies over time. Correlation and deviation values are not displayed. This view is primarily used to examine frequency and to select a range of documents from a given time period for further analysis in other views. For example, if you examine customer service transcripts, you might notice time periods where the frequency of calls is unexpectedly high. The Time Series view would let you filter and focus on these “spike” time periods, which you would then examine via other views and facets to understand why.

The Time Series view is updated whenever you filter the result set using a view or search expression, showing the new frequency values for that filter. For example, consider a situation where you select **vanilla ice cream** for the Product facet and add the keyword with the AND operator. In this case, you can see how the vanilla ice cream documents are distributed across the selected time scale (year, month, day) along with their computed frequencies.

In another example, the Time Series view in Figure 6-6 on page 170 shows the frequency of distribution when you select the Product facet and select Month for Time scale.

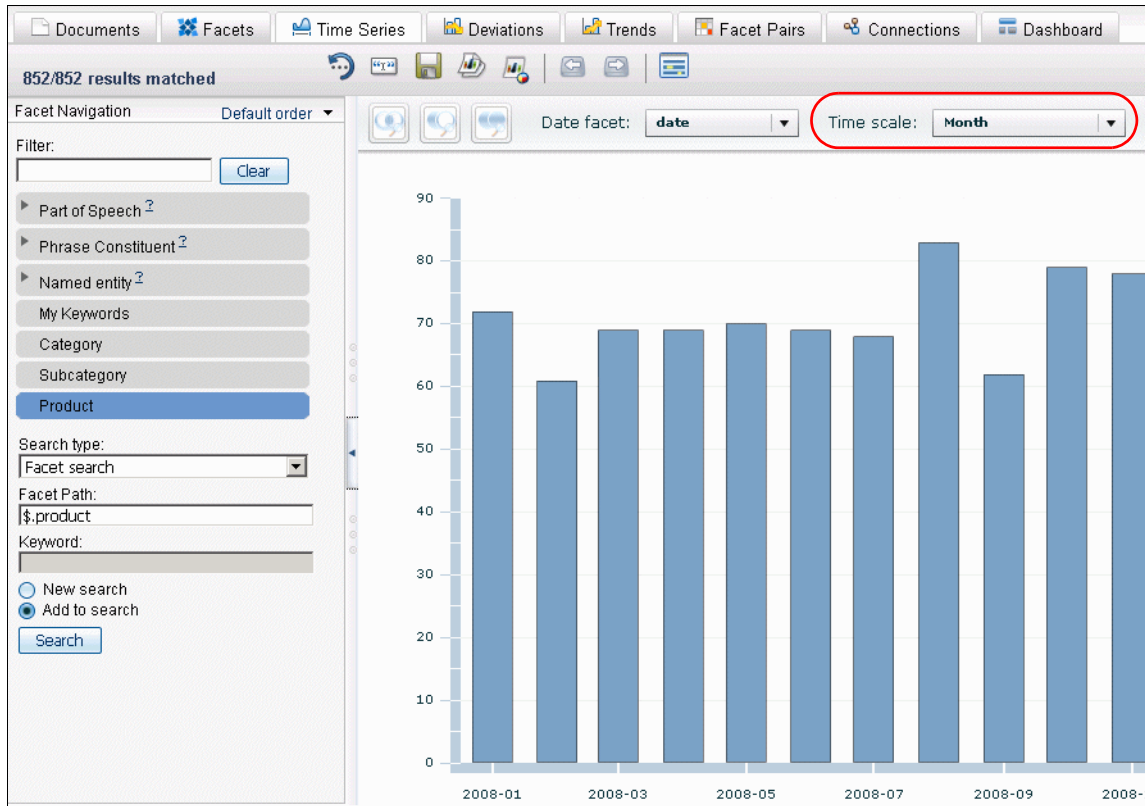


Figure 6-6 Time Series view showing results sorted by month

## 6.4.1 Features in the Time Series view

With the flexibility of the Time Series view, you can change the time scale, use the Zoom in and Zoom out features, and focus on specific date ranges. This section provides details about each of the features of this view.

### Changing the time scale

You can change the time scale of the graph when you analyze the data. From the drop-down menu, you can select basic time scales such as year, month, or day. You can also select time scales that reveal cyclical patterns, such as month of year, day of month, or day of week.

Each time scale calculates the sum of all documents for that particular time unit. For example, if you select “month” as the time scale, you see a bar for each month in the range of months as bounded by the documents that match your query. If your search result set spans two years, 24 months are shown.

If you select month of year, you see 12 bars, one for each month; if you select day of month, you see 31 bars; and if you select day of week, you see 7 bars. For each result, you see the sum of all documents that fall on that particular date increment. For example, if you select the day of week time scale, you see seven bars, with the first bar representing the total of all documents in the result set that fall on Sunday.

This feature conveniently shows the days of the week, month, or months of the year in which the most documents (or events) occur. If you are analyzing customer complaints about your product or service, for example, it might be useful to understand whether there are seasonal trends to these complaints, which the “months of the year” time scale can help to reveal.

### **Zooming in and out**

You can use the Zoom in and Zoom out feature especially when the bar chart becomes too busy to distinguish the exact values. As shown in Figure 6-7 on page 172, you can select an area that you want to look into by dragging and zooming in on that area. After you select the area, click the **Zoom in** icon.

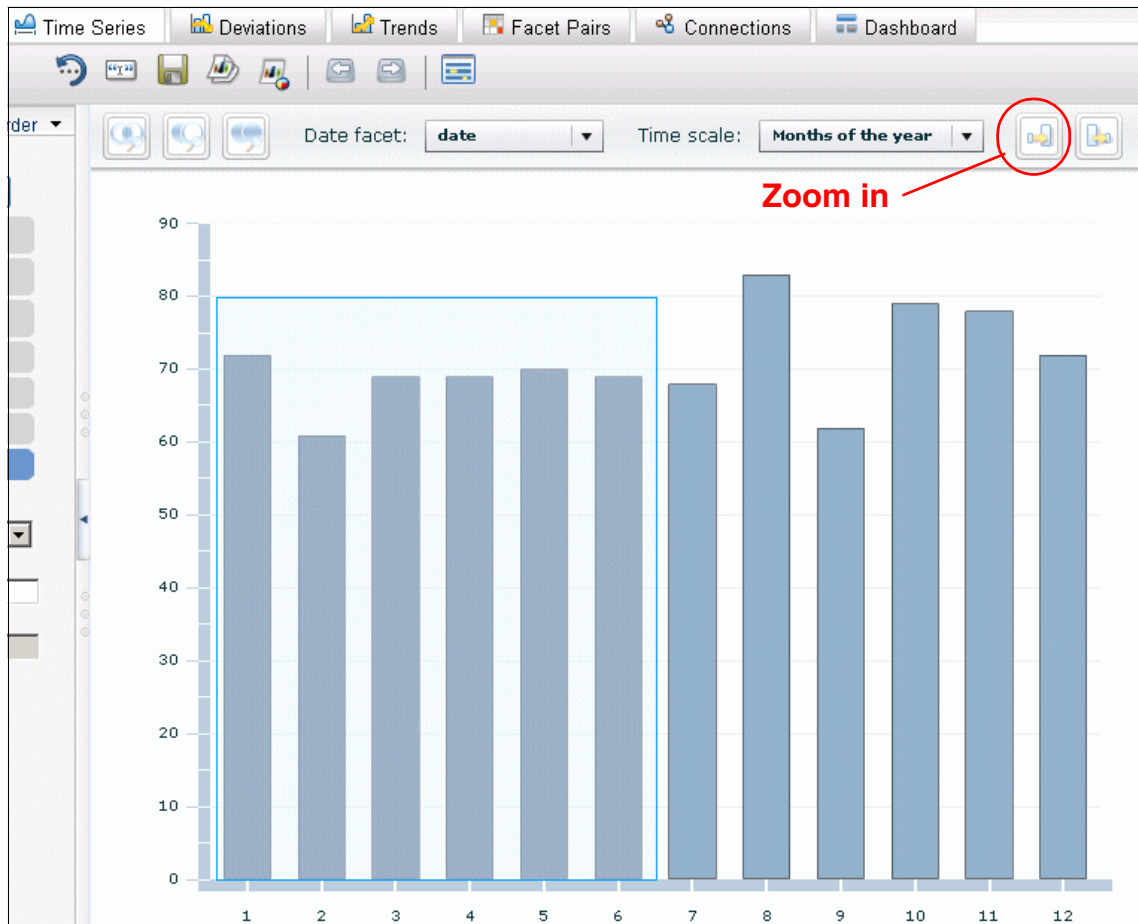


Figure 6-7 Time Series view showing the selected area to zoom in

Figure 6-8 shows the result of zooming in. When you click the **Zoom out** icon (highlighted in Figure 6-8), the view goes back to the original graph (Figure 6-7 on page 172).

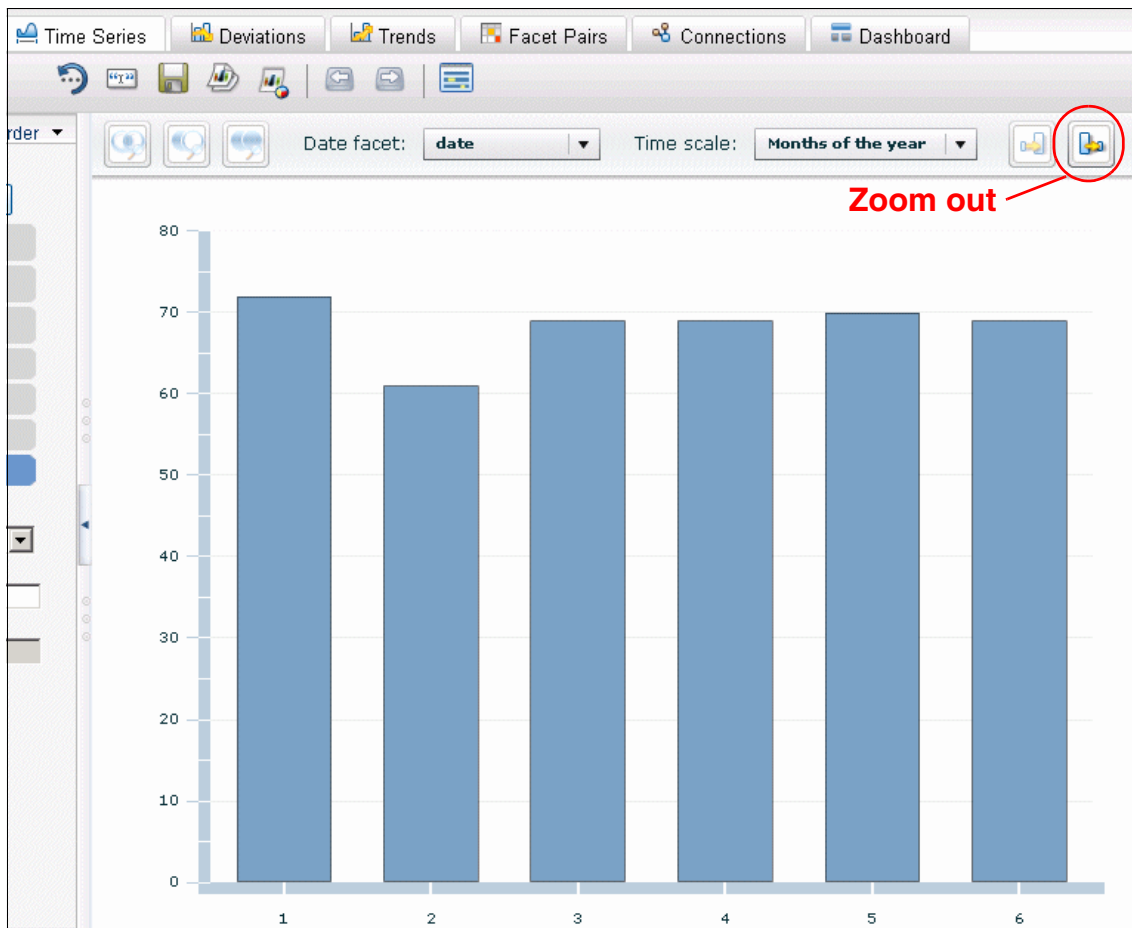


Figure 6-8 Time Series view showing the results of zooming in on the selected area

## Changing the Date facet

When you configure the Date facet to contain multiple date fields, you can select a different field for the Date facet to analyze the data in the Time Series, Trends, or Deviations views. By changing the field for the Date facet, the time scale for the graph is automatically updated to use the new date field.

For example, after the Time Series view is displayed based on the reported date of the document, you might want to see the same data in the Time Series view based on the date the incident occurred to give you another analysis perspective. Figure 6-9 shows selecting the Date facet value in the Time Series view.

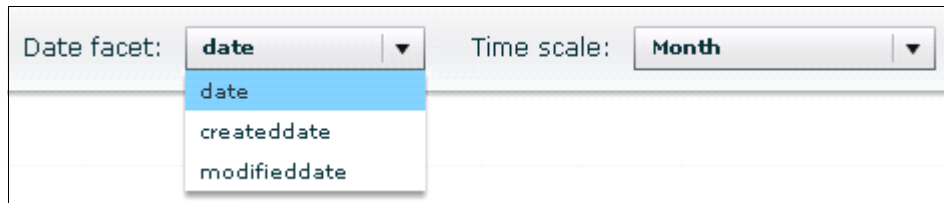


Figure 6-9 Selecting the Date facet in the Time Series view

**Configuring multiple date facets:** See “Optional: Configuring the data facet” on page 127 from the previous version of this book to configure date facets.

## 6.4.2 Understanding the Time Series view

The Time Series view shows the distribution of documents that match your query over a period of time. The y-axis shows the number of documents (frequency), and the x-axis shows the time scale that you selected.

When you hover the mouse cursor over a particular bar in the chart, a dialog window opens that details the data for that unit, namely the frequency and specific date value.

### 6.4.3 When to use the Time Series view

The Time Series view helps you to see the distribution of documents over different time scales, for the current filter. There are a number of ways this can be useful:

- ▶ You can build separate filters (using search expressions or combinations of facets) and compare the frequencies of occurrences between these filters. For example, you can compare the time series of one product versus another, or one category of complaint versus another. This approach also helps you to quickly see sudden increases for the particular category you have filtered on.
- ▶ You can use the Time Series view as a starting point in analyzing your collection. If you have a collection of problem reports, you might notice time periods where the frequency of these reports was unexpectedly high. The Time Series view would let you filter and focus on these time periods, which you would then examine via other views and facets to understand why.

Frequency counts alone, while useful, might not effectively reveal key patterns. The Trends and Deviations views add additional statistical information, as well as the ability to view multiple graphs at once, to more efficiently surface anomalies in your data. The Time Series view is more useful for selecting an initial filter leading to a more in-depth analysis in other views.

## 6.5 Trends view

The Trends view shows sharp and unexpected increases in frequency of a facet over time. The Trends view is similar to the Time Series view in that it shows the frequency of documents as a bar graph across a given time frame. You can change the time scale to year, month, or day. It also provides the same zoom in and zoom out functionality as the Time Series view. However, the Trends view has significant differences in helping you to gain additional insight from your data.

When you select a facet, the Trends view shows a list of individual bar graphs, one for each value of that facet. Each individual bar graph looks similar to a Time Series graph, but also includes additional statistical information to highlight where upward changes in frequency are higher than expected.

Figure 6-10 shows the Trends view when you select the Product facet and default date facet with the month time scale.

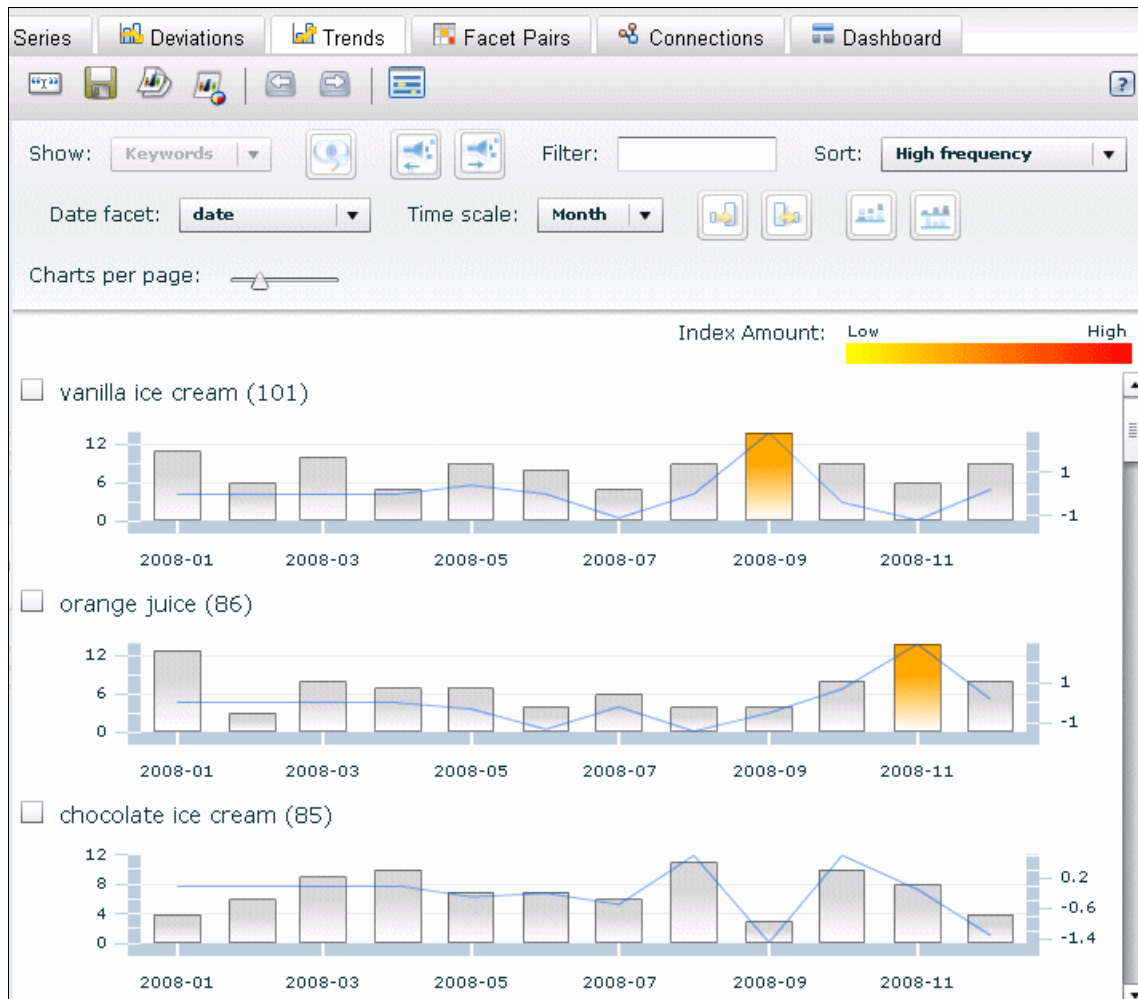


Figure 6-10 Trends view with the Product facet, sorted by high frequency and month



## 6.5.1 Features in the Trends view

The following features are the same as those features described in 6.4.1, “Features in the Time Series view” on page 170:

- ▶ Changing the time scale
- ▶ Zooming in and out
- ▶ Changing the Date facet

The time scale options do not include the ability to select cyclical views, such as month of the year, day of the month, or day of the week.

### Changing the Charts per page indicator

You can change the number of charts per page by sliding the bar as highlighted in Figure 6-11. This feature is helpful when you want to view and compare multiple charts at a time on a page or focus on only one chart. The size of the chart varies depending on the number of charts viewed per page.

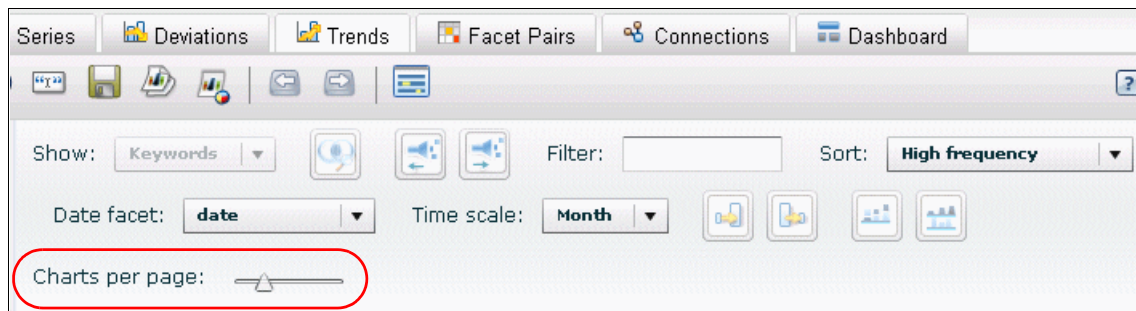


Figure 6-11 Trends view showing the Charts per page indicator

### Showing selected charts or showing all charts

Each individual bar chart comes with its own selection check box. The **Show selected charts** icon (highlighted in Figure 6-12 on page 178), when clicked, reduces the view of charts to only those charts that are selected. For example, you might have a total of 48 charts, and eight charts are shown per page on six pages. You are only interested in seven charts scattered across various pages. By selecting the charts that you are interested in using the check box and clicking the **Show selected charts** icon, only the selected charts are shown on a single page for comparison.

You can revert to the original chart view by clicking the **Show all charts** icon (also highlighted in Figure 6-12).

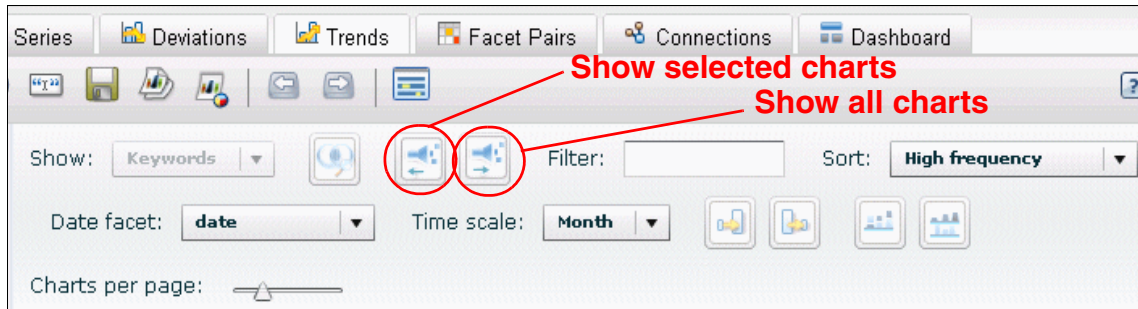


Figure 6-12 Trends view showing the Show selected charts and Show all charts icons

### Combining selected charts or showing separate charts

You can combine multiple selected charts into one chart by clicking the **Combine selected charts** icon (highlighted in Figure 6-13). This function aggregates all the charts into a single chart with each keyword given a different color.

You can revert to the original chart view by clicking the **Show separate charts** icon (highlighted in Figure 6-13).

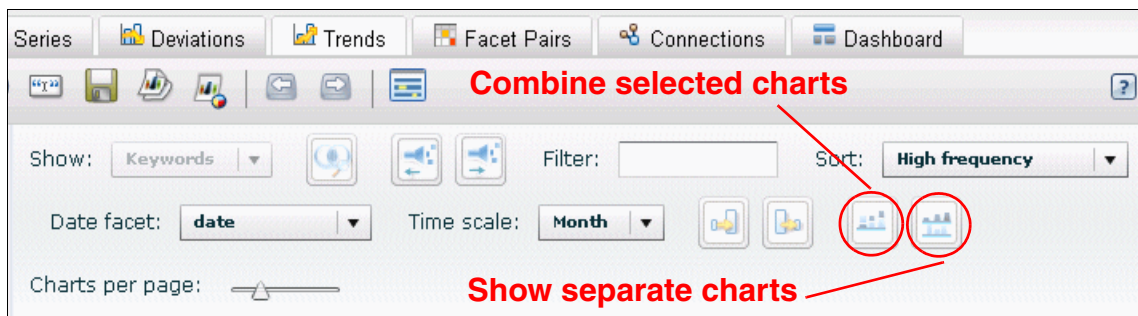


Figure 6-13 Trends view showing the Combine selected charts and Show separate charts icons

### Filtering the result by keyword

When you type a keyword in the Filter field box, the charts whose keywords contain the input filter are displayed in the view. The view is updated dynamically, so that you can see the filter result immediately.

## 6.5.2 Sort criteria

You can select from the following sort criteria to assist in the investigation of your data:

- ▶ Highest frequency (default)

This criterion lists, in descending order, those graphs with the highest frequency counts. The keywords graph at the top of the list contains a time period with the greatest frequency count. This method offers a quick way to see which keywords contain the most occurrences.

- ▶ Highest index

This criterion lists, in descending order, those graphs with the highest trends index values. This method offers a quick way to see which keywords are operating out of the norm the most. While frequency counts may not always be informative (higher sales of snow shovels in the winter would not provide much insight), index values are more likely to surface a real anomaly.

For example, you might have a facet that tracks the frequency of car part failures. The highest index might list, at the top of the list in descending order, the car part with the highest trends index from its expected trending average.

- ▶ Latest index

This criterion is similar to the highest index criterion but looks at the highest index values for only the most recent time unit (year, month, or day). This method is a quick way to see which keywords are most recently operating out of the norm.

- ▶ Name

This criterion alphabetically sorts the graphs by name in either ascending or descending order. This method is a quick way to find particular keyword values that you are interested in.

### 6.5.3 Understanding the Trends view

In the Trends view (Figure 6-14), the frequency of the selected time period is shown as a bar chart. It is scaled accordingly from 0 to the maximum frequency count along the vertical y-axis on the left side.

On the right side of the y-axis is a scale that represents the increase indicator. The *increase indicator*, also known as trend index, is a scale to measure the increase ratio of the frequency for a given time interval as compared to the expected average frequency that is calculated based on the changes in the past time interval frequencies. This expected change in frequency is estimated by using a modified Poisson distribution. The increase indicator is shown as a blue line graph in the chart, as shown in Figure 6-14.

The bar chart is in color (to highlight) whether the increase indicator is higher than what was expected within reasonable limits. This result means that the actual value is greater than the estimated value by a certain amount. Figure 6-14 shows a brighter orange color for December 2008. As you can see from the blue line in the graph, the increase indicator shows a sudden jump, which means that you might want to conduct additional investigation and analysis for that time period.

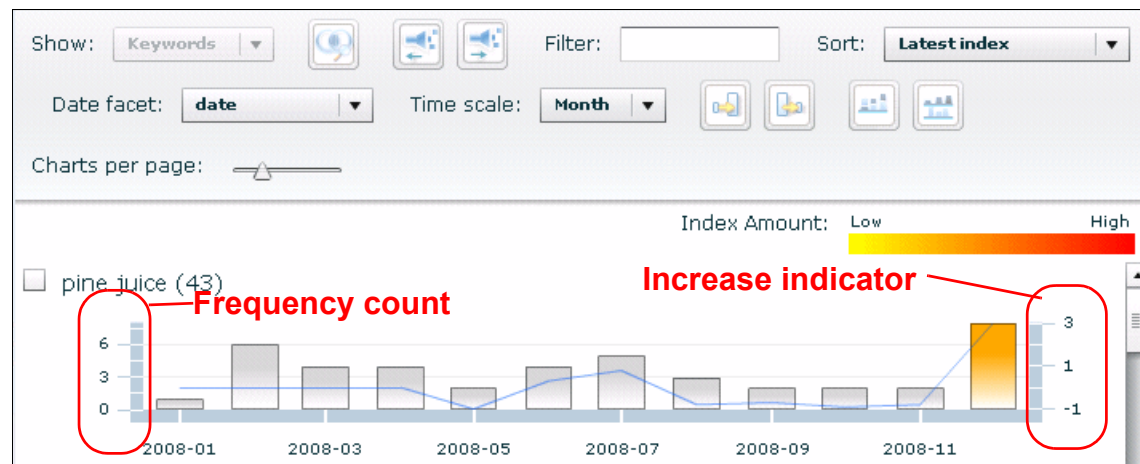


Figure 6-14 Trends view: Pine juice with frequency count and increase indicator

## Calculating the index in the view

How is the trend index value, which is shown as a blue line in the chart and determines how the graphs are highlighted for anomalies, calculated? Here is a high-level description of the elements that go into the calculation.

To generate the trend index values, you must calculate:

- ▶ The average global frequency, which is the average frequency of all searched documents over the given time period. This would be  $(\text{total count of documents in current results set}) / (\text{total number of time intervals})$  that is, years, months, or days.
- ▶ The average keyword frequency over the same period. This would be  $(\text{total count of documents in current results set for the keyword value graphed}) / (\text{total number of time intervals})$

Content Analytics also accounts for a decay factor to improve the validity of the calculation. *Decay* means that you want past data to decrease in relevance, the more distant it gets. To do this, each frequency is weighted according to a decay constant. For example, if the decay value is 0.85 (the default setting for the increase indicator in the content analytics miner), the frequency of the  $n$ th–4th date contributes less (about half) to the calculation as the frequency for the  $n$ th date. Both the global and the keyword average frequencies are weighted this way.

With this time series of weighted average frequencies, Content Analytics estimates the frequency counts for future dates. This estimate gives you the ability to collate the increase index, to the degree to which the frequency of this keyword has increased or will increase, for a particular date.

Since the algorithm needs the first four time intervals to begin calculation of the index (using an algorithm for Poisson distribution), these first four values for each graph will have a “standard” index value regardless of frequency count, and not be highlighted as an anomaly.

### 6.5.4 When to use the Trends view

With the Trends view, you can detect sharp and unexpected increases in the frequency of a given keyword value. Unlike Time Series view, which shows a basic frequency bar chart, the Trends view can quickly highlight anomalies for a number of keyword values on a single window.

Trends view can highlight key concerns such as an uptick in complaints about particular products or services, or positive trends as well. This can help you respond to a problem or opportunity at its early stages, when it can be most effectively addressed.

## 6.6 Deviations view

The Deviations view shows the deviation of keywords for a given time period. The Deviations view is similar to the Trends view in that it requires the selection of a facet and shows the corresponding individual graphs for each keyword. The controls across the top function are the same as in the Trends view with one exception. Three additional selections are available from the time scale pull-down menu, namely the month of year, day of month, and day of week. These additional selections provide greater insight into cyclic changes in your data.

The greatest difference between the two views is what the graphs are trying to convey when certain bars are highlighted, indicating something of interest. In particular, the Trends view alerts you when a keyword is trending up or down by an unexpected amount, and the expected amount is calculated based on the past history of frequency changes. This view is more focused on the trending of frequency counts over time.

The Deviations view is focused on how much the frequency of a given keyword deviates from the expected average for the given time period (not previous periods). The expected average takes into account all the averages of the other frequency counts for the given time period. The Deviations view is useful for identifying patterns that occur cyclically and alerts you when those cyclic patterns have an unexpected change. You can use this view to show seasonal patterns in your data or patterns that occur on a monthly or weekly basis.

Figure 6-15 shows the Deviations view when you select the Product facet with Time scale set to Month, the Date facet set to the default of date, and Sort set to High frequency.

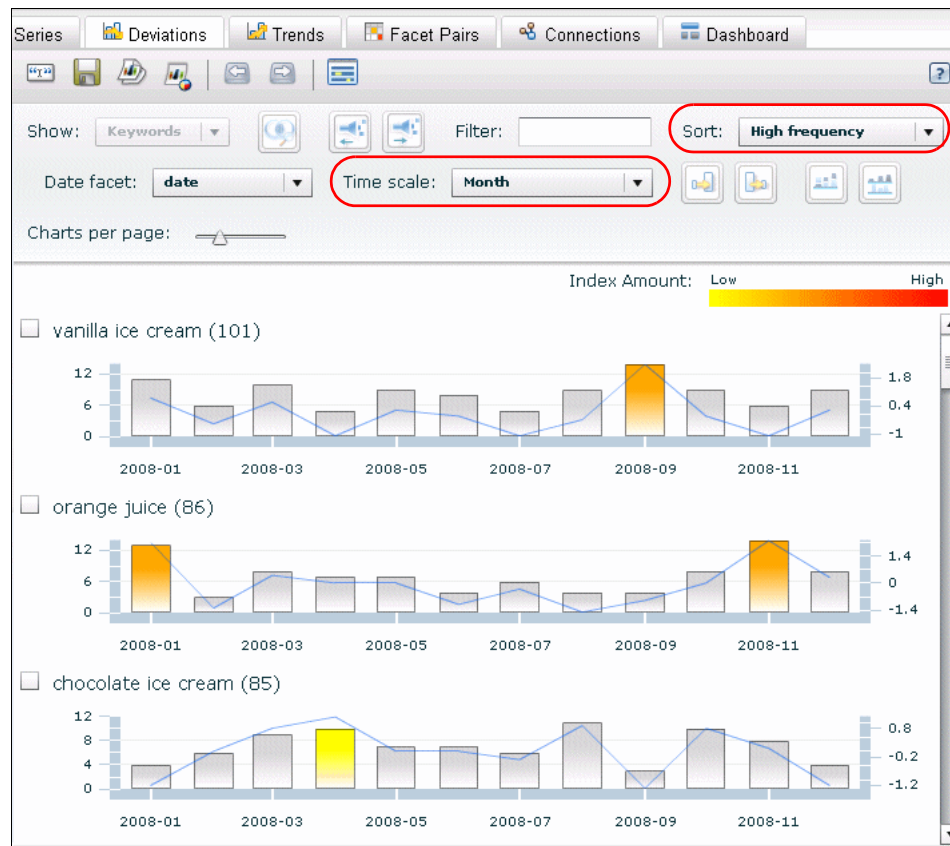


Figure 6-15 Deviations view showing sorting by high frequency and month

### 6.6.1 Features in the Deviations view

The following features are functionally the same as the features for the Trends view:

- ▶ Changing the Charts per page indicator
- ▶ Showing selected charts or showing all charts
- ▶ Combining selected charts or showing separate charts
- ▶ Zooming in and out

- ▶ Changing the Date facet
- ▶ Sort criteria
  - High frequency
  - High index
  - Latest index
  - Name (ascending)
  - Name (descending)

See 6.5, “Trends view” on page 175, for a detailed description of their functionality. Notice that the time scale feature in the Deviations view is identical to the same function in the Time Series view. As in the Time Series view, this time scale includes options for selecting the month of year, day of month, and day of week. You select these options when you want to see seasonal changes or monthly and weekly changes.

## 6.6.2 Understanding the Deviations view

In the Deviations view (Figure 6-16 on page 185), the frequency of the selected time period is shown as a bar chart and measured at the y-axis on the left side. The deviation index score is measured at the y-axis on the right side. The *deviation index score* is the standardized residual. It is referred to as *index* in the chart.

The deviation index score indicates how the actual value deviates from the expected value for a given time frame. The blue line in the chart shows the deviation index scores. The bar chart is displayed in color if the deviation index score is higher than the threshold, which indicates that the actual value is greater than the expected value.



For example, we select the Product facet and filter with the keyword “apple,” choose **High index** for Sort, and select **Month** for Time scale, as shown in Figure 6-16.

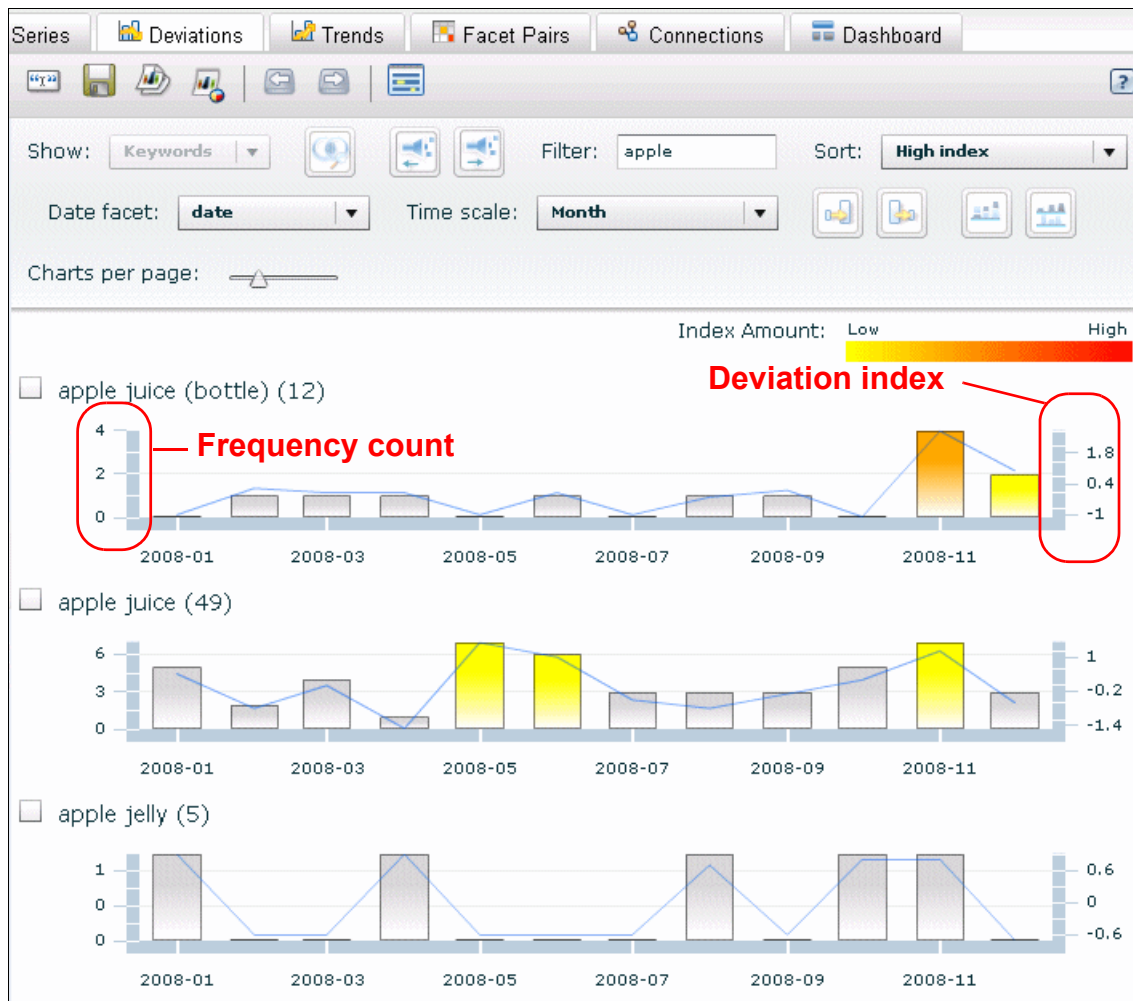


Figure 6-16 Deviations view showing the frequency count and deviation index

In the graph, the bar chart in 2008-11 (November 2008) for apple juice (bottle) is highlighted with orange, which indicates that the index amount is relatively high. Also, notice the yellow highlighted bars for 2008-12 for apple juice (bottle), and the graph below apple juice for the months 2008-05, 2008-06, and 2008-11.

Also notice that the frequency for apple juice (bottle) on 2008-11 is highlighted as orange with a frequency of 4. This frequency is less than the frequency of 7 for

apple juice on the same month 2008-11, which is highlighted in yellow. This result occurs because the deviation index score for apple juice (bottle) is higher than for apple juice and warrants the stronger red color.

How is the deviation index score calculated? It is calculated by using the frequency counts of each keyword in the given time period and the total number of frequency counts during the given time period.

If you go back to the Time Series view, select the **Product** facet, and select the Month as time scale. You see the following results:

- ▶ The total number of documents (all keywords) for 2008-11 is 78 from the Time Series view.
- ▶ The total frequency count (for the entire time period) for apple juice (bottle) is 12 from the Deviations view.
- ▶ The total frequency count (for the entire time period) for apple juice is 49 from the Deviations view.
- ▶ The frequency of 2008-11 for apple juice (bottle) is four from the Deviations view.
- ▶ The frequency of 2008-11 for apple juice is seven from the Deviations view.

The expected value for apple juice (bottle) and apple juice is calculated based on those values. It indicates which value is expected for the frequency of the keyword statistically. That is, the expected value multiplies the actual frequency of the selected keyword by the ratio of the selected keyword frequency in the entire frequency.

In this case, the expected value of 2008-11 for apple juice (bottle) is 1.0. (For 12 months, the total frequency count is 12, and therefore, the monthly count is about 1.0). The expected value of 2008-11 for apple juice is 4.5. (For 12 months, the total frequency count is 49. Therefore, the monthly count is a little more than 4).

The actual frequency value of 2008-11 for apple juice (bottle) is 4, which is greater than the expected value of the keyword “apple juice (bottle),” which is 1.0. Its delta ratio is greater than the one for apple juice. Notice that we do not compare the delta of the actual frequency and the expected value. Content Analytics calculates the deviation index score itself based on these values.

The bar chart is in color based on the deviation score index so that you can easily determine which keyword in the selected facet is worth further investigation.

### 6.6.3 When to use the Deviations view

The Deviations view is helpful when you want to see the deviation of the keyword for the selected facet within the given time period, such as month and day of the month. For example, you might want to see if the characteristics between Monday and Wednesday have any noticeable change when you look at the Product facet with day of the week selected as the time scale.

You can compare the deviation within the selected facet (that is the selected aspect). You can also see if you can find anything noticeable in that aspect with the given time scale, compared to the keywords that are found within the facet. In an earlier example, the deviation score index of 2008-11 for apple juice (bottle) is greater than the one for apple juice when you look at the data with the month time scale. Therefore, you might want to drill down the documents related to apple juice (bottle) and investigate why its deviation is noticeable compared to the other product found in the Product facet at 2008-11.

## 6.7 Facet Pairs view

The Facet Pairs view (Figure 6-17 on page 188) shows how the values of two different variables are related to each other. In this view, you select two facets from the Facet Navigation pane to see the correlations and frequencies of the different combinations of facet values.

This is a very useful and powerful view, especially as a starting point in your investigation process. With a couple of clicks, you can relate any two sets of values in a facet, such as showing a product list graphed against problem categories. In this scenario, you would quickly see how related particular problems are to specific products, seeing at a glance both frequency and (more importantly) correlation, for when a particular product sees an abnormally high number of instances for a problem type.

After you select two facets to analyze, you can also choose from the following three alternative displays of the facet pair comparison:

- ▶ Table view
- ▶ Grid view
- ▶ Bird's eye view

## 6.7.1 Table view

As shown in Figure 6-17, the Table view shows the selected two facets using a table style. By default, it is sorted by frequency. It is useful to sort by correlation so that you can focus on value pairs with the highest correlation values, which are more likely to represent interesting anomalies.

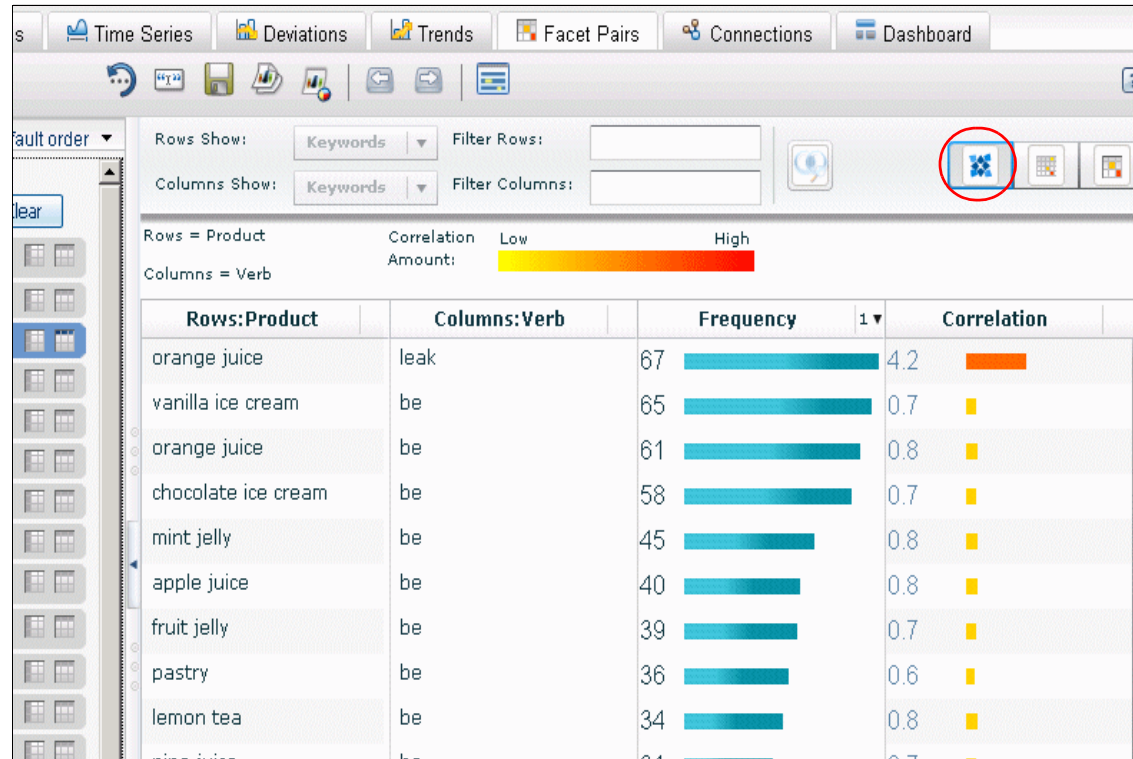


Figure 6-17 Facet Pairs view showing the Verb facet and the Product facet in Table view format

In Figure 6-17, we select the **Product** facet for Rows and the **Verb** facet for Columns and sort the result by frequency, not by correlation. As a result, the frequency of the document that contains both the keyword “orange juice” and “leak” is 67, and the correlation of those selected keywords is “4.2”. The Table view is most useful when you sort by frequency. The table view is also very useful to find which combination has the highest correlation value, when you sort by correlation.

The Table view is the default view when you initially select two facets to compare. You can change the default Facet Pairs view to another view format in the Preference window.

## 6.7.2 Grid view

You can see the correlation for the selected facets by using the Grid view. In the example shown Figure 6-18 on page 190, we select the **Product** facet for Rows and **Verb** facet for Columns. The cell that is the intersection of orange juice and leak is highlighted in orange, which indicates that the correlation value is greater than the threshold when compared to the other correlation values. That is, the leak issue has a high correlation with the orange juice product, or the orange juice product has a high correlation with the leak issue.

You might notice that two values found in the orange cell (67 and 4.2) are the same as those shown in the Table view in Figure 6-17 on page 188. The first row in the cell is the frequency (the number of documents) that contains both the selected keywords, and the second row in the cell is the correlation value.

You also see the numbers, such as 86 under the keyword “orange juice” and 123 under the keyword “leak.” These values are the frequency of each keyword and the number of documents returned by the selected keyword.

In this view, you can see the comparison values in table form by row and column, one facet for each dimension. The Grid view can calculate 100 x 100 cells of the table data, based on the highest frequency. Consequently, the top 100 most frequent keywords in one selected facet are displayed as rows. Also, the top 100 most frequent occurring keywords in the other selected facet are displayed as columns.

What if either of the selected facets has more than 100 keywords? The Facet Pairs view is constructed based on the assumption that the users want to see the highest frequency first because they usually contain the most interesting correlations. If you must oversee all of the data, instead consider using deep inspection, which is explained in 10.7, “Deep inspection” on page 387. Or, you can confirm the keyword connection by using the Connections view as explained in 6.8, “Connections view” on page 194.

Figure 6-18 on page 190 provides a Facet Pairs view showing the Verb facet and Product facet in the Grid view format.

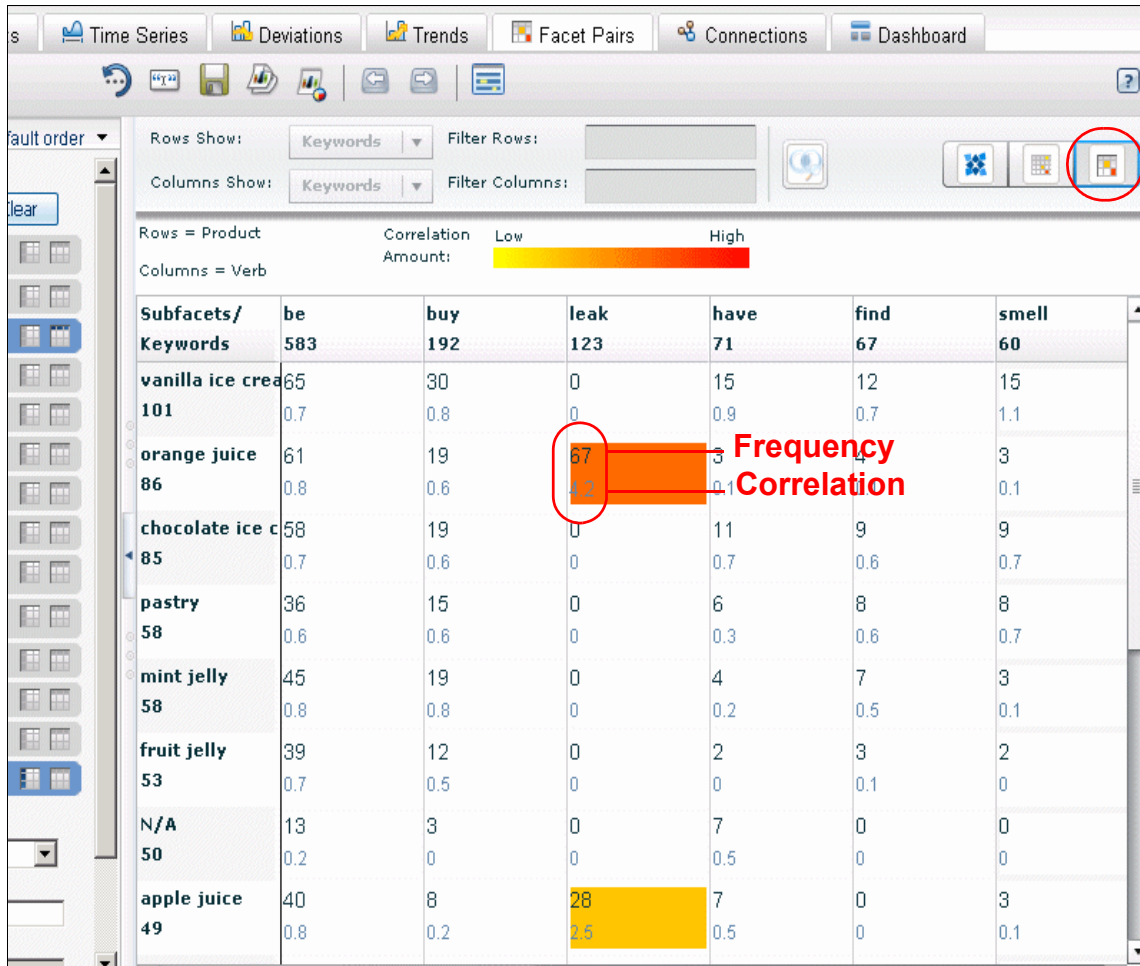


Figure 6-18 Facet Pairs view showing the Verb facet and Product facet in the Grid view format

### 6.7.3 Bird's eye view

By default, in the Grid view, you can only view a 15 x 15 celled area of the 100 x 100 table at a time. The use of the 15 x 15 viewing area is implemented for performance reasons to keep the number of required calculations low. With the Bird's eye view, you can select other areas of the table that you might want to see that are not in the default 15 x 15 viewing area.

As shown in Figure 6-19, the current 15 x 15 viewing area is displayed as a blue box in the upper left corner of the table. The dimensions of the table within the 100 x 100 limit are displayed as white cells. To select a different viewing area, move your cursor to where you want to start in the table, click and drag the blue box, and then click the table or the grid view to see the area.

Notice that the values in the individual cells (keywords, frequency, and correlation values) are displayed in a dialog box as you hover your cursor over the cell. Also, notice how the colors of highly correlated cells are maintained in the Bird's eye view to provide a convenient way to quickly locate areas of interest in the 100 x 100 table.

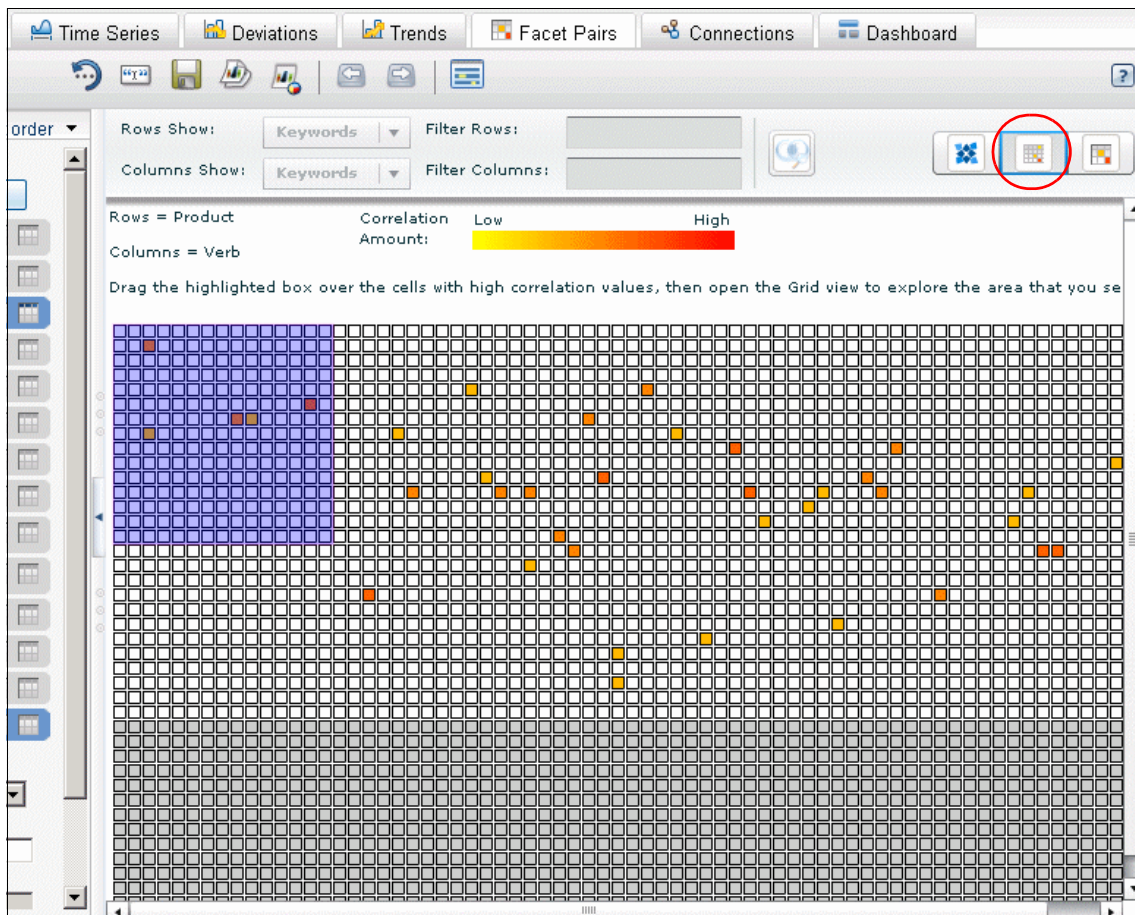


Figure 6-19 Facet Pairs view: Verb and Product facets in the bird's eye view format

## 6.7.4 Understanding the Facet Pairs view with correlation values

With the Facet Pairs view, you can identify a high correlation of keywords from the selected facets. Content Analytics requires two sets of search results to calculate a correlation. Accordingly, you select two facets that represent the two search result sets of the document set.

This section explains how the correlation value is calculated in Figure 6-19 on page 191 by using the Product facet as the row and Verb facet as the column.

### Calculating the correlation value

To calculate the correlation value, first you must know the following information:

- ▶ The total number of occurrences of “orange juice” is 86.
- ▶ The total number of occurrences of “leak” is 123.
- ▶ The entire number of documents in the corpus is 852.

Therefore, about 14% of the documents ( $123/852$ ) contain the keyword “leak.” This value is referred as the *density* of the keyword in a given document set.

Next, you must look at the intersection cells of the keywords, “orange juice” and “leak.” You see that the total number of occurrences of documents that includes both the keywords “orange juice” and “leak” is 67. Thus the density of the keyword “leak” in the document set for the keyword “orange juice” is 78% (that is  $67/86$ ).

When you calculate the correlation value, you must consider the ratio of these two density values as the correlation value:

- ▶ The density of the given set of documents that includes the specific keyword
- ▶ The density of the entire set of the documents in the whole collection

That is, you are interested in the ratio of the following items:

- ▶ The density of the keyword “leak” in the document set for the keyword “orange juice” ( $67/86 = 78\%$ )
- ▶ The density of the keyword “leak” in the entire document set ( $123/852 = 14\%$ )

In this example, the correlation value of orange juice and leak is calculated as 5.5 (roughly  $77\%/14\%$ ).

The correlation value calculated in this manner is not reliable especially when the number of documents, which includes both keywords, is relatively small. For example, the number of documents that includes keyword B is 2, while the number of documents that includes keyword C is 100. Then, consider the number of documents that include keyword A in both document sets. The density is 50% for both document sets. Which value is more reliable?



- ▶ The number of documents that includes keyword A and keyword B is 1 (50% of the document set that includes keyword B).
- ▶ The number of documents that includes keyword A and keyword C is 50 (50% of the document set that includes keyword C).

In this case, you must consider the first case (50% calculated by  $1/2$  is the document that includes keyword A and keyword B) is *less* reliable. Content Analytics takes into account such situations by applying a *reliability correction* that uses statistical *interval estimation* to make the calculated correlation more reliable. (Usually it makes the calculated correlation value smaller to some degree.)

With reliable correction, the earlier example correlation value of 5.4 becomes 4.2, as shown in Figure 6-17 on page 188 in the Table view. The correlation value 4.2 is higher than the normal threshold.

**Interval estimation:** The topic of interval estimation is outside the scope of this book.

**Correlation value:** In this example, the correlation value that is calculated from the given figure is much higher than the one that is displayed in the Table view. This result is normal because Content Analytics adjusts the correlation value to a smaller value to some degree by reliability correction. Remember that the data distribution is a sample data set and is not a real case. If the document set is relatively small and not reliable, the correlation value is reduced to be a more reliable value.

### 6.7.5 When to use the Facet Pairs view

The Facet Pairs view is useful when you want to compare facets of your collection and have Content Analytics show you how highly correlated they are to each other. Content Analytics highlights intersecting cells when two keywords are highly correlated. The Bird's eye view is used to review the entire table to quickly identify these highly correlated cells. You use the Grid view to focus on that specific area of intersecting cells.

After you discover the highly correlated keyword pairs, you can go back to the Documents view and look at the textual data (content of the document) and determine if there are any possible trends or insight there.

Remember that the Facet Pairs view only concentrates on the top most frequently occurring keywords. If you need to consider all of your data, use deep inspection as explained in 10.7, "Deep inspection" on page 387.

## 6.8 Connections view

The Connections view shows a graphical view of the relationship between keywords or subfacets within selected facet pairs. This view is another representation of the correlation of keywords within the selected facet pairs. The keyword is represented by a node, and the link between the nodes shows the correlation value between the two keywords.

Figure 6-20 shows an example of the Connections view when you select the Product facet and the Verb facet.



Figure 6-20 Connections view when selecting Product facet and Verb facet

**Representation in the Connections view:** Depending on the browser window size or your operation, the representation in the Connections view can vary even though the same facet pairs are selected.

In the Connections view, you can determine the highly correlated keyword pairs by focusing on the size of the node, link color, and link length:

- ▶ The Node shows the keyword or subfacet in the selected facet pairs:
  - Node size represents the frequency of the keyword or subfacet. The larger node size represents a higher frequency count in the entire document corpus.

- Node color indicates the selected facet where the keyword is located. Our example has two node colors: light blue and dark blue. The keywords “leak”, “use,” and “find” belong to the Verb facet (light blue), while the keywords “orange juice”, “apple juice,” and “chocola” belong to the Product facet (dark blue).
- When you move the mouse pointer over a particular node, the facet name, keyword, and frequency are displayed, as shown in Figure 6-21.

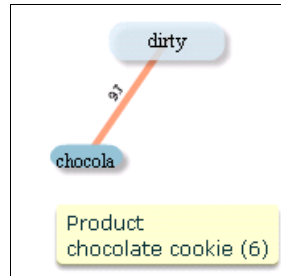


Figure 6-21 Tooltip showing the facet name, keyword, and frequency of the node

- ▶ Link color shows the rank of the correlation index. The link in red has a higher correlation value than a yellow link color.
- ▶ Link length reflects how tightly the two nodes are correlated. The higher that correlation is between two nodes, the shorter the length of the link is.

You can modify the rendering behavior of the Connections view from the **Connections** tab in the Preferences window. The following settings are among some of the settings that are configurable:

- ▶ Do not allow nodes to overlap
- ▶ Link length corresponds to correlation values
- ▶ Node size corresponds to frequency values

**Configuration in Preferences:** From the **Connections** tab in Preferences, you can also select the following attributes:

- ▶ Number of results to show for Facet1 per page
- ▶ Number of results to show for Facet2 per page
- ▶ Show the keywords or Sub facets by default for Facet1
- ▶ Show the keywords or Sub facets by default for Facet2

The Number of results to show for Facet1 or Facet2 is the number of facets used for the analysis. By default, 50 is set for both facets, which does not mean that 50 nodes are displayed in the search results. Fewer nodes are displayed in the search result if fewer nodes are involved in the correlation. Increasing the value can affect the performance of showing the Connections view user interface.

## 6.8.1 Features in the Connections view

The Connections view has the following features:

- ▶ Creating a window capture and saving it as an image file
- ▶ Resuming or pausing rendering
- ▶ Zoom in and Zoom out
- ▶ Zoom to fit the viewing area
- ▶ The AND operator
- ▶ Filter by correlation
- ▶ Node labels
- ▶ Highlight mode

### Creating a window capture and saving it as an image file

You can create a window capture when you want to save a specific connection representation by clicking the **Camera** icon (highlighted in Figure 6-22). Then, a window opens where you can select the location to save the image file and select the image file format.

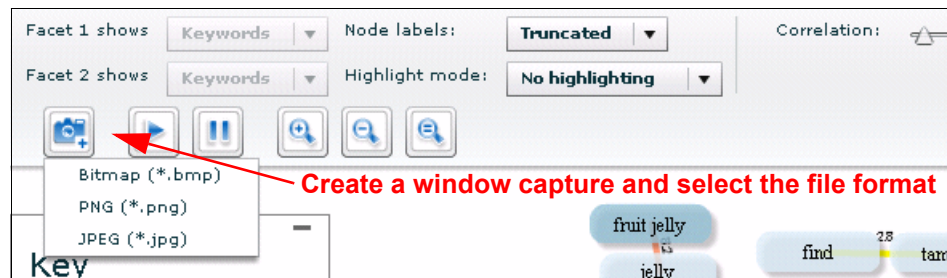


Figure 6-22 Create a window capture and save as image file features

## Resuming or pausing rendering

When you select two facets, the Node connections are animated, which takes a while to complete. You can resume or pause the animation in the middle of rendering when you click the **Resume rendering** or **Pause rendering** icons, which are shown in Figure 6-23.

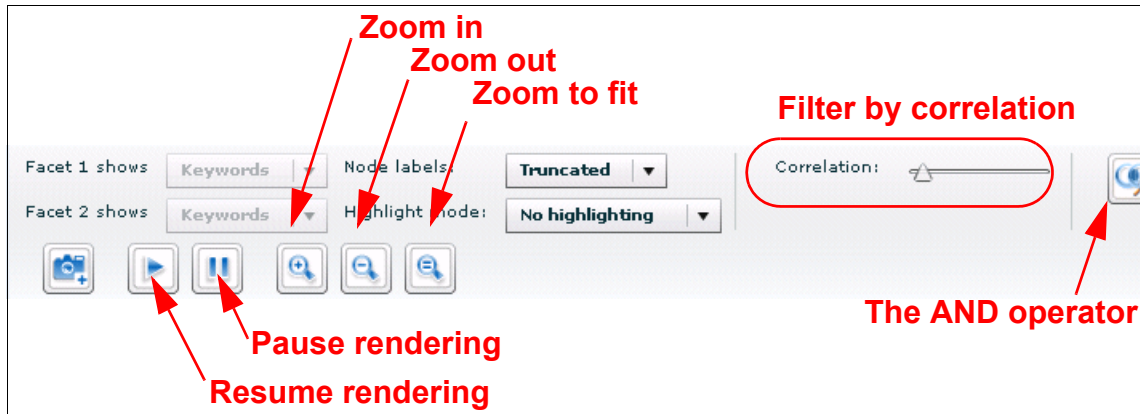


Figure 6-23 Features in the Connections view

## Zoom in and Zoom out

Sometimes the Connections view becomes busy when many nodes are to be displayed. You can zoom in and zoom out for specific node connections.

## Zoom to fit the viewing area

After you change the browser window size, the Connections view is redrawn if you select the Zoom to fit the viewing area feature. Content Analytics redraws the Connections view to fit the new size of the viewing area.

## The AND operator

Similar to other views, you can perform the AND operator search to narrow down the scope. The AND operator is helpful in narrowing down the data before interacting with the other views. First, select a specific keyword, and then click the **AND operator** icon.

## Filter by correlation

Sometimes the Connections view becomes very cluttered with values. By default, all keywords that have a correlation value greater than 2.0 are displayed. You can set the filter by a higher correlation value by using the slide bar. Only keyword pairs that contain a correlation value greater than the correlation value that you set are displayed in the Connections view.

For example, when you want to view the keywords that have a correlation value greater than 5.0, set the filter to 5.0. As a result, fewer nodes are displayed in the Connections view (Figure 6-24) than before the filter change (Figure 6-23 on page 197).

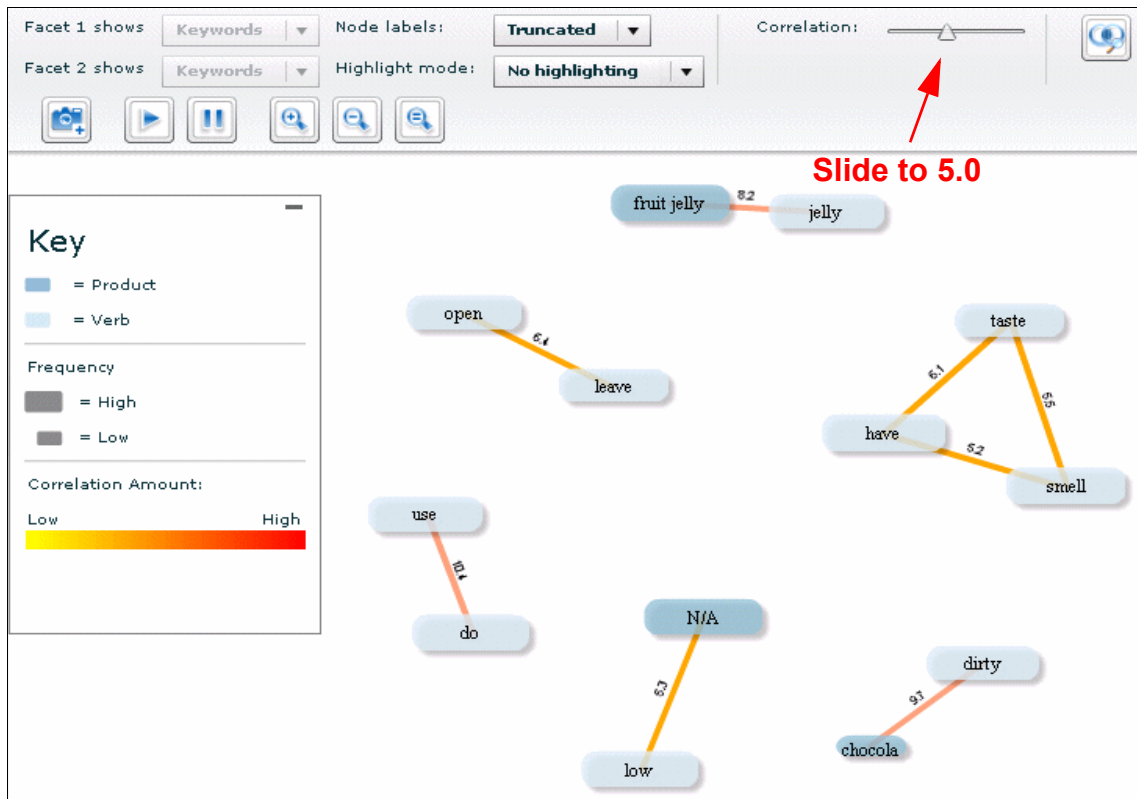


Figure 6-24 Connections view filtered by a correlation value of 5.0

## Node labels

You can select how the Node label is displayed. The options are Complete, Truncated, or None. By default, the **Truncated** option is selected. To display the full name of the keyword in the node, select the **Complete** option.

For example, the keyword “chocola” and the keyword “dirty” are displayed in the Connections view, as shown in Figure 6-24 on page 198. If you set the Node label field to **Complete**, the keyword “chocola” is changed to “chocolate cookie”, as shown in Figure 6-25.

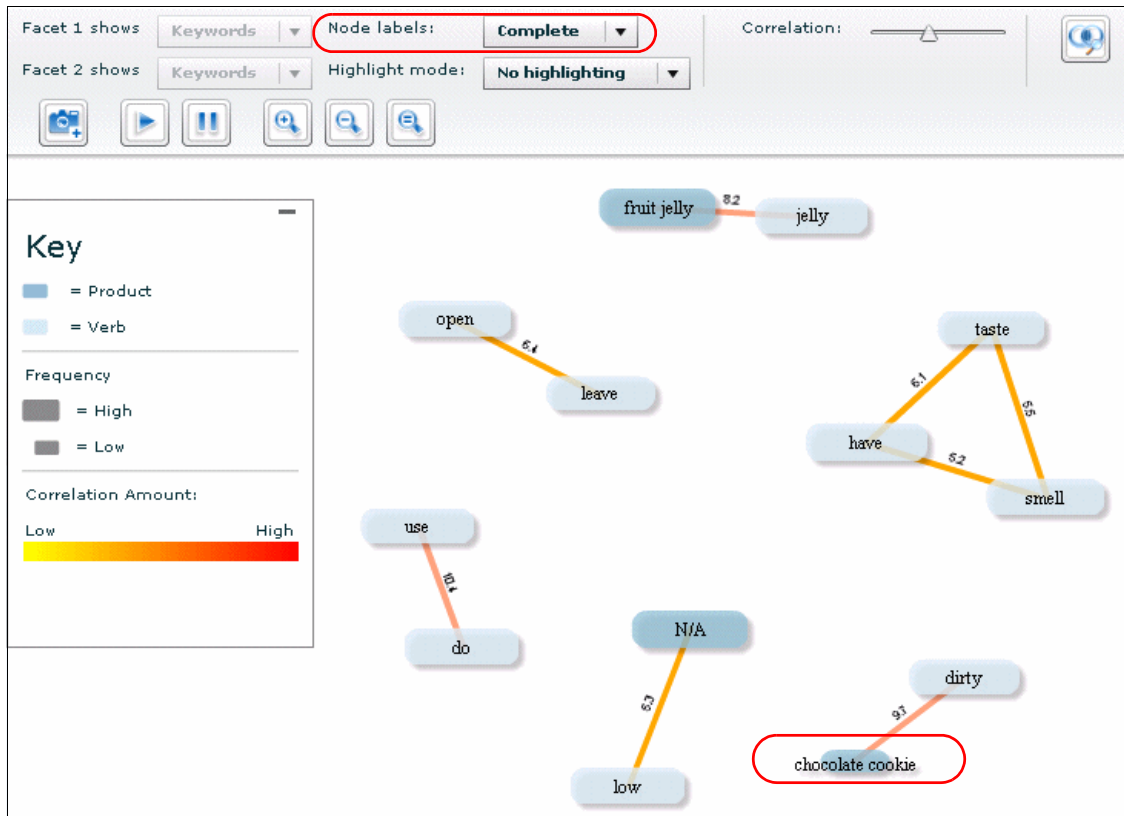


Figure 6-25 Selecting Node label as Complete in the Connections view

## Highlight mode

Changing the Highlight mode is useful when you want to focus on specific keywords in the Connections view. By default, the **No Highlighting** option is selected.

Suppose that you focus on the keyword “smell.” To see the direct connection to the keyword, select **Direct links only** for the Highlight mode field. In Figure 6-26, the direct connection to the keyword in the view is now highlighted, and other connections are unavailable.

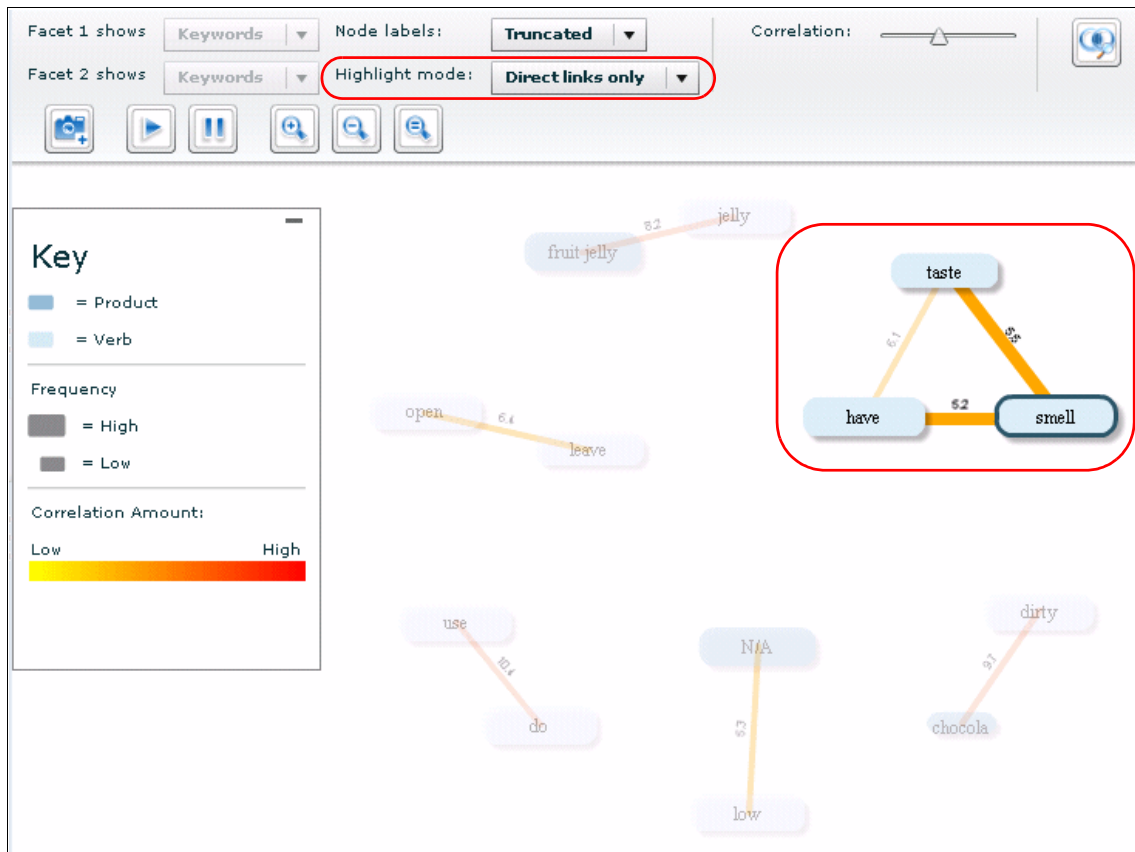


Figure 6-26 Selecting Direct links only for Highlight mode to view the node “smell”



In addition, when you select **All Links** for the Highlight mode field, all the links that are connected to the keywords are highlighted, as shown in Figure 6-27. In our example, the connection between the keyword “have” and the keyword “taste” is now highlighted.

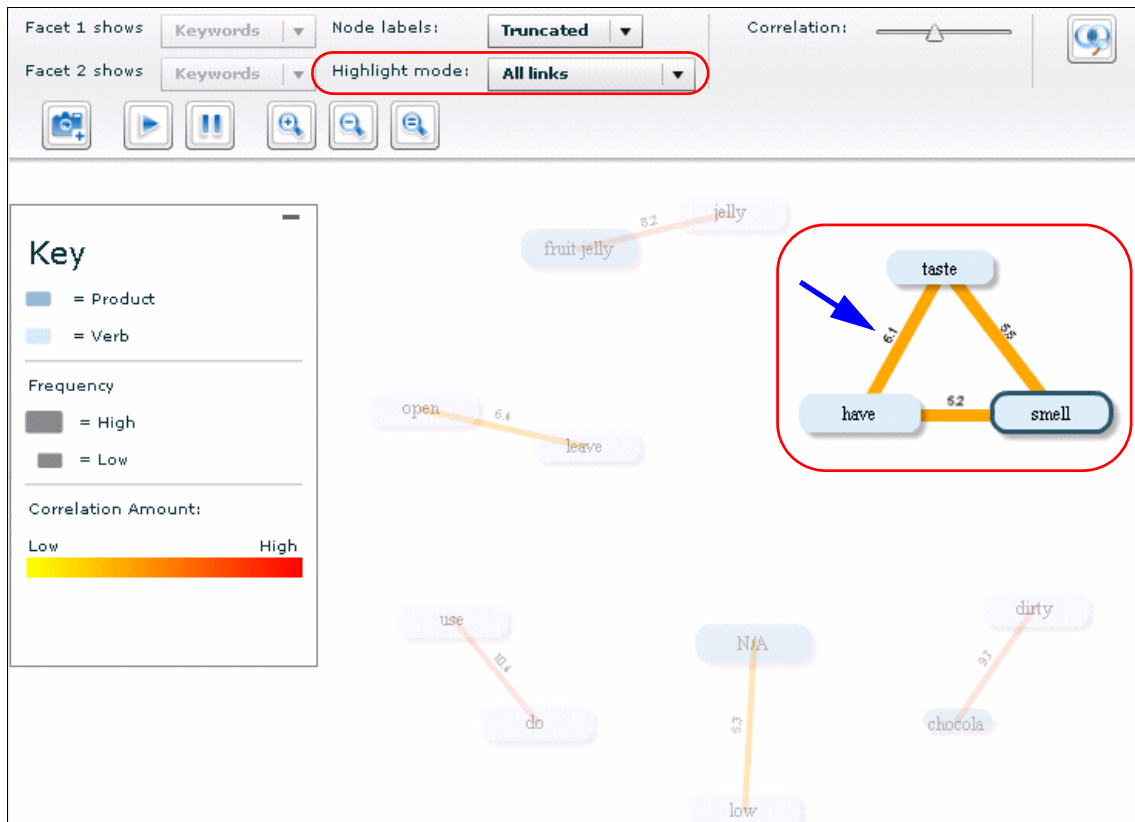


Figure 6-27 Selecting All links for Highlight mode to view the node “smell”

## 6.8.2 Understanding the Connections view

The Connections view helps you to identify the high correlated keywords based on the selected facets. As explained in the 6.7, “Facet Pairs view” on page 187, Content Analytics requires two sets of search results to calculate the correlation value.

This section helps you to interpret the results of the Connections view and understand how the Connections view is created. In the example in this section, two facets are selected: Product and Verb.

## Understanding the values that are displayed

This section explains the meaning of the values that are shown in the Connections view. In Figure 6-28, the keyword “leak” has a higher frequency compared to the other keywords shown in the figure. The node size represents the number of occurrences for the keyword in the result set.

The keywords “orange juice” and “apple juice” have the same blue color, while the keywords “leak” and “drink” have the same light blue color. The color indicates which facet the keywords represent. Thus, you can conclude that the keywords “orange juice” and “apple juice” belong to the same facet (Product). The keywords “leak” and “drink” belong to the same facet (Verb), but the facet is different from the Product facet.

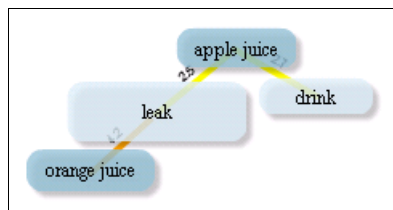


Figure 6-28 Connections view example

When you see the connection between the nodes, you notice that the connection between the keywords “orange juice” and “leak” is an orange color. It is also shorter as compared to the other connections (between the keywords “leak” and “apple juice” or between the keywords “apple juice” and “drink”). Thus, the correlation value between the keywords “orange juice” and “leak” is higher than others. You might want to investigate this relationship further.

## How the Connections view is created

As described in the previous section, Content Analytics requires two sets of search results to calculate the correlation value. Content Analytics uses the same calculation mechanism in the Connections view as described in 6.7, “Facet Pairs view” on page 187. However, it considers all possible combinations of the selected facet pairs.

For example, consider calculating the correlation value between the nodes “orange juice” and “leak” and between the nodes “apple juice” and “leak”, as shown in Figure 6-28. You can see the same correlation value between these keywords when you select the Product facet and the Verb facet in the Facet Pairs view, as shown in Figure 6-29 on page 203. Likewise, the correlation value between the nodes “apple juice” and “drink” is calculated with the same facet pairs, Product facet and Verb facet.

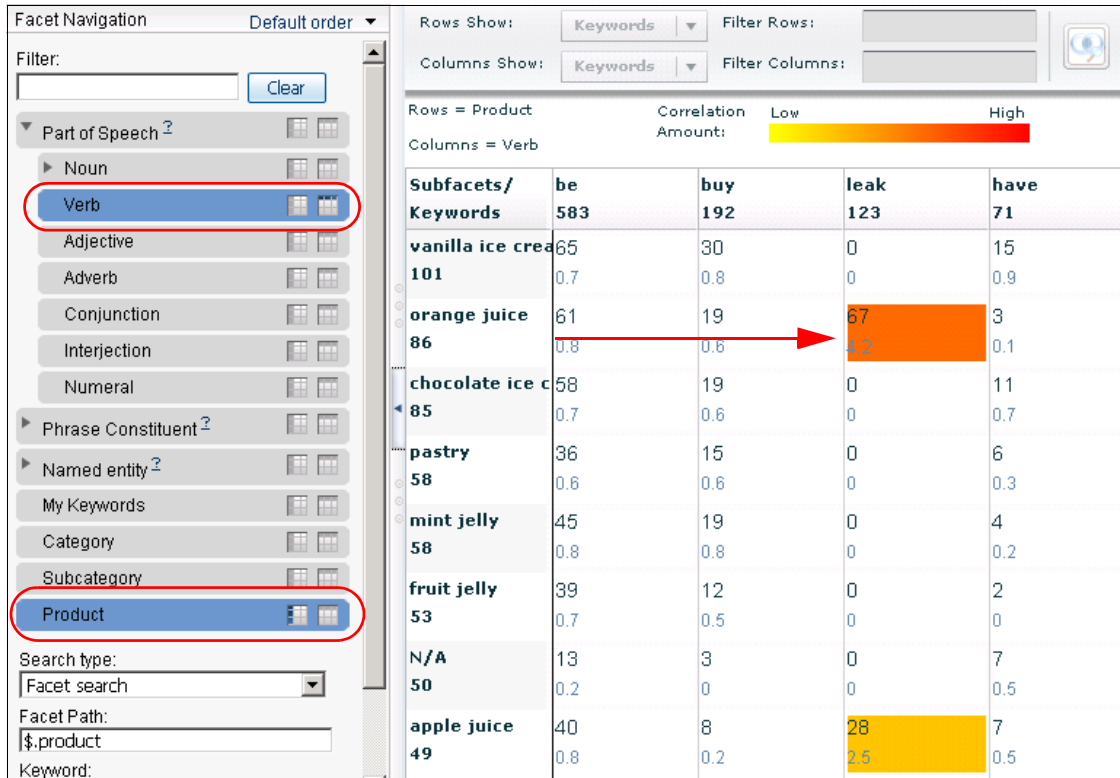


Figure 6-29 Grid view in the Facet Pairs view when selecting Product and Verb

However, the connections sometimes have a different correlation value than the Facet Pairs view. The difference in values occurs when keywords from the same facet are compared against each other, such as “smell” and “taste”, “smell” and “have”, “have” and “taste” from the Verb facet, as shown in Figure 6-30.

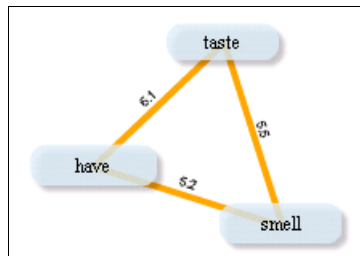


Figure 6-30 Connections view in the same Verb facet

How is the Connections view created in this case? Content Analytics uses the same calculation as the Facet pairs view, but considers all three combinations for the selected facet pairs. In this case, Content Analytics uses the facet pairs, but selects the Verb facet for both the vertical facet and horizontal facet, as shown in Figure 6-31.

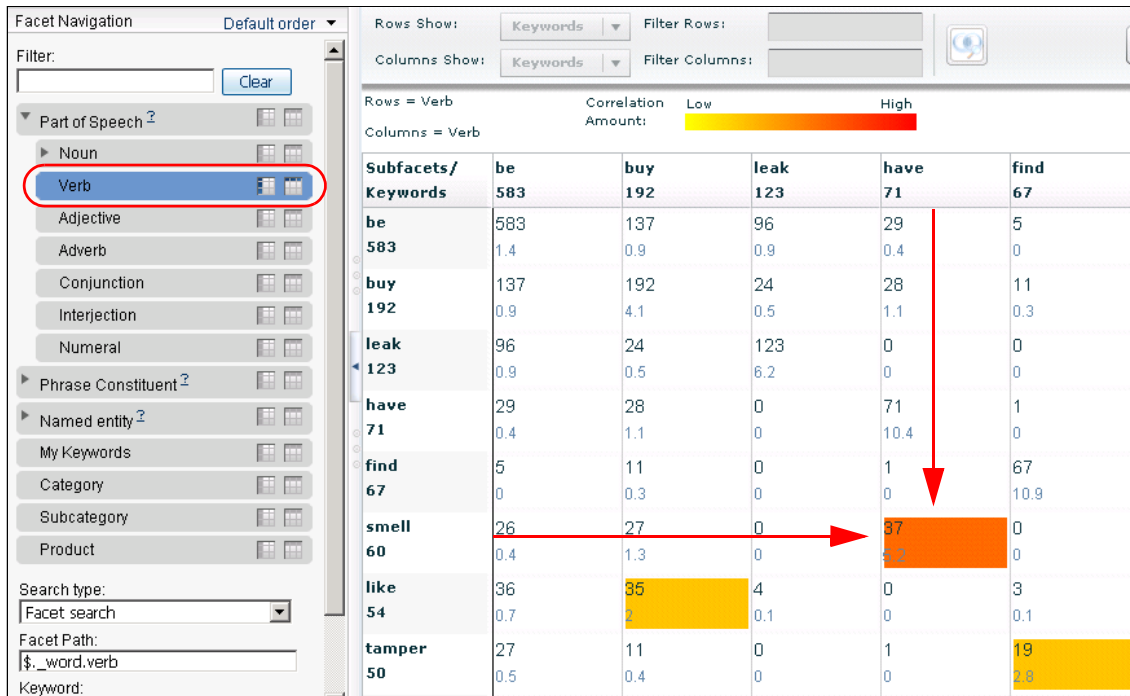


Figure 6-31 Grid view in the Facet Pairs view when selecting Verb and Verb

The remaining combination of the facet pairs in this example relates to selecting the Product facet for both the vertical facet and the horizontal facet. However, as shown in Figure 6-32, these correlation values are not higher than the threshold (2.0 by default). As a result, the connection between these keywords from the Product facet is not displayed.

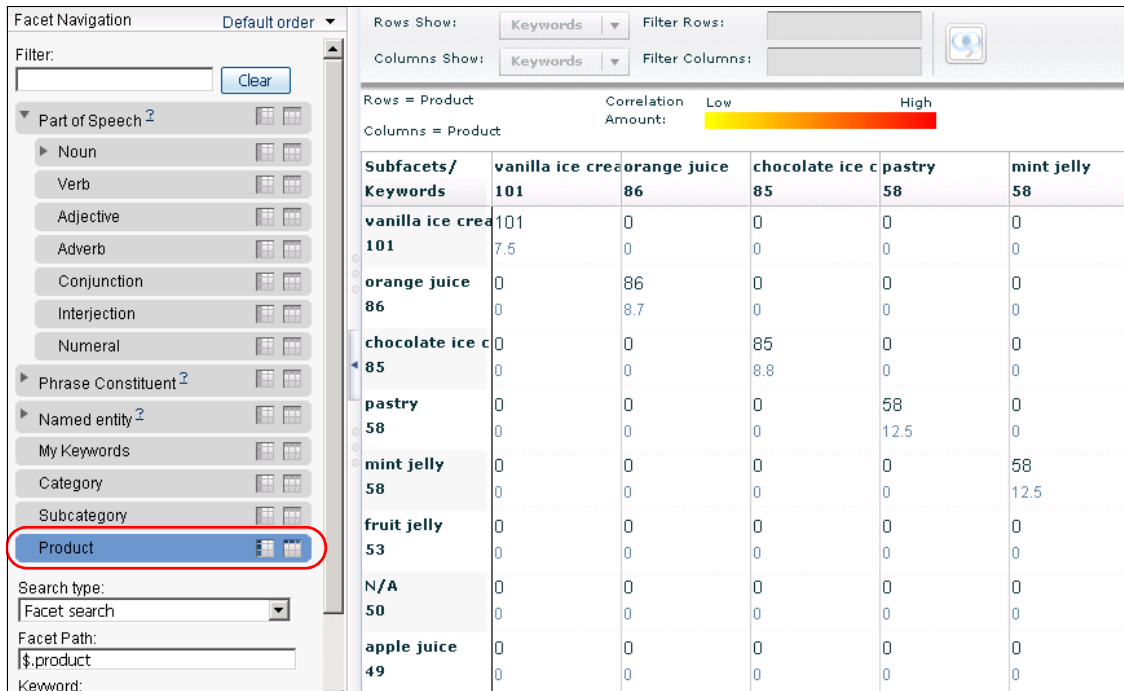


Figure 6-32 Grid view in the Facet Pairs view when selecting Product for the horizontal and vertical facet

The Connections view shows the correlation value automatically. You do not need to open the Grid view in the Facet Pairs view and select a different facet each time. You can see how the keywords are correlated with each other immediately so that you can concentrate on highly correlated keywords.

### 6.8.3 When to use the Connections view

With the Connections view, you can see the keyword cluster based on the correlation. In addition, you can find the hints or “hidden connections” between keywords because the Connections view shows the correlation value for the selected facet pairs and all possible variations of the selected facet pairs.

For example, consider the case when you select the Category facet and the Adjective facet in the Connections view for the sample collection used in this

chapter. There is a high correlation between the “strange” keyword and the “Taste/smell” keyword. A high correlation between the “sour” keyword and the “Taste/smell” keyword is displayed. These same high correlation keyword pairs are shown in the Facet Pairs view. However, in the Connections view, you also see the correlation between the “strange” keyword and “sour” keyword. In this case, the three keywords “Taste/smell”, “strange,” and “sour” are connected. Therefore, you can easily see that the user reports a problem when the product taste or smell is sour.

Based on this information, you might drill down into the documents to understand the situation further. When you individually look at the keyword pairs of “sour” and “Taste/smell” or “strange” and “Taste/smell,” you might not think that the user’s report of a “sour” taste as being something unusual. However, in the Connections view, you can easily see the connection through the visual representation of the keyword connections.

You can see the same results in the Facet Pairs view as described earlier. However, you cannot see the correlation of keywords at one time when you use the Facet Pairs view. Sometimes you cannot see the keyword pairs with the default configuration in the Grid view (such as “apple juice” and “drink”). To determine this information using the Facet Pairs view, you must use the Bird’s eye view, which involves extra steps. However, you can find the highly correlated keywords in the Connections view without changing windows.

The Connections view shows the “connection” based on the correlation value between all keywords found in the selected facets, and it is easy to filter the result by correlation value. You might see the trends of keyword clusters. As in the Facet Pairs view, after you discover the highly correlated keyword pairs, you can open the Documents view to further investigate the content of the document.

## 6.9 Dashboard view

The Dashboard view shows various predefined charts and tables in a single text mining view. With this view, you can visualize various aspects of the data to quickly interpret, analyze, and further investigate. In addition, you can save the images in the bitmap, PNG, or JPEG format so that you can easily share the data with other people for collaboration purposes. Unlike a static report, the results of these Dashboard charts are dynamically updated whenever you change a filter (through a view or search expression), which updates the current results set.

The administrator can customize the predefined Dashboard layouts. Then, the user can select the Dashboard layouts for viewing, saving them as images and analyzing them further. By default, you can use two preconfigured layouts. These default layouts include Layout 1 and Layout 2. The administrator can add

additional layouts and customize them based on business requirements. Figure 6-33 shows an example of the default layout, Layout 2, as it is displayed in the Dashboard view.

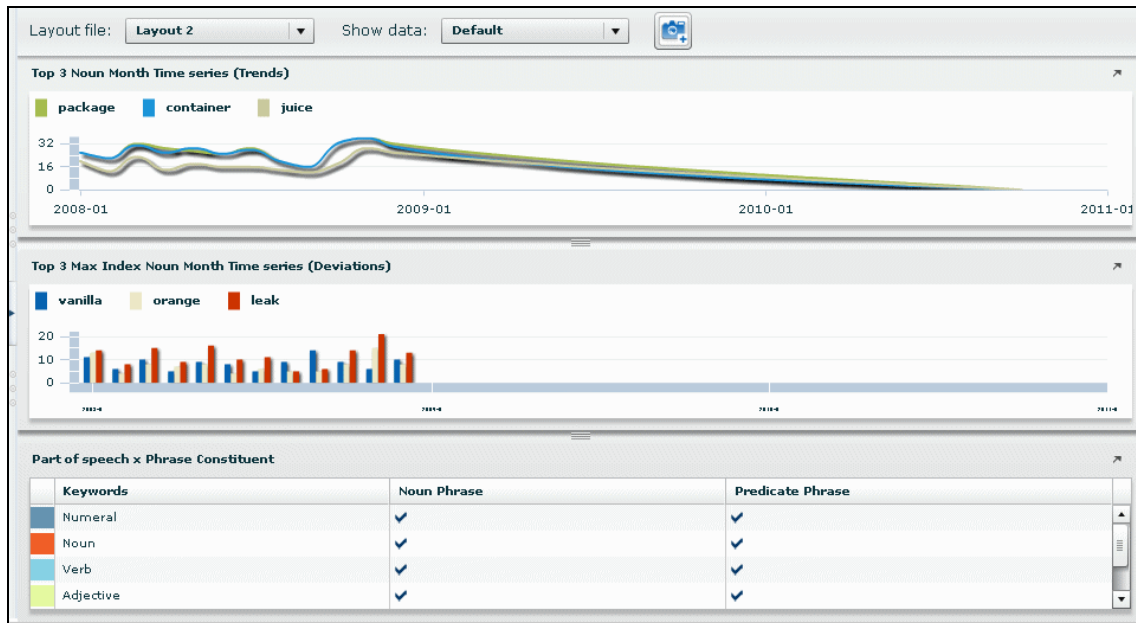


Figure 6-33 Default Layout 2 in the Dashboard view

## 6.9.1 Configuring the Dashboard layout

In this scenario, a new Dashboard layout is configured that contains four separate charts or tables to display data from the sample collection. The new Dashboard layout includes a bar chart, facet table, pie chart, and column chart. In this scenario, we set up each chart to contain separate data.

### Creating a layout

The Dashboard layout consists of one or more horizontal or vertical containers. First, you must set up how you want the containers organized in the layout and add panels to each of them. In this scenario, we create a layout that consists of a chart or table in each cell of the layout containing two columns and two rows.

To create a layout, follow these steps:

1. In the administration console, click the **Analytics Customizer** tab to open the Analytics Customizer application.
2. Click the **Dashboard** tab.

3. On the Dashboard page (Figure 6-34), complete the following actions:
  - a. Select the collection that you want to associate with the new layout in the Collection name field. In this scenario, for Collection name, select **Sample Text Analytics Collection**.
  - b. For Layout, select **New**.
  - c. For Title, type Problem Report as the name for the layout.

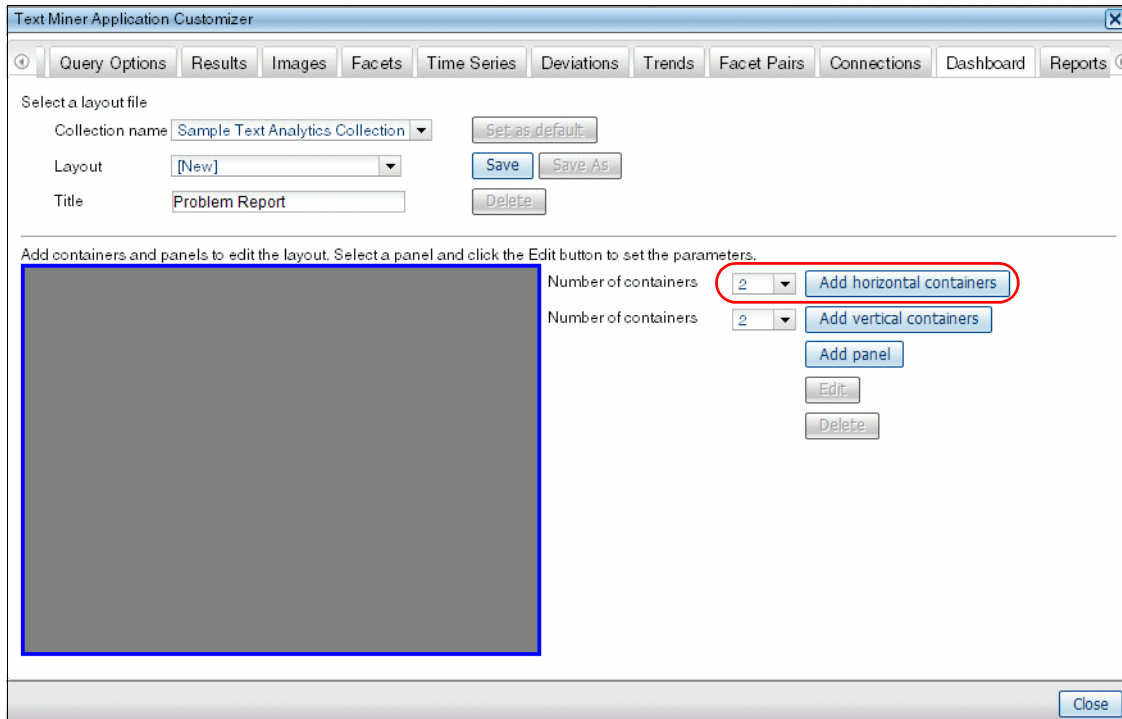


Figure 6-34 Adding two horizontal containers to the new Problem Report Dashboard layout

The layout consists of containers. Each container needs to contain a panel. If you do not add a panel to each container, you cannot save the new layout.

- d. For Number of containers for the horizontal field, select **2**.



- e. Click **Add horizontal container**. You now have two containers of equal size, as shown in Figure 6-35.
- f. Click the leftmost horizontal container that you just created to select it, and select **2** for the Number of containers field for the vertical container (Figure 6-35).

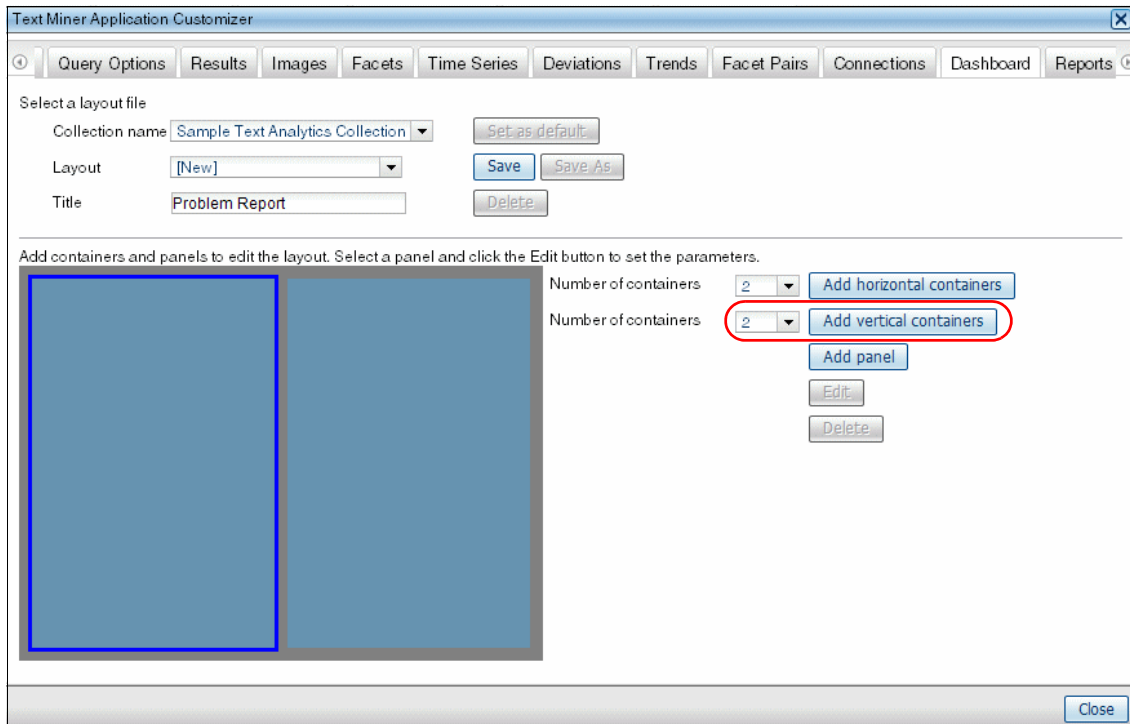


Figure 6-35 Adding two vertical containers to the left container

4. Click **Add vertical container**. You now have three containers. For this scenario, we want to show four charts. Therefore, you need to add one more container to the layout.

5. Click the rightmost horizontal container to select it, and select **2** for the Number of containers for the vertical container, as shown in Figure 6-36.

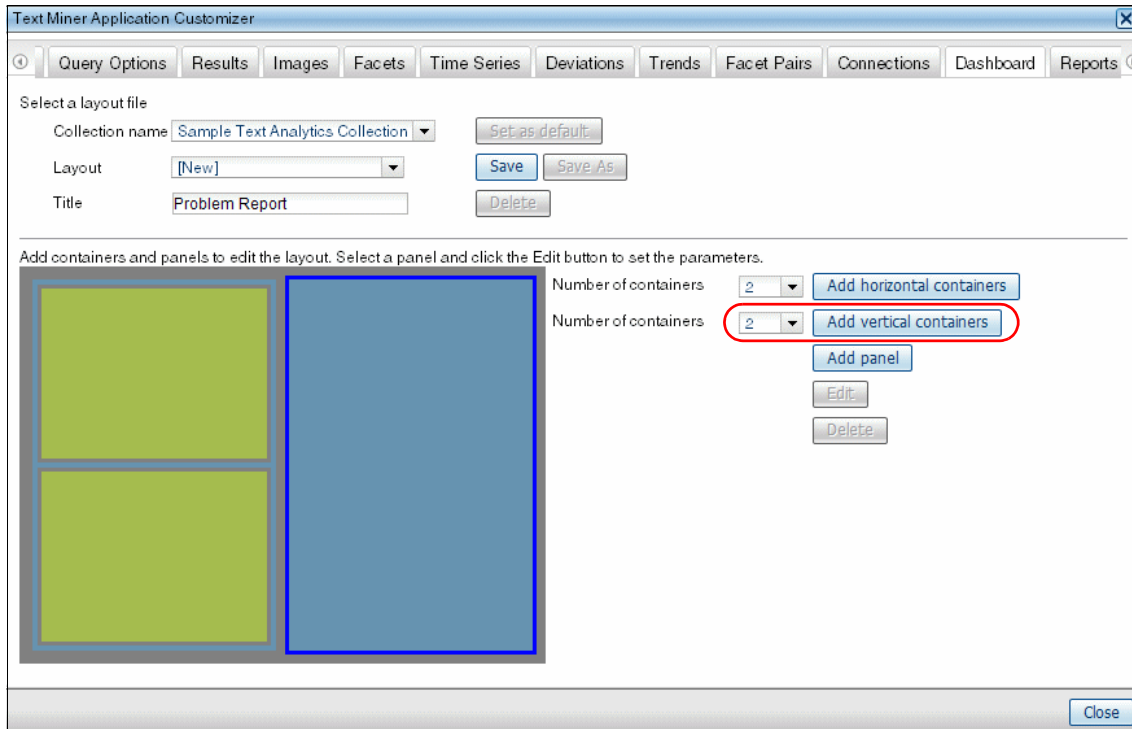


Figure 6-36 Adding two vertical containers to the right container to create a four-container layout

6. Click **Add vertical container**. You now have four containers of equal size, as shown in Figure 6-37.

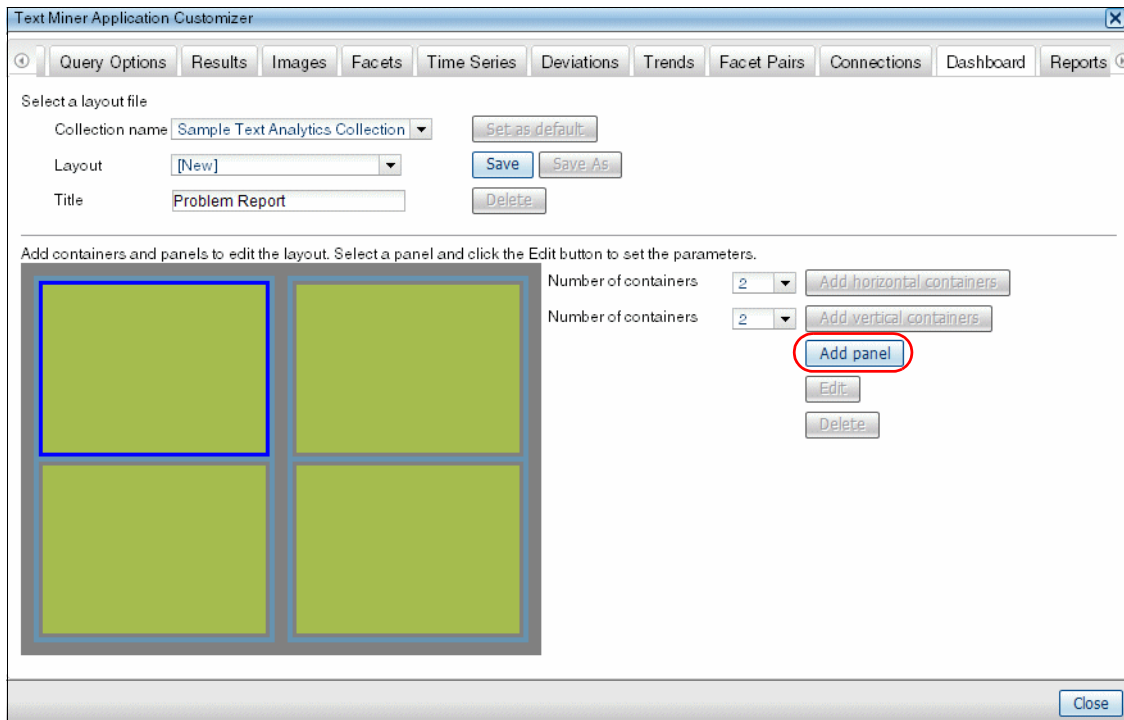


Figure 6-37 Adding a panel to the upper left container of the new layout

7. Click the upper-left container, and click **Add panel**.
8. Repeat step 7 for each container so that each of the four containers contains a panel. This action adds a panel to each container with the default settings for a bar chart.

The following sections explain how you can edit the panel to create your specific charts and tables.

### Creating a bar chart

In this section, you configure the new layout to contain a bar chart in the upper-left panel of the new layout view. You set the data to be displayed in the bar chart to be the frequency of the top category facet values. For more information about the Dashboard configuration options, see the following address:

<http://www-01.ibm.com/support/docview.wss?uid=swg21420024>

To create the bar chart, follow these steps:

1. Click the upper-left panel to select it and click **Edit** (Figure 6-38).

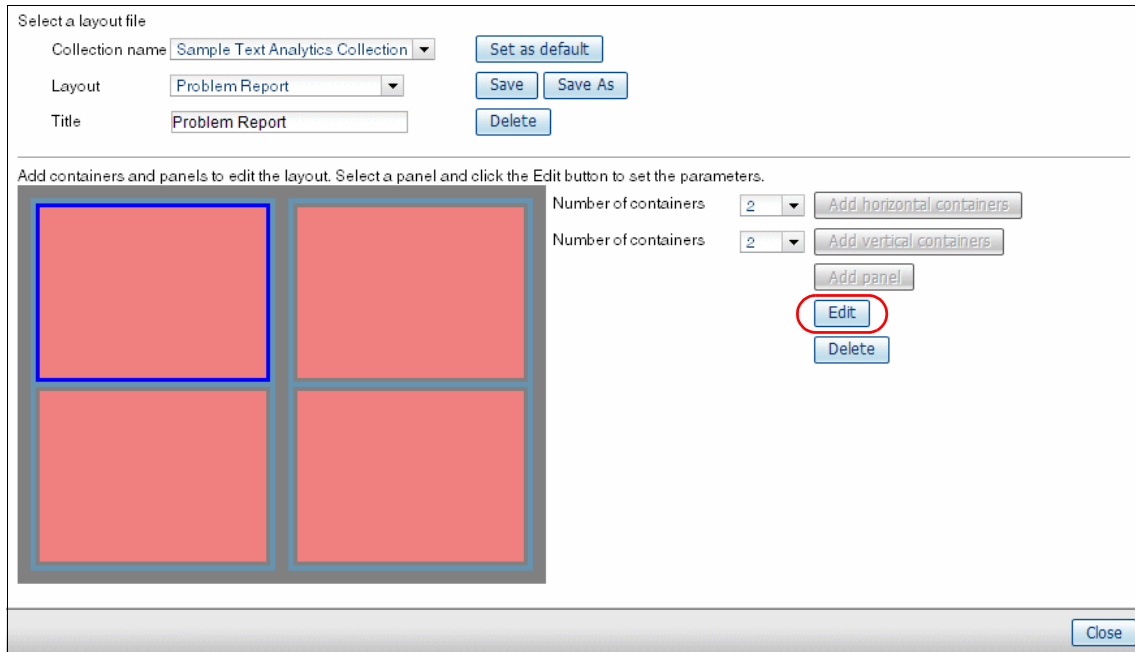


Figure 6-38 Dashboard layout panel to configure the bar chart

2. In the Panel properties window (Figure 6-39), complete the following steps:
  - a. For the Title field, type Top 5 Frequent Categories.
  - b. For the Type field, select **Bar chart**.
  - c. For the Facet ID field, click **Select** and click **Category**.
  - d. Click **OK**.

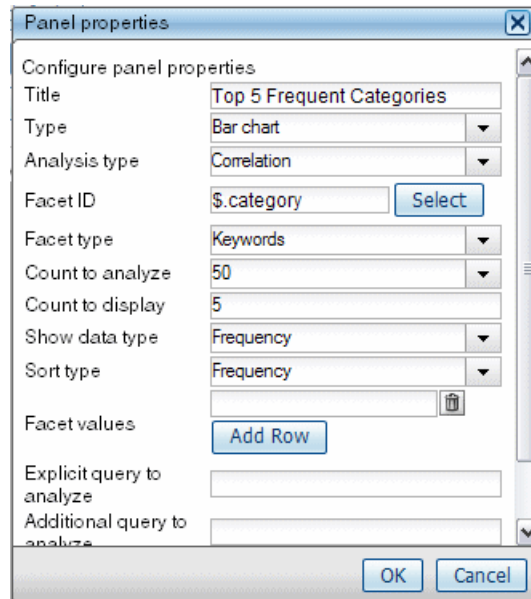


Figure 6-39 The bar chart panel properties

You do not see the results of the bar chart until all containers are configured. For an example of the completed bar chart, see the upper-left container in Figure 6-43 on page 219.

## Creating a facet table

In this section, you configure the new layout to include a facet table in the upper-right panel of the layout view. You set the data to be displayed in the facet table as the top correlated verbs.

To create a facet table, follow these steps:

1. Click the upper-right panel to select it and click **Edit** (Figure 6-40).

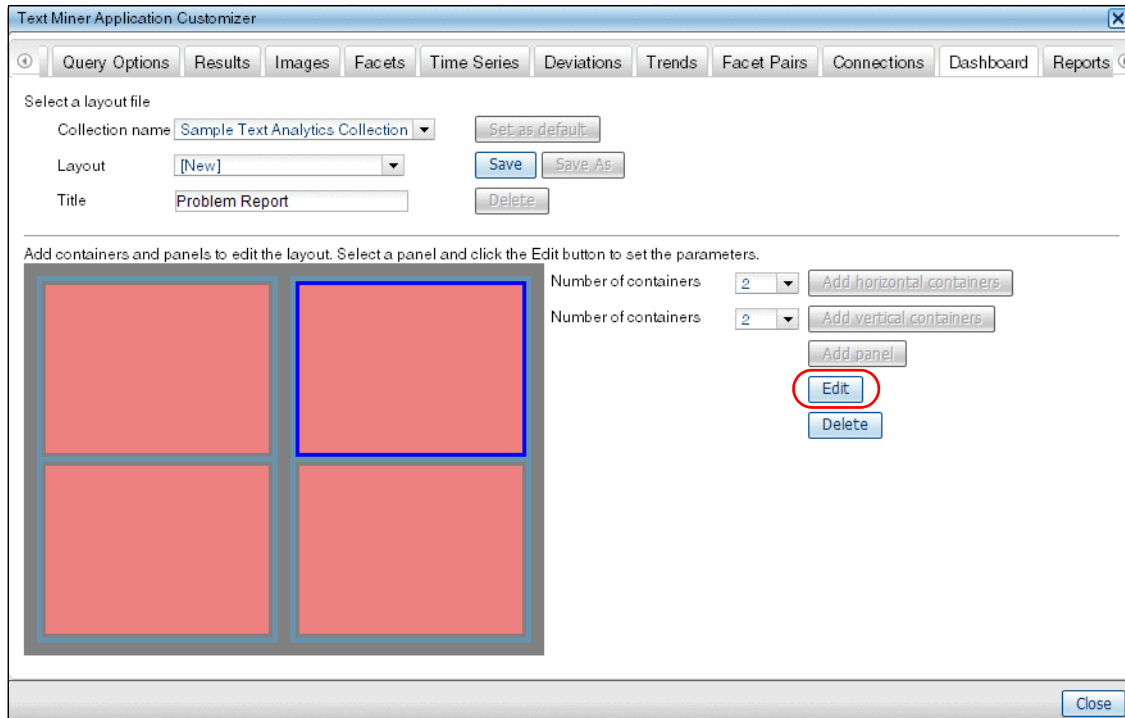


Figure 6-40 Dashboard layout panel to configure the facet table

2. Add the field values as shown in Table 6-1. Keep the default values for all other fields.

Table 6-1 Panel properties for the top five correlated verb facets

Field	Value
Title	Top 5 Correlated Verbs
Type	Facet Table
Facet ID	Select <b>Part of Speech</b> → <b>Verb</b>
Show data type	Correlation or index
Sort type	Correlation

3. Click **OK**.

You do not see the results of the facet table until all containers are configured. For an example of the completed facet table, see the upper-right container in Figure 6-43 on page 219.

## Creating a pie chart

In this section, you configure the new layout to contain a pie chart in the lower-left panel of the layout view. You set the data to be displayed in the pie chart as the correlated values of the Product facet.

To create a pie chart, follow these steps:

1. Click the lower left panel to select it and click **Edit** (Figure 6-41).

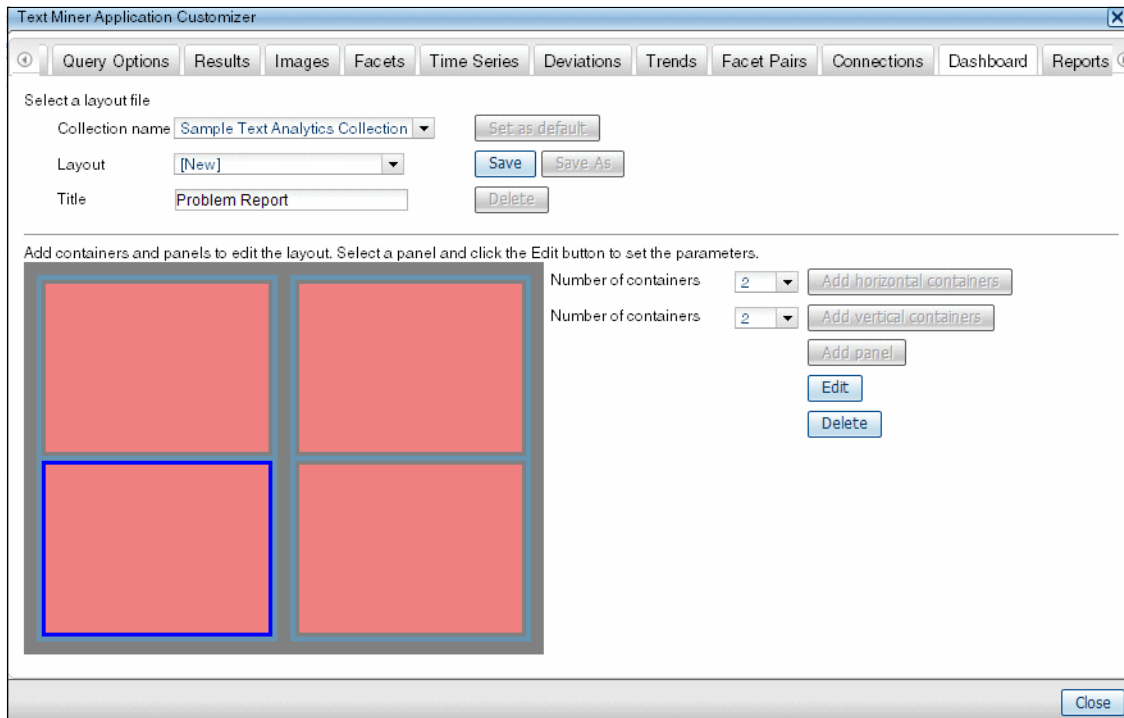


Figure 6-41 Dashboard layout panel to configure for the pie chart

2. Add the field values as shown in Table 6-2 and keep the default values for all other fields.

*Table 6-2 Panel properties for the top five correlated products pie chart*

<b>Field</b>	<b>Value</b>
Title	Top 5 Products by Correlation
Type	Pie Chart
Facet ID	Select <b>Product</b>
Show data type	Correlation or index
Sort type	Correlation

3. Click **OK**.

You do not see the results of the pie chart until all containers are configured. For an example of the completed pie chart, see the lower-left container in Figure 6-43 on page 219.

### **Creating a column chart**

You configure the fourth panel to contain a column chart in the lower-right panel of the layout view. You set the data to be displayed in the column chart to the frequent values of the subcategory facet for documents that contain the term “juice”. The data set is narrowed to documents that contain the term “juice.” Then, the top 50 most frequent subcategory values are analyzed, and the top three subcategories are displayed in the chart.



To create a column chart, follow these steps:

1. Click the lower right panel to select it and click **Edit** (Figure 6-42).

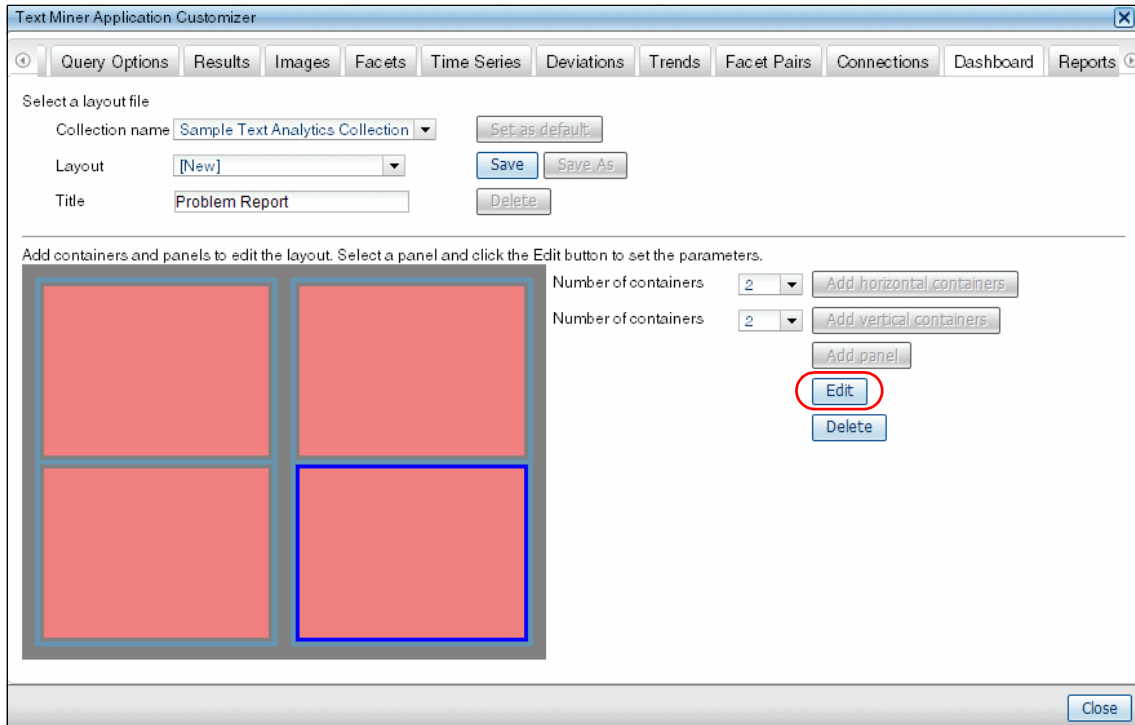


Figure 6-42 Dashboard layout panel to configure for the column chart

2. Add the field values as shown in Table 6-3 and keep the default values for all other fields.

Table 6-3 Panel properties for the top three frequent subcategories column

Field	Value
Title	Top 3 Frequent Subcategories Containing Juice
Type	Column Chart
Facet ID	Select <b>Subcategory</b>
Count to display	3
Explicit query to analyze	juice

3. Click **OK**.

You do not see the results of the column chart until all containers are configured. For an example of the completed column chart, see the lower-right container in Figure 6-43 on page 219.

### **Saving the Dashboard layout**

Now that you configured the new Dashboard layout, you need to save it:

1. Click **Save**.
2. Click **Close** to close the Dashboard configuration window.
3. Click **Save Changes**. Otherwise, the changes might not fully save to the content analytics miner.
4. Click **Exit** to exit the Analytics Customizer application.
5. In the message window that indicates that the window will close, click **OK**.

## **6.9.2 Viewing the Dashboard**

After an administrator sets up a new dashboard layout, a user can view it and work with it through the content analytics miner:

1. Open the content analytics miner.
2. For this scenario, select the **Sample Text Analytics Collection**. If this collection is not displayed at the top of the content analytics miner, click the **change** link and select the **Sample Text Analytics Collection**. Click **Save**.
3. Click the **Dashboard** tab. In the Layout File field, select the **Problem Report** layout.

The layout that you created in 6.9.1, “Configuring the Dashboard layout” on page 207, is now shown in the window. The four charts and tables contain the analytic data of the collection, as shown in Figure 6-43. If a data value is listed in more than one chart or table, the color associated with that data value is the same across the multiple charts and tables.

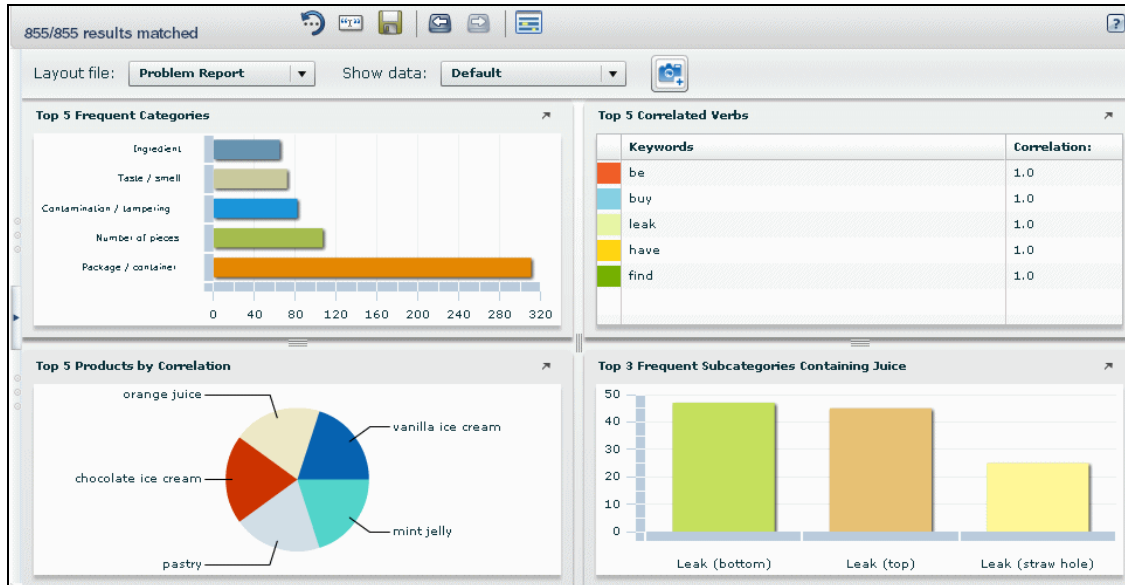


Figure 6-43 Problem Report layout in the Dashboard view

## 6.9.3 Working with the Dashboard

The Dashboard provides additional useful functionality. For example, you can further narrow down the data set, display correlation and frequency values for a data point, enlarge a table or chart, and set user preferences.

### Narrowing the search results

With the charts and tables in the Dashboard layout, you can further narrow down your data set to focus on an area of interest. In this scenario, you click the **Package/container** bar in the “Top 5 Frequent Categories” chart. As a result, the following syntax is automatically added to the query statement:

```
/"keyword$.category"/"Package / container"
```

Notice that all the charts and tables have changed, except for the column chart. Also, only the documents that contain a category equal to Package/container are now displayed. Because the column chart has the term “juice” set as the explicit query to analyze, it is not modified by any additional query statements that a user includes. If you want the user’s query statement to be appended to the query defined for the chart, set the query in the additional query to analyze the field when editing the panel in the dashboard layout.

To view the query syntax area, click the **Expand this Area** icon at the top of the window. Figure 6-44 on page 221 shows the addition to the query syntax and the updated charts and tables.

### Frequency and correlation display

When you move the mouse pointer over a data point on the chart or table, you see the frequency and correlation values for that particular data point. In this scenario (Figure 6-44 on page 221), move the mouse over the **Minerals** pie piece in the “Top 5 Products by Correlation” pie chart to view the frequency of 39 and correlation of 1.9.

### Expanding and minimizing a chart

To see only one chart at a time in a larger view, click the **expand** icon in the upper-right corner of the chart or table (Figure 6-44 on page 221).

After the chart is expanded, click the **minimize** icon in the upper right corner of the chart or table to go back to the main layout view that shows all of the charts or tables.

### Changing the chart or table size

To change the size of the charts, you move the mouse pointer over the border of the chart, and drag the frame of the chart to your wanted location. With this action, you keep all the charts in the layout viewable while making a chart larger. Figure 6-44 on page 221 shows an example of changing the frame of a chart. Notice that the “Top 5 Frequent Categories” and “Top 5 Products by Correlation” charts are wider than the other charts and tables.

### Dashboard preference settings

To change the default layout that is shown in the Dashboard, select **Preferences** → **Dashboard**. For the sample collection, select **Problem Report** to be the default layout. Now the Problem Report layout is displayed every time that you open the Dashboard view for the sample collection.

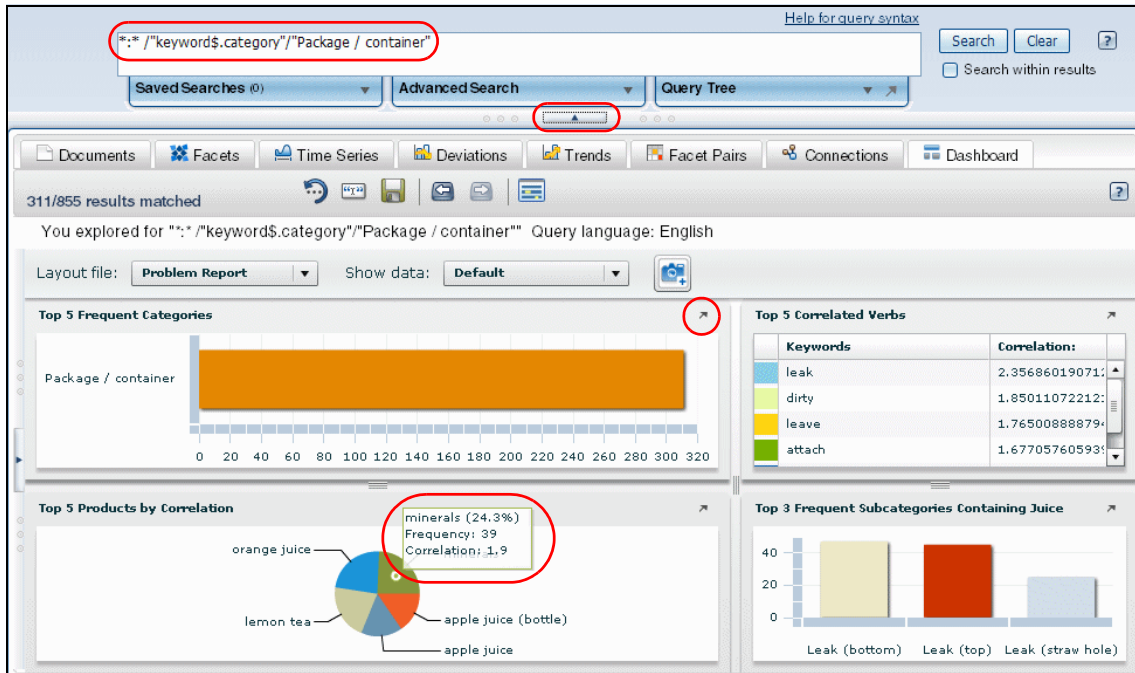


Figure 6-44 Problem Report layout with results for the Package/container category

## 6.9.4 Saving Dashboard charts as images

With the Dashboard, you can save all of the charts and tables in the selected layout as an image, or you can save an individual expanded chart or table. You can use the bitmap, PNG, and JPEG formats to save the images. You save an image by clicking the **image** icon, as shown in Figure 6-45, and selecting your wanted image format. After the image is saved, you can share it with coworkers for further collaboration.

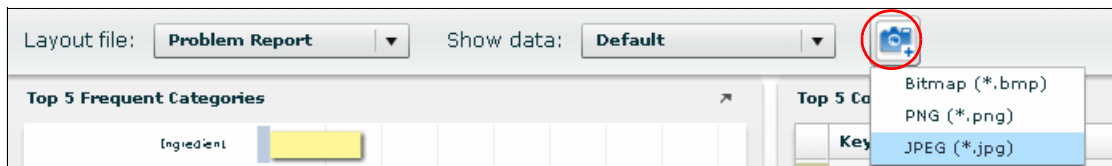


Figure 6-45 Saving the Dashboard charts by clicking the image icon

## 6.10 Sentiment view

The Sentiment view provides a dashboard to reveal the positive and negative language in your collection documents. If you are looking to mine “voice of the customer” (VOC) information and other similar types of data, the Sentiment view can help you understand user attitude towards your products and services for key quality control and marketing insights.

The Sentiment view is activated when you enable “Sentiment Analysis” for your content analytics collection. Enabling Sentiment Analysis also enhances the existing Document view as well. This section describes the sentiment enhancements in the Document view, the features available in the Sentiment view, and how to customize sentiment analysis to flag particular words and phrases as being positive or negative. For the steps to configure Sentiment Analysis for your content analytics collection, refer to the Content Analytics Administration Guide.

### 6.10.1 Document view with Sentiment Analysis enabled

When you enable Sentiment Analysis for a content analytics collection, the Document view now clearly identifies phrases and sentences that contain positive, negative, or ambivalent (both positive and negative) sentiment. See Figure 6-46 on page 223.

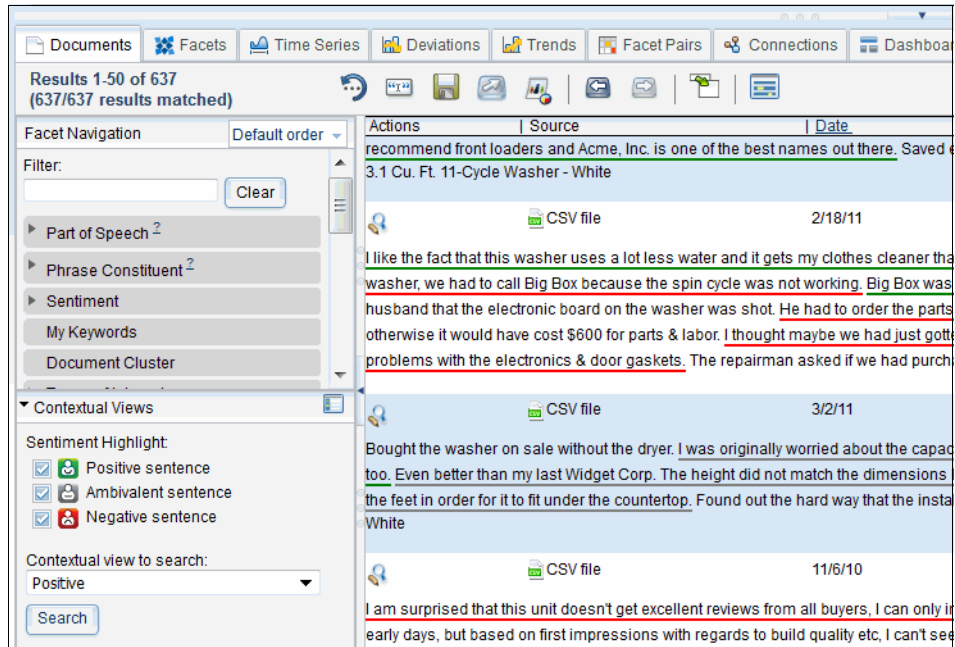


Figure 6-46 Document view with Sentiment Analysis enabled

For example, in Figure 6-46, a customer comment starting with “I like the fact that...” is underlined in green, reflecting a phrase with positive sentiment. The following sentence is underlined in red because it contains “...is not working”, a negative sentiment. Another product review in the collection contains the phrase “Dries well but is much noisier than my old machine”, an ambivalent statement about the product that Content Analytics underlines in gray.

You can control which types of sentiment are highlighted. For example, if you are performing quality control analysis to understand what products and features users are complaining about, it may be useful to only highlight and focus on negative comments. On the leftmost column of the Document view, the “Contextual Views” pane lets you select or clear the positive, negative, or ambivalent sentence types. See Figure 6-47 on page 224.

The screenshot displays the IBM Watson Content Analytics interface. At the top, there are navigation tabs: Documents, Facets, Time Series, Deviations, Trends, Facet Pairs, Connections, and Dashboard. Below these, it shows 'Results 1-50 of 629 (629/637 results matched)'. The main area is divided into a left sidebar and a right main pane.

**Facet Navigation:** Includes a 'Filter:' input field with a 'Clear' button. Below are expandable categories: Part of Speech<sup>2</sup>, Phrase Constituent<sup>2</sup>, Sentiment, My Keywords, Document Cluster, and Terms of Interest. The 'product' term is selected under Terms of Interest.

**Contextual Views:** A section for filtering by sentiment. It includes 'Sentiment Highlight:' with three options:
 

- Positive sentence
- Ambivalent sentence
- Negative sentence

 Below this is a 'Contextual view to search:' dropdown menu set to 'Positive' and a 'Search' button.

**Main Results Table:** A table with columns for 'Actions', 'Source', 'Date', and 'Title'. It lists several document entries, each with a CSV file icon, a date, and a snippet of text. Some text in the snippets is underlined in red, indicating sentiment highlights.
 

Actions	Source	Date	Title
[Icon]	CSV file	8/7/10	it
[Icon]	CSV file	8/18/10	ho
[Icon]	CSV file	7/11/10	He
[Icon]	CSV file	8/21/10	4
[Icon]	CSV file	7/6/10	ve

Figure 6-47 Document view where you can select positive or negative sentiment

If you have a current search expression in place, you can target the search for only those entries identified as positive or negative. For example, the word “fast” might be positive or negative depending on the context. In our appliance review collection, it typically is part of a positive comment (“my clothes dry fast!”). If you want to instead look for the term “fast” only within negative comments, you would enter that search term, and then use the “Contextual view to search” option to search the “negative” context. See Figure 6-48 on page 225.



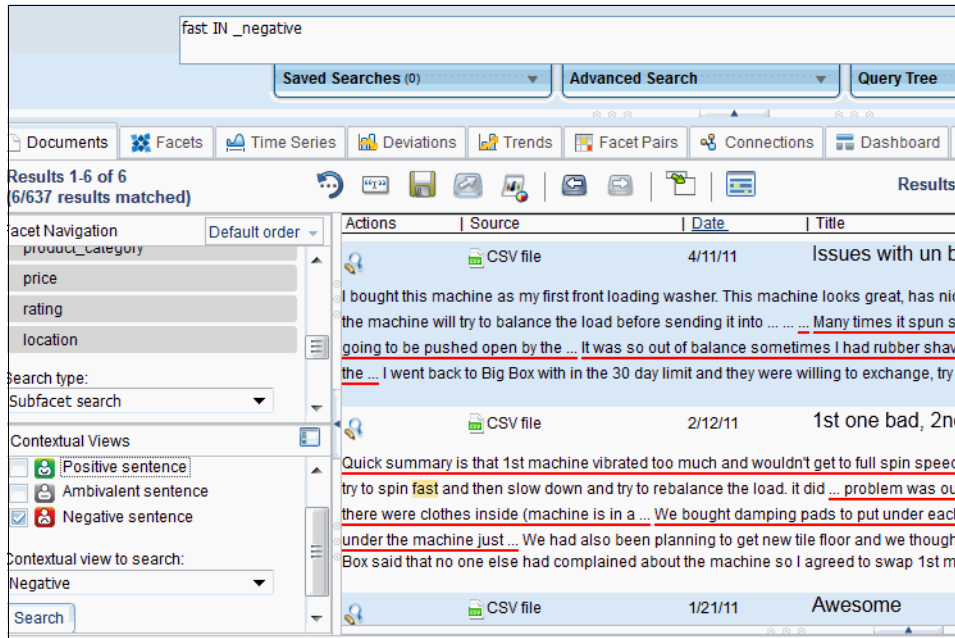


Figure 6-48 Showing a search term “fast” with negative sentiment

To launch the preview dialog pop-up window, click the preview icon under the Source column. For example, in Figure 6-48, you click the **CSV file** label under the Source column.

With Sentiment Analysis enabled, the “Document Analysis” dialog window within the Document view adds sentiment-related words and phrases, beyond the default list of words, phrases, and metadata. See Figure 6-49 on page 226.

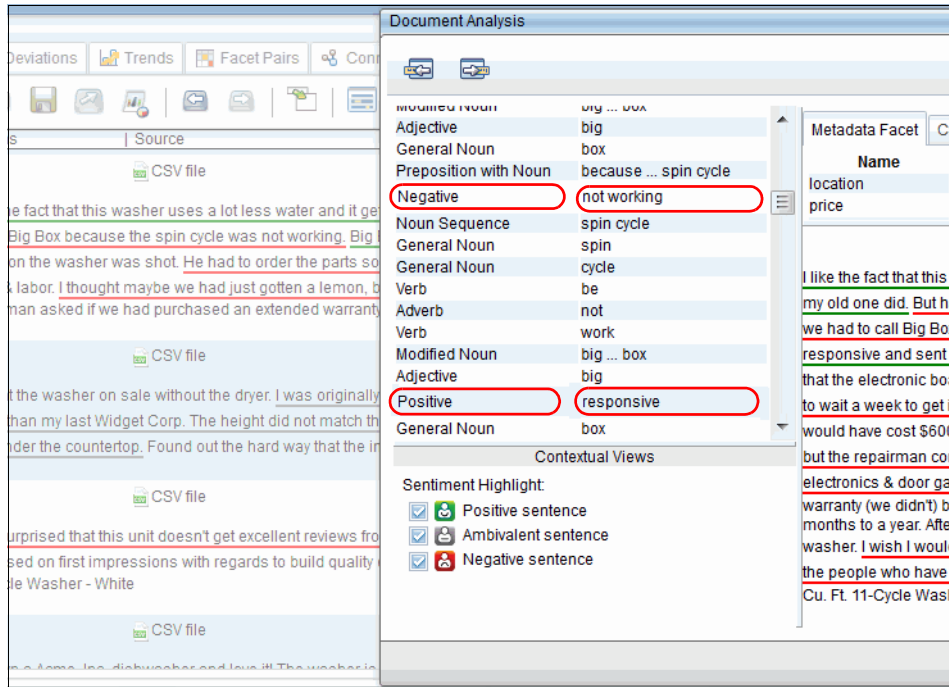


Figure 6-49 Adding sentiment-related words and phrases in Document view

## 6.10.2 Understanding the Sentiment view

While the Document view highlights positive and negative phrases when Sentiment Analysis is enabled, the Sentiment view provides a targeted dashboard with additional tools for exploring sentiment-related content. See Figure 6-50 on page 227.

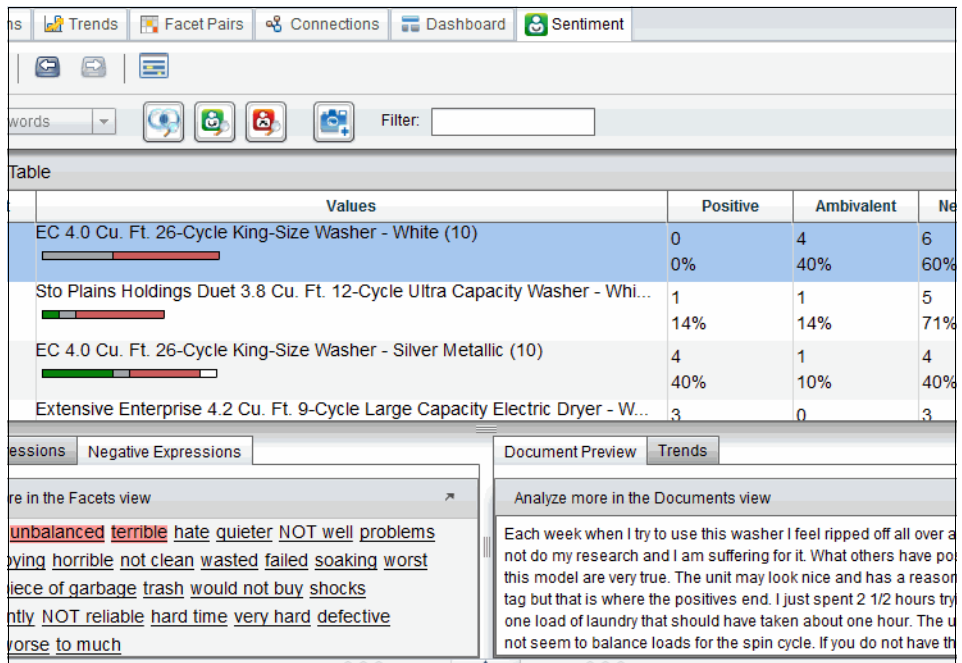


Figure 6-50 Sentiment view with additional tools for exploring sentiment content

The main window of the Sentiment view contains the following panes:

- ▶ An enhanced filter bar with two new icons to allow quick filtering of positive or negative documents.
- ▶ A main window showing values for the currently selected facet, along with information to see at a glance the relative frequency of positive and negative documents that are associated with that value.
- ▶ An expressions pane that shows either the specific positive or negative words and phrases found in documents for the currently selected facet value.
- ▶ A pane to quickly show either a detailed document view or trend graph for the current selection.

Selecting a value listed in the main sentiment window updates the expressions pane to display keywords that are found in those documents. See Figure 6-50.

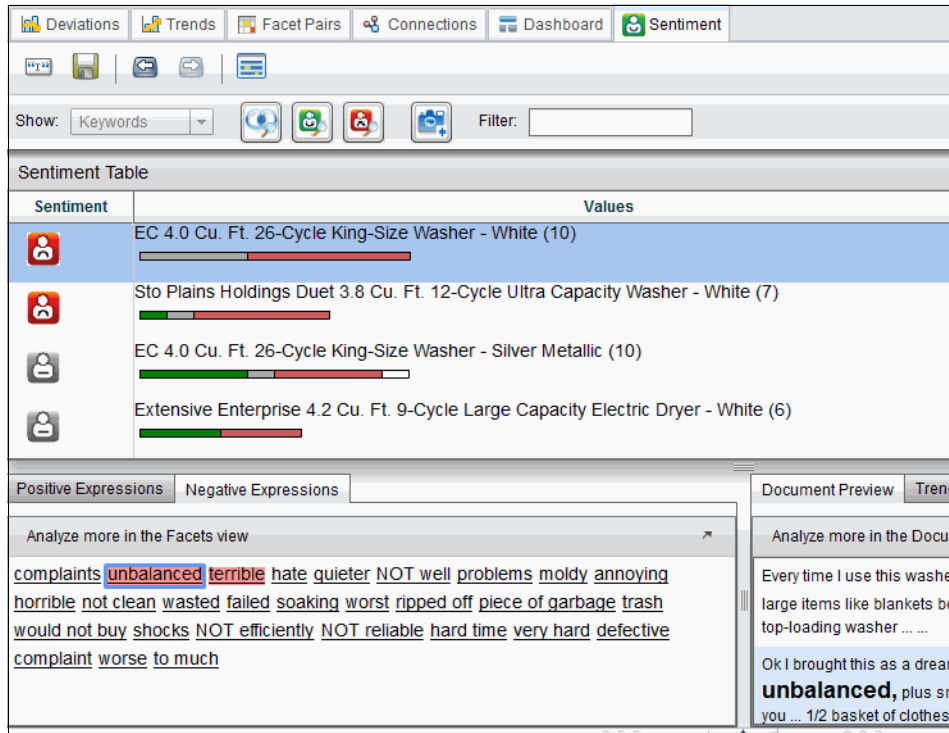


Figure 6-51 Updated expression pane when selecting a value in the sentiment window

You can review the keywords in the Expressions pane and further target particular terms for investigation. For example, when the word “unbalanced” is selected in Figure 6-51, the Document Preview pane to the right shows phrases including “unbalanced” in enlarged bold font. You can scroll down the “Document Preview” list to see these comments, and select the arrow at the upper right of that pane to maximize that view. See Figure 6-52.

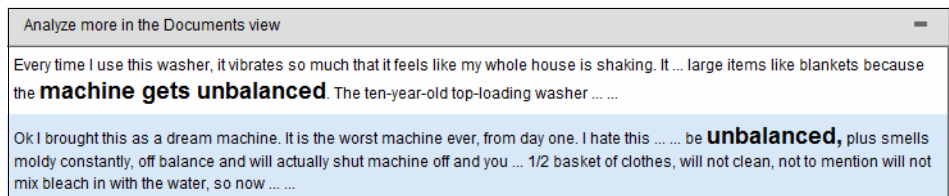


Figure 6-52 Document Preview pane showing “unbalanced” word found documents

If you click the **Analyze more in the Documents view** link, you will be directed outside of the Sentiment view to the standard Document view, with a filter applied for your selected search term. See Figure 6-53 on page 229.

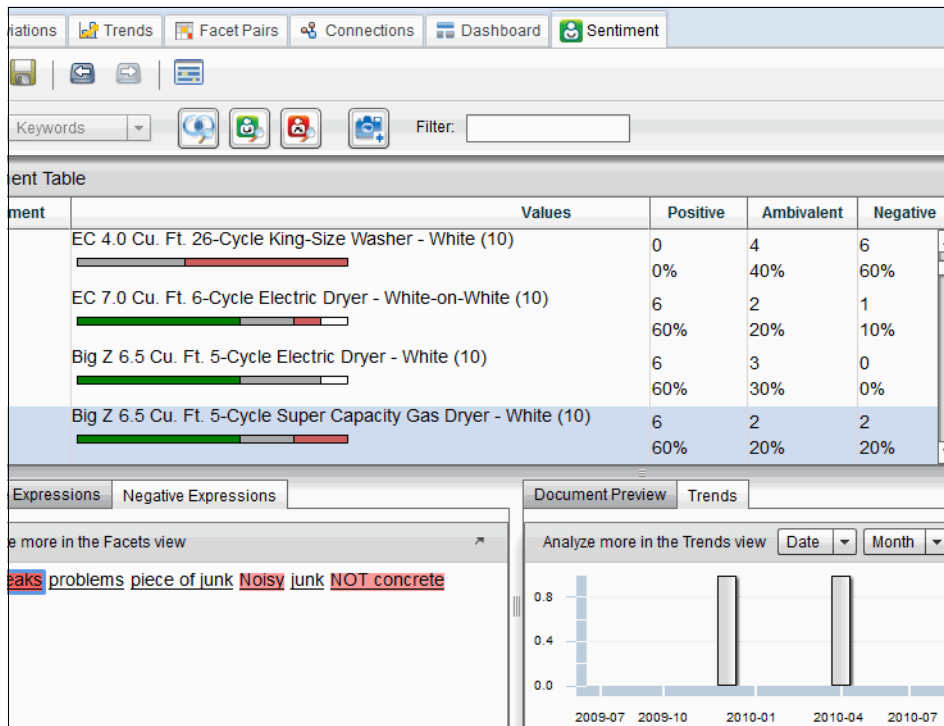


Figure 6-53 Display when clicking the “Analyze more in the Documents view” link

From Figure 6-53, you can also see how your selected keyword has been trending over time, without leaving the Sentiment view. Next to the Document Preview tab in the lower-right portion of the window, there is a Trends tab available for selection. This displays the frequency of documents containing the target phrase for particular time periods. Just as you can with the Document Preview pane, you can select the arrow in the upper right of that pane to maximize this trend graph. You can also select **Analyze more in the Trends view** to be directed outside of the Sentiment view to the main Content Analytics Trends tab, described earlier in this chapter.

If you want to see various facet values for the documents that you have been investigating in Sentiment view, select “Analyze more in the Facets view”, which directs you to the Facets view with a filter applied for the currently selected documents. See Figure 6-54 on page 230.

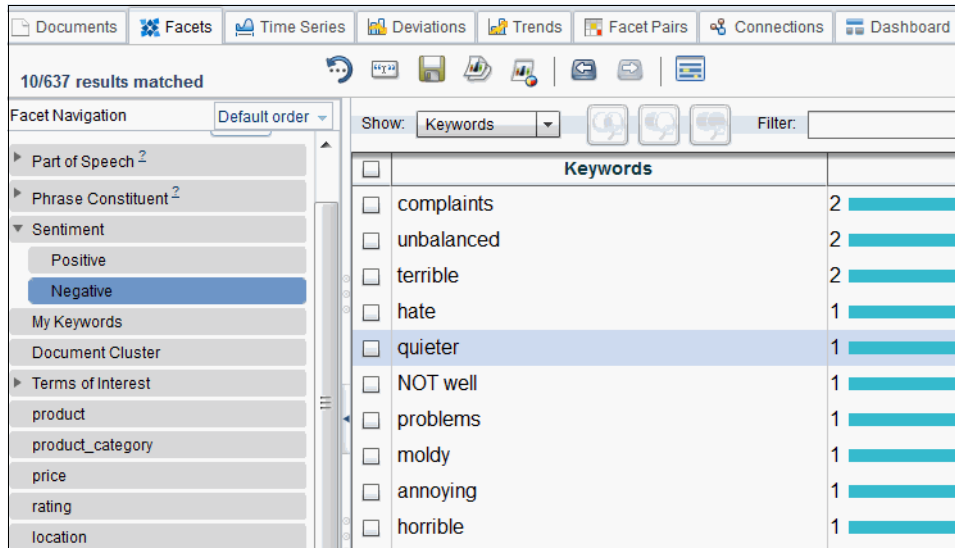


Figure 6-54 Facets view with sentiment information

In conclusion, the Sentiment view can help highlight whether particular items (products, stores, service reps, and so on) generate significant positive or negative feedback, help you drill down to particular aspects of the complaint (or compliment), and then help you move seamlessly to other Content Analytics views for further investigation.



## Performing content analysis

Chapters 5 and 6 provide information about the basic features and operations of the content analytics miner. Chapter 5 covers the search and discovery features, while chapter 6 covers the views that are available for analysis. This chapter provides additional guidance on performing content analysis with IBM Watson Content Analytics (Content Analytics) by discussing a range of techniques to generate useful facets for analysis, and ways to effectively use the out-of-box views. We show various techniques that you can use with the sample analytics collection. This collection is easily installable through the “First Steps” application, and a useful sandbox for initially exploring product capabilities. Then, we cover various techniques to generate useful facets.

This chapter includes the following sections:

- ▶ Working with the sample collection
- ▶ Content analysis scenarios for the sample collection
- ▶ Overview of techniques to create facets for analysis
- ▶ Preferred practices

## 7.1 Working with the sample collection

The content analytics miner packaged with Content Analytics is a powerful tool for content analysis, which is ready for immediate use. This section explains how to use the content analytics miner to discover actionable insight from your textual data.

Throughout this section, the sample collection, Sample Text Analytics Collection, is used. This collection is created when you select **Text Analytics Tutorial** in the First Steps program. Create this collection before you proceed with the rest of the section. You can manually build the content analytics collection with the same data and configuration as explained in Chapter 4, “Installing and configuring IBM Content Analytics” on page 85 from the previous version of this book (which you can download as part of the additional material for this book).

This section also shows techniques that can be used with this and other collections, including use of a custom dictionary and custom text analysis rules.

### 7.1.1 The sample data

The data used in this example is in the `ES_INSTALL_ROOT/samples/firststep/data/xml/xmls.tar.gz` file. When you extract each XML data file from the `.tar` file, you see the content similar to what is shown in Example 7-1.

*Example 7-1 XML data used in the sample collection*

---

```
<?xml version="1.0" encoding="UTF-8"?>
<doc>
  <id>00000000</id>
  <title>lemon tea - Package / container</title>
  <date>2008-01-01</date>
  <timestamp>1199186392296</timestamp>
  <category>Package / container</category>
  <subcategory>Straw</subcategory>
  <product>lemon tea</product>
  <text>[Pack] The straw was peeled off from the juice pack.</text>
</doc>
```

---

This data simulates the transcript of customer feedback regarding issues with food products purchased in grocery stores. Note the fields between the `<doc>` tags to understand the starting point of data we are working with. In this case, the data includes product, date, problem category, and subcategory, as well as the free-form comment text.



In this case, the source data happens to be XML files. However, the techniques we show would apply regardless of the source, whether our data is coming from a database, a delimited flat-file, a content management system, or one of the many other source types that can be imported.

**Sample data source:** The sample data is bundled with Content Analytics.

When you run the Text Analytics Tutorial from the First Steps program, the sample collection is created for you with the sample data. The following steps are automatically executed:

1. Collection Index fields are created.
2. A crawler is created, with mappings defined between source fields and collection index fields.
3. Facets are created, with mappings defined between certain index fields and facets.
4. The crawler and indexing-related tasks are executed.

At this point, no custom dictionaries or text analysis rules have been created on the sample collection. We show some analysis steps that you can perform with the initial collection configuration, as well as how to create dictionaries and rules to improve the set of facets that are available for analysis.

Figure 7-1 on page 234 shows the mapping named *sample\_mapping* that defines the mapping between the XML element name and Field name. This window is specific to crawling XML files from a file system. Other source types will have crawler configuration windows that differ depending on the nature of the source, though they all allow the ability to control how source fields are mapped to index fields.

[Collections](#) > [Sample Text Analytics Collection : Parse](#) > [Map XML elements to search fields](#) > [sample](#)

### Edit an XML Field Mapping

[Help for this page](#)

You can change the name of this set of mapping rules, map additional XML elements to fields, and change on

\* XML root element name:

\* XML mapping name (Valid characters are: a-z, A-Z, 0-9, underscore(\_), hyphen(-), and spaces.):

[+ Add Field](#)

* XML element name	XML attribute name	XML attribute value	Field name	Returnable	Facet search
<input type="text" value="id"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="doc_id"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="title"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="title"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="timestamp"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="date"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="category"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="doc_category"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="text" value="subcategory"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="doc_subcategory"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="text" value="product"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="doc_product"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[OK](#) [Cancel](#)

Figure 7-1 Mapping the XML element to the search field for sample data

Figure 7-1 also shows that the sample XML document has the element and field associations that are shown in Table 7-1.

Table 7-1 Element and field associations for the sample XML document

Element	Field
id	doc_id
title	title
timestamp	date
category	doc_category
subcategory	doc_subcategory
product	doc_product

Figure 7-2 shows the facets that are created for you. Three default facets are created by the tutorial: the Category facet, the Subcategory facet, and the Product facet. These facets are mapped with the following fields:

- ▶ The Category facet is mapped with the doc\_category field.
- ▶ The Subcategory facet is mapped with the doc\_subcategory field.
- ▶ The Product facet is mapped with the doc\_product field.

The screenshot shows a web interface titled "Configure the facet tree" for a "Sample Text Analytics Collection". The main area displays a "Facet Tree" with a hierarchical structure: Root, My Keywords, Category (selected), Subcategory, and Product. To the right, there are two panels: "Add a facet" and "Edit a facet".

**Add a facet:**

- \*Facet path: [text input]
- Facet name: [text input]
- Facet type: [Standard facet]
- Visible in the text miner:
- Add counts to parent facet:
- [Add button]

**Edit a facet:**

- \*Facet path: [category]
- Facet name: [Category]
- Facet type: [Standard facet]
- Visible in the text miner:
- Add counts to parent facet:
- Facet order: [text input]
- Field mappings: [doc\_category]

Figure 7-2 Mapping between the facets and search fields

Notice that the XML element “text” in Example 7-1 on page 232 is intentionally *not* associated with any facet. Useful facets contain a useful categorization of your data. Values such as product type, problem type, location (and many others) are useful categories. Free-form sentences and paragraphs of content are not typically useful categories. Other data elements that might not be useful as facets are fields where the values are unique for every document item. If each comment in our collection had a unique serial number, that field would not be useful as a facet since it is not helpful for grouping documents into categories for analysis.

**Best practice:** Do not use fields as facets if the fields are unique for every document item. This is because these type of fields will not be helpful to group documents into categories for analysis.

## 7.1.2 Getting insights from the sample collection

The sample collection includes key metadata to make it ready for analysis, including product, problem category and subcategory, and date. You can derive many insights from the sample collection in its initial state.

When you explore the sample collection, or your own data collections, you can start with a specific question in mind, such as: “Why did support calls increase last month?” or “What problems are we seeing with product X?”. Alternatively, you might not have a specific starting point for your investigation, and want to mine for unexpected patterns and trends that provide actionable insights. Let us first look at the latter case, where you do not have an initial focus, and want to use content analytics miner to highlight areas of potential interest in your data.

There are several views that are particularly useful to quickly spot anomalies in the data. The Trends view, the Deviations view, the Facet Pairs view, and the Connections view all highlight statistical anomalies or patterns in the data, and can be useful even before an initial filter is applied.

For example, in the Trends view, we can see in one screen the charts for a number of products. Figure 7-3 on page 237 shows the result when you select the Product facet and sort it by the latest index. Notice that the number of calls (as indicated by the number of cases in the sample data) that are related to pine juice increases in December 2008 (highlighted in Figure 7-3 on page 237). This and other products are highlighted at certain time periods as data points of interest. This tells us that the frequencies of items related to these items have notably increased over a period.

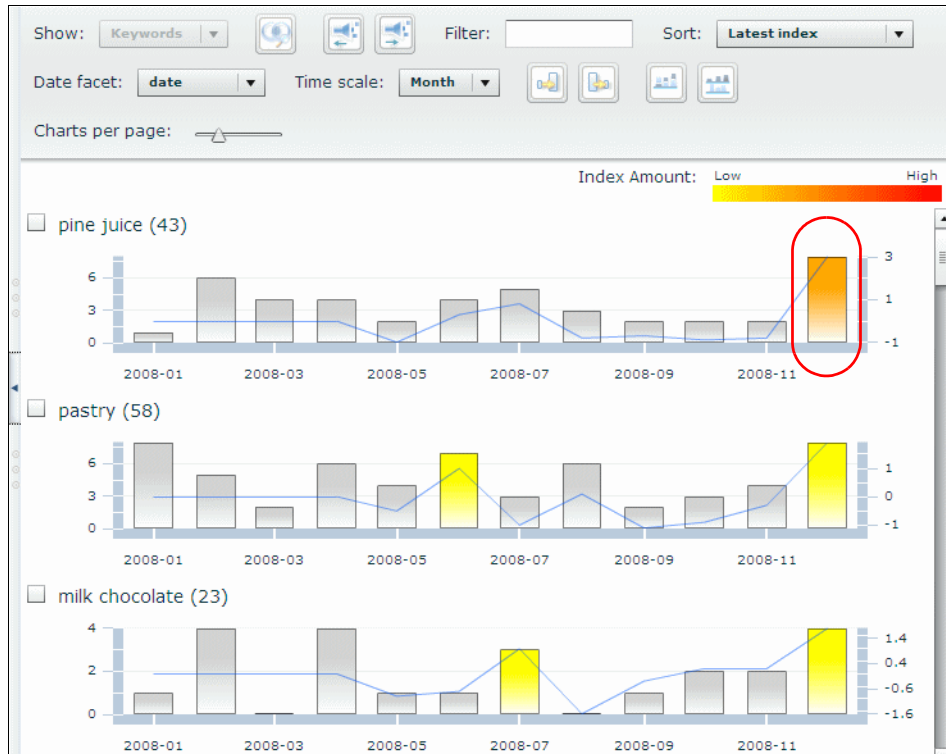


Figure 7-3 Trends view showing the Product facet sorted by the latest index

In the Facet Pairs view, you can also see which keywords are highly correlated. Figure 7-4 shows the result when you select the Product facet and the Noun facet and sort them by correlation.

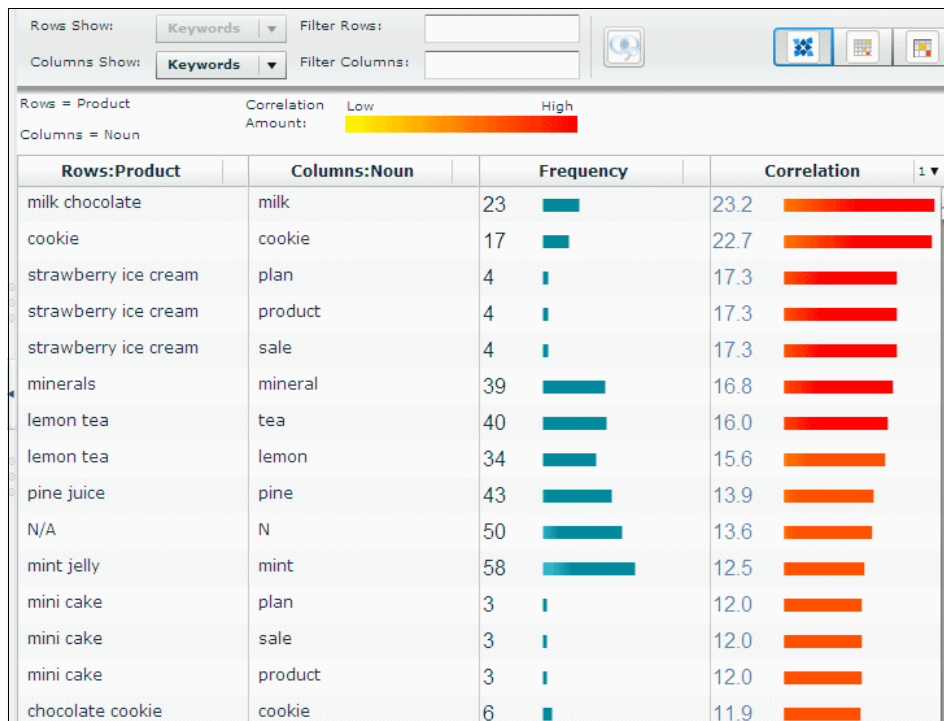


Figure 7-4 Facet Pairs view showing the Product facet and Noun facet selected

In addition to keywords, you can see the verbs that are highly correlated with a facet. Figure 7-5 shows the result when you select the Product facet and the Verb facet and sort them by correlation.

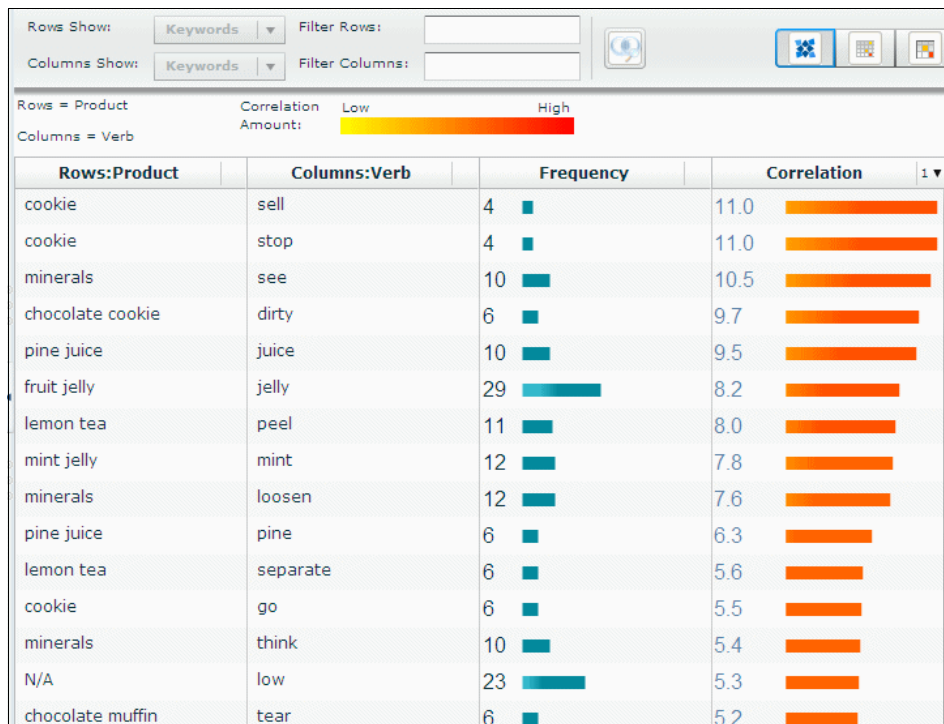


Figure 7-5 Facet Pairs view showing the Product facet and Verb facet selected

Depending on the facets you select and the data you have, you might discover interesting insights without any customization. Alternatively, you might not discover anything that requires special attention immediately.

### 7.1.3 Considerations about what you want to discover from the data

Even though you can find useful insight from the data with various views, you might want additional facets to provide new and valuable insights for your data set. For the sample collection, we have certain facets such as Category, Subcategory and Product, defined. What if we wanted to look at the sample data in the context of which stores saw the most problem reports, or if we wanted to refine the categories to make them more meaningful?

Unless the data is already available in structured form to map to a field and facet, we need to find that data elsewhere. In many cases it is embedded in free-form

text. Techniques such as custom dictionaries and syntax rules can help extract this valuable data from free-form text, and make it available in a facet for new ways to explore your data set.

The next section describes several scenarios, explaining how you can define a dictionary or custom text analysis rules with the sample data to potentially discover additional insights from your data.

## 7.2 Content analysis scenarios for the sample collection

To better help you understand how to perform content analysis, this section includes the following scenarios, which include use of the content analytics miner views, and iterative improvement of facets:

- ▶ Scenario 1: Using a custom dictionary to discover package- related calls
- ▶ Scenario 2: Using custom text analysis rules to discover trouble-related calls
- ▶ Scenario 3: Discovering the cause of increasing calls

For the procedures to create the associated custom dictionary and custom text analysis rules, see 8.2.2, “Configuring the Dictionary Lookup annotator” on page 282, and 8.3, “Configuring the Pattern Matcher annotator” on page 291.

Before you proceed with the scenarios, see Chapter 3, “Designing content analytics solutions” on page 41 and Chapter 4, “Understanding content analysis” on page 59, if you have not already done so.

### 7.2.1 Scenario 1: Using a custom dictionary to discover package-related calls

This scenario explains how to use a custom dictionary to discover package-related calls to the call center. The Facet Pairs view is used for this scenario.

Though the sample collection does include problem category and subcategory facets, we will imagine (for the sake of this tutorial) that this data does not initially exist, and that we need to somehow create it by looking for package-related words in our collection.

**Tip for finding candidates for a dictionary entry or rule:** The best resource for finding words and patterns is your collection data. The predefined Part of Speech and Phrase Constituent facets provide good candidates. The Terms of Interest facet also shows a candidate list if it is enabled.



## Considering the words to register with the dictionary

First, you want to determine the words that might indicate a package-related call. To determine these words, from the Facet Navigation pane, follow these steps:

1. Expand the **Part of Speech** facet, and select the **Noun** facet.
2. Go to the Facets view. Notice that many words are related to packages. These words include package, container, pack, bottle, and cup, as shown in Figure 7-6.

Track all the words that you identify that are related to the package calls, and add them to your custom dictionary. We select to track the words **bag**, **bottle**, **cap**, **container**, **cup**, **material**, **pack**, **package**, **shape**, **spoon**, **straw**, and **top**.

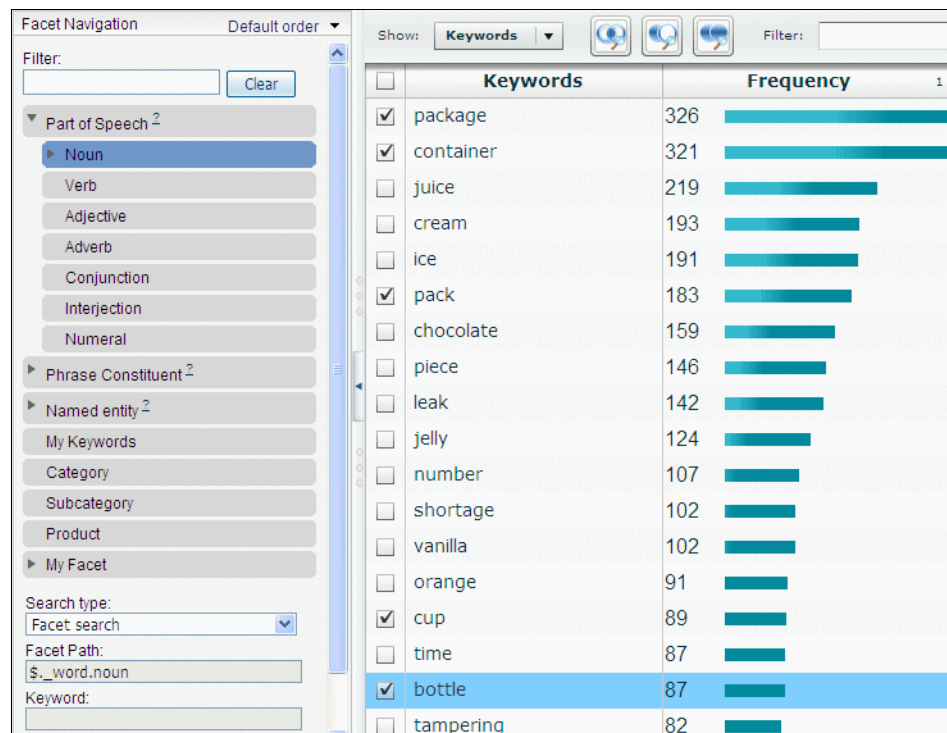


Figure 7-6 Facets view showing the Noun facet

## Updating the facet tree

After you decide the words that you want to add to the custom dictionary, create a facet to associate these words. In this example, we create the My Facet facet to distinguish it from the predefined facets. We create two additional facets under My Facet called “Package” and “Troubles”, as shown in Figure 7-7.

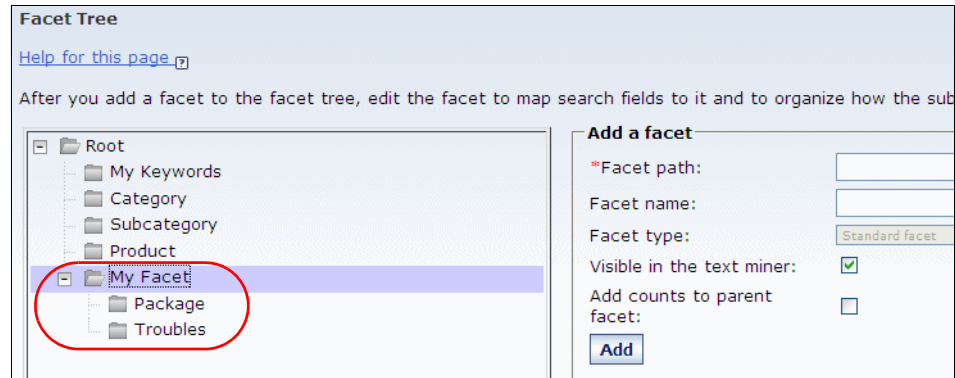


Figure 7-7 Facet tree after defining the Package facet and Troubles facet

## Creating a custom dictionary and associating the words with a facet

Now, you are ready to create a custom dictionary and associate the words that you identified earlier with the new facet. This can be done either by using ICA Studio, or more simply by creating a user dictionary within Content Analytics administration interface. ICA Studio provides the ability to mix dictionaries and syntax rules for sophisticated analysis, and will often be the tool of choice for customized text analytics. You can learn more about ICA Studio in Chapter 11, “Customizing content analytics with IBM Content Analytics Studio” on page 405.

In this scenario, we use the approach of creating a custom dictionary from within the administrative interface. We associate the words that we selected earlier with the Package facet, as shown in Figure 7-8 on page 243. Now, whenever any of the words are displayed in a call, the call is logged with the Package facet.

























Applied Words (12) <<< 1 / 1 >>>				
Select	Parts of speech	Word	Synonym	Facet
<input type="checkbox"/>	Noun	bag		Package 
<input type="checkbox"/>	Noun	bottle		Package 
<input type="checkbox"/>	Noun	cap		Package 
<input type="checkbox"/>	Noun	container		Package 
<input type="checkbox"/>	Noun	cup		Package 
<input type="checkbox"/>	Noun	material		Package 
<input type="checkbox"/>	Noun	pack		Package 
<input type="checkbox"/>	Noun	package		Package 
<input type="checkbox"/>	Noun	shape		Package 
<input type="checkbox"/>	Noun	spoon		Package 
<input type="checkbox"/>	Noun	straw		Package 
<input type="checkbox"/>	Noun	top		Package 

Figure 7-8 Custom dictionary showing associating keywords with the Package facet

To associate special nouns with a particular facet, you define them in the custom dictionary. If you want to associate verbs or phrases with a facet, you use custom text analysis rules as explained in 7.2.2, “Scenario 2: Using custom text analysis rules to discover trouble-related calls” on page 244.

For details about how to create a custom dictionary, see 8.2.2, “Configuring the Dictionary Lookup annotator” on page 282.

## Deploying resources and rebuilding the index

After the dictionary is ready, you must deploy the resources from the administration console. After the resource is deployed successfully, you can see the updated facet tree in the Facet Navigation pane in the content analytics miner.

The resource deployment does not update the documents that are already indexed. To view the changes in the dictionary, rebuild the index to apply the changes to the documents already in the index.

## Confirming the result

After you rebuild the index, go back to the content analytics miner, and confirm the results. If necessary, perform further analysis.

Go to the Facet Pairs view, select the **Product** facet for the row and the **Package** facet for the column. Sort the result by correlation, as shown in Figure 7-9 on page 244.

As shown in Figure 7-9, a product, such as lemon tea, has a high correlation with the word “straw.” You can look through the documents to see if you can discover any actionable insights. For this scenario, we add the Boolean search condition AND and see the result documents in the Documents view for analysis.

After we review each result document, we notice that several reports indicate that the package of lemon tea did not include a straw with its package or that a straw was separated from the package itself. These reports indicate that the lemon tea product might have a problem with its packaging. This finding is the result from the dictionary definition for this scenario. With a custom dictionary, you can discover the potential defect.

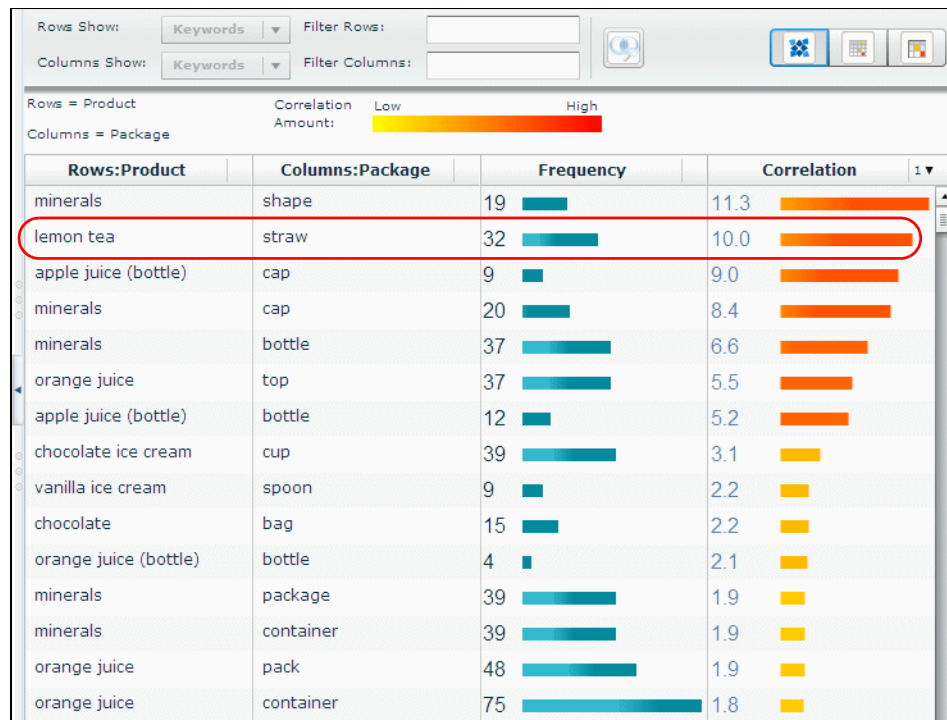


Figure 7-9 Facets Pair view showing the Products facet and the Package facet

## 7.2.2 Scenario 2: Using custom text analysis rules to discover trouble-related calls

This scenario explains how to use custom text analysis rules to discover trouble-related calls for the call center. We will use the Facet Pairs and Connections views for this scenario.

## Considering the patterns to register as rules

User dictionaries only work with word sequences. However, a word sequence is not always sufficient to identify a condition. The comment “Jane did not provide very good service” might be counted improperly as a positive event if you are simply looking for the phrase “good service”. In this section, we focus on using custom Text Analysis rules deployed via the administration console to help us identify critical product problems in our sample analytics collection.

To determine the verbs or phrases that you want to consider for pattern matching, expand the **Part Of Speech** (POS) facet, and in the Facet Navigation pane, select the **Verb** facet. Then go to the Facets view.

We notice verbs, such as leak, smell, dirty, loosen, and lower, that indicate some kind of trouble. Therefore, you want to identify the verbs that indicate trouble. For our scenario, we select the verbs leak, smell, dirty, loosen, lower, peel, expire, clump, and detach as trouble words.

In the custom text analysis rule, you can also add nouns as patterns. For our scenario, we select the nouns leak, shortage, contamination, tampering, hole, dirt, prank, clump, hair, thread, empty, bruise, nail, and chunk, which also indicate trouble.

## Updating the facet tree

After you decide the verbs and nouns for the custom text analysis rules, create a facet to associate with the results of those rules. In this scenario, we want to create a Troubles facet to associate pattern matching. Because we already created it earlier, we do not need to update the facet tree now. Figure 7-7 on page 242 shows the facet tree.

## Creating custom text analysis rules

From the administration console, you can add the custom text analysis rules. However, before adding the custom text analysis rules, you must create a rule file that is written in XML. You can create your own rule file that is based on keywords or phrases.

In this scenario, we create the `trouble.pat` rule file (Example 7-2).

*Example 7-2 The trouble.pat rule file*

---

```
<?xml version="1.0" encoding="UTF-8"?>
<pattern-list lang="en">
<mi category="$.myfacet.troubles" value="{1.lex}">
  <w id="1" lex="leak"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
```

```

    <w id="1" lex="smell"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="/^dirt"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="loosen"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="lower"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="peel"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="expire"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="clump"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="detach"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="shortage"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="contamination"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="tampering"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="hole"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="prank"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="clump"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
    <w id="1" lex="hair"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">

```

```
<w id="1" lex="thread"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
  <w id="1" lex="empty"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
  <w id="1" lex="bruise"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
  <w id="1" lex="nail"/>
</mi>
<mi category="$.myfacet.troubles" value="{1.lex}">
  <w id="1" lex="chunk"/>
</mi>
</pattern-list>
```

---

**Tip to create a rule pattern file:** To create a rule pattern file, start with an existing rule pattern file and make a copy of the file. Then, edit from the copy and use it in your system. You do not need to create the file from scratch.

For details about how to configure the custom text analysis rules using the user interface, see 8.3, “Configuring the Pattern Matcher annotator” on page 291.

## Deploying resources and rebuilding the index

After the dictionary and pattern rules are ready, deploy the resources from the administration console. When the resource is deployed successfully, you can see the updated facet tree in the Facet Navigation pane of the content analytics miner.

The resource deployment does not update the documents that are already indexed. To reflect the changes in the dictionary, you must rebuild the index to apply the changes to the data that is indexed already.

## Confirming the result

After the index is rebuilt, go back to the content analytics miner, and confirm the results. If necessary, perform further analysis.

Go to the Facet Pairs view again. Select the **Product** facet for row and the **Troubles** facet for column. Sort the result by correlation.

As shown in Figure 7-10, you can see that a product, such as chocolate cookie, has a high correlation with words such as “dirty” or “dirt.”

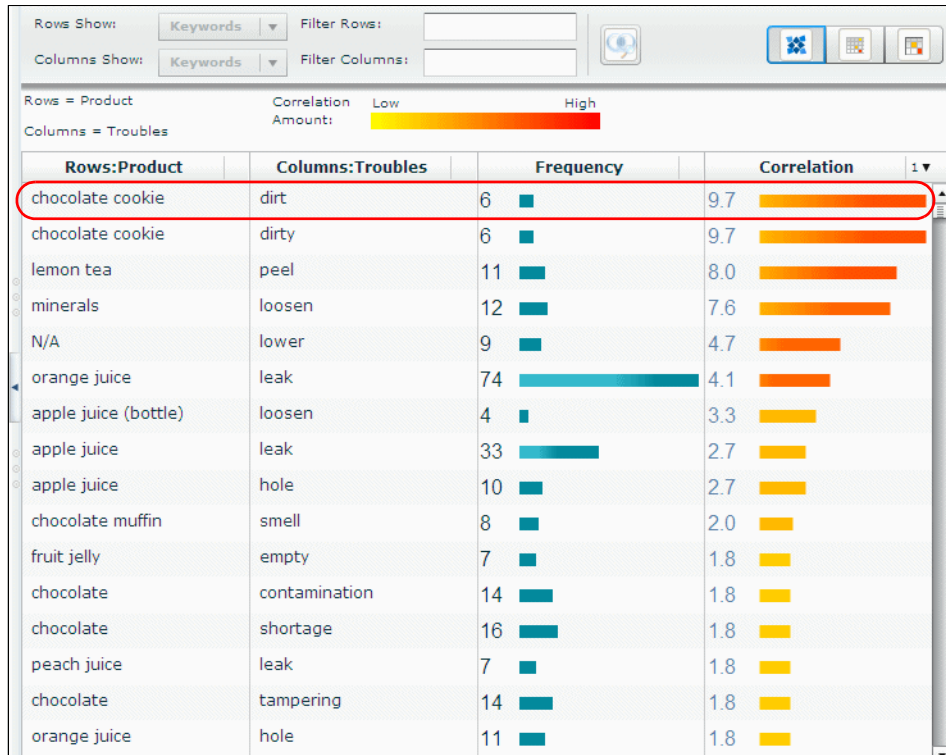


Figure 7-10 Facet Pairs view showing the Product facet and the Troubles facet



You can also see the relationships between a pair of facets by using the Connections view, as shown in Figure 7-11.

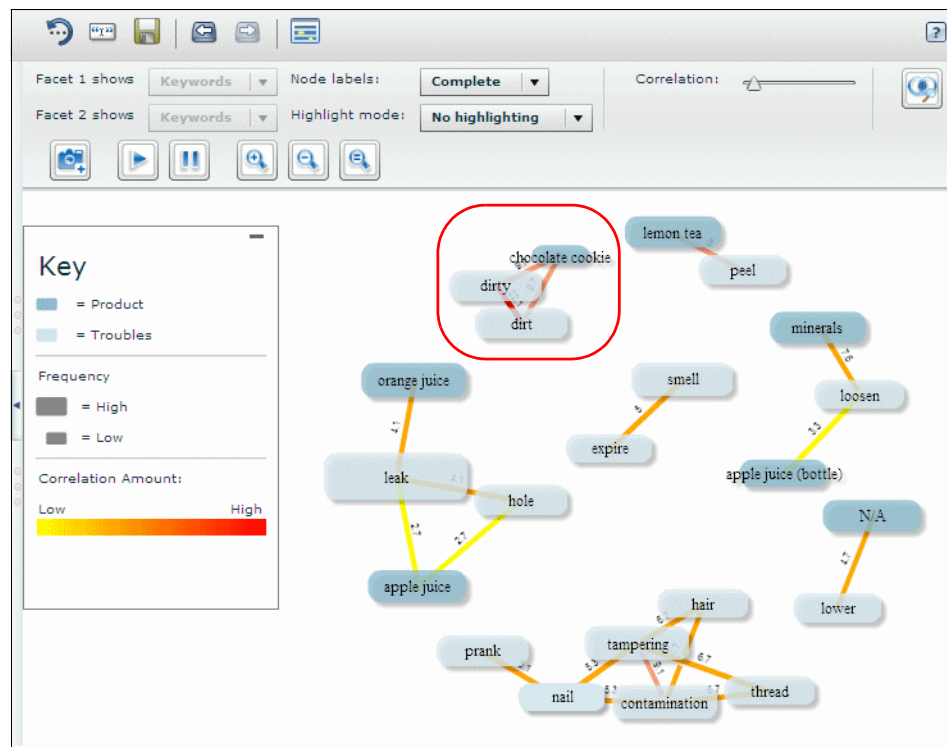


Figure 7-11 Connections view showing the relationships between the Product facet and the Troubles facet

We again add the condition as a Boolean search AND, and review the result documents in the Documents view for details. After we review each document, we notice that some reports indicate that the container inside was dirty for some reason. With pattern matching, you can discover potential problems with the container.

### 7.2.3 Scenario 3: Discovering the cause of increasing calls

This scenario is a bit different from the first two scenarios. As shown in Figure 7-3 on page 237, when you see the Trends view with the Product facet sorted by the latest index, you see that the calls related to pine juice increase in December 2008. A sharp increase in the Trends view usually indicates something that you must investigate.

In this scenario, we consider investigating the cause of the increasing calls by using the dictionary that we defined previously to see if we can find anything noticeable from the product package aspect.

**Custom dictionary:** We use the same custom dictionary as defined in 7.2.1, “Scenario 1: Using a custom dictionary to discover package- related calls” on page 240. If you want to analyze the data from a different aspect, you can update your custom dictionary.

In this scenario, we consider the possible cause of the increasing calls for pine juice. For this purpose, package-related calls might be helpful because “package” is commonly used with other products, and there might be a correlation here.

### Confirming the result of Scenario 3

In the Trends view with the Package facet, we noticed that the calls related to “straw” and “bag” increased in December of 2008, as shown in Figure 7-12.

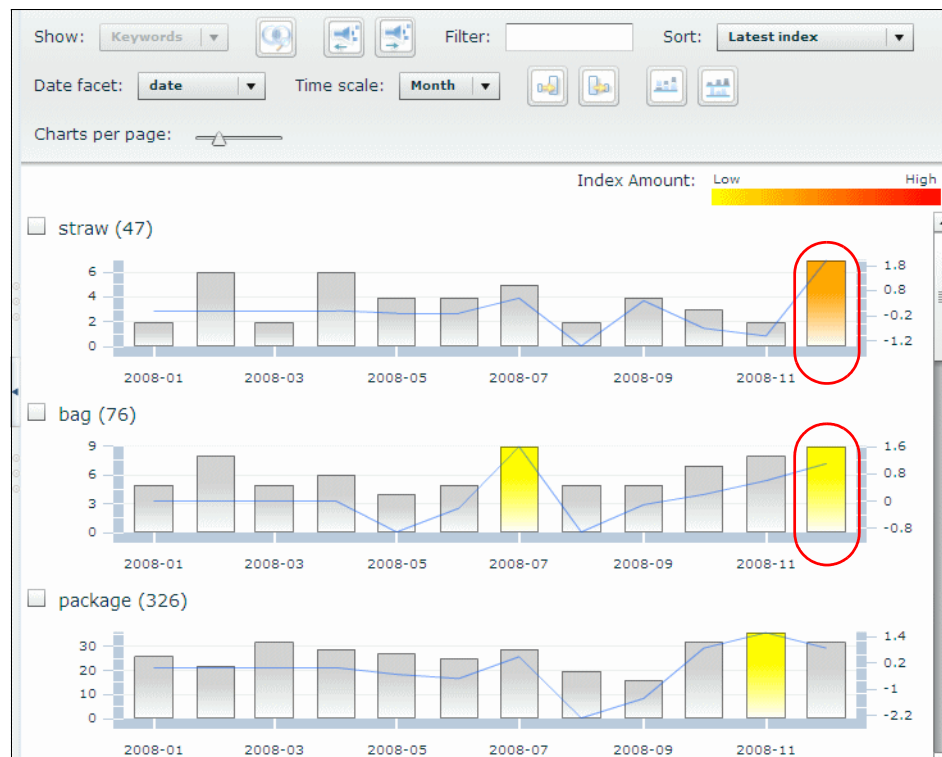


Figure 7-12 Trends view showing the Package facet sorted by the latest index

With the custom dictionary defined earlier, we have another analytical aspect of the package that can be related to pine juice and other products if those items are commonly used as pine juice. We discover the potential package-related problem with the Trends view, with the help of pattern matching rules and a custom dictionary.

## 7.2.4 Conclusion

As you can see by the scenarios in this section, you can use the content analytics miner to determine the symptom or the trend. You can also use the content analytics miner to determine what to analyze, select the candidates for the custom dictionary, or define the custom text analysis rules to discover the insight from various aspects. You can also use the content analytics miner to discover noticeable events, detect anomalies at the beginning when the problem occurs, or predict future trends.

As shown in this section, when you perform analysis with the content analytics miner, you typically perform the following steps:

1. Analyze the data, and review the results with the content analytics miner.
2. Consider which words you want to analyze. That is, select the candidate words for the custom dictionary or the custom text analysis rules.
3. Define a facet tree so that you can see the data from the specific aspect.
4. Define a custom dictionary or the custom text analysis rules, and associate them with the facet.
5. Deploy the resources, and rebuild the index.
6. Confirm the result with the content analytics miner. See if you can discover insights from a defined custom dictionary or custom text analysis rules.

If not, consider iterating the process until you can discover what you want with different dictionary keywords or different custom text analysis rules.

The analysis is an iterative process. You need to employ a trial-and-error approach until you gain interesting insights that help your business.

Remember that the content analytics miner does not provide what you need to focus on automatically. You must be conscientious of how you create facets via the various techniques that are available, including dictionaries, pattern rules, and other approaches. See Chapter 4, “Understanding content analysis” on page 59 for more information about the overall concept of analysis.

The rest of this chapter provides information about the techniques that are available to create facets, including custom dictionary and custom text analysis rules.

## 7.3 Overview of techniques to create facets for analysis

As you saw in the previous scenarios for the sample collection, the out-of-box views can be very effective in surfacing insights in a data set. They can highlight key patterns and trends using the available facets that you have configured for your collection. This leads to a key inference: You need to have facets available that have valuable information related to the business questions and insights that you would like to address.

In many real-world instances, key information for analysis is not readily available in the source data. For social media information for example, there may be very little associated structured data. In order to generate insights, there will be the need to extract key elements from the unstructured text. These elements could range from basic product or service references, problem types, product features, names of service or sales representatives, or any other entity or concept that might explain why there is a problem (or opportunity), and lead you to a conclusion of what you might do to mitigate (or exploit) the situation. This information needs to be extracted and mapped to facets in order to enable you to analyze the collection as effectively as possible.

A content analytics collection will always automatically extract key parts-of-speech and phrase elements that will be available to you in out-of-box facets. Though these facets are valuable in themselves, there are a host of other techniques to provide more targeted groupings of words and phrases, or other useful categories to use in your analysis.

In this section, we provide an overview of techniques, roughly sorted by the level of effort required to implement, though that level of effort varies depending on how the technique is used. In some cases, the techniques are out-of-box options that can be turned on with a few mouse-clicks. In other cases, there are many different configuration options, and there will likely be testing to determine the optimum configuration of those options to reach your business goals.

### 7.3.1 Named Entity Extraction component

Content Analytics comes with several out-of-box analytics that you can quickly and easily enable in order to extract certain types of information from your unstructured text. The first of these analytics that we mention here is the *Named Entity Extraction* component. When this collection option is enabled, a “Named Entity” facet is created with three subfacets, each of which displays one of the three types of proper name information that this module captures: Person, Location, and Organization. Figure 7-13 on page 253 shows a sample Named entity view.

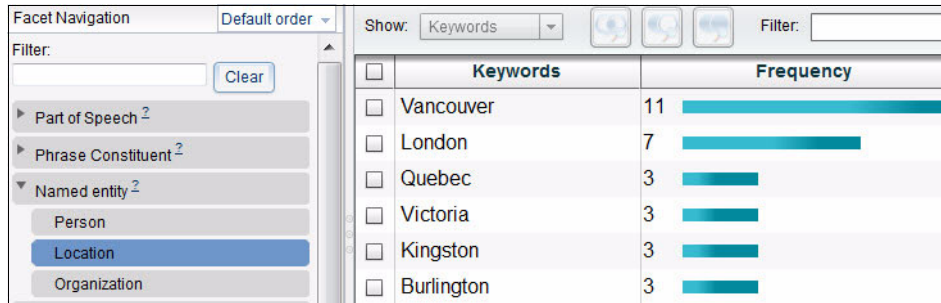


Figure 7-13 Named entity view

The administration interface allows you to modify which values are flagged (or not flagged). See Figure 7-14 for the configuration of the named entity recognition annotator.

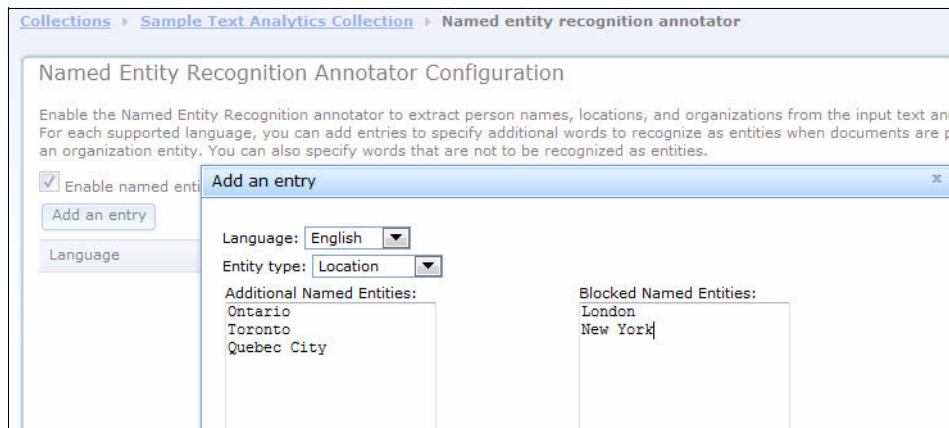


Figure 7-14 Named entity recognition annotator configuration

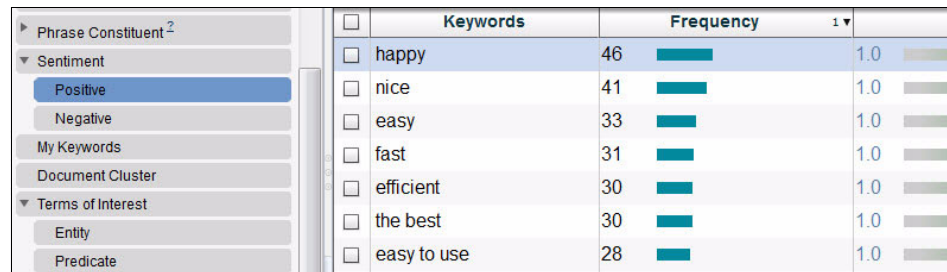
This facet information can be used as-is, or it can be used to provide candidate information for custom dictionaries that you might want to create. If your content does not have references to entities with proper names, this option will not provide value. For working with a custom dictionary for analytics, see 8.2, “Configuring dictionary-driven analytics” on page 277.

**Note:** The Named Entity Recognition annotator currently supports a subset of source languages. See the following link for current language support for this module:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/index.jsp?topic=%2Fcom.ibm.discovery.es.ta.doc%2Fiiystasystem.htm>

## 7.3.2 Sentiment

In use-cases with voice-of-the customer (VOC) comments or other opinionated content, sentiment analysis is usually a valuable tool to deploy. It can quickly highlight positive or negative phrases in the language to give you an at-a-glance view of overall polarity (that is, whether feedback is positive or negative) for a particular entity such as a product or service. Figure 7-15 shows a sample Sentiment view.



The screenshot shows a web interface for sentiment analysis. On the left is a navigation pane with options: Phrase Constituent<sup>2</sup>, Sentiment (expanded), Positive (selected), Negative, My Keywords, Document Cluster, Terms of Interest (expanded), Entity, and Predicate. The main area displays a table with columns for Keywords, Frequency, and a score of 1.0. Each row has a checkbox on the left.

<input type="checkbox"/>	Keywords	Frequency	1.0
<input type="checkbox"/>	happy	46	1.0
<input type="checkbox"/>	nice	41	1.0
<input type="checkbox"/>	easy	33	1.0
<input type="checkbox"/>	fast	31	1.0
<input type="checkbox"/>	efficient	30	1.0
<input type="checkbox"/>	the best	30	1.0
<input type="checkbox"/>	easy to use	28	1.0

Figure 7-15 Sentiment view

Sentiment Analysis is turned on for a collection via the administration console. As with Named Entity Recognition, the list of flagged items is configurable: You can add words, and specify whether they should be considered positive or negative. This is important because many words that might be negative in one context (“execution” in the law enforcement and corrections domain) might be positive in another (“execution” of financial transactions).

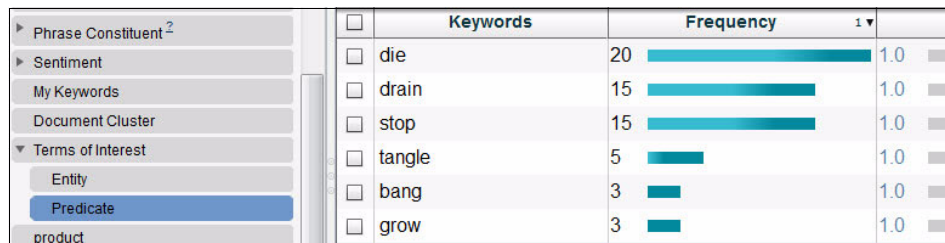
**Note:** The Sentiment Analysis annotator currently supports a subset of source languages. See the following link for current language support for this module:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/index.jsp?topic=%2Fcom.ibm.discovery.es.ad.doc%2Fiiysatasentiment.htm>

See 6.10, “Sentiment view” on page 222 for more details about enabling the sentiment facets.

### 7.3.3 Terms of interest

If your collection contains a significant number of problem descriptions, the terms-of-interest option can help to highlight problem-related terms found, and list those terms into separate subject and predicate facets. The algorithm does this by looking at the items surrounding adverbs that typically relate to problems (“suddenly”, “improperly”) as opposed to those that typically describe a normal situation (“properly”). You can use these facets as-is, or pick items from the list that are interesting to you, and use them to create your own dictionaries or pattern rules. Figure 7-16 shows a sample Term of Interest view.



<input type="checkbox"/>	Keywords	Frequency	1
<input type="checkbox"/>	die	20	1.0
<input type="checkbox"/>	drain	15	1.0
<input type="checkbox"/>	stop	15	1.0
<input type="checkbox"/>	tangle	5	1.0
<input type="checkbox"/>	bang	3	1.0
<input type="checkbox"/>	grow	3	1.0

Figure 7-16 Term of Interest view

Terms of interest can be enabled very easily with the administration console, and there is no additional configuration required. For more details about this feature, see section 8.1, “Terms of interest” on page 266.

### 7.3.4 Custom dictionaries

Custom dictionaries can be created to flag particular words and their synonyms, creating facet values for dictionary items found in each document. You can create a dictionary that is based on candidate words and phrases that are found in other facets, such as these out-of-box facets:

- ▶ Part of speech
- ▶ Phrase Constituent
- ▶ Named Entity
- ▶ Terms of interest
- ▶ Sentiment

These facets use out-of-box rules, and you might want to refine this output to remove “noise” to create a facet with a more focused view of the data. For example, you might want to remove references to any nouns except nouns that are related to your product or its features. Or, you might want to normalize and consolidate redundant items, such as recognizing that “New York” and “NY” are equivalent entries and should be counted in the same facet entry.

Lists of words and phrases can be deployed to a dictionary in several ways, including:

- ▶ Adding words in phrases one by one in the Content Analytics administration console. See 8.2, “Configuring dictionary-driven analytics” on page 277 for more information.
- ▶ Importing a comma-delimited list of words and phrases into a dictionary that has been initialized in the administration console.
- ▶ Creating a dictionary in ICA Studio and embedding the dictionary into an analytics package, which is deployed to the analytics pipeline for a collection. See Chapter 11, “Customizing content analytics with IBM Content Analytics Studio” on page 405 for more information.

The latter two options provide the ability to import a list of keywords that you might already have available, such as a company or product-related word list. When the list is converted to a supported format (such as a comma-delimited list), it can be easily deployed for use with your collection.

Another benefit of using ICA Studio for deploying custom dictionaries is that ICA Studio can draw from an online catalog of analytics modules, among which are a number of custom dictionaries. Depending on the domain and type of language you are analyzing, you might find a ready-to-use dictionary using the ICA Studio online catalog.

### 7.3.5 Facet ranges

Numeric and date values that are mapped to facets display by default as distinct facet values. For example, Figure 7-17 on page 257 shows how a facet with numeric information might look in an out-of-box configuration.



















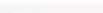
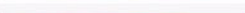
















<input type="checkbox"/>	Keywords	Frequency	Correlation
<input type="checkbox"/>	1500	124 	0.8 
<input type="checkbox"/>	750	270 	0.8 
<input type="checkbox"/>	1800	67 	0.7 
<input type="checkbox"/>	4200	9 	0.5 
<input type="checkbox"/>	3000	24 	0.5 
<input type="checkbox"/>	2700	14 	0.5 
<input type="checkbox"/>	1000	229 	0.5 
<input type="checkbox"/>	400	99 	0.5 
<input type="checkbox"/>	800	224 	0.4 
<input type="checkbox"/>	4800	19 	0.4 
<input type="checkbox"/>	1200	496 	0.4 
<input type="checkbox"/>	5800	6 	0.4 
<input type="checkbox"/>	1450	36 	0.4 
<input type="checkbox"/>	3600	34 	0.4 
<input type="checkbox"/>	3300	9 	0.3 
<input type="checkbox"/>	2300	8 	0.3 
<input type="checkbox"/>	2400	83 	0.3 

Figure 7-17 Default facet view with numeric values

In this case, showing every value is not the best way to display the data. If there are many different numeric values, it is generally more useful to aggregate the values into ranges, which makes them easier to understand and use during investigation. Figure 7-18 shows how similar data could look after organizing it into facet ranges.













<input type="checkbox"/>	Ranges <small>1 ▾</small>	Frequency	Correlation
<input type="checkbox"/>	I. > 1500	664 	0.5 
<input type="checkbox"/>	H. 1250-1499	45 	0.2 
<input type="checkbox"/>	G. 1000-1249	732 	0.4 
<input type="checkbox"/>	F. 750-999	498 	0.6 
<input type="checkbox"/>	E. 500-749	101 	0.2 
<input type="checkbox"/>	D. 250-499	228 	0.3 

Figure 7-18 Modified facet views using facet ranges when dealing with numeric values

### 7.3.6 Field Filters

In some cases, the source data includes fields that need manipulation or cleansing before they would be useful in a facet. For example, imagine you were analyzing “voice of the customer” data for retail goods, and wanted to understand

trends by country. If your source data did not have a “country” field, and instead had a single “retail store ID” field with country information embedded (sample values might be “US-1234”, “US-3456”, “UK-2345”, “JP-5678”, and so on), how could you create a facet that only displayed the country values?

One approach would be to preprocess your data before adding it to the content analytics collection using some manner of extract, transform, and load (ETL) tool. If this is not an option, the Field Filters capability can perform some of this functionality directly within the administration console. Field Filters can split field values into separate fields, concatenate field values, and a number of other operations. Figure 7-19 shows the configuration for a Field Filter.



Figure 7-19 Field Filter configuration window

### 7.3.7 Rule-based categories

Rule-based categories allow you to create facets, which group documents together based on queries or URI patterns. Content Analytics offers a powerful query syntax similar to popular Internet search engines, and creating queries can sometimes be the most straightforward way to group documents for particular goals.

For example, let us say that you wanted to identify documents with any reference to your chief competitor, the (fictional) XYZ corporation. There could be references in the documents to “XYZ”, or “XYZInc”, or “XYZCorp”, or “XYZWidgetsInc”. One approach would be to create a dictionary that captures all of these variations as synonyms. However, with this method you would run the risk of missing some of the variations.

Using Rule-Based Categories, you could easily create a query using wildcards, such as the query **'XYZ\*'**. If these company references were in a structured field, you could also use a field-based query syntax, such as **'companyname:XYZ\*'**. If you want to add references to another competitor (ABC Widgets Inc.) in the same facet, you can create a separate rule, or expand the query for the initial rule with an OR condition: **'XYZ\* OR ABC\*'**. Figure 7-20 shows the configuration window where you can add the rules.

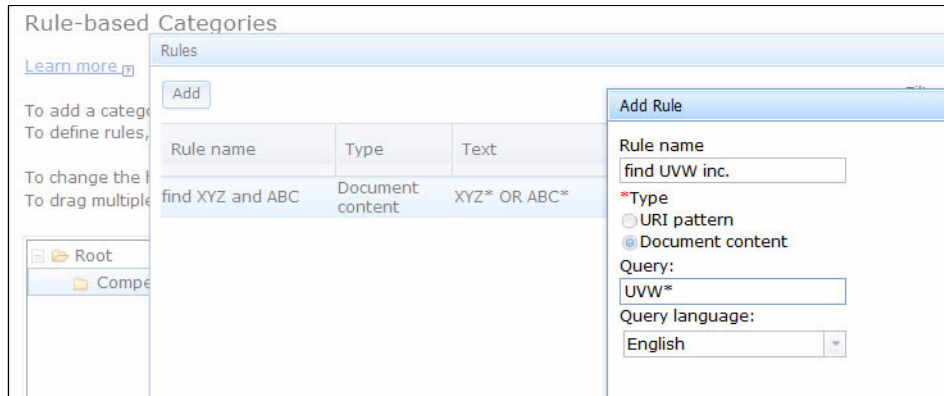


Figure 7-20 Adding rules for Rule-based Categories for analysis

See 5.5, “Rule-based categories with a query” on page 135 for more information about how to implement rule-based categories.

### 7.3.8 Syntax Pattern Rules

As mentioned in 7.2.1, “Scenario 1: Using a custom dictionary to discover package-related calls” on page 240, dictionary-based techniques only work with word sequences. However, a word sequence may not be adequate by itself to accurately signal a condition. For example, if you want to identify positive customer interactions, you might create a dictionary capturing the noun phrases “good service” or “very good service”. Unfortunately, this approach will result in false positive hits for comments such as “Jane did not provide very good service.”

To more precisely capture the concepts you are looking for, context is often required, which means you need to capture a syntax pattern, instead of a simple sequence of words. For the previous example, “Jane did not provide very good service”, we could use a dictionary of positive phrases which would have “good service” in the list, but we can extend the logic to look for accompanying phrases that would modify the intent. The nearby use of the word “not” would suggest that the concept of “good service” was being negated, and the sentiment is actually a negative as opposed to a positive comment.

Content Analytics offers tools to allow the creation of syntax pattern rules. As with dictionaries, pattern rules can be configured either directly within the administrative interface, or can be created with ICA Studio. Within the administration console you can create custom Text Analysis Rules, used in the scenario covered in 7.2.2, “Scenario 2: Using custom text analysis rules to discover trouble-related calls” on page 244, and covered in more detail in 8.3, “Configuring the Pattern Matcher annotator” on page 291.

ICA Studio provides a powerful interface with the ability to create very sophisticated parsing rules. In ICA Studio, rules can build off the results of previously configured dictionaries and rules, forming a powerful environment for identifying many types of text patterns in a flexible and efficient manner. See Chapter 11, “Customizing content analytics with IBM Content Analytics Studio” on page 405 for more information about ICA Studio.

### 7.3.9 Document clustering and classification

For many kinds of documents, the associated metadata and basic features that can be extracted via dictionaries and pattern rules provide plenty of useful facets to use for investigation. Moderate-length comments, such as Voice-of-the-Customer feedback, will often mention a specific product or problem, and might be accompanied by sufficient category metadata that can be mapped into facets to help your investigation.

Some documents are not as easy to categorize. How would you go about analyzing a set of larger documents, such as multi-page documents with a lot of text content, if little metadata was available? This is often the case with documents hosted on file systems, or other loosely managed repositories.

At the very least, it would be extremely useful to have a categorization of each document, as a starting point for analysis. For example, we might want to know the overall topic of each document, which could be a product area, or line-of-business. It would also be useful to understand the type or class of each document, to distinguish marketing presentations from legal contracts, or user documentation from internal engineering specifications.

A dictionary match or phrase pattern might not be very accurate for this categorization. A long marketing presentation could mention a number of products; simply identifying that ProductA is mentioned does not indicate that the document is primarily *about* ProductA.

To address these situations, there are a couple of techniques that take the entirety of the text content for each document into consideration, instead of scanning for predefined words and phrase patterns. They use sophisticated statistical techniques to determine the level of similarity between documents, or

the level of similarity between a particular document and a set of categories, based on example documents in each of those categories.

The following two techniques are used:

- ▶ Built-in Document Clustering
- ▶ Content Analytics integration with IBM Content Classification

Document Clustering allows you to select an arbitrary number of clusters for the system to automatically create. The clusters are created based on documents currently in the collection, and you can also choose the number of document samples, which will be used as input to create the statistical signatures that define each cluster. After defining the number of clusters and number of sample documents to use, the system will present you with a set of proposed clusters, as shown in Figure 7-21.

Parameters used by the document clustering task			
Name	Number of clusters	Number of samples	Clustering algorithm
Sample Test	30	1000	K-means
Proposed clusters:			
Select	Cluster	Cluster name	Number of documents Words in th
<input checked="" type="checkbox"/>	1	itim,ix,dn,xi,xii,specifying,gskit,ikeyman,ldap,characters	48
<input checked="" type="checkbox"/>	2	healthcare,sciences,pharmaceutical,biology,navigation,computational,proteomics,drug,clinical,rese	49
<input checked="" type="checkbox"/>	3	financingleasing,disposal,literatureanalyst,contentnorth,financing,buyback,retired,disposalenvironr	21

Figure 7-21 Sample document cluster proposed by the system

The name of each cluster will initially be set to a list of words that are strongly tied to that cluster, though you can change the name to a shorter, easier to use name.

Though Document Clustering provides a simple method for assigning a document to a category, there is limited control over the exact set of categories that can be used. A “cluster proposal” is created automatically, and may not match your wanted category set (that is, taxonomy).

In contrast, the integration with IBM Content Classification provides the ability for you to explicitly define the categories, based on a corporate taxonomy or other means. You would use IBM Content Classification user interfaces to configure your taxonomy and provide example documents to statistically define each category. You can even apply additional rules in a “Decision Plan” to allow the use of confidence thresholds or other criteria to influence how the Classification

server responds to requests. See Figure 7-22 as an example of a Content Classification Quick Start Tool interface for defining classification behavior.

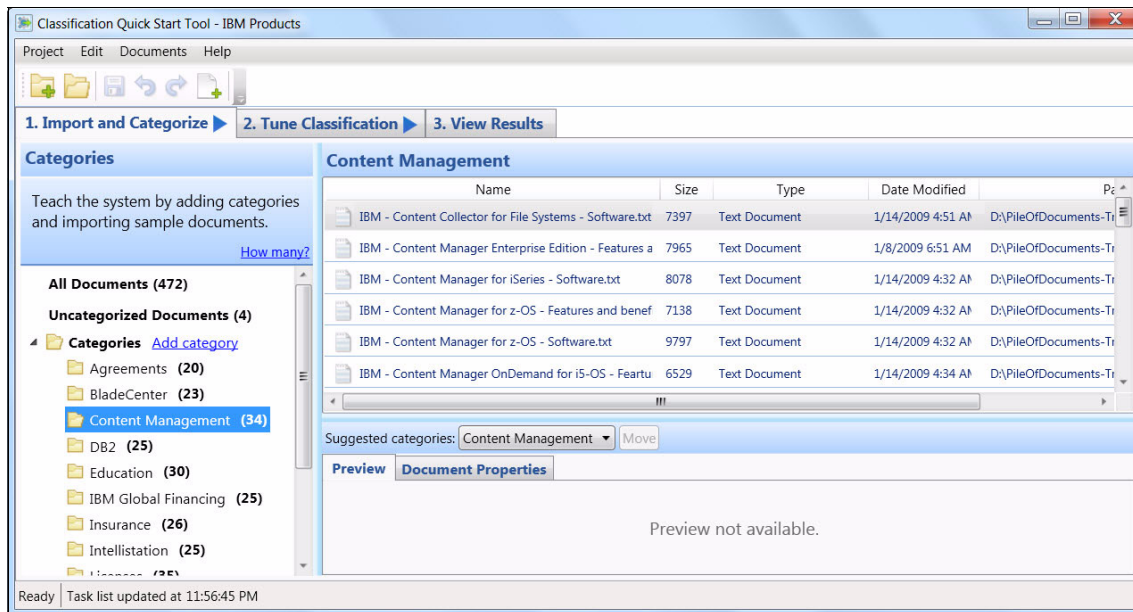


Figure 7-22 Content Classification Quick Start Tool user interface

See Chapter 9, “Content analysis with IBM Content Classification and document clustering” on page 305 for additional details about using Document clustering and classification techniques with Content Analytics.

## 7.4 Preferred practices

Text mining and gaining insight into your content is an iterative process. The following best-practice approach is based on years of field experiences:

- ▶ Assess your content and goals to determine which out-of-box analytics (Named Entity Recognition, Sentiment, Terms of Interest, and so on) will be valuable to configure.
- ▶ Start your analysis with a small collection. This makes it quicker to index the collection and allows more iterations of changes and tests. Gradually increase collection size as you get closer to a configuration that meets your goals.
- ▶ Follow the normal procedure several times:
  - Crawl, parse, index, and inspect the content by using the content analytics miner.

- Define new dictionaries, and inspect the content by using the content analytics miner.
- Define new syntax pattern rules, and inspect the content by using the content analytics miner.
- ▶ When you discover the need for more sophisticated analysis, use tools such as the following:
  - IBM Content Classification. See Chapter 9, “Content analysis with IBM Content Classification and document clustering” on page 305.
  - ICA Studio. See Chapter 11, “Customizing content analytics with IBM Content Analytics Studio” on page 405.







## Performing content analysis with built-in annotators

Chapter 5, “Content analytics miner: Basic features” on page 85 and Chapter 6, “Content analytics miner: Views” on page 159 describe basic features and operations of the content analytics miner including search and discovery features and views that are available for analysis. Chapter 7, “Performing content analysis” on page 231 provides additional guidance on performing content analysis by describing an overview of the range of techniques available to generate useful facets for analysis, and ways to effectively use the out-of-box views. In this chapter, we cover particular techniques in more depth.

Although certain advanced use-cases may be addressed by tools configured primarily outside of the IBM Watson Content Analytics (Content Analytics) administration console (such as IBM Content Analytics Studio (ICA Studio) and IBM Content Classification that have their own powerful task-specific configuration), this chapter focuses on several powerful built-in annotators that can be configured directly from the web-based administration interface.

This chapter covers the following sections:

- ▶ Terms of interest
- ▶ Configuring dictionary-driven analytics
- ▶ Configuring the Pattern Matcher annotator

*Terms of interest* is an available annotator that can help to identify words that appear to be associated with problem descriptions. The built-in *dictionary lookup* annotator (enabled by default) allows you to quickly apply domain-specific dictionaries to the analysis process. Finally, the *Pattern Matcher* annotator (enabled by default) allows you to create text analysis rules that look for syntax patterns in language, which can add more precision to the text analysis than dictionary matches alone.

**Note:** Dictionaries and pattern-matching rules can be created with the methods described in this chapter, or with a tool such as ICA Studio. The configuration steps in this chapter might be more efficient for implementing relatively straightforward dictionaries and rules directly in the Content Analytics administration console. In contrast, ICA Studio provides a separate developer-friendly interface to manage more sophisticated use-cases.

## 8.1 Terms of interest

You can derive various insights from a large amount of textual data. However, most novice users tend to analyze data without a customized dictionary. They also tend to perform aimless analysis of their data by using a broad range of expressions such as all the nouns that might not lead to any valuable results.

To help users gain valuable insight, Content Analytics automatically identifies candidates that might be of interest for content analysis. We call these terms, the *terms of interest*. Content Analytics provides terms of interest as an available annotator to help to identify words that appear to be associated with problem descriptions.

### Enabling terms of interest for new collections

To enable terms of interest for new collections, you can enable the option when creating your collection. To do so, follow these steps:

1. Click **Create Collection**, as shown in Figure 8-1 on page 267.

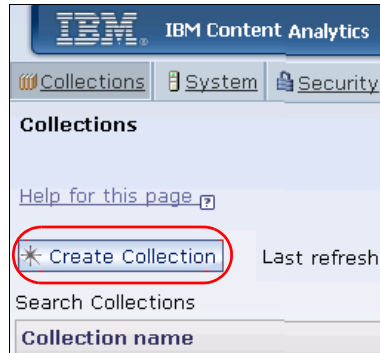


Figure 8-1 Create Collection button on the system using administration console

2. In the Create a Collection panel, click **Advanced options**, as shown in Figure 8-2.

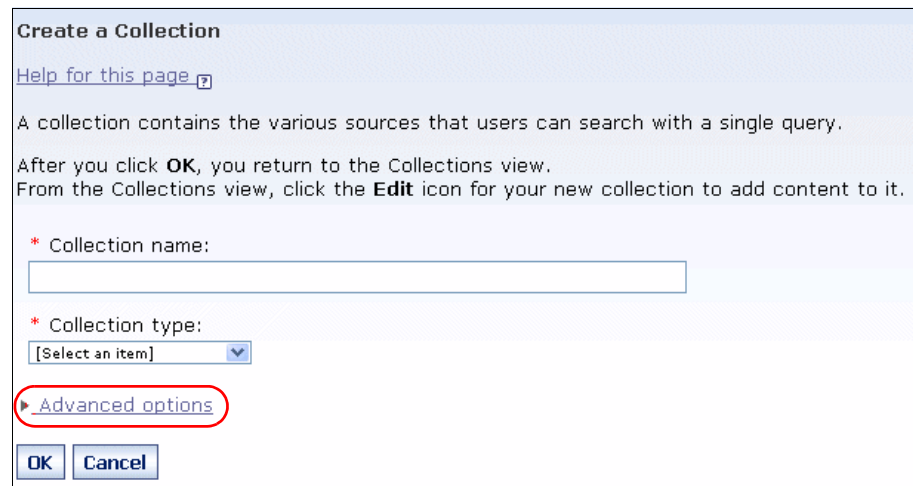


Figure 8-2 Selection of Advanced options

3. Under Advanced options, for Terms of interest, select **Enable automatic identification of terms of interest** to activate the terms of interest feature as shown in Figure 8-3 on page 268.

Advanced options

Description:

Estimated number of documents:  
 (This value is used to estimate resources, not to enforce a limit.)

Collection security (required for enforcing document-level security):

Document importance (static ranking model):

Duplicate document detection:

Optional facet index:

Terms of interest:

Figure 8-3 Selecting “Enable automatic identification of terms of interest”

## Enabling terms of interest for existing collections

For collections that are already created, you can enable this option by following these steps:

1. From Content Analytics Administration console, select **Actions** → **Settings** → **Edit collection settings**, as shown in Figure 8-4.

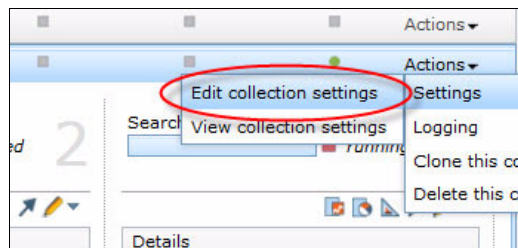


Figure 8-4 Select to edit collection settings

2. In the **Edit Collection Settings** window, under the Terms of interest section, select **Enable automatic identification of terms of interest**, as shown in Figure 8-5.

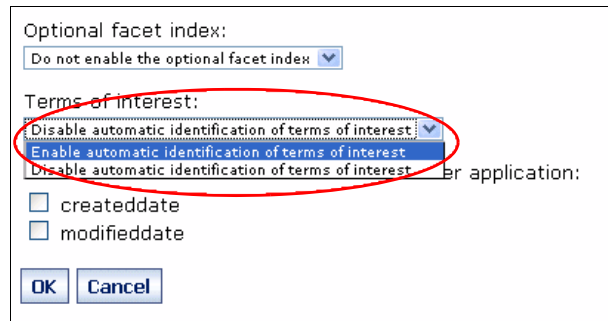


Figure 8-5 Enable automatic identification of terms of interest for a collection

After enabling automatic identification of terms of interest, make sure to deploy the resource and rebuild the full index to make this effective.

### 8.1.1 Basic algorithm for identifying terms of interest

Problem identification has been one of the most popular use-case scenarios for content analytics because it usually leads to action for reducing and preventing problems.

#### Predicates in terms of interest

In the complaint data about cars, problems are often described with predicates (verbs), such as “fail,” “stop,” and “leak.” To recognize complaints involving such predicates, you can prepare dictionary and pattern matching rules to identify these problem situations such as “fail,” “damage,” and “complain.”

Figure 8-6 on page 270 shows the distribution of general problem-type verbs in the nearly 620,000 car complaint records. Many of these verbs do not indicate specific problems, but rather indicate the existence of a potential problem. Moreover, specific problems usually depend on the domain context of the data. This domain dependence makes it impractical to predefine such problem expressions for diverse domains.

Keywords	Frequency	1 ▼
fail	71502	
lose	14581	
leak	12319	
damage	7386	
complain	7060	
malfunction	6678	
injure	6328	
crash	4863	
frustrate	856	
dissatisfy	334	
waste	257	
poison	15	

Figure 8-6 Distribution of predefined general problem verbs within car complaints

Another approach for capturing problem expressions is to use certain patterns such as verbs followed by “cannot,” “failed to,” and “not able to”, as shown in Figure 8-7. Although this approach might capture a wider variety of expressions, the frequency of many of such expressions is not very high, and the uses of low frequency expressions are not suitable for analysis of deviations and changes of trends.

Keywords	Frequency	1 ▼
unable to find	5077	
unable to duplicate	2680	
unable to get	2098	
unable to determine	2074	
unable to afford	1686	
unable to fix	1210	
unable to do	952	
unable to believe	756	
unable to locate	558	
unable to identify	553	
unable to help	512	
unable to turn	429	
unable to figure	406	

Figure 8-7 Distribution of expressions with predefined patterns within car complaints

A feature of Content Analytics is the automatic identification of terms of interest. Terms of interest are determined by using the nature of problem expressions that tend to co-occur with specific adverbial expressions such as “suddenly” and “often.” However, they do not co-occur with specific adverbial expressions such as “correctly” and “normally.”

For example, problems are typically reported in textual data with expressions, such as “my machine suddenly freezes,” indicating that “freeze” is the problem. Yet, non-problem expressions can be also modified by “suddenly” such as in “it suddenly worked,” indicating a recovery from some previous problem. On the contrary, problem expressions are seldom modified by “correctly” and “normally.” For example, “It freezes correctly” sounds odd if “freeze” indicates a problem.

Thus, by calculating the co-occurrence ratio of verbs with such adverbial expressions within an entire collection of data, Content Analytics can identify problem expressions automatically. It can also list them in the Predicate subfacet under the Terms of Interest facet. Because such adverbial expressions are limited and their concepts are language independent, Content Analytics provides this function for all supported languages.

“Predicate in Terms of Interest” usually contains keywords that indicate potential problems compared to “Verbs in Part of Speech” that lists all verbs without focusing on problem areas. Figure 8-8 shows the predicts in terms of interest such as “kill,” “destroy,” and verbs from parts of speech such as “have” and “drive”.

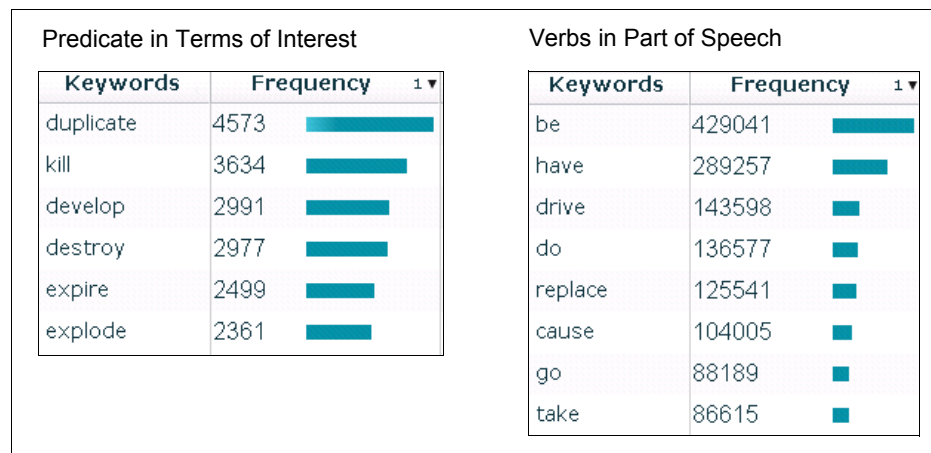


Figure 8-8 Comparing predicts (terms of interest) and verbs in car complaints collection

In general, this algorithm works well for a collection that contains many records with problem descriptions. For a collection with a relatively small number of records or with textual data that does not contain many problem descriptions, this

feature might not identify any terms of interest. Also, the facet might be sparsely populated. In particular, if none of the records contain “suddenly,” “often,” “sometimes,” or “frequently” in the case of English texts, this feature will not identify any terms of interest, and the Predicate subfacet under Terms of Interest facet will be empty.

## Entities in terms of interest

When “Predicates in Terms of Interest” are determined as candidate expressions, Content Analytics looks for nouns that tend to be associated with these predicates. These nouns are typically their subjects and direct objects and can be classified as candidates of entities that relate to the problem. As a result, identified nouns are listed in the Entity subfacet under the Terms of Interest facet.

“Entities in Terms of Interest” usually contains keywords that relate to a set of problems as compared to the “Nouns in Part of Speech” that lists all nouns without bringing attention to specific problem areas. Figure 8-9 shows entities in terms of interest, such as “pavement”, “road condition”, and nouns from parts of speech such as “vehicle” and “dealer”.

Entities in Term of Interest			Nouns in Part of Speech		
Keywords	Frequency	1 ▾	Keywords	Frequency	1 ▾
pavement	2898		vehicle	255098	
road condition	2475		Dealer	166452	
curb	2434		problem	162866	
shake	2201		car	106293	
			time	102844	
			consumer	94688	

Figure 8-9 Comparing entities (terms of interest) and nouns in car complaints collection

“Predicates in Terms of Interest” represent candidate problems. “Entities in Terms of Interest” represent potential entities that relate to problems, such as the cause of the problem and targets of the problem.

Similar to the case of a predicate, this algorithm generally works well for a collection that contains many records with problem descriptions. For a collection with a relatively small number of records or with textual data that does not specify any problem descriptions, this feature might not identify any terms of interest.

Because keywords in the Entity facet are derived from keywords in the Predicate facet, the values in the Entity facet are empty if the values of the Predicate facet are empty. If none of the records contain “suddenly,” “often,” “sometimes,” or



“frequently” in English texts, this feature does not identify any terms of interest, and both the facets will be empty.

## 8.1.2 Limitations in using automatic identification of terms of interest

The feature for identifying terms of interest can help users to identify valuable insights. The algorithm used to identify terms of interest is based on the statistical distribution of keywords associated with a set of specific adverbs. Therefore, the quality of the result might vary depending on the textual data within each collection. In particular, it requires a reasonable amount of predicate descriptions associated with “suddenly,” “often,” “sometimes,” “frequently,” “correctly,” “firmly,” “normally,” “properly,” and “securely,” and corresponding expressions for other languages.

In addition, because the algorithm is robust and language independent, the presence of noise (non-problem terms) is unavoidable. Therefore, we recommend treating terms of interest as *candidate* for terms of interest.

Based on our experiments, this algorithm works well for a collection that consists of a million records or more and that contains problem descriptions within a small domain where the use of each term is relatively consistent. It is important to understand the limitation of the terms of interest feature. Terms of interest might not work well for the following types of collection:

- ▶ A collection with few records
- ▶ A collection without problem descriptions
- ▶ A collection with textual content in a diverse domain where many of the keywords are polysemous with multiple meanings

For example, both uses of “freeze” for “refrigerate” and “halt” are observed in textual data within the same collection.

## 8.1.3 Preferred use of terms of interest identified automatically

To help users to gain valuable insight through Content Analytics, the keywords listed in the Terms of Interest facet might be used directly for problem detection and for correlation analysis to identify specific problems. In addition, the list of keywords provides good candidates to include in a dictionary.

### Terms of interest for problem detection

Although not all terms in Predicate of Terms of Interest are relevant to a specific problem, going through the list of keywords in the Predicate through Facet view can lead to actionable insights.

For example, through the analysis of a collection of customer satisfaction surveys, the keyword “change” was listed in the Facet view of Predicate under Terms of Interest. By focusing on the textual data containing “change,” we saw that a sudden change and frequent changes of their customer representative had a negative impact on customer satisfaction. This kind of insight leads to action, resulting in a modification to the customer relationship management strategy.

Therefore, looking at each keyword in Predicate of Terms of Interest can lead to the identification of noteworthy problems.

## Terms of interest for correlation analysis

The skill for acquiring valuable and actionable insights by using Content Analytics might require practice. It requires good sense, imagination, deep domain knowledge, and patience. You can improve this skill based on your experience. By using terms of interest, you can acquire valuable and actionable insights relatively easily.

When analyzing your collection, follow these steps:

1. Open the Facet view and select **Predicate under Terms of Interest**.
2. If you see a reasonable number of keywords listed and some of them seem to indicate a potential problem, open the Facet Pairs view by selecting **Predicate** under Terms of Interest in the Column.
3. Select other facets as the Rows and sort the Facet Pairs view by Correlation, as shown in Figure 8-10.

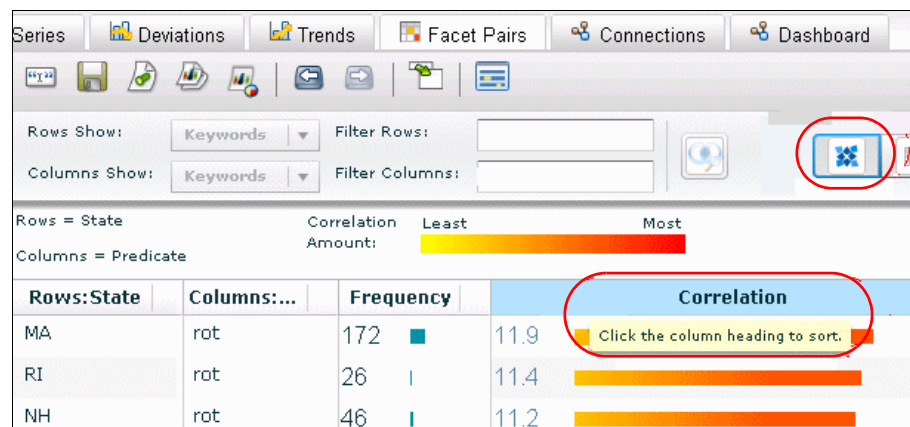


Figure 8-10 Sorting the Facet Pairs view by Correlation

4. If you find any noticeable correlation pairs, select the pair to focus on the data set associated with both keywords, as shown in Figure 8-11 on page 275.

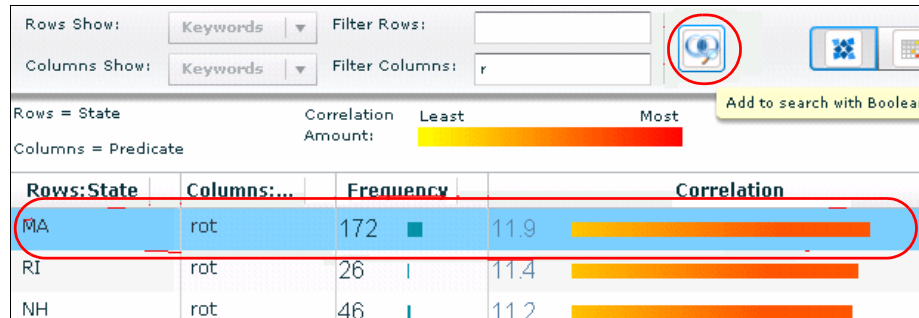


Figure 8-11 Selecting noticeable correlation pairs

5. Check the details of the data with the Facets view by selecting **Parts of Speech** and sorting the list by Correlation, as shown in Figure 8-12.

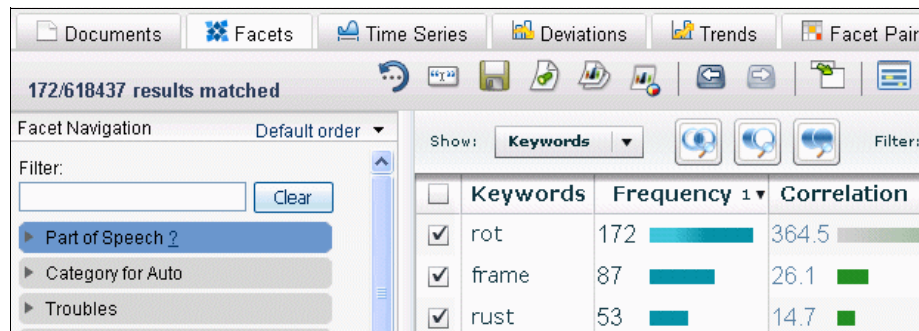


Figure 8-12 Analysis of features of the data focused at the Facet Pairs view

6. Verify the potential problem area.

In this example, the data indicates “rot” problems (with “frame” and “rust”) are typical in the northern eastern states of the US (New Hampshire, Road Island, and Massachusetts). You can verify this assumption by selecting the keywords “rot,” “frame,” and “rust,” and checking the Document view, as shown in Figure 8-13.

SEVERE UNDER BODY FRAME ROT. THE UNDERCARRIAGE AND FRAME OF THE VEHICLE IS SEVERELY CORRODED. THERE ARE NUMEROUS ROT HOLES IN THE BED FLOOR TO THE EXTENT THAT

Figure 8-13 Keywords identified from the Facet Pairs view and Facet view

In general, selection of product names often leads to valuable and actionable insights because it indicates problems that are specific to some products. For example, in the analysis of the complaint data about cars, you can find product-specific problems similar to those problems shown in Figure 8-14 on

page 276. The window displayed in Figure 8-14 shows a Facet Pairs view with facets from Model and Predicate in Terms of Interest.












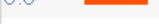

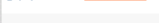

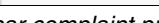
Rows:Vehicle...	Columns:Predicate	Frequency	Correlation
Model ABC	puncture	159 	10.2 
Model BBC	fade	47 	10.2 
Model CBC	consume	61 	10.1 
Model ABB	rot	44 	9.3 
Model ACC	flicker	112 	8.8 
Model ACB	rot	45 	8.6 
Model BBB	obstruct	29 	8.4 
Model CBA	lurch	220 	8.4 

Figure 8-14 Car models and their predicates for potential car complaint problems

As a result, you might find that many drivers of Model CBA reported that the car lurches when braking around pot holes, as illustrated in Figure 8-15. The keywords shown in Figure 8-15 are shown in the Facet Pairs view through the Facet view for **Phrase Constituent** → **Noun Phrase** → **Noun Sequence**.







Keywords	Frequency	Correlation
braking problem	14 	78.2 
pot hole	13 	29.2 
braking issue	6 	48.9 

Figure 8-15 Keywords showing potential problems

It is important to try the various facets in step 3 on page 274 with patience and imagination.

### Terms of interest as candidates for dictionary creation

As explained in 8.2, “Configuring dictionary-driven analytics” on page 277, a customized dictionary and pattern matching rules enrich your analysis and often lead you to better insights. It is always easier and better to make a dictionary and pattern matching rules from a meaningful list of candidate keywords instead of working from scratch.

The list of keywords in the Predicate of Terms of Interest facet is a good resource for creating pattern matching rules for a facet that might represent potential problems. To deal with a conjugated form, register predicates as pattern matching rules rather than registering them in a dictionary where terms are treated as nouns.

Compared to all the verbs in a collection, Predicates from Terms of Interest help narrow down the keywords that indicate potential problems. As an example, you can compare the keywords for all verbs in a collection with the keywords that indicate some problems in Figure 8-8 on page 271. Therefore, the use of keywords in Predicate of Terms of Interest makes the creation of pattern matching rules (as shown in 7.2.2, “Scenario 2: Using custom text analysis rules to discover trouble-related calls” on page 244) productive and effective.

It is important to understand that the Predicate of Terms of Interest might not contain all of the keywords that indicate potential problems. That is, pattern matching rules based on the keywords in just the Predicate facet might not be a complete set of pattern matching rules. However, creation of a dictionary and pattern matching rules is an iterative procedure. You must create a dictionary and pattern matching rules for a meaningful facet quickly. You must also apply the dictionary and pattern matching rules to your collection, and test how they lead to valuable insights.

After you understand the value of the dictionary and pattern matching rules for analysis, go through a longer list of keywords in a facet under Part of Speech, such as Noun and Verb, to enrich the dictionary and pattern matching rules.

Like the list of keywords in Predicate, the list of keywords in Entity of Terms of Interest is a good resource for creating dictionaries. In the case of keywords for the Entity facet, you might want to define multiple facets, such as “parts” or “components” in the car complaint scenario that are damaged and “causes” that create the problem.

Again, do not treat the list of keywords in the “entity” facet as a complete list. It might contain noise that is not relevant to any of the facets that are defined. Also, it might not contain all the keywords to be registered under each facet. Therefore, it is important to create a tentative dictionary quickly with minimum effort and apply it to your collection to check the feasibility to acquire such valuable insights.

## 8.2 Configuring dictionary-driven analytics

To get valuable insights from your data, it is important to identify the appropriate keywords and analyze their deviations and changes. Aimless analysis of facets with a broad range of values, such as all the nouns in your data collection, is not a good approach because the number of irrelevant terms (or noise) often masks the most important insights. Therefore, it is essential to select appropriate keywords for each specific purpose of analysis. It is also necessary to register them into a dictionary and pattern matching rules that are associated with their proper facets.

The best resource of terms to be registered in the dictionary and pattern matching rules is the collection that you are working with. To identify deviations and changes in keyword distribution that lead to valuable insights, the keywords in the dictionary must appear frequently in the textual data of the collection. If none of the terms in the dictionary appear in your data, the dictionary is useless. Therefore, instead of building a dictionary from scratch, always start with a list of keywords that are extracted from Content Analytics.

A list of nouns in the Facet view is a typical candidate list of keywords to be in many facets and in the Noun Sequence facet under the Noun Phase of Phrase Constituent. This list can be easily exported to your machine in the comma-separated value (CSV) format so that you can edit it by using a conventional spreadsheet application.

This section describes a scenario where a customized dictionary enhances the investigation of voice of the customer (VOC) information.

## **8.2.1 Multiple viewpoints for analyzing the same data**

An important step for taking advantage of Content Analytics is to set appropriate objectives for analysis. That is, match what you want to do with the data and what the data allows you to do. In our experience, even when the data does not provide the answers you need, it often discovers valuable insights that are unexpected. A dictionary plays a key role in customizing Content Analytics for each of your analysis objectives.

### **Analysis of complaint data for problem identification**

Consider an example of analyzing complaint data about cars that consists of problem reports from various drivers. In fact, such data is maintained and made public in many countries including the US, Japan, China, and France. This type of data typically consists of a textual description of each problem. It is accompanied by structured information, such as the report date, model of car, name of automotive company, and the area where the driver lives.

Many of the customer contact records, containing VOC data, share some essence with this complaint data about cars in terms of the textual description associated with various structured information.

A major use of complaint data for an automotive company is to identify critical defects that need fixing hopefully in their early stages. For this analysis, it is important to analyze the type of problems that occurred with which components.

Therefore, for this analysis, the following facets might be appropriate:

- ▶ The Problem facet, which consists of keywords that describe the nature of the problem, such as “leak,” “crack,” “fire,” and “blow”.
- ▶ The Component facet, which consists of keywords that describe the car components that are involved, such as “brake,” “engine,” “transmission,” and “steering wheel”.

By applying correlation analysis based on these two facets using the Facet Pairs view, you can identify notable defects for a specific car model compared to other models. For example, in Figure 8-16, which shows the correlation between car models and components, Model00077 has a strong correlation with the windshield wiper.











Model00077	windshield wiper	17		14.2	
Model00142	heater	22		11.3	
Model00142	pump	22		7.4	
Model00077	module	17		7.3	
Model00031	speedometer	37		7.3	

Figure 8-16 Facet Pairs correlation analysis between car models and components

The correlation index indicates that this model tends to have a problem with the wiper approximately 14 times higher than other models. By focusing on the 17 reports for Model00077 that describe a problem with the windshield wiper, you can check the Problem facet view to analyze the kind of problem that is typically reported on a wiper of this model. In this case, most of the 17 reports claims a similar phenomenon. Their windshield wipers did not always work when the switch was turned on, and they did not always stop when the switch was turned off.

## Analysis for additional insights from various viewpoints

The use of complaint data about cars is not limited to the analysis of car problems. In the complaint data, each problem is usually described in context, such as who is involved and in which circumstances.

### **Weather**

Consider a case where most of the data contains information about the weather conditions. By defining a Weather facet that consists of keywords such as “rain” and “snow,” you can analyze which cities are strongly associated with what type of weather. The example shown in Figure 8-17 on page 280 indicates that Buffalo has a stronger association with snow compared to other cities.

Rows: City	Columns: Weather	Frequency	Correlation
BUFFALO	snow	15	2.2
DENVER	snow	16	1.7
AURORA	snow	12	1.3
MILWAUKEE	snow	12	1.2
SAN ANTONIO	rain	33	1.1
JACKSONVILLE	rain	33	1.1

Figure 8-17 Facet Pairs view for analyzing the correlation between city and weather

By analyzing the distribution of states with the month of the year in the Deviations view after focusing on the data with rain, you can see the rainy season for each state, as revealed in Figure 8-18.

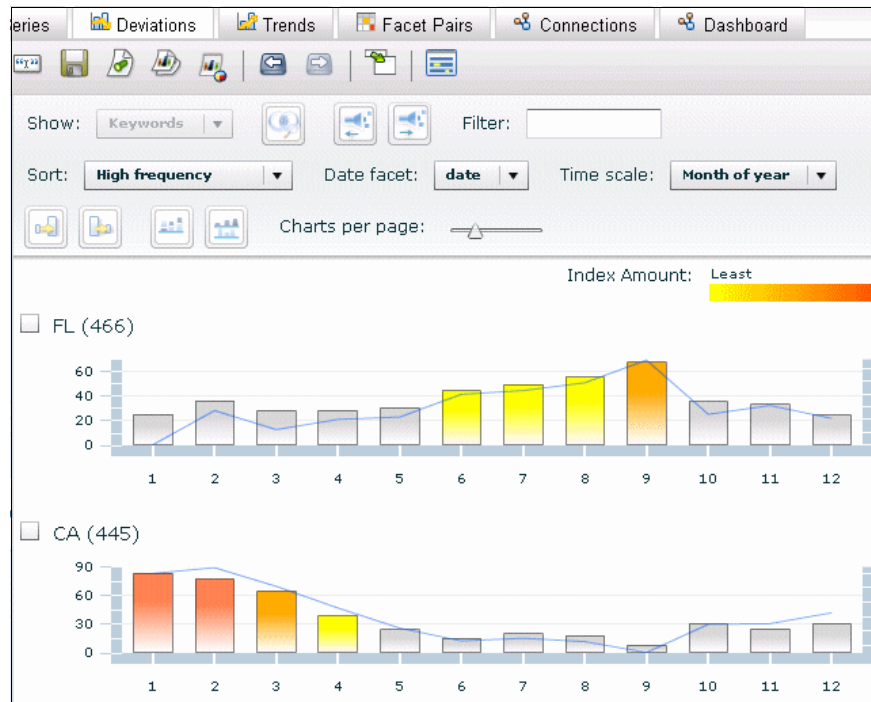


Figure 8-18 Deviations view for analyzing rain data by state

### Activities and families

Some of the data also contains information related to driver activities, such as shopping, vacation, and school. Figure 8-19 on page 281 indicates the high season within a year for each activity.



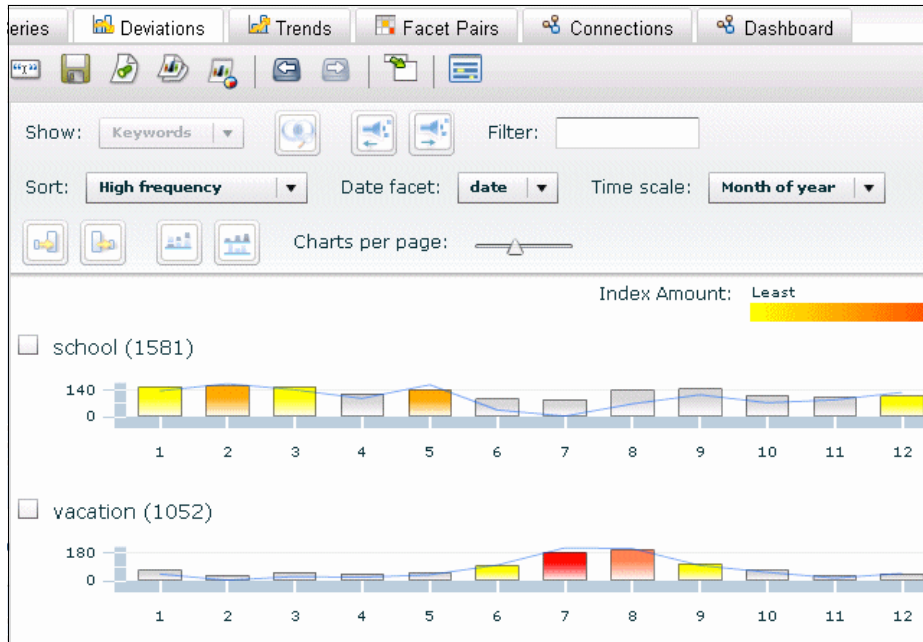


Figure 8-19 Deviations view for analyzing activities based on user-defined keywords

Figure 8-20 indicates which models are highly associated with each activity.

Rows: Vehicle/Equip. Model	Columns: activity	Frequency	Correlation	1 ▼
Model 001	vacation	16	3.1	
Model 007	school	30	2.7	
Model 005	shopping	23	2.4	

Figure 8-20 Facet Pairs view for analyzing the correlation between model and activity

Likewise, much of the data also contains information about families such as wife, kids, and parents. With this information, you can also analyze which cars are correlated to which family type and which activity.

## Extending use cases

By expanding the viewpoints with new facets, you can extend use cases. In the complaint data about cars, typical usage of car models might be identified, and such information might be valuable for product developments and target marketing. Analysis of environments might help manufacturers find good places and seasons for field-test environments.

Moreover, information from complaint data about cars can benefit car manufacturers, their dealers, and the following groups:

- ▶ Drivers to identify potential problems of their own cars
- ▶ Used-car dealers to estimate car conditions by model and manufacturing year
- ▶ Insurance companies to estimate the risk of each car

## 8.2.2 Configuring the Dictionary Lookup annotator

The Dictionary Lookup annotator matches words and synonyms from dictionaries with words in your text. The annotator also associates the keywords with user-defined facets.

Keywords are a critical element in text analysis. When you know particular terms in a specific domain, for example, product names, they are useful for extracting documents that belong to a specific domain. Keywords can be grouped by concept type and then used to identify documents with interesting combinations of these concepts.

The Dictionary Lookup annotator finds user-defined keywords in a document and associates the words with user-specified facets. The Dictionary Lookup annotator is a simple but powerful way to identify particular keywords.

Dictionary Lookup annotator is enabled by default. If you use the Dictionary Lookup annotator, you must also enable the Pattern Matcher annotator.

**Dictionary Editor:** The Dictionary Editor supports nouns only. You cannot use the Dictionary Editor to add other parts of speech such as verbs, adjectives, or adverbs. To capture other parts of speech, such as verbs and adjectives, use the Pattern Matcher annotator.

**Noun identification:** A document has zero to many fields. Each field has some attributes such as analyzable. For noun identification, text analytics is applied to all analyzable fields and content. XML documents are processed as content and thus can also be analyzed.

## 8.2.3 When to use the Dictionary Lookup annotator

Consider using the Dictionary Lookup annotator in the following cases:

- ▶ You want to see particular noun terms as a facet in the content analytics miner.
- ▶ You want to add new nouns to enhance the linguistic analysis process.

See 7.2.1, “Scenario 1: Using a custom dictionary to discover package- related calls” on page 240, for the reason why you might want to use the Dictionary Lookup annotator to create a custom dictionary.

## 8.2.4 Configuring custom user dictionaries

To create a custom user dictionary and to add, edit, and delete keywords and their synonyms, you can use the administration console. By using the scenario described in 7.2.1, “Scenario 1: Using a custom dictionary to discover package-related calls” on page 240, we show how to add the nouns **bag**, **bottle**, **cap**, **container**, **cup**, **material**, **pack**, **package**, **shape**, **spoon**, **straw**, and **top** that we selected from that scenario.

### Adding nouns to a custom dictionary

To add the nouns to your custom dictionary, follow these steps:

1. From the administration console, click **Parse and Index** from the collection to which you want to add the custom user dictionary.
2. Select **Text Analytics** → **Edit** → **Configure user dictionaries**.
3. In the Configure user dictionaries panel, complete these steps:
  - a. Enter the custom dictionary name. We type **package**.
  - b. For Language, select the language that the dictionary will be applied to. The drop-down list shows the languages that you selected when creating a collection. We select **English**.
  - c. Click **Create Dictionary**. See Figure 8-21.

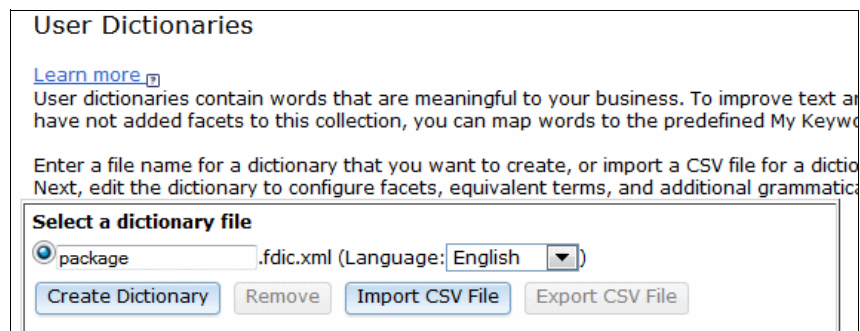


Figure 8-21 User Dictionaries configuration window

4. In the Dictionary Editor, create new keywords:
  - a. Click **Add Words** from the window that is shown in Figure 8-22.

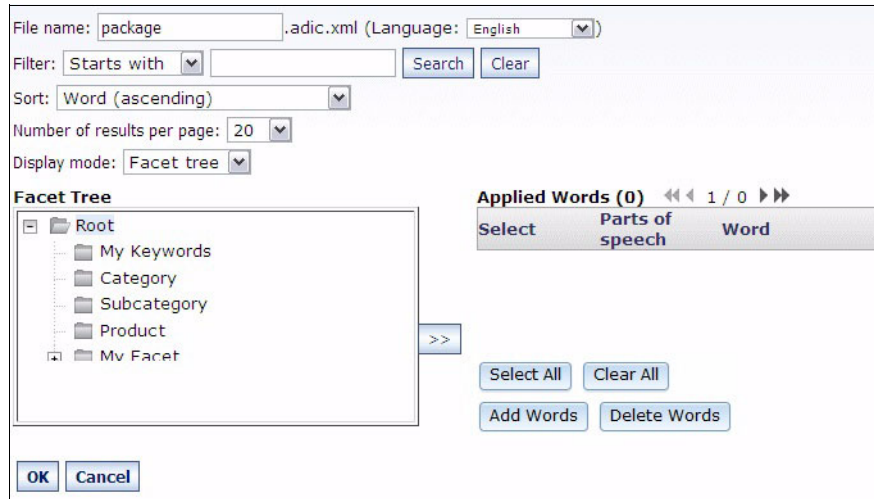


Figure 8-22 Add keywords

- b. In the Add Words window, enter keyword strings as shown in Figure 8-23. You can add multiple keywords at a time. The format is one keyword per line. We enter package, container, and other keywords. Click **Add**.

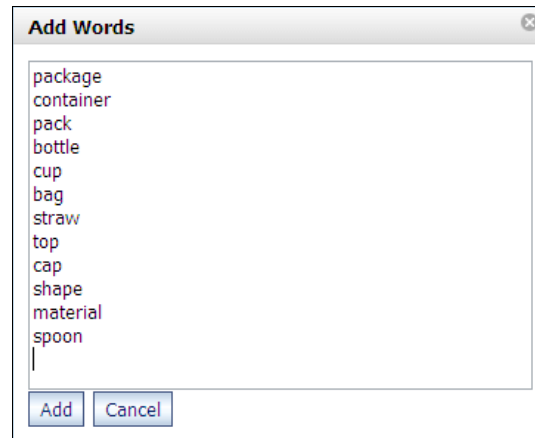


Figure 8-23 Adding new keywords in the dictionary

The keyword editor reflects the new keywords as shown in Figure 8-24 on page 285.

Applied Words (9) <<< 1 / 1 >>>					
Select	Part of Speech	Word	Equivalent Term	Facet	Word Options
<input type="checkbox"/>	Noun	bag			
<input type="checkbox"/>	Noun	bottle			
<input type="checkbox"/>	Noun	cap			
<input type="checkbox"/>	Noun	container			
<input type="checkbox"/>	Noun	package			
<input type="checkbox"/>	Noun	shape			
<input type="checkbox"/>	Noun	spoon			
<input type="checkbox"/>	Noun	straw			
<input type="checkbox"/>	Noun	top			

Figure 8-24 Keywords created

In many cases, keywords come with alias names and abbreviations. Keywords can also have variant forms (inflected forms). You can add multiple synonyms that have an identical meaning for a defined keyword. The Dictionary Lookup annotator captures synonyms and treats them as a single keyword.

5. To create an equivalent term for a keyword, from the Applied Words list, follow these steps:
  - a. Click the **Equivalent Term** icon for a given keyword.
  - b. Enter a new equivalent term in the text box. Click **Add**.
  - c. Repeat these steps until you add all synonyms for the given keyword.

For example, we add the abbreviation “pkg” to the keyword package, as shown in Figure 8-25.

Add Equivalent Terms				
New Equivalent Term: <input type="text" value="pkg"/>		<input type="button" value="Add"/>		
Facet Value	Word	Used as Facet Value	Used as Equivalent Term	Delete
<input checked="" type="radio"/>	package			
<input type="radio"/>	pkg			

Figure 8-25 Adding “pkg” as an equivalent term for the keyword “package”

In Figure 8-25 on page 285, the first entry, package, is selected as the keyword, which means that package is the normal form of these terms. If a word is already added as a normal form of the other term, then “Used As Facet Value” is selected. If a word is already added as an equivalent term for the other term, the “Used As Equivalent Term” is selected. These two features are warning signs for conflicts with other entries.

d. Click **OK**.

Dictionary Editor now reflects the new equivalent term, as shown in Figure 8-26.

Applied Words (9) <<< 1 / 1 >>>					
Select	Part of Speech	Word	Equivalent Term	Facet	Word Options
<input type="checkbox"/>	Noun	bag			
<input type="checkbox"/>	Noun	bottle			
<input type="checkbox"/>	Noun	cap			
<input type="checkbox"/>	Noun	container			
<input type="checkbox"/>	Noun	package	 pkg PKG		
<input type="checkbox"/>	Noun	shape			
<input type="checkbox"/>	Noun	spoon			

Figure 8-26 Dictionary reflecting “pkg” as equivalent term for “package”

**Equivalent Terms:** Synonyms are not displayed as discrete words with their associated facet in the content analytics miner. Only the keyword (not its synonyms) is displayed (in this example, package) in the Facets view.

6. Associate the defined keywords to a facet. If you have not created a facet, follow the steps in “Creating facets and mapping search fields to facets” on page 120 in the previous version of this book, which can be downloaded as part of the additional material for this book.

To associate keywords to a facet, follow these steps:

- a. In the Facet Tree view, select a facet.
- b. In the Applied Words table, select the keywords that are associated with the selected facet.
- c. Click the **Assign** button (>>).

In this example, we assign all keywords to the Package facet under My Facet, as shown in Figure 8-27.

The screenshot shows the Dictionary Editor interface. On the left is the 'Facet Tree' with a hierarchy: Root > My Keywords > Category > Subcategory > Product > My Facet > Package. The 'Package' facet is selected. On the right is the 'Applied Words (12)' table, which lists 12 words, all of which are checked in the 'Select' column. Below the table are four buttons: 'Select All', 'Deselect All', 'Add Words', and 'Delete Words'.

Select	Parts of speech	Word
<input checked="" type="checkbox"/>	Noun	bag
<input checked="" type="checkbox"/>	Noun	bottle
<input checked="" type="checkbox"/>	Noun	cap
<input checked="" type="checkbox"/>	Noun	container
<input checked="" type="checkbox"/>	Noun	cup
<input checked="" type="checkbox"/>	Noun	material
<input checked="" type="checkbox"/>	Noun	pack
<input checked="" type="checkbox"/>	Noun	package = PKG = pkg
<input checked="" type="checkbox"/>	Noun	shape
<input checked="" type="checkbox"/>	Noun	spoon
<input checked="" type="checkbox"/>	Noun	straw
<input checked="" type="checkbox"/>	Noun	top

Figure 8-27 Associating keywords to a facet

Dictionary Editor now reflects the facet that you associated with the keywords (Figure 8-28 on page 288).

Applied Words (9) << 1 / 1 >>						
Select	Part of Speech	Word	Equivalent Term	Facet	Word Options	
<input type="checkbox"/>	Noun	bag		Package		
<input type="checkbox"/>	Noun	bottle		Package		
<input type="checkbox"/>	Noun	cap		Package		
<input type="checkbox"/>	Noun	container		Package		
<input type="checkbox"/>	Noun	package	 pkg PKG	Package		
<input type="checkbox"/>	Noun	shape		Package		
<input type="checkbox"/>	Noun	spoon		Package		
<input type="checkbox"/>	Noun	straw		Package		
<input type="checkbox"/>	Noun	top		Package		

Figure 8-28 Facets associated with keywords

- d. Click **OK** to save the dictionary.
7. If you have other word lists that you identified from another perspective, add multiple dictionaries for a collection. Repeat step 3 on page 283 through step 6 on page 286. Figure 8-29 shows another dictionary created to capture keywords related to “flavor.”

**Select a dictionary file**

.fdic.xml (Language: English )

flavor.fdic.xml

package.fdic.xml

Figure 8-29 Dictionary that captures keywords related to “flavor”

**Conflicts:** Dictionary Editor only checks conflicts within the same dictionary. It is important to store keywords that belong to the same facet in a single dictionary.

You can select an existing dictionary and select “Export CSV File” to generate a CSV equivalent of the dictionary, which can facilitate comparing multiple dictionaries with a text editor or other similar tool.



## Applying and removing words for text analysis

Applied words are the keywords that are used for text analysis. If the words fall into the candidate words list, these words are not used for text analysis. Moving words to the Candidate Words list is useful when evaluating potential words to include in your dictionaries. Figure 8-30 shows the Candidates mode. In this example, we move the keyword “box” from the Applied Words list to the Candidate Words list. This way, the noun “box” will not be used in future analysis. In moving a keyword from the Applied Words box, notice that the associated equivalent terms also move with the selected word.

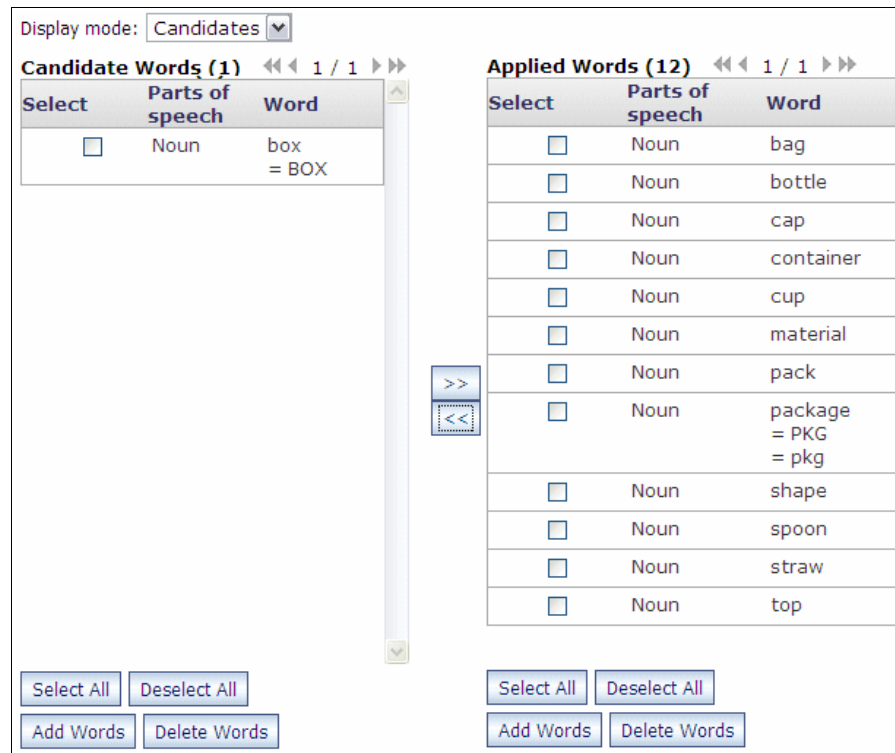


Figure 8-30 Candidate mode of Dictionary Editor

## Importing CSV files

In addition to adding keywords and their synonyms using the Dictionary Editor, you can also import a comma-separated values (CSV) file that lists words that you want to add in a dictionary. To import a CSV file, follow these steps:

1. From the User Dictionaries configuration window (Figure 8-21 on page 283), click **CSV Import**.

2. In the CSV Import window (Figure 8-31), complete the following steps:
  - a. Specify an Adic file name. You can select an existing adic file to update or create a dictionary.
  - b. For Language, specify the language that the dictionary will apply when creating a dictionary.
  - c. Select a CSV file. The first column in each row is a keyword. The rest of the columns are treated as synonyms. Example 8-1 shows the CSV file that contains two keywords (package and container). PKG and pkg are synonyms of the keyword “package.”

*Example 8-1 CSV file that contains two keywords*

---

```
package,PKG,pkg
container
```

---

- d. Select an appropriate encoding for the CSV file.
- e. Select a facet to map if necessary.
- f. Click **OK**.

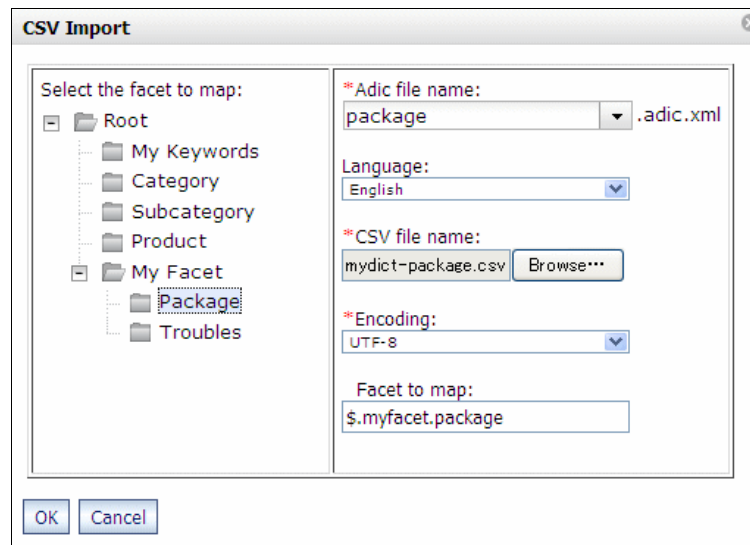


Figure 8-31 Importing a CSV file

After importing the CSV file, you can add, edit, and delete keywords and their synonyms using the Dictionary Editor.

For more information about Dictionary Editor, see the “Configuring user dictionaries” topic in the IBM Content Analytics Information Center at the following address:

<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iisatauserdict.htm>

After you create a user dictionary, you must deploy the text analytics resources. You must also rebuild an existing index to update the results in the content analytics miner.

## 8.2.5 Validation and maintenance

The easiest way to confirm that the words in your dictionary match the words in your textual content is to use the Facets view in the content analytics miner. When you add keywords and associate them with a particular facet, you can see the added keywords in the Facets view if your collection contains documents with these keywords.

You can easily add, edit, and remove keywords by using the Dictionary Editor. Remember to deploy resources and rebuild the index when you update the dictionary so that the latest dictionary is reflected in the content analytics miner. The rebuild task might take a long time depending on the size and number of documents in your content analytics collection.

**Tip for building a dictionary:** When building your dictionary, build it iteratively with a small subset of your content. You can start from a small document set and check the results to make sure that they are what you expected. You can also find other keywords to include in your dictionary during this process. Also, enhance your dictionary iteratively. When you think you are done with the iteration process, you can then apply your dictionary to the entire collection.

## 8.3 Configuring the Pattern Matcher annotator

The Pattern Matcher annotator identifies patterns in your text by using the rules that you defined. The annotator also associates the patterns with user-defined facets.

Using keywords for text analysis is a simple and powerful way to identify documents that contain particular keywords and their synonyms. However, using individual keywords alone is not enough to discover and normalize concepts and ideas. For example, the keyword “milk” can help us easily identify all documents that contain the word milk. However, matching single words is not enough to

extract the true context of the documents in many other cases. For example, consider the following sentence:

The product is not broken.

When you try a traditional search with the keywords “product” and “broken,” any document that contains this sentence is returned in the search results. However, this sentence does not indicate a product problem. We want to distinguish concepts that a document presents, such as broken or not broken. In this case, the following rules can help to decode the actual information in a document:

```
[product name] + [be] + [negative term] = product problem  
[product name] + [be] + [not] + [positive term] = product problem
```

The first rule states any product name that is followed by a be type of verb and a negative term is considered a product problem. The second rule states that any product name that is followed by a be type of verb, a not type of word, and a positive term is considered a product problem. This product problem is then considered as a keyword phrase that can be associated with a facet and can be used for analysis. In this scenario, documents that contain any of the following sentences are displayed as having a product problem in the analysis:

ProductA is broken.

ProductBs are not working.

The Pattern Matcher annotator recognizes the sequences of words that are defined in the rules and associates them with specified facets. With the Pattern Matcher annotator, you can add custom rules.

The Pattern Matcher annotator is enabled by default. It produces the Part of Speech and Phrase Constituent facets, which are predefined by default.

**Disabling Pattern Matcher:** If you disable the Pattern Matcher annotator, the predefined Part of Speech and Phrase Constituent facets do not show any results.

### 8.3.1 When to use the Pattern Matcher annotator

For reasons why you might want to use the Pattern Matcher annotator, see 7.2.2, “Scenario 2: Using custom text analysis rules to discover trouble-related calls” on page 244.

In general, you use the Pattern Matcher annotator in the following cases:

- ▶ You want to extract sequences of words (single word and multiple words).
- ▶ You want to capture patterns that are constructed by multiple words.

An alternative approach to use Pattern Matcher annotator is using ICA Studio.

### 8.3.2 Configuring custom text analysis rules

To construct rules for Pattern Matcher annotator, a certain degree of linguistic knowledge is required.

To create a custom rule file, and edit text analysis rules, you can use the administration console. Using the scenario in 7.2.2, “Scenario 2: Using custom text analysis rules to discover trouble-related calls” on page 244, we add the rules as shown in Example 8-2 on page 294 to extract terms that are possible signs of troubles.

To create the rules, follow these steps:

1. From the administration console, select the **Parse and Index** pencil icon of the collection that you want to add a custom rule file, and select **Analytic Resources** → **Custom text analysis rules**, as shown in Figure 8-32.

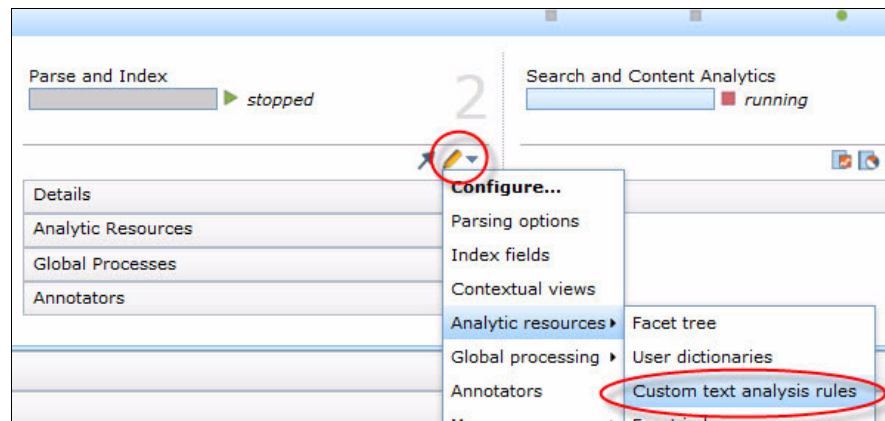


Figure 8-32 Creating custom text analysis rules

- In the Text Analysis Rules window as shown in Figure 8-33, enter the custom rule file name and click **Open**.

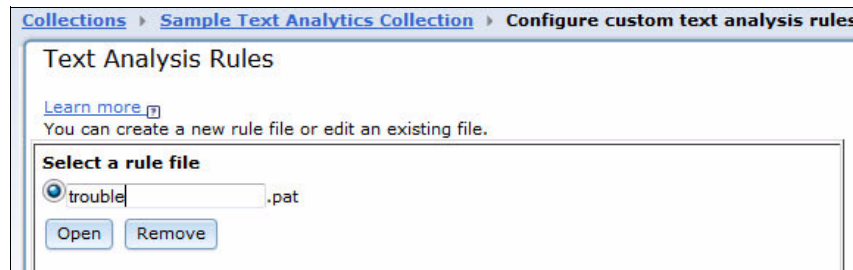


Figure 8-33 Entering a custom rule file name

- Add the custom rules. Example 8-2 shows the rule structure in the .pat format.

Example 8-2 Pattern rule syntax

---

```
<pattern-list lang="en">
  <mi category="$.myfacet.favorable" value="able to ${4.lex}">
    <w id="1" lex="be" str="!/^(was)|(WAS))$/"/>
    <w id="2" lex="able"/>
    <w id="3" lex="to"/>
    <w id="4" pos="verb"/>
  </mi>
</pattern-list>
```

---

To create the XML to define the rules, follow these steps:

- Add the top-level element pattern list, which must specify the target language as the lang attribute.
- Add a <mi> element that represents a rule. You need to specify the facet path as the category attribute. The facet path must start with \$. The dot (.) is the path separator. For example, for the word favorable under myfacet, use \$.myfacet.favorable. Also give the value attribute that is shown in the content analytics miner as the keyword.

In a <mi> element, the actual pattern consists of one or more <w> elements that represent tokens. The <w> element must have a token ID. You can add several constraints by using the str, lex, pos, ftrs, category, and guard attributes. With the Pattern Matcher annotator, you can use regular expression matching in the constraints.

For more information about the rule development, see 8.3.3, “Designing the custom text analysis rules” on page 296.

Figure 8-34 shows the added custom rules in the administration console.

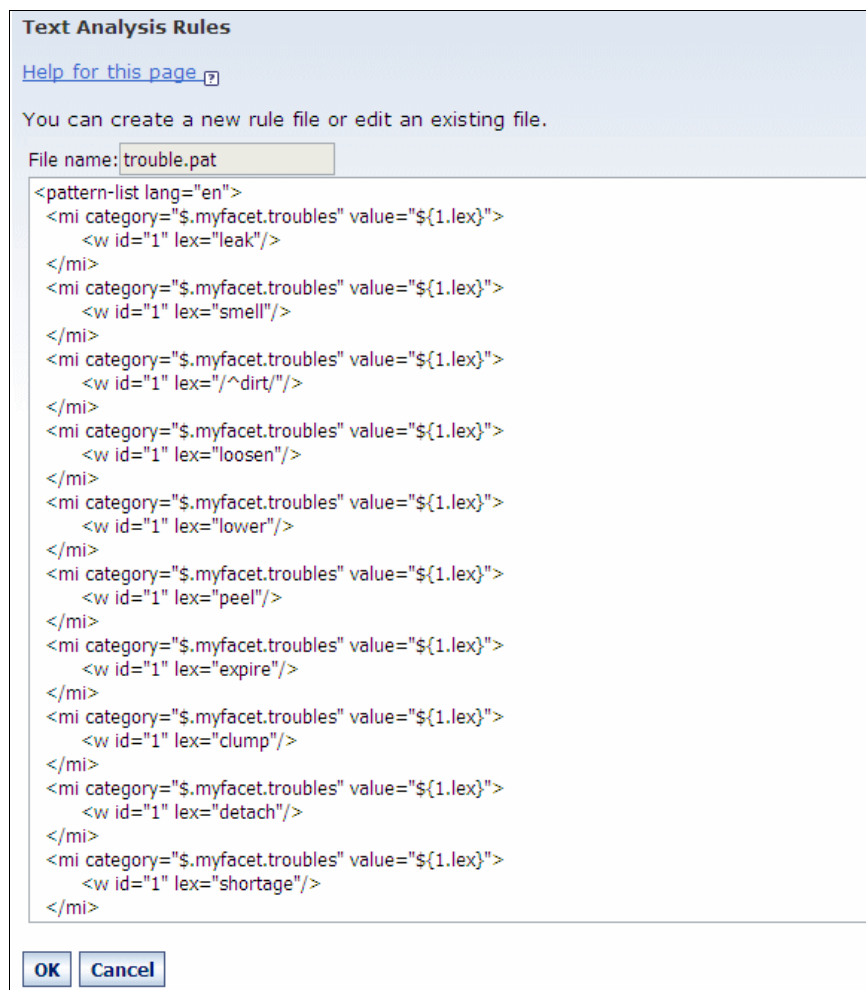


Figure 8-34 Text Analytics Rules Editor

c. Click **OK** to save the rule file.

You can add multiple rule files for a collection.

After creating the rule file, deploy the text analytics resources. In addition, rebuild an existing index to update the results in the content analytics miner.

**Hints:** Copy the sample pattern file, and edit it by using an appropriate application that can assist in XML editing. Writing analytic rules from scratch can cause problems even though the XML syntax is simple. The sample pattern files for VOC analysis are in the ES\_INSTALL\_ROOT/samples/voc/pattern/ directory.

### 8.3.3 Designing the custom text analysis rules

By defining your own text analysis rules, you can extract concepts that are expressed in natural language.

This section focuses on designing custom text analysis rules. Here, we do not explain the details of the text analysis rule syntax. Instead, we pick up several situations in our analysis of the voice of the customer and provide some perspectives about text analysis rule development. For more information about the custom analysis rule syntax, see the “Custom rule files for content analytics collections” topic in the IBM Content Analytics Information Center at the following address:

<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ta.doc/iiysatextanalrules.htm>

You can also see the practical pattern files for voice of the customer analysis that are in the ES\_INSTALL\_ROOT/samples/voc/pattern/ directory.

#### **Sample rule: Capturing question-related information**

In VOC data, you often encounter questions posed by the customer. For example, you might find that a customer says “I need information about ...”. This phrase seems to be a common request. Therefore, you want to capture these types of questions where customers have asked for information associated with a particular subject. You want to create a text analysis rule that gathers these kinds of phrases. You start by creating a custom text analysis rule to capture this simple expression, as shown in Example 8-3.

*Example 8-3 Simple rule to capture the expression “I need information”*

```
<mi category="$ .voc.question" value="Question ${4.str} ${5.str}
${6.str}">
  <w id="1" str="I"/>
  <w id="2" str="need"/>
  <w id="3" str="information"/>
  <w id="4"/>
  <w id="5"/>
  <w id="6"/>
```



</mi>

---

Each <w> element defines a token to be extracted and analyzed. Tokens #1, #2, and #3 use a simple string constraint that matches an actual word. Token #4, #5, and #6 do not have any constraint, meaning they can be any token in a sentence.

In the <mi> element, you must specify the facet path as the category attribute and the value attribute that is showing in the application as the keyword. The value attribute is a template of the output keyword.

After you save this rule within the Text Analytics Rules Editor, you must apply it and verify results. In this case, assume that the pattern matcher annotator examines the sentence “I need information about Text Mining.” You can see that the annotator captures this sentence and assigns it to the VOC/Question facet with a keyword of “Question about Text Mining” as shown in Figure 8-35.

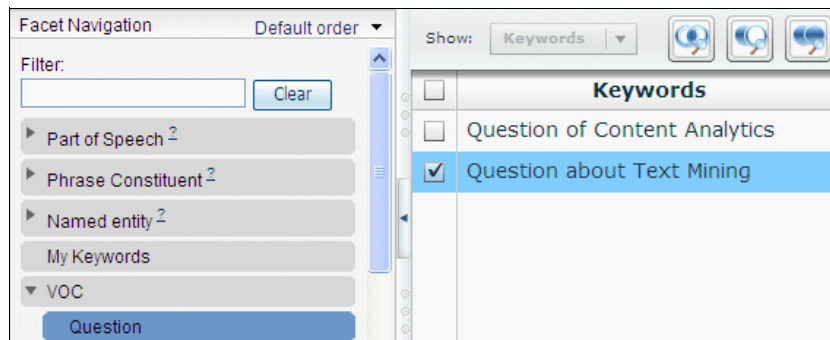


Figure 8-35 Facets view showing results of the sample rule

You also notice another generic expression that asks a question directly. To see the actual sentences, you can attempt a simple search with the keyword “question” and find several sentences in your collection:

I got a question regarding ...  
I have a question about ...  
I have several questions on ...

To capture this type of expression, we add a rule as shown in Example 8-4.

Example 8-4 Rule to capture question-related information

---

```
<mi category="$.voc.question" value="Question ${5.lex} ${6.lex}
${7.lex}">
  <w id="1" str="I"/>
  <w id="2" lex="/(^have$)|(^get$)/"/>
  <w id="3"/>
```

```
<w id="4" lex="question"/>
<w id="5"/>
<w id="6"/>
<w id="7"/>
</mi>
```

---

The rule starts with a simple string constraint that matches “I.” Token #2 uses the lex attribute that stands for the lemma (linguistic normalized form of the word). With this attribute, you can write a constraint without listing all possible inflected forms of a word. For example, `lex="get"` matches “get,” “gets,” “getting,” “got,” and “gotten” that have the normalized form “get”. The lex value of token #2 is using regular expression matching. All rules with the forward slash (/) operator are evaluated by using `java.util.regex` classes. This lex value uses the pipe (|), ^, \$, and () operators, so that it matches “have”, “had”, “get”, and “got”. It does not match “haven’t” and “forget” for example.

The list below contains various `java.util.regex` operators for custom rules:

- ▶ / operator: The / operator itself is not a part of the Java regular expression syntax. It is used only to invoke regular expression processing on a rule.
- ▶ | operator: The pipe (|) operator is interpreted as the Boolean OR operator. This operator is a literal constraint that is not interpreted by the regular expression interpreter. Matches occur when a document contains one of the words in the list of words to be evaluated.
- ▶ ^ operator: This operator matches only at the beginning of a line. For example, `str="/^love/"` matches loves and lovely, but does not match beloved or glove.
- ▶ \$ operator: This operator matches only at the end of a line. For example, `str="/love$/"` matches love and glove, but does not match lovely or loves.
- ▶ () operator: Parentheses are the grouping operator. Use this operator when you want more than one word and variations of those words to be evaluated by the Java regex interpreter.

Token #3 does not have any constraints. This token can be anything in a sentence. It is a place holder for any article or numeral such as “a,” “the,” “one,” or “some”. Token #4 also uses a lex constraint to ignore a difference between a singular form (question) and a plural form (questions). Tokens #5, #6, and #7 can be any token.

The value attribute in `<mi>` element uses the lex variable instead of the str variable. It intends to create facet values with no distinction, for example, between “I have a question about Text Mining” and “I have a question about text mining.”

After adding this rule, you must apply them and verify the results again. Rule development is an iterative process. Start with a small set of rules. Then check the results to make sure that they work as expected. Continue to enhance the rules. The rules can be iteratively improved.

### Sample rule: Capturing people who are unable to do something

Next, you want to understand what customers cannot do. For example, you might look for question-type phrases that are expressed indirectly, such as “I cannot install the application” or “I could not find a catalog.” You first start with the simple rule shown in Example 8-5.

**Cannot to can not:** If a sentence contains “cannot”, the document processor breaks it down into “can” and “not”.

*Example 8-5 Simple rule to capture an expression “someone cannot do something”*

---

```
<mi category="$.voc.question" value="Unable to ${3.lex} ${4.lex}
${5.lex}">
  <w id="1" lex="can"/>
  <w id="2" lex="not"/>
  <w id="3"/>
  <w id="4"/>
  <w id="5"/>
</mi>
```

---

After applying this rule, you confirm that the example sentences are captured as expected. However, this rule also captures system error messages, such as “Process cannot be terminated in.” For your purposes, you want to distinguish between question-type phrases and error messages. We change the rule as shown in Example 8-6 to accommodate this case.

*Example 8-6 Improved rule to capture an expression “someone cannot do something”*

---

```
<mi category="$.voc.question" value="Unable to ${3.lex} ${5.lex}">
  <w id="1" lex="can"/>
  <w id="2" lex="not"/>
  <w id="3" lex="!be"/>
  <w id="4"/>
  <w id="5" pos="noun"/>
</mi>
```

---

Token #3 uses the lex constraint with the logically not (!) operator. It intends not to match phrases that are in the passive voice. Also we use the part-of-speech

constraint in Token #5. The last token has to be a noun. The *pos* attribute stands for part-of-speech. The Pattern Matcher annotator supports 11 parts of speech:

- ▶ adjective
- ▶ adposition
- ▶ adverb
- ▶ conjunction
- ▶ determiner
- ▶ interjection
- ▶ noun
- ▶ numeral
- ▶ pronoun
- ▶ residual
- ▶ verb

Consider this sentence, for example:

Ah, product #200 is great but really expensive for me!

The document processor categorizes tokens into the corresponding part of speech (Table 8-1). You can write a rule that pertains to a particular part of speech.

*Table 8-1 Part-of-speech values*

Part-of-speech	Token
pronoun	me
verb	is
noun	product
adjective	great expensive
adverb	really
adposition	for
interjection	Ah
conjunction	but
determiner	the
numeral	200
residual	, # !

Token #4 can be anything, such as “a” or “the.” This token is not important in categorizing the common phrase and is, therefore, no longer part of a keyword value template.

Then, you must apply the rule again, and check the results. The rule captures the example sentences and ignores the system error messages as expected. However, the rule might not capture sentences such as “I cannot open files.” If you must capture such sentences as this example, you can add a more generic rule (Example 8-7) in addition to the previous one.

*Example 8-7 Additional rule to capture common phrases*

---

```

<mi category="$.voc.question" value="Unable to ${3.lex}">
  <w id="1" lex="can"/>
  <w id="2" lex="not"/>
  <w id="3" lex="!be"/>
</mi>

```

---

Example 8-6 on page 299 and Example 8-7 on page 301 can apply to the same sentence.

### Sample rule: Capturing negative phrases

To capture the mood of customers, you might want to create rules that are associated with negative phrases. Typical phrases that customers use when they are unhappy or unsatisfied are “... not happy with something” or “... not satisfied with something.” To capture this type of expression, you can use the rules that are shown in Example 8-8.

#### *Example 8-8 Rules to capture negative phrases*

---

```
<mi category="$.voc.negative-phrase" value="Not ${2.lex} with
${4.lex}">
  <w id="1" lex="/(not$)|(isnt$)|(arent$)/"/>
  <w id="2" lex="/^((happy)|(satisfy)|(satisfactory))$"/>
  <w id="3" lex="with"/>
  <w id="4" pos="noun"/>
</mi>
<mi category="$.voc.negative-phrase" value="Not ${2.lex} with
${5.lex}">
  <w id="1" lex="/(not$)|(isnt$)|(arent$)/"/>
  <w id="2" lex="/^((happy)|(satisfy)|(satisfactory))$"/>
  <w id="3" lex="with"/>
  <w id="4" pos="/^((determiner)|(adjective))$"/>
  <w id="5" pos="noun"/>
</mi>
```

---

“isnt\$” and “arent\$” in the token #1 capture “isnt” and “arent” that are denoted literally in a sentence. For example, the rules in Example 8-8 capture a sentence, such as “this is not satisfactory” and “this isn’t satisfactory.” They also capture a sentence such as “this isnt satisfactory.”

The difference between the first rule and second rule is if there is a determiner or an adjective before a noun.

To add more flexibility to the rule, we can refer to a dictionary instead of hardcoding a set of words. For example, in the rules above we specifically look for variations of “happy”, “satisfy”, and “satisfactory” by explicitly referencing those words. If this set of words were expected to change, we could maintain them separately in a dictionary, replacing the token...

```
<w id="2" lex="/^((happy)|(satisfy)|(satisfactory))$"/>
```

with a reference to the “happy\_terms” dictionary-based facet as follows:

```
<w id="2" category="$.happy_terms"/>
```

In this way, we could add values such as “please” or “impress” in a referenced dictionary without further complicating the rule.

### 8.3.4 Validation and maintenance

Similar to the Dictionary Lookup annotator, you can check the result of your custom text analysis rules in the Facets view of the content analytics miner. One alternative to validate the results is to use Real-time Natural Language Processing (NLP) capability, which is useful for evaluating rules interactively.

Develop your rules iteratively. Do not try to stretch a rule to make it solve every problem. Complicated rules make maintenance difficult. It is important to keep rules short and simple.







# Content analysis with IBM Content Classification and document clustering

IBM Content Classification can classify document content into categories. The Content Classification annotator that is available through IBM Watson Content Analytics (Content Analytics) enables automatic text classification and context-based text-understanding. This chapter provides details about the Content Classification annotator and explains how to use the annotator for content analysis.

This chapter includes the following sections:

- ▶ The Content Classification annotator
- ▶ Fine-tuning your analysis with the Content Classification annotator
- ▶ Creating and deploying the Content Classification resource
- ▶ Validation and maintenance of the Content Classification annotator
- ▶ Preferred practices for Content Classification annotator usage
- ▶ Document clustering

## 9.1 The Content Classification annotator

The Content Classification annotator is integrated inside the Unstructured Information Management Architecture (UIMA) document processing pipeline of Content Analytics. The Content Classification annotator uses the capabilities of Content Classification to classify content into categories and generate metadata information that can be used for facets or keywords in Content Analytics.

Content Classification uses sophisticated natural language processing and semantic analysis algorithms to determine the true intent of words and phrases. It then uses that knowledge to automate classification.

Accuracy improves over time because the system adapts to your content by identifying different categories from examples that you provide. When you provide feedback, the system adjusts in real time and immediately incorporates any corrections that you make. The accuracy of the classification results keeps pace with changes in your content and environment.

The Content Classification annotator combines contextual statistical analysis with a rule-based, decision-making approach. For example, the system can identify keywords, patterns, and words within a certain proximity of each other. When content that matches a condition in a rule is detected, the action defined for the rule is applied, and content is classified accordingly.

In addition to organizing information by policies or keywords, the Content Classification annotator can also assign metadata that is based on the full context of the document. The classification process searches for a single word or phrase. It also analyzes the entire document, distills the main point of the text, and assigns the text to a category.

### 9.1.1 When to use the Content Classification annotator

The classification capability of the Content Classification annotator is used to categorize text or make correlations between text and objects (for example, personalization or general data classification applications). Search, text mining, and classification are often integrated together in a single system. They are synergistic for several reasons.

Search, text mining, and classification provide complementary mechanisms for describing documents. *Search* and *text mining* find and describe documents based on a small set of words supplied by users (such as the query “energy bill”). *Classification* attempts to describe the overall document based on a set of descriptors supplied by the taxonomy (for example, one of the subjects in a

subject taxonomy). That is, if a search engine supplies the category to the user, it can be easy for the user to distinguish which search results are relevant.

For example, if the user query is “energy bill,” some of the results are marked as a piece of energy legislation, but others are marked as a monthly electric or gas bill. A user seeing this mixture of topics can then refine the query to select just the ones intended by this ambiguous query.

Search and classification can be paired in the following ways:

- ▶ Search within a category. You can select a category and then search only documents that are both within the category and that match your query.
- ▶ Facet search. In this method, you can specify several different facets (or characteristics) of a document to a search engine (for example, “search for all PDF documents about databases from last year”). This search is a generalization of “search within a category,” where multiple criteria that might or might not be categories from a taxonomy can be combined.
- ▶ Taxonomy browsing. Some or all of the documents on a website are displayed as a taxonomy that can be navigated, with each document assigned to one or more nodes of the taxonomy.
- ▶ Classifying search results. The results of a search are displayed together with their assigned categories. Categories can be used to group or sort result sets.

### 9.1.2 The Content Classification technology

Understanding and classifying text is an old problem in the field of artificial intelligence. Determining the most likely category in which to classify a new content item is not trivial.

With IBM Content Classification technology, applications can understand and classify unstructured free-form text. The Content Classification annotator attempts to understand information based on its existing knowledge. It “learns” how to distinguish and classify data based on its acquired information. For example, the technology learns how to distinguish between text about dogs and cats, after you provide it with example text about dogs and another set of example text about cats. Then, the system attempts to correctly classify the new text as being related to dogs or cats.

The Content Classification annotator learns from real-world examples and stores classification information into what is referred to as a *knowledge base*. The Content Classification annotator consults the knowledge base to classify new text into categories based on their similarity to the text seen in the past.

To create the knowledge base, the Content Classification annotator is first trained by using a body of sample data, such as emails, documents, or other text, that has been preclassified into appropriate categories. This body of data is known as a *corpus*. The corpus consists of sample text that represents the kind of information that the system is expected to classify. It creates statistical models that make up the knowledge base of a system.

After the Content Classification annotator is trained, new text can be submitted for classification by using a process called *matching*. The Content Classification annotator analyzes the new text and computes relevancy scores for each category in the knowledge base, as a measure of how closely the text matches each category.

After creating and training a knowledge base, you can build a Content Classification decision plan that contains rules that refer to the knowledge base suggestions. The decision plan can also extract metadata that is associated with the text.

Figure 9-1 shows an example of importing a sample set of documents to the Content Classification annotator to create and train a knowledge base to recognize different types of burn documents. The types of burns include corneal burns, fire burns, and chemical burns.

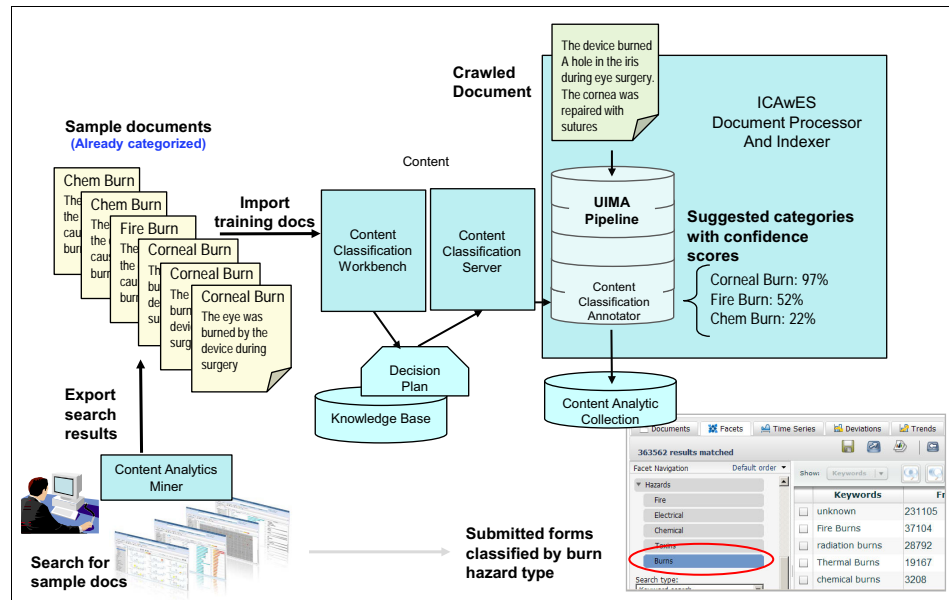


Figure 9-1 Integration of Content Classification and Content Analytics

After the knowledge base is trained, you use the Content Classification annotator to categorize all the documents in terms of various burns with a metadata set on the documents. The metadata information is then used for Hazard type → Burn facet (which is also called the *Burn Hazard facet* for simplicity). You can then use Content Analytics to analyze the different types of burns and any correlation to other factors and discover important insights related to the burns or other hazard-related matters.

## 9.2 Fine-tuning your analysis with the Content Classification annotator

This section explains how to use the Content Classification annotator to fine-tune your content analysis.

### 9.2.1 Building your collection

The first step when working with the Content Classification annotator is to gather sample content and categorize the content into buckets that represent different categories. The content and the associated categories are then used to train a Content Classification knowledge base.

The example uses a set of documents that contain information related to a Fictitious Medical Devices Company A. The data set describes various adverse events for medical devices. The company wants to gain insight into its content and wants to quickly identify quality control issues and fix them. The company also wants to be prepared for legal and regulatory actions.

To analyze the content, follow the standard procedure for building a content analytics collection by using the documents of the manufacturer:

1. Build a collection from the documents. The documents come with a list of fields with well-defined values, such as equipment name and type. They also come with a series of textual fields that contain free format text information.
2. Define facets and associate them with data fields and textual fields from the content.
3. Define new facets and associate them with dictionary entries that are relevant to the content case of the manufacturer. For this example, you define a new facet called *Hazards* that categorizes all hazards caused by the failing devices: Fire, chemical, toxic, or electrical. You also create synonyms to be used in the dictionary. For example, for fire-related hazards, you create the synonyms “burn,” “smoke,” “blaze,” and “flame” for “fire.”

4. Use the content analytics miner of Content Analytics to find problems related to the medical devices and procedures described in the documents.

By looking at the documents under the Fire Hazards facet, you see several documents that do not refer specifically to hazards caused by fire. However, these documents are in the view because the word “burn” was found in them, and they were labeled as such. Finding documents that have the words, but where the usage is unrelated, is one of the drawbacks of the rule-based dictionary approach to classification. In this example, you find a series of documents related to “corneal burn,” similar to the example in Figure 9-2, that do not have anything to do with a fire hazard in these scenarios.

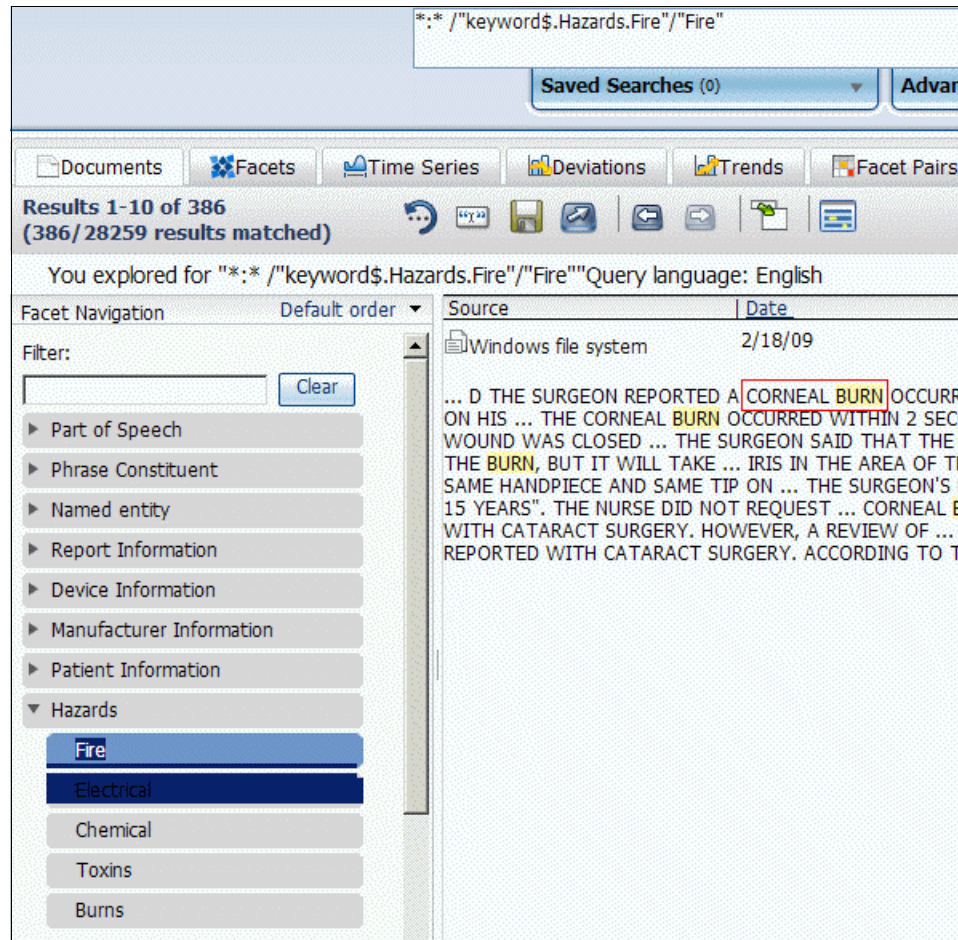


Figure 9-2 Example of an unrelated search result in the Fire Hazards facet view

## 9.2.2 Refining the analysis

Because you encountered several documents unrelated to the Fire Hazard facet, you must refine the analysis and look for more accurate techniques for text mining. The Content Classification annotator can help you to improve the overall accuracy of classification. Content Classification has the unique ability to consider the entire context of the document and not just a few words that are in the document. With this ability, Content Classification can differentiate better between the different types of burns (chemical burns, radiation burns, and thermal burns). In the example, you use the Content Classification annotator to learn the difference between the different types of burns and to correctly categorize them.

First, you gather a few examples of documents for training the Content Classification annotator to identify these different types of *burn hazards*. For the example, you use the export capabilities of Content Analytics Version 3.0 (as explained in 10.6.3, “Exporting search result documents to the file system for IBM Content Classification” on page 376). You use them to gather relevant content for training the knowledge base and designing the decision plan in Content Classification:

1. Use the content analytics miner in Content Analytics to identify sample documents for each of the categories: Chemical burn, radiation burn, fire burn, and thermal burn.
2. Search for “chemical burn” and review the documents.
3. After you are satisfied with the example documents that you find in the content analytics miner, click **Export**. Make sure to provide a meaningful name for the XML file and Description fields. The name that you assign for the Description field becomes the Content Classification Category label for the current set of examples. Therefore, the name must be meaningful and exact.
4. Repeat steps 2 and 3 to gather examples for all the categories that you plan for Content Classification to learn. When you finalize this process, you have a directory with a series of XML files that contain sample documents for training a Content Classification knowledge base. The `catalog.xml` file contains the information regarding the sample data fields names.

To use the Content Classification annotator, create the following resources in Content Classification and publish them to the Content Classification server:

- ▶ Knowledge base
- ▶ Decision plan that is related to the knowledge base

### 9.2.3 Using a conceptual search for advanced content discovery

In a traditional text search, both documents and queries are regarded as sets of terms. A document is a good match for a search query if the document contains terms of that query. Documents with more terms in common with the query are ranked higher in the search result. If you need to discover documents that are conceptually similar, even if the query terms do not exactly match the document content, you can enable a conceptual search within the collection.

For more information about setting up the conceptual search, go to IBM Content Analytics Information Center at the following address, and search on *configuring Content Classification search fields and scores*:

<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/index.jsp>

Through the process of a concept-based search, you can discover documents that are most similar to the query in terms of their classification results. Content Classification is used in Content Analytics to categorize documents and assign them a relevancy score for each category suggested in the results set. Based on this information, the conceptual search returns its results.

In addition, the document categorization can be based on the clustering discovery. The conceptual search can be run against the categorization determined by clustering for the collection to further refine the categorization.

#### **Using a conceptual search in Content Analytics**

Conceptual searches can be used in Content Analytics for the following reasons:

- ▶ To improve search ranking based on conceptual resemblance. Documents that conceptually resemble a search query are regarded as highly relevant in the search results. This type of search helps to improve the quality of the search results ranking and helps users to find documents they are looking for without knowing the exact terms contained in the documents.
- ▶ To filter documents that conceptually match a query regardless of whether these documents contain the search query terms. If documents do not contain any query terms, but contain terms that are used frequently with the query terms, they are included in the search results.



**Reference materials:** For information about building and maintaining a decision plan and knowledge base in Content Classification, see the following resources:

- ▶ IBM Content Classification tutorial

<http://publib.boulder.ibm.com/infocenter/classify/v8r8/index.jsp>

For hints and tips about building a knowledge base and decision plan, see *IBM Classification Module: Make It Work for You*, SG24-7707.

## 9.3 Creating and deploying the Content Classification resource

This section explains the step-by-step procedures for creating the knowledge bases and decision plan in Content Classification and for deploying them into the Content Analytics document processing pipeline. It does not attempt to teach you how to do everything in Content Classification. For comprehensive Content Classification product usage, see the materials referenced in the previous shaded box.

Creating and deploying Content Classification resources entails the following tasks, which are explained in the sections that follow:

1. Starting the Content Classification server
2. Creating and training the knowledge bases
3. Creating a decision plan
4. Deploying the knowledge base and decision plan
5. Configuring the Content Classification annotator

### 9.3.1 Starting the Content Classification server

Before you create the knowledge base and decision plan, start the Content Classification server:

1. Click the **Services** icon in your taskbar to open the Services Management Console in Windows.
2. Make sure that the IBM Content Classification Process Manager is running. If it is not running, in Windows, right-click **IBM Content Classification Process Manager** and select **Start the service**.
3. Open the Content Classification Management Console by selecting **Start** → **All Programs** → **IBM Content Classification 8.8** → **Management Console**.

The Content Classification Management Console is the Content Classification Server administration tool with which you can manage all the knowledge bases and decision plans. In this chapter, Content Classification Management Console is used to ensure that the decision plans and knowledge bases that are needed for the content analytics collections are installed on the server and running.

### 9.3.2 Creating and training the knowledge bases

To create a knowledge base, follow these steps:

1. Start the Classification Workbench by selecting **Start** → **All Programs** → **IBM Content Classification 8.8** → **Classification Workbench**.
2. Select **Project** → **New** → **Knowledge Base**.
3. In the New Project window (Figure 9-3), type the name of the knowledge base. For this example, enter Burn Hazard KB. Then, click **Next**.

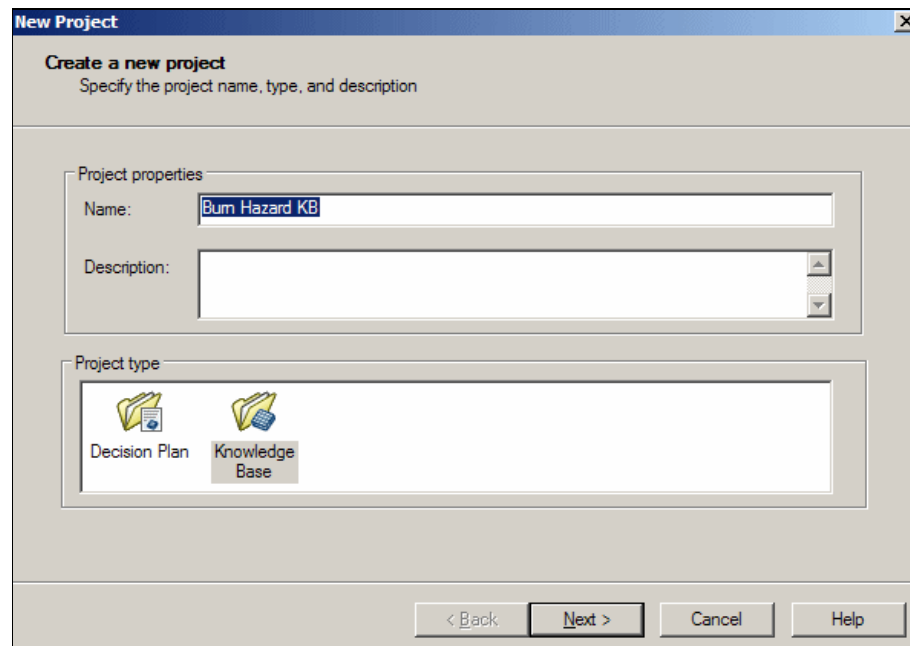


Figure 9-3 Creating a Content Classification knowledge base project

4. Import the XML files that were exported by Content Analytics. Follow the defaults of the wizard until you reach the next step to import the content set.

- In the Import Content Set window (Figure 9-4), select **XML** and click **Next**.

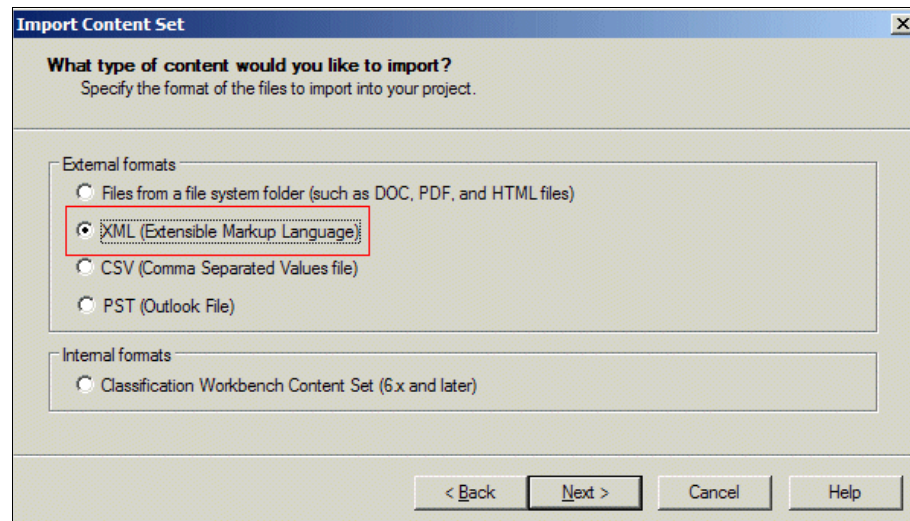


Figure 9-4 Importing XML files that were exported by Content Analytics

- Select the XML folder where you exported the data from Content Analytics. Because you have the `catalog.xml` file, Content Classification Workbench has all the necessary information regarding the fields. Click **Next**.
- In the next window, click **Finish**.

Figure 9-5 shows the full content set. The Category field is highlighted in pink.

ID	Body	Category	device...
1		Chemical	
2	1414082 1348858 21...	Chemical	No
3	1367584 1304417 30...	Chemical	No
4	1402208 1338048 16...	Chemical	Yes
5	1386570 1293853 M...	Chemical	No
6	1381548 1318058 16...	Chemical	No
7	1336973 1274682 17...	Chemical	No
8	1327332 1265255 M...	Chemical	No
9	1291183 1231294 30...	Chemical	No
10	939339 901413 MWS...	Chemical	No
11	931244 893377 1022...	Chemical	No
12	950737 912770 1610...	Chemical	No
13	950735 912768 1610...	Chemical	No
14	920473 882667 2431...	Chemical	Yes
15	914199 876433 1610...	Chemical	Yes
16	871643 828462 1022...	Chemical	No
17	906974 869223 1527...	Chemical	No

Figure 9-5 Content Classification Workbench showing the imported sample data

- Open the Create, Analyze and Learn Wizard to train the new knowledge base.

9. In the “Specify options for the selected process” window (Figure 9-6), select **Create using all, analyze using all** and click **Next**.

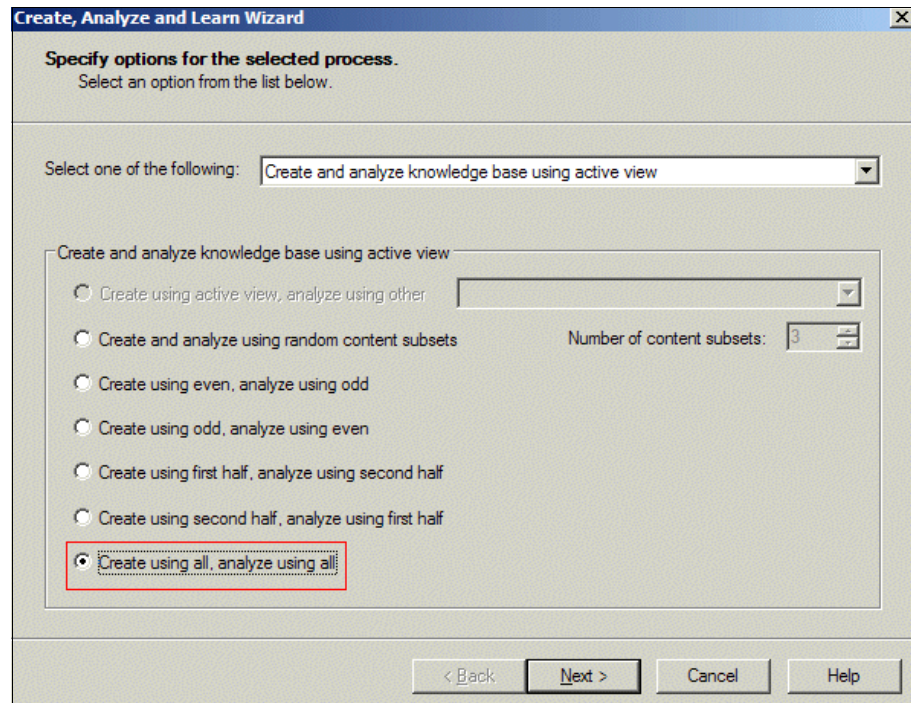


Figure 9-6 Training the Content Classification knowledge base

10. For the next windows, use the default settings until you reach the Status window.

11. In the Status window (Figure 9-7), click **Close**. You have finished creating and training your new knowledge base.

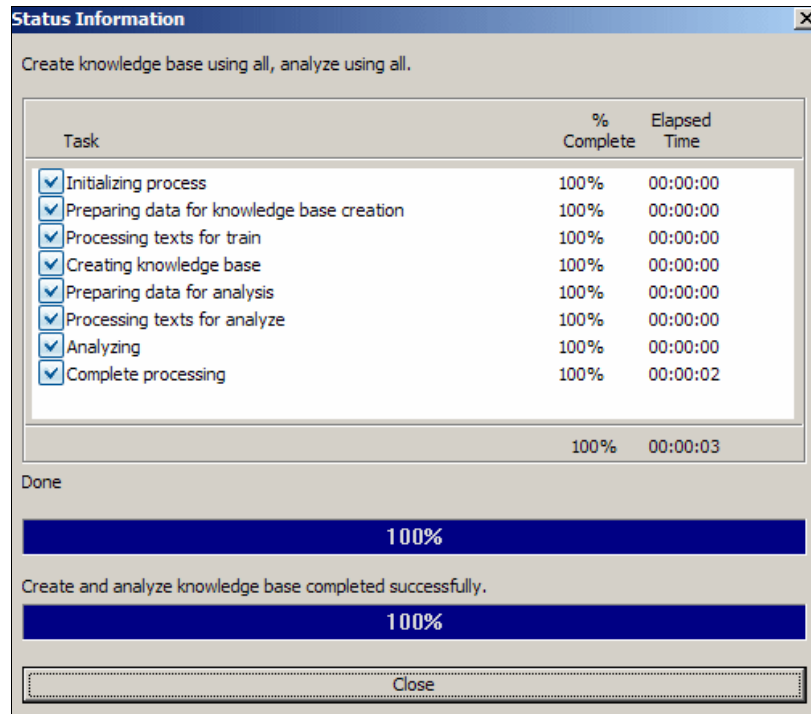


Figure 9-7 Status window showing training is complete

### 9.3.3 Creating a decision plan

To create a Content Classification decision plan, follow these steps:

1. From Classification Workbench, select **Project** → **New** → **Decision Plan**.
2. In the New Project window (Figure 9-8), type the decision plan name. For this example, enter Burn Hazard DP. Then, click **Next**.

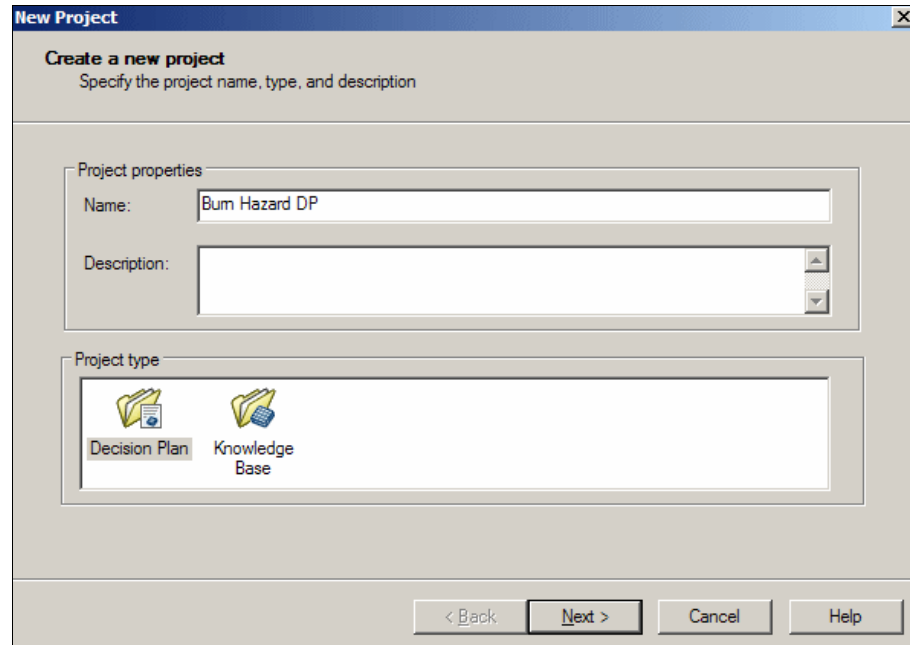


Figure 9-8 Creating a Content Classification decision plan project

3. Import the data that was exported by Content Analytics. See 10.6.3, “Exporting search result documents to the file system for IBM Content Classification” on page 376, for details about exporting.
4. When you foresee that your data will be in several languages, go to **Project Explorer**, and select **Project Options**.

5. In the window that opens (Figure 9-9), select all the languages that are relevant for your project. Then, click **OK**.

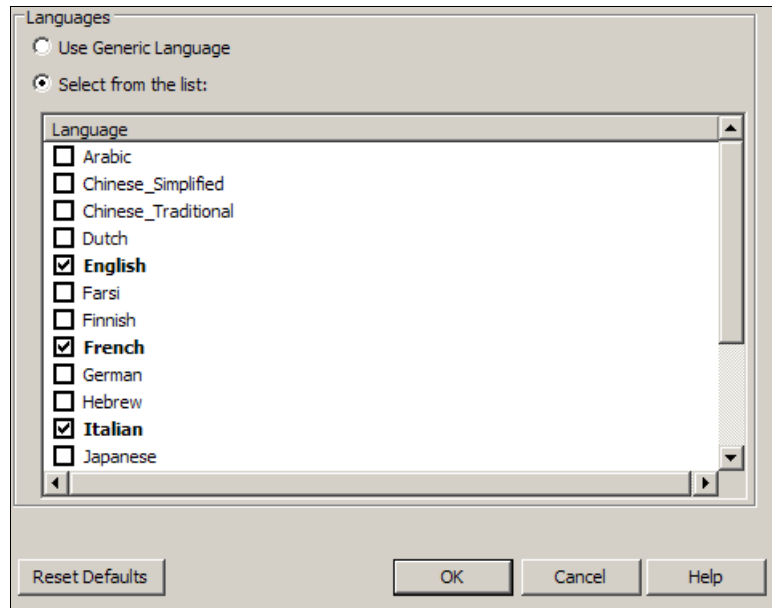


Figure 9-9 Setting the languages for a Workbench project

6. To use the previously created knowledge base, go to **Referenced Projects** and click **Add Project**. Add your knowledge base to the project. For this example, add the **Burn Hazard knowledge base** to the project (Figure 9-10).

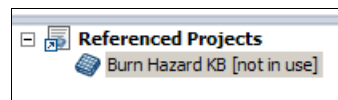


Figure 9-10 Decision plan under Referenced Projects referring to a knowledge base

7. Define a new rule that sets a new field called *burntype* if the category suggested by the Burn Hazard KB knowledge base has a confidence score above 95:
  - a. Right-click **New Group**, and select **New Rule**.
  - b. Select **Trigger**. Click **Condition** and choose **Trigger always**.
  - c. Select **Actions**. Click **Add** and select **General actions**. Select **Set the value for a content field** and click **Next**.
  - d. In the “Set the value of this content field” window (Figure 9-11), complete these steps:
    - i. Type the content field name. For this example, enter *burntype*.
    - ii. Choose your knowledge base. For this example, select **Burn Hazard KB** as the knowledge base.
    - iii. Choose **All categories whose score is above this percentage**. Type the confidence score. For this example, enter 95.

Figure 9-11 shows the new rule.

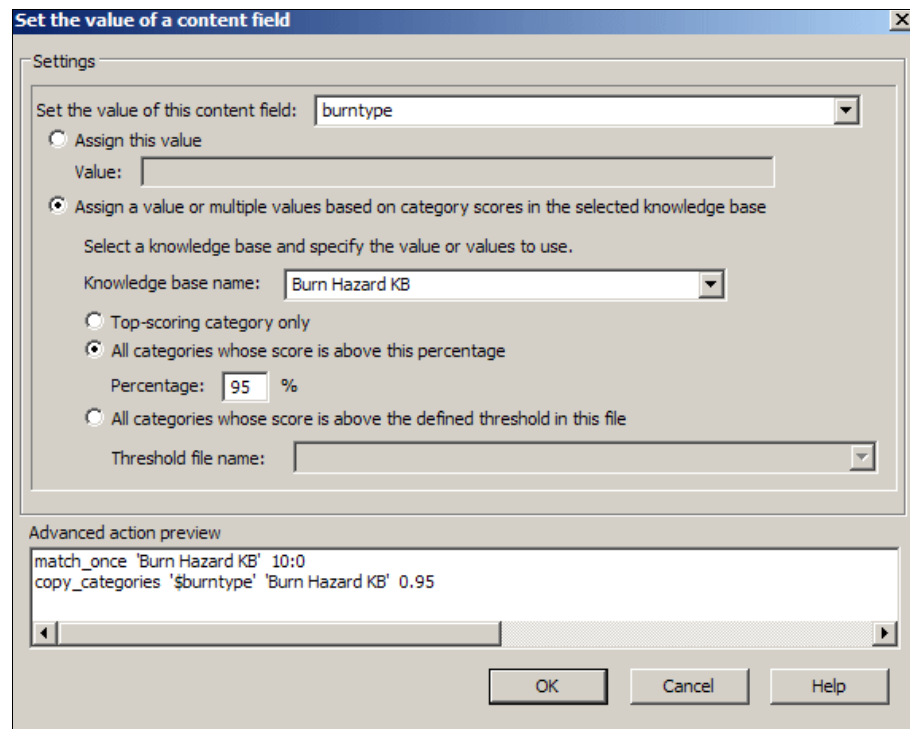


Figure 9-11 Choosing *burntype* with a confidence score greater than 95%



8. Define a new rule to set the burntype field to the value Unknown when the confidence score is not above 95. Because you set the field in cases when the score is above 95, you know that, in all other cases, you have an empty burntype field. To define the rule, follow these steps:
  - a. Create a rule. Type a new rule name. For this example, enter Set\_BurnType\_Unknown.
  - b. Select **Trigger**. Click **Condition** and choose **Advanced**.
  - c. Select **number = number**.
  - d. Click the first number link. Choose **Size content field**. Click **Content field** and choose **burntype**.
  - e. Click the second link and type 0.
  - f. Set the trigger size to  $(\$burntype) = 0$ .
  - g. Select **Actions**. Click **Add** and choose **General actions**. Select **Set the value for a content field** and click **Next**.
  - h. For the content field name, type burntype.
  - i. Type the value Unknown.

Figure 9-12 shows the new defined rule.

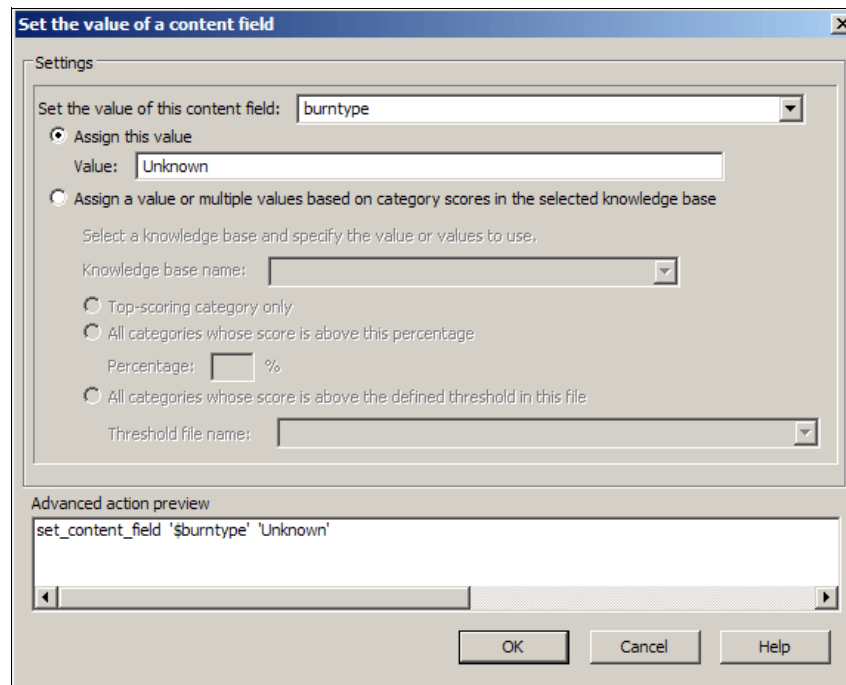


Figure 9-12 Setting the burntype field to Unknown for low confidence scores

### 9.3.4 Deploying the knowledge base and decision plan

To use the knowledge base and decision plan that you created, you must deploy them to the Content Classification server:

1. Select **Project** → **Export**.
2. Select **Decision Plan**, and click **Next**.
3. Select **IBM Content Classification Server**, and click **Next**.
4. In the Connection window (Figure 9-13), specify the Content Classification Server machine name and port. Click **Next**.

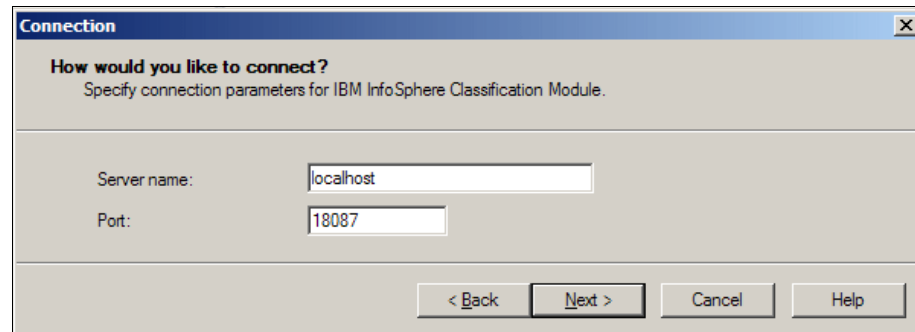


Figure 9-13 Workbench connecting to the Content Classification server

5. In the Publish Decision Plan window (Figure 9-14), select **Create a new decision plan on the Classification Module server**, and for “Specify a name for the new decision plan”, enter Burn Hazard DP. Then, click **Next**.

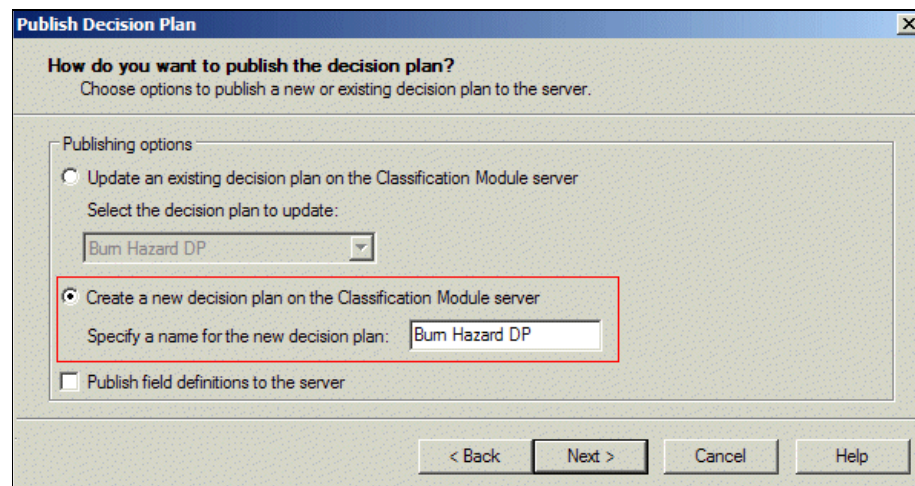


Figure 9-14 Publishing the decision plan to the Content Classification server

6. Select the knowledge base and click **Next** to publish the associated knowledge base (Figure 9-15).

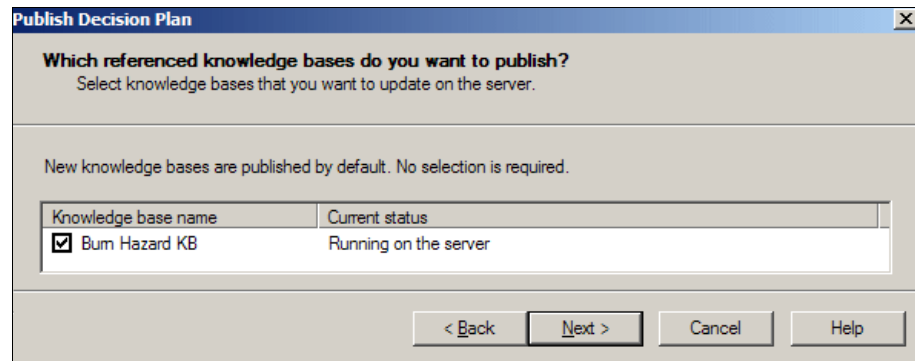


Figure 9-15 Publishing the knowledge base to the Content Classification server

7. In the last window, click **Finish**.

**Tip:** Launch the Content Classification Management Console to check the status of the decision plan and knowledge base. Check that the list of languages for each knowledge base and decision plan corresponds to the languages that you foresee in your data (see Figure 9-16 on page 324).

**Continuous operation:** The decision plans and knowledge bases in Content Classification that you need to use with Content Analytics must run continuously. If you made changes to the system, such as adding a Content Classification catalog field, you must restart them accordingly.

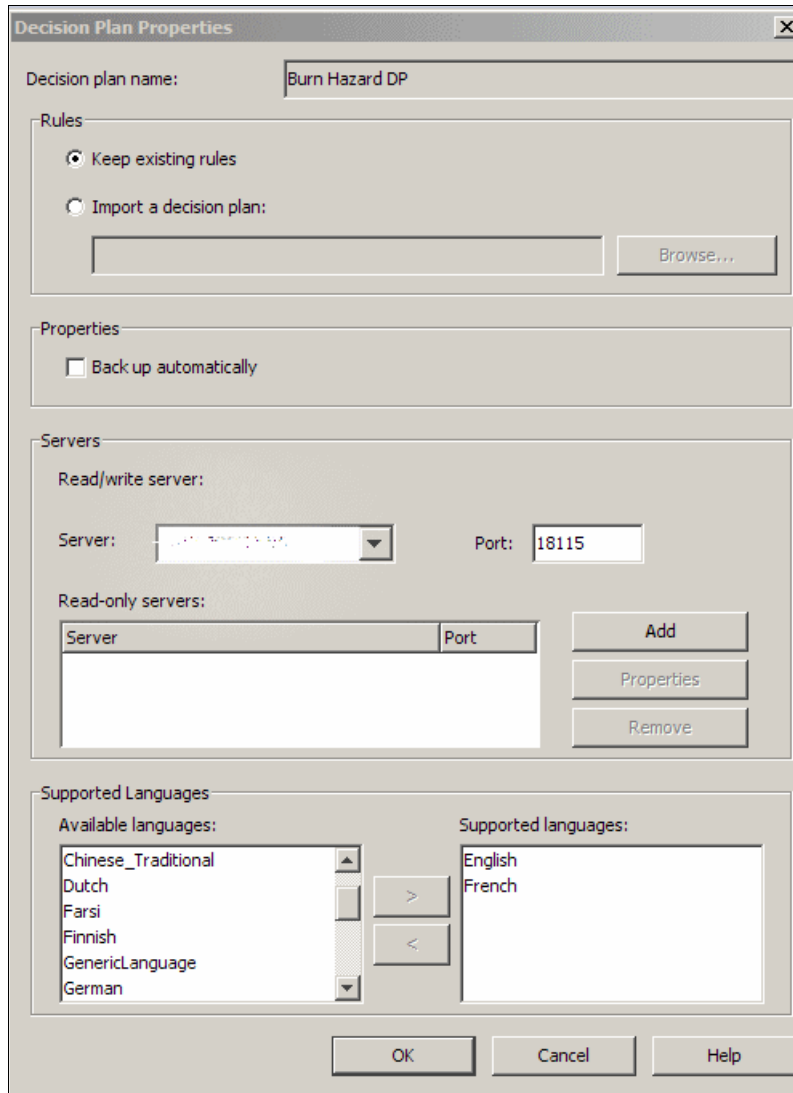


Figure 9-16 Content Classification Management Console: Set languages

For more information about building a Content Classification knowledge base, see the technote *Building a knowledge base for IBM Content Classification V8.8* at the following address:

<http://www.ibm.com/support/docview.wss?uid=swg27020839>

## Taxonomy Proposer

Content Classification provides clustering services with its *Taxonomy Proposer* application. You can use the data exported from Content Analytics and cluster it to discover groups of documents that share similar concepts and to define a new set of categories and a new taxonomy. You can then use the new categories to train a Content Classification knowledge base and use it to generate new, interesting facets in Content Analytics.

## Content Analytics clustering

The Content Classification clustering technology is integrated into the content analytics miner of Content Analytics. You can cluster a subset of the documents in the collection and deploy a categorization task to annotate the entire collection. See 9.6, “Document clustering” on page 330. You can also export a training set based on clustering suggestions (as facets).

## Quick Start Tool

Content Classification 8.8 introduces a new tool called the *Quick Start Tool*. The Quick Start Tool takes the user through small and clearly defined steps, suggesting actions to improve the quality of the training set of documents. Each one of these proposed actions is named a task. It generates a Decision Plan and a Knowledge Base that can be exported to Classification Workbench.

### 9.3.5 Configuring the Content Classification annotator

To configure the Content Classification, start by configuring the decision plan:

1. Start the Content Classification server.
  - a. Go to the Content Classification Server panel (Figure 9-17).
  - b. Type the URL of the Content Classification server and click **Next**.

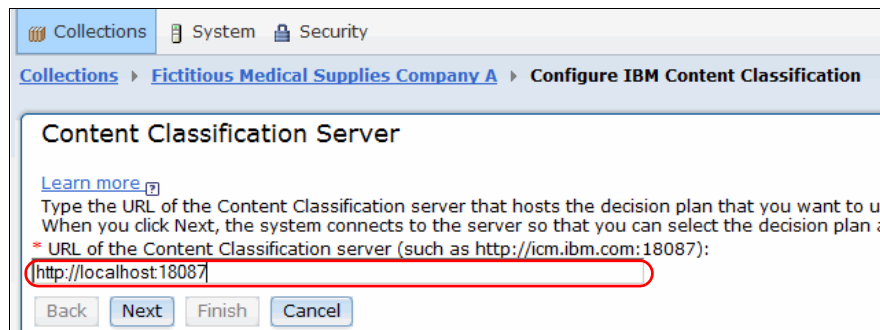


Figure 9-17 Content Classification URL setting

2. Associate a decision plan with your collection:
  - a. In the Decision Plan panel (Figure 9-18), from the decision plan drop-down list, select the decision plan that you want to associate with your collection. For this example, select **Burn Hazard DP**.

Figure 9-18 Selecting the decision plan

- b. Map the Content Classification fields. In the Content Classification fields list (Figure 9-19), you see all the results fields that Content Classification can generate after classifying your data. In this case, select **burntype**, which is the only option available to select. In a case where you have more results fields, click **Add Field**, and choose all the fields that you want to use in Content Analytics (for facets association and others).

Figure 9-19 Mapping the Content Classification field

- c. Optional: Import category scores to enable category-based scoring or conceptual search.
3. Create a facet that is based on the fields generated by Content Classification:
  - a. From the Content Analytics administration console, go to the parse configuration for your collection (for our example, Fictitious Medical Supplies Company A), and click **Edit**.
  - b. Click **Configure facets**.

- c. Under the Hazards facet, add a facet:
  - i. In the Add a facet panel, enter Burns as the new facet name and BURNTYPE for the facet path.
  - ii. Click **Add** to complete the addition of the facet.
  - iii. Select the **Add counts to parent facet** check box.
  - iv. Clicking **Edit** next to the Fields mapping field to enable field mapping and choose **burntype**.
  - v. Click **OK**.

**Enabling field mapping:** The Content Classification fields are automatically enabled for field mapping.

**After you make your changes:** You must deploy the resources for the changes that you made to be reflected in the index.

After you successfully reindex the content, you can review the facets that are generated by the Content Classification annotator in the content analytics miner. For this example, go to the Facets view and choose **Burn** to see the documents. You will find documents related to “chemical burns” that do not necessarily include the word “burn.” This result occurs because Content Classification looks at the entirety of the document to classify the results properly instead of just relying on a particular keyword.

**Classification capability of Content Classification:** The Content Classification annotator uses the entire set of words in a document and can classify the documents related to a certain category based on understanding the full context.

## 9.4 Validation and maintenance of the Content Classification annotator

You use Classification Workbench to create and train a knowledge base and to create a decision plan. You also use Classification Workbench to tune and maintain your decision plan and knowledge base:

1. Import the data exported by Content Analytics. Then, use Content Classification Workbench to train and fine-tune the knowledge base:
  - Use only part of the imported data to train the knowledge base and use the remaining part to analyze and tune it.

- Use the Content Classification Workbench reports to assess the accuracy of the knowledge base.
- Use the Content Classification Workbench reports to check the decision plan rules. In the Workbench decision plan projects, you have reports on the rules behavior for the overall content set. You can also trace for a specific content item that the rules have triggered and view their results by using the “Run item through decision plan” functionality.

Go to the IBM InfoSphere Classification Module Information Center at the following address, and search on *decision plan analysis*:

<http://publib.boulder.ibm.com/infocenter/classify/v8r8/index.jsp>

**More information:** For more tips about tuning Content Classification (previously known as Classification Module), also see the IBM Redbooks publication, *IBM Classification Module: Make It Work for You*, SG24-7707.

2. Import the XML with the suggested scores results to view the XML export data from Content Analytics that contains the analysis results.

The Classification Workbench of Content Classification can generate accuracy reports from the data that contains the categories and the scores. For more information, search on the following topics in the IBM InfoSphere Classification Module Information Center:

- “Analyzing a knowledge base in production”
- “Analyzing a decision plan in production”
- “Sample XML output from saved analysis data”

<http://publib.boulder.ibm.com/infocenter/classify/v8r8/index.jsp>

### 9.4.1 Using the Content Classification sample programs

After deploying the decision plan and knowledge base to the Content Classification server, you can validate their behavior by invoking one of the sample applications that is installed with Content Classification. For example, you can use the `.\Samples\Java\JavaGUIDecide` application to connect to the decision plan that you created and published to the Content Classification server. You can introduce text or a document and observe the results.

**Tip:** Consult the “Content Classification Tutorial” that is installed with Content Classification. To access the tutorial, select **Start** → **All Programs** → **IBM Content Classification 8.8** → **Content Classification Tutorial**.



## 9.4.2 Content Classification annotator validation techniques

The Content Classification annotator supports the following validation techniques:

- ▶ Verify the connection to the Content Classification server.
- ▶ Use the Content Classification samples to test that your decision plan acts as designed.
- ▶ Make sure that you reindex after deploying the Content Classification annotator and configuring the facets mapping.

## 9.5 Preferred practices for Content Classification annotator usage

Text mining and gaining insight to your content is an iterative process. The following guidelines have been developed based on the field experiences of the authors of this book:

- ▶ Start your analysis with a small collection.
- ▶ Follow the normal procedure several times:
  - Crawl, parse, and index, and inspect the content using the content analytics miner.
  - Define new dictionaries and inspect the content using the content analytics miner.
  - Define new pattern rules and inspect the content using the content analytics miner.
  - When you discover the need for a more sophisticated analysis, engage Content Classification.
- ▶ Training and tuning with Content Classification and defining decision rules can be a small iterative cycle in itself. Use the Content Classification Workbench to tune a knowledge base or refine the decision plan rules.
- ▶ Use the Quick Start Tool or the Taxonomy Proposer to refine your categories or to discover new categories.
- ▶ Use the Content Analytics clustering to discover classes of documents inside the collection and deploy a categorization task to annotate the entire collection accordingly.

Content Classification can invoke external hooks. When you need to engage Content Classification for more sophisticated text analysis, you can also use (if needed) its extensibility points to customize the text processing.

## 9.6 Document clustering

The document clustering functionality in Content Analytics is empowered by the Content Classification clustering algorithms. With document clustering, you can obtain insight quickly into a large data set of unstructured data without setting up dictionaries or defining rules. It provides a potential categorization of your documents. You can use automatic cluster-based categorization to navigate through your content to gain insight into your content.

The categorization is based on a sample content set of 1000 to 10,000 documents. Document clustering does not require that you install and configure Content Classification. However, the cluster categorization results for document clustering might be less informative or detailed.

To run document clustering, you need to use a large subset of data for Content Analytics to determine meaningful categories. After Content Analytics determines the clusters, you can review, rename, remove, or add clusters. When you are satisfied with the defined clusters, you can apply them to the entire collection. Then, a facet that represents the document cluster results is generated.

The document clustering workflow consists of the following tasks, which are explained in the sections that follow:

1. Setting up document cluster
2. Creating a cluster proposal
3. Refining the cluster results
4. Deploying clusters to a category

### 9.6.1 Setting up document cluster

To use the document clustering functionality, you must specify that the collection supports document clusters for the categorization type when you create the collection. There are four categorization type options:

- ▶ None
- ▶ Rule based
- ▶ Document clusters
- ▶ Rule based and Document clusters

The Document clusters option enables document clustering for the collection. With the Rule-based and Document clusters, you can define rules and perform document clustering for the collection.

After the collection is created, and documents are crawled and indexed, you can perform document clustering. To configure document clustering, follow these steps:

1. From the administration console, click the **Collections** tab.
2. Click **Create Collection**.
3. In the Create a Collection pane, complete the following fields.
  - a. For the Collection name field, type `LargeSampleCollection`.
  - b. For the Collection type field, select **content analytics collection**.
  - c. For the Categorization type field, select either *Document clusters* or *Rule-based and document clusters* to enable document clustering. For this scenario, select **Document clusters**.

**Content analytics collection:** Document clustering can only be enabled for a content analytics collection because it is not available for enterprise search collections.

- d. Click **OK**.
4. In the Collections view, click the **Edit** icon (Figure 9-20) for the collection that you want to edit. We click the **Edit** icon that corresponds to the `LargeSampleCollection` collection.

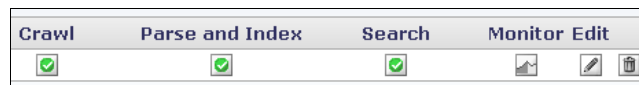


Figure 9-20 Collections view showing the editing and monitoring options

5. Click the **Crawler** tab.
6. Click **Create crawler** and create a crawler to a content repository that contains at least 1000 documents.

**Index for document clustering:** Document clustering requires 1000 to less than 10,000 documents in the index. You must wait until the index has completed before creating the cluster proposal.

**Content body and document clustering:** Documents that can be clustered must have content body. If data is crawled from a database or imported from CSV files, and there is no column that is set as body or that column is not made as analyzable, documents do not have content body and thus clustering would not work.

7. Click the **Parse and Index** tab.
8. Click **Actions** then **Edit** collection.
9. In the **Document clustering** field, select the option **Enable document clustering**, and click **OK** (Figure 9-21).

The screenshot shows the 'Edit Collection Settings' page for 'Sample Text Analytics Collection'. The page includes a breadcrumb trail: 'Collections > Sample Text Analytics Collection > Edit collection settings'. Below the title, there is a 'Learn more' link and a paragraph explaining that settings like date facet fields and time scales cannot be configured. The form contains several fields: 'Collection name' (Sample Text Analytics Collection), 'Description' (Collection created by SI-API client), 'Document importance (static ranking model)' (Do not apply any static ranking), 'Duplicate document detection' (Do not enable duplicate document detection), 'Rule-based categorization' (Enable rule-based categorization), 'Document clustering' (Enable document clustering, highlighted with a red box), and 'Optional facet index' (Do not enable the optional facet index).

Figure 9-21 Enabling document clustering

If you change the categorization type from the **None** or **Rule based** options to **Document clusters** or **Rule based and Document clusters**, rebuild the index. Click **Restart a full index build**.

However, because we already enabled document clustering when we created the collection, click **Cancel** to return to the Parse and Index edit pane.

## 9.6.2 Creating a cluster proposal

To create a cluster proposal, follow these steps:

1. In the Parse and Index tab (Figure 9-22), choose **Global processing** then **Document clusters**.

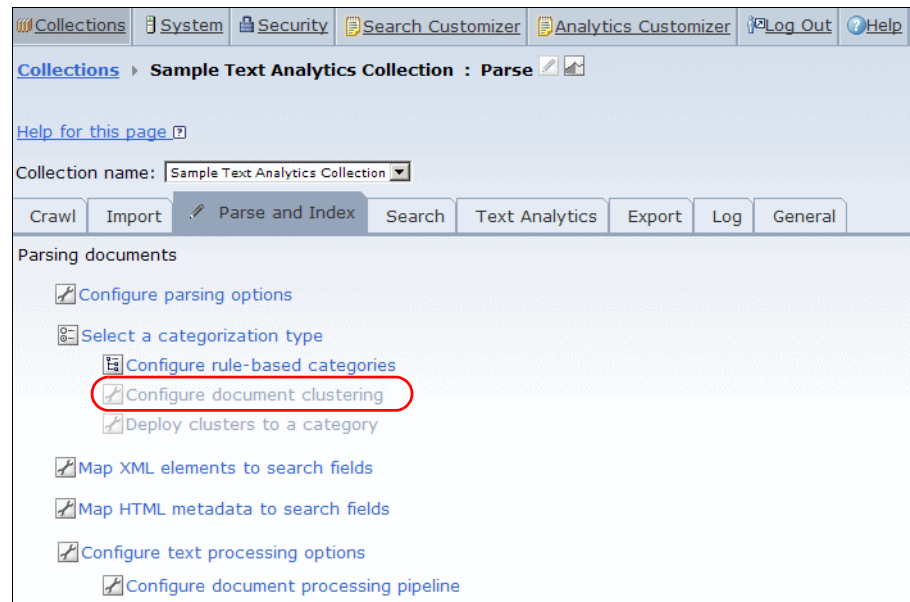


Figure 9-22 Configure Document Clustering

2. In the Document Clustering Tasks pane, set up the task with the following values and then click **Start** (Figure 9-23 on page 334):
  - In the Name field, type Sample test.
  - In the Number of clusters field, type 30.
  - In the Number of samples field, type 1000.
  - In the Clustering algorithm field, select **K-means** from the following four options not shown in the figure:
    - K-means
    - Latent Dirichlet Allocation (LDA)
    - Latent Dirichlet Allocation - detect clusters by samples, learn by all
    - Latent Dirichlet Allocation - detect clusters in partitions, learn by all

### Document Clustering Tasks

[Learn more](#)

Enter parameters for a new document clustering task and click **Start** to create a cluster proposal. A cluster proposal is created by sampling documents, extracting terms, and proposing candidate terms.

**Important:**

- Before starting a document clustering task, start the parse and index services.
- The **Number of clusters** value should be much smaller than the **Number of samples** value.
- Because it takes time and memory to sample documents, do not specify a large **Number of samples** value.
- For some algorithms, you can specify the number of documents in an extended set of samples. If **Number of clustered documents** is blank, all documents in the index are included when the task is run.

Last refreshed: Saturday, November 2, 2013 7:04:57 PM EDT

Name	Number of clusters	Number of samples	Number of clustered documents
Sample test	30	1000	

Figure 9-23 Setting up the parameters in the document clustering task pane

**Number of samples:** The number of samples value must be less than the number of documents in the index. It must be a number 1000 - 10,000.

3. Monitor the clustering process by clicking **Refresh** until the process is finished.

### 9.6.3 Refining the cluster results

After inspecting the initial document clustering results by reviewing the proposed clusters, you can further refine the results:

1. When the cluster task is complete, click the **Edit** icon associated with the Sample Test cluster task to edit a cluster proposal.

The Edit a Cluster Proposal pane (Figure 9-24) is now displayed. The cluster name usually refers to the most popular typical terms in the cluster. The names give you a general idea about the cluster content.

Edit a Cluster Proposal			
<a href="#">Learn more</a>			
The changes that you can make depend on whether you click <b>Start</b> or <b>Edit</b> on the Document Clustering task. After you refine the cluster proposal, start the <b>Global Processes &gt; Cluster deployment</b> task to categorize documents.			
Parameters used by the document clustering task			
Name	Number of clusters	Number of samples	Number of clustered documents
Sample test	30	1000	
Proposed clusters			
Cluster	Cluster name	Number of documents	
1	buyer,manuals,quadrant,incident,newsletters,buyers,tivoli,netcool,entitlements,m...	15	
2	survival,rup,hebner,rbde,delivery,rmc,cobol,sdp,betas,pl	48	
3	maximo,gas,gasoline,stocks,oil,industriescontent,petroleum,station,chemical,petr...	36	
4	professionalsskip,tabsoverview,newsibm,papersidc,futurefeatured,sandbox,sma...	28	
5	datasheet,ultrium,archival,restore,ultriumtm,opentm,backup,tape,lvd,adheres	50	
6	ts,asm,bin,mhz,iii,clusterware,rac,infotype,subtype,htmlfid	26	
7	jet,damage,nontraditional,frequency,enovia,prototypes,mutually,furthermore,roll,j...	31	

Figure 9-24 Reviewing and editing a clustering proposal result

- To rename a document cluster, type the new name in the cluster name field. The words listed under “Words in the cluster” contain the typical terms in the cluster and are ranked in popular order. The terms that are most popular are listed first.
- To remove any cluster group, clear the check box in the **Select** column that is associated with the cluster you want to remove.
- After you complete the changes for the cluster names and delete any unnecessary cluster groups, click **OK**. See Figure 9-25 on page 336.

[Help for this page](#)

**Important:** After adding clusters or editing words in a cluster, you must run the document clustering task to apply the changes that you made.

Parameters used by the document clustering task

Name	Number of clusters	Number of samples
SampleTest	30	1000

Proposed clusters

Select	Cluster	Cluster name	Number of documents	Words in th
<input checked="" type="checkbox"/>	1	Ldap	48	characters
<input checked="" type="checkbox"/>	2	healthcare,sciences,pharmaceutical,biology,naviga	49	healthcare
<input checked="" type="checkbox"/>	3	financingleasing,disposal,literatureanalyst,contentr	21	financingleasi
<input checked="" type="checkbox"/>	4	petroleum,oil,chemicals,gas,navigationchemicals,ir	35	petroleum
<input checked="" type="checkbox"/>	5	islands,pierre,leoneslovakia,helenast,fasoburundic	29	islands
<input checked="" type="checkbox"/>	6	Miscellaneous	34	framemaker
<input checked="" type="checkbox"/>	7	requirementslibrarycompetitive,notescompare,buy	19	6887

Figure 9-25 Renaming clusters

5. In the Document Clustering Tasks pane, select the **Sample Text** task. Then, click **Start** to start the document clustering task.  
Alternatively, you can click **Cancel**. In this case, no other clustering task is run. The changes you made become effective after you deploy the clustering proposal.
6. Click **Refresh** until the process displays are complete.
7. Click the **Edit** icon associated with the Sample Test cluster task to edit a cluster proposal.
8. In the “Edit a cluster proposal” pane, add a word to the document cluster:
  - a. In the Words in cluster column, select the new word in the text field, and click **Add a Word**.



- b. To delete defined words in the cluster, under Words in cluster, select the word from the list (see Figure 9-26). Click the **Delete** icon associated to that particular cluster. As a result, the cluster is refined to better represent your use case.
- c. Optional: Add a cluster by clicking **Add a Cluster**.
- d. After you make the wanted changes to the cluster proposal, click **OK**.

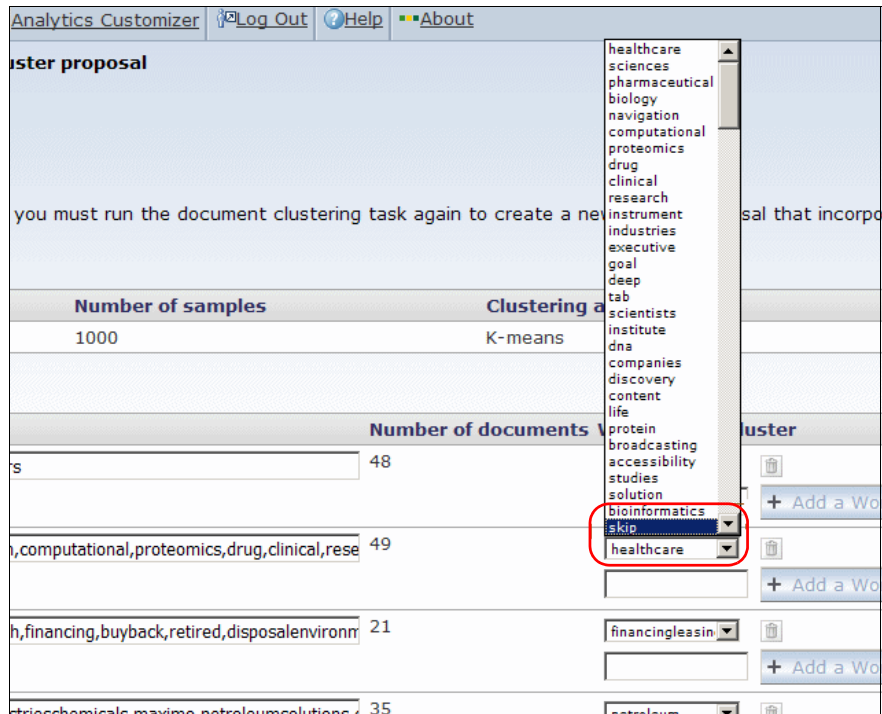


Figure 9-26 Refine the document clustering

9. Rerun the document clustering process by selecting the cluster name and clicking **Start** (Figure 9-27 on page 338). Click **Refresh** to monitor the clustering run. See Figure 9-27 on page 338.

[Collections](#) > [LargeSampleCollection : Parse](#) > **Configure document clustering**

**Document Clustering Tasks**

[Help for this page](#)

Enter parameters for a new document clustering task or select an existing task to refine a cluster proposal. A cluster proposal defines how documents in the index are to be sampled to extract candidate terms.

**Important:**

- Before starting a document clustering task, monitor the collection and start the **Parse and Index** task.
- The **Number of clusters** value should be much smaller than the **Number of samples** value.
- Because it takes time and memory to sample documents, do not specify a large **Number of samples**.
- If you modify a document clustering task, click **Start** to run the task and refresh the cluster proposal.

Last refreshed: Saturday, November 20, 2010 7:08:56 AM EST [Refresh](#)

Select	Name	Number of clusters	Number of samples	Clustering algorithm
<input type="radio"/>	<input type="text"/>	<input type="text" value="20"/>	<input type="text" value="5000"/>	Latent Dirichlet Allocation
<input type="radio"/>	Sample Test	30	1000	K-means

[Start](#) [Return](#)

Figure 9-27 Monitoring the document clustering run

The process of refining the document cluster might take more than one iteration. Continue to refine the clusters until you are satisfied with the result.

## 9.6.4 Deploying clusters to a category

Document categorization that is based on document clustering involves the following tasks:

- ▶ Configuring the system to detect clusters by sampling a subset of documents and extracting words. See 9.6.1, “Setting up document cluster” on page 330, and 9.6.3, “Refining the cluster results” on page 334.

**Renaming cluster names:** Before you annotate the entire collection, decide on the most appropriate cluster names. Rename them by double-clicking their names and typing a new value.

- ▶ Deploying a document categorization task to add metadata to documents based on the cluster analysis. In this process, the internal Content Classification knowledge base is created. The knowledge base is used to classify all documents in the index into rule-based categories.

To deploy a document categorization task (annotate a full collection), follow these steps:

1. Click the **Parse and Index** tab.
2. If you are not in edit mode, click the **Edit** icon.
3. Click **Global Processes** then **Cluster deployment**.
4. In the Deploy clusters to a category pane (Figure 9-28), enter the following information:
  - a. In the Category name field, type MyDocCluster.
  - b. Select the cluster set that you want deployed. For our scenario, we select **Sample Test**.
  - c. Select a categorization type. For our scenario, we select **Categorize to a top relevant cluster above the threshold value**.
  - d. Click **OK**. See Figure 9-28.

**Deploy Clusters to a Category**

Help for this page [?](#)

Select the cluster proposal that you want to use to apply and specify your preferences for ca

Input category

Category name

Cluster proposals

Select	Name	Number of clusters	Number
<input checked="" type="radio"/>	Sample Test	30	1000

Categorization

Categorize to a top relevant cluster:

Categorize to clusters above the threshold value:

Categorize to a top relevant cluster above the threshold value:

Threshold value:

Figure 9-28 Deploying the Categorization task

5. Restart the document categorizer:
  - a. After this process is finalized, click the **Parse and Index** tab.
  - b. Click the **Monitor** icon.
  - c. Click **Details**.

- d. In the “Document categorizer for clustering status summary” section (Figure 9-29), click **Start**. Wait until the document categorizer for clustering process is complete. See Figure 9-29.

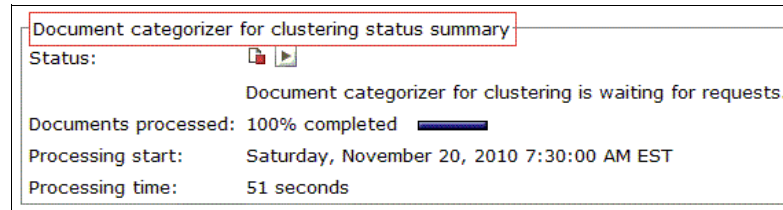


Figure 9-29 Document clustering summary

6. Deploy the resources to update a category label by clicking **Start** in the Resource deployment status section. Wait until the process is complete.

**Rebuilding the index:** It is not necessary to rebuild the full index. However, you can choose to start an index rebuild for another reason by clicking **Restart a full index build**.

## 9.6.5 Working with the cluster results

You can now work with the cluster in the enterprise search application and content analytics miner. The deployed cluster analysis can help narrow down your content to help you gain insight by working with the categorized documents:

1. Open the content analytics miner.
2. Click the **Facets** tab.
3. Click the **Document Cluster** facet. Document Cluster is the default name for the cluster facet.

4. Click the MyDocCluster value that you want to view further. For our scenario, we select **LDAP**. See Figure 9-30.

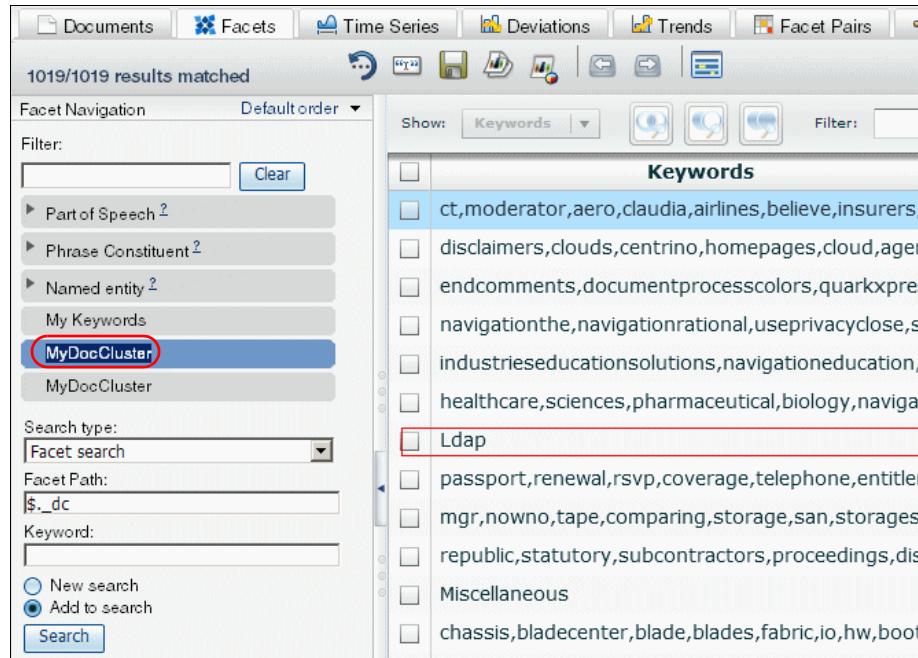


Figure 9-30 Cluster facets on Content Miner application

5. Click **Add to search with Boolean AND** to add the LDAP facet to the search query. Now the documents in the result set are limited to those documents within the LDAP cluster group.
6. Click the **Documents** tab to view these documents further.

In addition, if a conceptual search is enabled for the collection, you can search documents that conceptually match the query terms. See 9.2.3, “Using a conceptual search for advanced content discovery” on page 312.

## 9.6.6 Creating and deploying the clustering resource

When you decide that the document cluster generated the expected results, you can use them in Content Classification with the annotator. You can build the Content Classification resource and enable the Content Classification annotator so that future documents are annotated with it.

The document clustering results are stored and presented to the user as facets. The default name of the clustering facet is Document Cluster. When you deploy

the categorization task, you have the option of choosing a new name for this dedicated facet. In our scenario, we named the document cluster facet MyDocCluster. You can use the information for this facet to build the Content Classification resources:

1. Export documents under the MyDocCluster facet by using the search export that is configured to export documents as an XML file for Content Classification. This exported data can be used to train the knowledge base. Figure 9-31 shows the search export configured to export as XML files for Content Classification.

**Options to Export Searched Documents**

[Learn more](#)

**Searched document export options** Users can export documents after searching a collection. You can co

Do not allow documents to be exported

Options for searched document export

Export documents as XML files

Export documents as XML files for IBM Content Classification

Output file path for searched document metadata:

Export documents into a relational database

Export documents as CSV files

Export documents by using a custom plug-in

OK Cancel

Figure 9-31 Exporting XML files for Content Classification

For further information about exporting to Content Classification, see 10.6.3, “Exporting search result documents to the file system for IBM Content Classification” on page 376.

2. Import of the exported XML files into Content Classification Workbench.
3. Create and tune a knowledge base.
4. Build a decision plan to use the document clustering results that are based on information in the knowledge base and any rules that you create.
5. Export the decision plan to the Content Classification.
6. Enable the Content Classification annotator in Content Analytics.

In Content Analytics, map the fields. See 9.3.5, “Configuring the Content Classification annotator” on page 325.

## 9.6.7 Preferred practices

This section provides guidelines for working with clustering.

### When to use document clustering

You might want to use document clustering when you have a lot of unstructured content and little knowledge about it. Without knowing your content, it might be difficult to create dictionaries or obtain valuable insight with the content analytics miner. Often you must obtain more insight into the content before being able to use the sophisticated text analysis tools.

Document clustering provides insight to a large set of unstructured content without having to configure Content Classification. Content Analytics offers many tools for text analysis, and clustering is one of them. You are encouraged to use any of the following tools to gain comprehensive insight of your content:

- ▶ Leverage the Content Classification annotator built on the top of the clustering-based knowledge base, after tuning it by using Content Classification Workbench.
- ▶ Build dictionaries using Content Analytics tools.
- ▶ Build pattern rules in Content Analytics.
- ▶ Generate terms of interest.
- ▶ Refine the search results by clustering facets

### Number of documents to use in clustering

The granularity of document clustering can influence the nature and quality of the insight. To take advantage of the fullest potential of document clustering insight, follow these steps:

1. Run two or three cluster proposals with different granularity.
2. Inspect the cluster names and refine the results as needed.
3. Inspect the documents represented by a cluster in the content analytics miner.
4. Decide the best cluster groups to use.
5. Deploy the categorization based on clustering to annotate the entire collection.
6. Use the Content Analytics tools to continue.

Document clustering offers insight about unstructured content that is further used with other text analytics tools. The suggested cluster categories and names offer knowledge to further build dictionaries or patterns rules.







## Importing CSV files, exporting data, and performing deep inspection

IBM Watson Content Analytics (Content Analytics) supports importing comma-separated value (CSV) files into a collection so that you can quickly add content to the collection without setting up a crawler or accessing a content repository. It also provides a set of export capabilities to help you take advantage of the discoveries made by Content Analytics in your textual data with other tools.

This chapter provides information about CSV file import and the export features with several scenarios of how and why you might use export. It also explains deep inspection, which is a form of export that extends the content analysis capabilities of the content analytics miner to all of your data.

This chapter includes the following sections:

- ▶ Importing CSV files
- ▶ Overview of exporting documents and data
- ▶ Location and format of the exported data
- ▶ Common configuration of the export feature
- ▶ Monitoring export requests
- ▶ Enabling export and sample configurations

- ▶ Deep inspection
- ▶ Creating and deploying a custom plug-in

## 10.1 Importing CSV files

With the import functionality, you can add records in a CSV file to a content analytics collection so that the information is searchable when users work with the content analytics miner. The import functionality is easy to use. With it, you can quickly add content to the collection without setting up a crawler or accessing a content repository. Sometimes, hardcopy documents need to be addressed by Content Analytics. IBM Datacap Taskmaster Capture provides means to scan and OCR key fields in the documents. It can export the CSV files, which can be imported into Content Analytics for rapid insights.

The CSV files need to conform to the RFC 4180 standard:

- ▶ The files must contain one or more records where each record is on a separate line and delimited by a line break (carriage return (CR), line feed (LF), or carriage return-line feed (CRLF)).
- ▶ The fields within a record are delimited by a comma, white space, tab, or semicolon.
- ▶ The fields that contain escaped line breaks, quotation marks, or field delimiters must be enclosed in quotation mark characters.
- ▶ The file cannot contain more than 128 columns or be greater than 512 KB per record. Files greater than 128 MB are not supported for CSV file import when you select to upload a local file to the server.

In this scenario, you import a CSV file that contains two records. You import this file into the collection created, which is the Sample Text Analytics Collection in this case:

1. Open the administration console and go to the **Crawl and Import Settings** for the Sample Text Analytics Collection, and click the **Import** bar (plus sign) (Figure 10-1 on page 347).

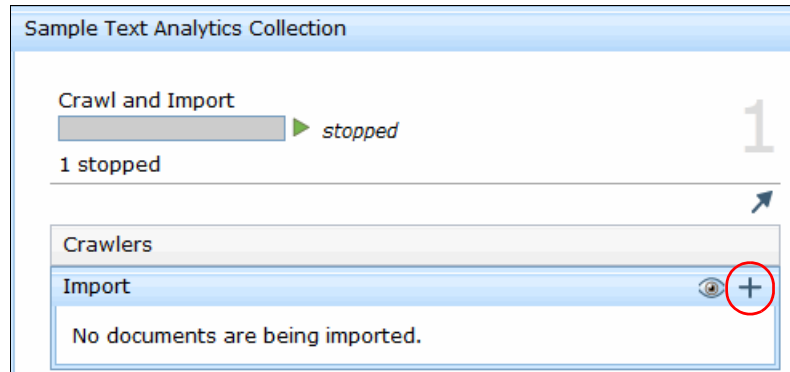


Figure 10-1 Selecting the "Import into the collection" option

2. Create a file that contains the following text and name it `FoodData.csv`:  
Category,Product,Subcategory,Title,Body  
Price,cookie,Price (general),Cookie - Price (general),I was charged a higher price than what was advertised for my box of cookies.  
Price,cookie,Price (general),Cookie - General Price,I was charged twice for the same box of cookies.

3. In the Specify CSV Files to Import panel (Figure 10-2), for the File name field under Local path, click **Browse** and select the **FoodData.csv** document that you created in step 2.

**Importing more than one CSV file:** To import more than one CSV file, select a directory path in the File name field. This action results in adding all CSV files (\*.csv) in the directory path to the index of the collection.

Specify the CSV Files to Import

[Learn more](#)

Specify the CSV files to import and whether you want to reuse previously saved settings for this import task.  
**Note:** A file with more than 128 columns or a record with a size greater than 512 KB cannot be imported.

CSV file path

Local path (The file size should be smaller than 128 MB.)

File name:

C:\CSV Import\FoodData.csv

Index server path (Specify a file path or directory path. If you specify a directory, all the files under the directory will be imported.)

File name:

CSV import settings

Use the system default values for the configuration

Reuse the values of previously saved settings for the new configuration

Reuse the values from a property file for the new configuration

Figure 10-2 Specify the CSV Files to Import panel

4. Select the **Use the system default values for the configuration** radio button and click **Next**.

**Reimporting a CSV file:** To reimport a CSV file and use the settings that you previously saved during import, select the **Reuse the values of previously saved settings for the new configuration** radio button. To reimport a CSV file and use the settings that you previously downloaded to a property file, select the **Reuse the values from a property file for the new configuration** radio button.

5. In the Specify Options to Read CSV Files panel (Figure 10-3), enter the field values that are defined in Table 10-1. Then, click **Next**.

Table 10-1 Specify Options to Read CSV File field values

Field	Value
Encoding character set	windows-1202
Delimiter	Comma
Starting line number	1
Read the starting line as a header	(Select the check box)

### Specify Options to Read CSV Files

[Learn more](#)

Specify options for importing CSV files. If you change an option, the preview is updated automatically.

The preview shows how the content of the file will be imported, beginning with the first line that is to be read.

Encoding character set:

Delimiter  
 Comma     Space     Tab     Semicolon

\*Starting line number:

Read the starting line as a header, which enables the column names to be mapped to index fields

Preview:

1:	Category	Product	Subcategory	Title	Body
2:	Price	cookie	Price (general)	Cookie - Price (gen	I was charged a hi
3:	Price	cookie	Price (general)	Cookie - General P	I was charged twic

Figure 10-3 Specify Options to Read CSV Files panel

6. In the Specify the Columns to Import panel (Figure 10-4 on page 350), complete these steps:
  - a. Select the Search Field Name for the particular column based on the values listed in Table 10-2.

Table 10-2 Field values for the Specify the Columns to Import panel

Column	Search Field Name
Category	doc_category
Product	doc_product

Column	Search Field Name
Subcategory	doc_subcategory
Title	title
Body	body

- b. In the Import Space ID field, which must have a unique value, for this scenario, type 1/FoodData.csv.
- c. Click **Next**.

**Specify the Columns to Import**

[Learn more](#)

Map the columns that you want to make searchable to a predefined or new index field

Columns to import:

	Import Column	Index field name		Returnable	Faceted search
<input checked="" type="checkbox"/>	<b>All</b>				
<input checked="" type="checkbox"/>	Category	doc_category	doc_category	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Product	doc_product	doc_product	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Subcategory	doc_subcategory	doc_subcategory	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Title	title	title	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Body	body	body	<input type="checkbox"/>	<input type="checkbox"/>

Import options

\*Import Space ID:  
This ID is used to distinguish imported documents from other documents in the index

1/FoodData.csv

Figure 10-4 Values used for the Specify the Columns to Import panel

**Format of date and number values in data:** If your data contains date values, set the Date Format and Time Zone fields (under the Advanced Options section) to the format of your date values. If your data contains number values, set the Number Format field (under the Advanced Options section) to the format of your number values. These formats are applied to all columns in the CSV file, which are mapped to date or number fields in the CSV file (that is, fields with “Parametric search” enabled).

7. In the “Specify whether or not to save the current settings” panel (Figure 10-5), complete these steps:
  - a. Choose one of the following options for the current settings:
    - Save the configurations that you have defined as a property file by clicking **Download the current settings as a property file**.
    - Save the configurations to the server to be used later by selecting **Save the current settings to the server**.

In this scenario, do not save or download the import settings.

**Saving the imported settings:** You can only save the import settings in the “Specify whether or not to save the current settings” panel. If you plan to import the same CSV file more than once or to reuse the import settings for other files, save the settings at this time.

- b. Click **Finish**.

Specify whether or not to save the current settings

[Learn more](#)

Save the current settings

Download the current settings as a property file

Save the current settings to the server

Back Next Finish Cancel

Figure 10-5 “Specify whether or not to save the current settings” panel

8. Now that you have set up the CSV file import, validate that the records were imported by reviewing the CSV document import history. To open the import history, click the **Monitor** icon.
9. Click **View the CSV file import history** (Figure 10-6).

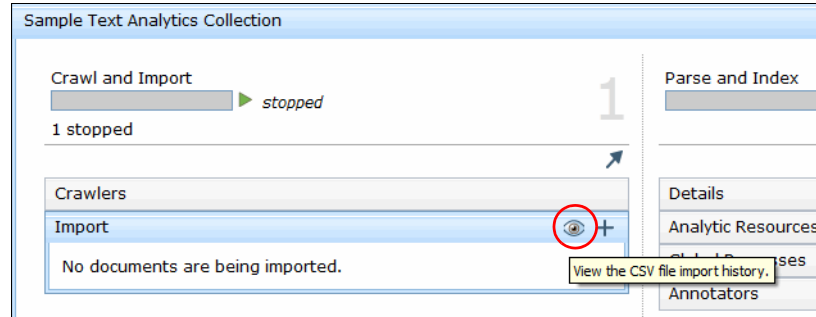


Figure 10-6 CSV import summary panel

The row that contains the Import Space ID that matches the value used in step b on page 350 shows the results of the import. The value in the Number of records column indicates the number of records that were imported. In this scenario, two records were added to the index within the collection, as shown in Figure 10-7.

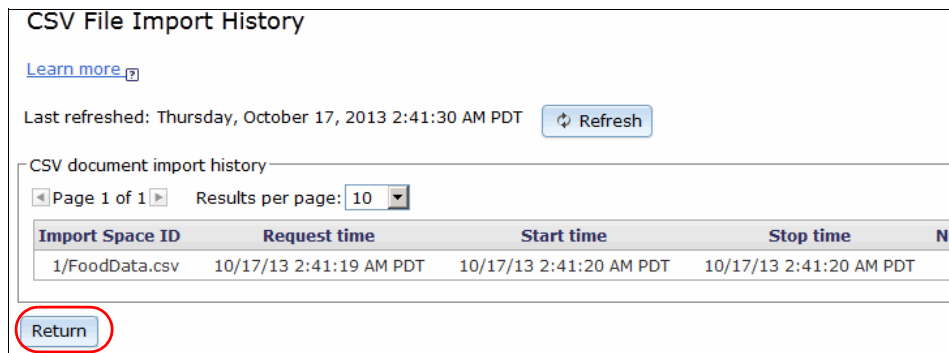


Figure 10-7 History of the 1/FoodData.csv file import

**Import CSV function:** The import CSV file function does not keep track of modifications to the file. If you modify the CSV file, you must import the CSV file again, which reimports all of the records in the CSV file into the collection.

**Deleting the import history task:** If you delete the import history task, the documents that were added to the index based on that particular import task are deleted from the index.



## 10.2 Overview of exporting documents and data

Content Analytics provides the powerful capability of analyzing structured and unstructured data (textual data) so that users can obtain actionable insight. In addition, with Content Analytics, users can also export data so that they can use the results of their analysis by using other applications such as data warehouse, business intelligence, or classification applications. Many IBM products can import and use this exported information, including IBM Content Collector, IBM Content Classification, and IBM Cognos Business Intelligence (BI).

### **Rationale for exporting data from Content Analytics**

You might want to export data from Content Analytics for several reasons. For example, when two companies merge, both companies have data stored in different sources at different locations. In this situation, you can use Content Analytics to crawl data from various sources and then export the data to the file system. This exported data can be used later by Content Collector for archiving to a wanted location.

Content Analytics can export data to a relational database in a star schema model. By using business intelligence or data warehouse applications, which access data from a relational database, analysts can gain a unique advantage of analyzing both structured and unstructured content together.

### **Exporting points (stages) in Content Analytics**

You can export data from Content Analytics during the following stages as illustrated in Figure 10-8 on page 354:

- ▶ Export point 1: After documents are crawled.
- ▶ Export point 2: After documents are analyzed (processed and indexed).
- ▶ Export point 3: After a search is performed. You can then export the search result.

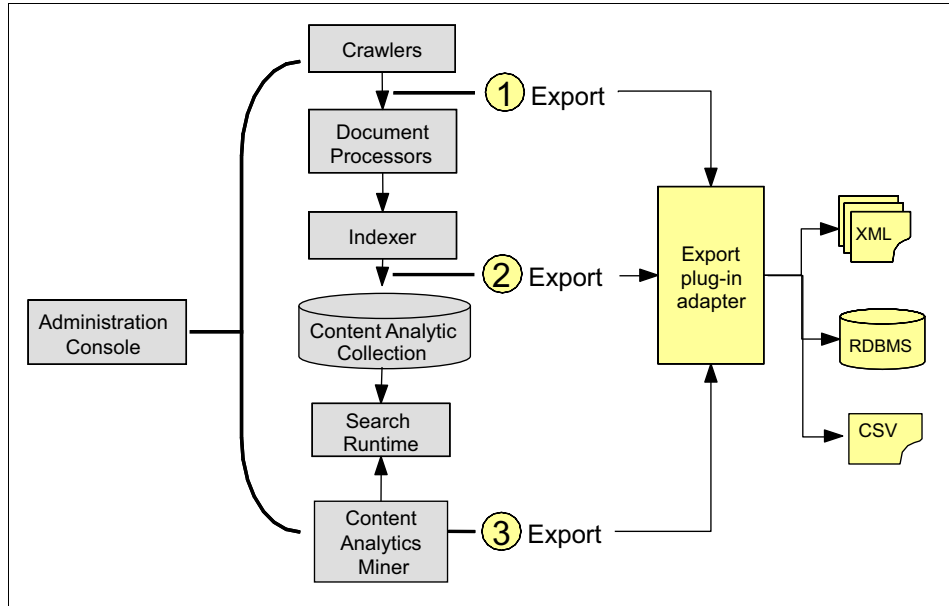


Figure 10-8 Export points in Content Analytics

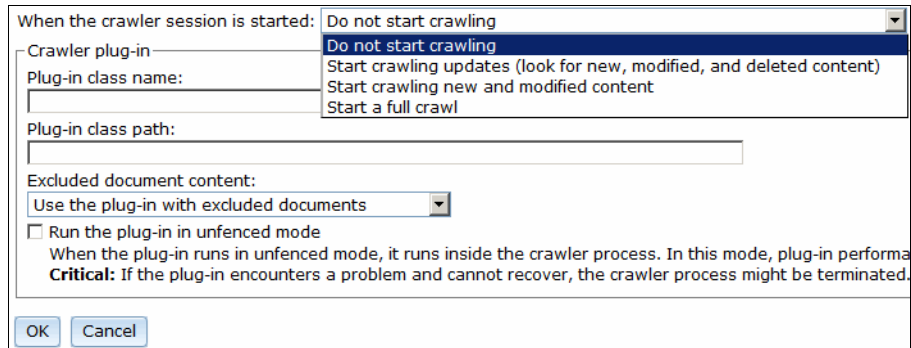
At each of these stages, you have the option to export data to either a file system or a relational database. Additionally, you can configure the deep inspection feature on a large content analytics collection to export the analysis results.

**Import feature:** Content Analytics does not provide an import feature to add the previously exported data from Content Analytics into the same or different collection.

## 10.2.1 Crawled documents

Crawled documents are documents that have been retrieved from their data source but are not yet parsed or analyzed by Content Analytics. You can export them from Content Analytics. When you export crawled documents, you can choose to export metadata, binary content, or both. Additionally, you can export data so that it can be imported later by Content Collector.

The configuration of a crawler determines which crawled documents are to be exported. As shown in Figure 10-9, a crawler session can be configured to crawl data in different ways when it is started.



The screenshot shows a configuration dialog box for a crawler session. It contains the following elements:

- A dropdown menu labeled "When the crawler session is started:" with the selected option "Do not start crawling".
- A section labeled "Crawler plug-in" containing:
  - A dropdown menu with the selected option "Do not start crawling".
  - A list of three options: "Start crawling updates (look for new, modified, and deleted content)", "Start crawling new and modified content", and "Start a full crawl".
- A text input field labeled "Plug-in class name:".
- A text input field labeled "Plug-in class path:".
- A section labeled "Excluded document content:" containing:
  - A dropdown menu with the selected option "Use the plug-in with excluded documents".
- A checkbox labeled "Run the plug-in in unfenced mode".
- A warning message: "When the plug-in runs in unfenced mode, it runs inside the crawler process. In this mode, plug-in performance is degraded. **Critical:** If the plug-in encounters a problem and cannot recover, the crawler process might be terminated."
- "OK" and "Cancel" buttons at the bottom.

Figure 10-9 Configuration options for the crawler session

When you select **Start a full crawl** for the crawler, all the documents are recrawled, and thus all of the documents are exported.

When you select **Start crawling all updates** for the crawler, the crawler only retrieves new, modified, and deleted documents since the last crawl. In this case, Content Analytics exports new and modified data and information about the deleted data. For deleted documents, the value of the `/Document@Type` element in exported data is set to `DELETED`. When a document is deleted from the source, Content Analytics *does not* delete already exported data to synchronize with the source. In this case, you must be aware of the deleted documents and handle synchronization systematically or manually if required.

When you select **Start crawling new and modified data** for the crawler, the crawler only crawls new and modified data. The crawler does not check to see if any previously crawled documents were deleted. Therefore, Content Analytics exports only new and modified data. It does *not* export information about deleted documents because it does not have knowledge about deleted data.

## 10.2.2 Analyzed documents

Analyzed documents contain metadata, textual data, and any annotations that are added during the Unstructured Information Management Architecture (UIMA) document processing pipeline. You can export analyzed documents from Content Analytics. When you export analyzed documents, you have the option of exporting metadata and facets, Common Analysis Structure (CAS), extracted text, or both. You can use this option to perform advanced analysis on structured documents. For example, Content Analytics can export facets to a relational database in a star schema model so that it can later be used by IBM Cognos Business Intelligence to build meaningful reports.

Additionally, exported data at this stage contains a consistent and unified view of metadata from the different crawled data sources. Companies can often employ different systems to manage their information depending on the business need. For example, you might have active design or enhancement discussions stored in an IBM Lotus Notes database and have documents related to released products stored in a content management system such as IBM FileNet Content Manager.

Often fields that are semantically the same are named differently between data sources. In this scenario, the discussion documents in a Lotus Notes database might have a native field named “From” to identify the initiator of the discussion. In this same scenario, documents in the content management system might have the native field named “Author” to identify the creator of the document. Both of these fields represent the owner of the document. Assuming these native fields are mapped to a single search field named “Owner in Content Analytics,” they become normalized. When you export the analyzed documents, the values of the From and Author native fields are exported as values for the Owner field. This method relieves you of exported data from knowing the details of the individual data sources and allows normalized analysis of data from multiple disparate sources.

## 10.2.3 Search result documents

Search result documents are the set of documents returned for the query you execute by using the content analytics miner. You can export these documents from Content Analytics. You can also schedule an export of the search result documents on a recurring basis. By using the scheduling capability, you can periodically export the data of interest without having to manually mine the collection every time. For example, if new data is being added to the collection on a regular interval, you can use the *incremental export* feature where only documents added and updated after the last export are exported.

## 10.2.4 Exported data manifest

Depending on which stage you are at when you export your data, you can export different types of data, including metadata, binary content, CAS, and extracted text.

### Metadata

Metadata entails the properties associated with documents. File size, date created, and author can be thought of as conventional metadata for files such as PDF files. Content Analytics also considers native fields either intrinsic to the document or managed externally from the document by the data source as metadata.

When you export analyzed documents, metadata includes facets populated by annotators in the UIMA processing pipeline. Additionally, if native fields are mapped to search fields, the exported metadata contains the names of search fields instead of the names of native fields. When exporting analyzed or search result documents, metadata includes only search fields that are configured as returnable.

### Binary content

Binary content is the unstructured part of a document. Binary content is maintained in its native format such as a Microsoft Word document or PDF document.

### Common Analysis Structure

When you export analyzed documents to a file system, you can also export the output of the UIMA document processing pipeline known as the CAS format. The CAS data is formatted as XML data and contains extracted text, annotations, facets, and other results of analysis. Exported CAS data to file system can also be used to validate the output of custom annotators. The CAS format conforms to the UIMA standard and is subject to change with future releases of UIMA.

### Extracted text

Text extracted from the binary content is referred to as *extract text*. Extracted text is also referred to as *parsed content*. You can export some or all of these types of data at each stage of export.

Table 10-3 lists the configuration options at each stage. Even though you can configure the same options at different stages, the output at each stage can be different. For example, you can configure exporting metadata after both the crawled document and analyzed document stages. However, exported metadata for documents after the crawled document stage only contains conventional metadata. Exported metadata for analyzed documents also contains analyzed facets in addition to the conventional metadata.

*Table 10-3 Export configuration options*

<b>Exported data</b>	<b>Crawled documents</b>	<b>Analyzed documents</b>	<b>Search result documents</b>
To file system or relational database	Yes	Yes	Yes
Metadata	Yes	Yes	Yes
Binary content	Yes		Yes
CAS		Yes	
Extracted text		Yes	Yes
For Content Collector integration	Yes	Yes	Yes
For Content Classification integration			Yes
Schedulable			Yes
Customize export	Yes	Yes	Yes

## 10.3 Location and format of the exported data

The type and format of data that is exported and the location to which the data is exported depends on the stage that you selected for export. This section explains where the documents are exported and the format in which they are exported.

### 10.3.1 Location of the exported data

When exporting data to a file system, you are required to provide a path to an existing directory for metadata and content. When the export service runs, a new folder is created under that path.

The name of the folder is based on the date and time that the export occurs. The name of the created folder is in the *yyyymmddhhmm* format. For example, if the export service starts on 17 October 2013 at 5:25 p.m., the folder name 201310171725 is created. After that, subdirectories are created for every 1000 documents exported that start with 0 and are incremented sequentially. The number 1000, indicating the number of documents in a folder, is a nonconfigurable parameter.

Example 10-1 shows the directory structure if you configure the export path of crawled documents as C:\Export\CrawledDataExport.

*Example 10-1 Output for crawled documents into one directory*

---

```
C:\Export\CrawledDataExport\201310171725\0
    0000.xml
    0000.dat
    0001.xml
    0001.dat
    ....
C:\Export\CrawledDataExport\201310171725\1
    0000.xml
    0000.dat
    0001.xml
    0001.dat
    ....
```

---

Additionally, when you configure the export options, you can provide different paths for the metadata and content. Example 10-2 shows a directory structure where metadata is exported into the C:\Export\CrawledDataExport\metadata directory, while content is exported into the C:\Export\CrawledDataExport\content directory.

*Example 10-2 Output for crawled documents into separate directories*

---

```
C:\Export\CrawledDataExport\metadata\201310171725\0
    0000.xml
    0001.xml
    ....
C:\Export\CrawledDataExport\metadata\201310171725\1
    0000.xml
    0001.xml
    ....
C:\Export\CrawledDataExport\content\201310171725\0
    0000.dat
    0001.dat
```

```
.....  
C:\Export\CrawledDataExport\content\201003301725\1  
0000.dat  
0001.dat  
.....
```

---

### 10.3.2 Metadata format

Metadata is generally the structured part of a document and is often referred to using native fields. Metadata is exported in the XML format to a file system or mapped to relational database columns. The name of the native field is preserved in the XML file when exporting crawled documents. When Content Analytics analyzes documents, the native fields are mapped to the search fields, and the name of the search fields are used instead.

For example, a native field named *timestamp* can be mapped to the search field *date* in Content Analytics. When you export metadata for the analyzed and search result documents, the name of the search fields is used, not the name of the native fields. In this example, the exported metadata file contains the field *date*, not the time stamp. Furthermore, when exporting analyzed or search result documents, the metadata file only contains search fields that are configured to be returnable in the Content Analytics administration console.

The metadata format remains the same for Content Collector integration. However, the attributes and field names are converted to the XML element to allow XML metadata mapping in Content Collector.

#### Metadata file name

When a user configures Content Analytics to export metadata to a file system, the metadata is exported as XML files. The name of the first exported XML file begins with `0000.xml` and increments sequentially. For example, if the source file is `sample.doc`, the metadata file name is `0000.xml`. The XML file contains the original file name as a portion of the metadata.

#### Metadata in a relational database

Metadata is added to the database as columns of the table.

### 10.3.3 Binary content format

Binary content contains text or the unstructured part of the document and is exported in the original format such as Word or PDF.



When exporting crawled documents to a file system, by default, Content Analytics exports content as .dat files *and* preserves the binary format of the file. For example, you export a .doc source file in the .dat format and rename the .dat format to the .doc format. In this case, the resulting .doc file is the same as the original .doc file. However, when you export to a .csv file, the binary content of the document is not exported.

When you configure the export of crawled content for Content Collector integration, Content Analytics preserves the extension from the source (if available) and exports content with the original extension. For example, if the source document has the document sample.doc file, the exported content also has the sample.doc file name. For situations where Content Collector integration is enabled, but the source document does not provide a file extension, the document is exported as a .dat file.

### **Binary content file name**

When you configure Content Analytics to export content to a file system, the name of the XML file begins with 0000.dat and increments sequentially. For example, if the source file is the sample.doc file, the metadata file name is 0000.dat or 0000.doc if Content Collector integration is enabled.

### **Binary content in the relational database**

Binary content is stored in the relational database as a binary large object (BLOB).

## **10.3.4 Common Analysis Structure format**

CAS is a data structure for representing information that is gathered during the analysis of document such as annotations, tokens, and facets. When you export the CAS format to a file system, the data is exported in the XMI format as .xmi files. The CAS format conforms to UIMA standards and is subject to change with future releases of UIMA.

When Content Collector integration is enabled, the format of the CAS file does not change.

### **Common Analysis Structure file name**

The name of the XMI file begins with 0000.xmi and increments sequentially until the name reaches 9999.xmi. When more than 10,000 documents are exported, a new folder is created with file names beginning with 0000.xmi.

### **Common Analysis Structure in relational database**

CAS export to relational database is not supported.

### 10.3.5 Extracted text format

Extracted text is the unstructured part of the document. The text contains extracted characters from binary content.

When you export extracted text to a file system, the data is exported in the same XML file that contains the metadata. It is included in the `<content></content>` element. When Content Collector integration is enabled, the format of the extracted text does not change.

#### Extracted text file name

Extracted text is part of the metadata. See “Metadata file name” on page 360 for more information.

#### Extracted text in a relational database

When you export analyzed documents to a relational database, the extracted text is exported as a character large object (CLOB) into a column.

## 10.4 Common configuration of the export feature

The three export stages share common configuration features when exporting crawled, analyzed, or searched documents. This section provides details about these common configuration options.

### 10.4.1 Document URI pattern

Content Analytics supports limiting which documents are exported based on the composition of the Uniform Resource Identifier (URI) of the document. You can enter a list of regular expression patterns as a value of this configuration property, and Content Analytics only exports those documents whose URI matches one of these patterns. For example, if you provide the following example as a value for the Document URI patterns field, only PDF and Word documents are exported:

```
.*.pdf  
.*.doc
```

### 10.4.2 Exporting XML attributes and preserving file extensions

If you plan to import into Content Collector the crawled, analyzed, or search result documents that are exported from Content Analytics, you must select the **Use field**

**name or facet path as XML element** check box. By selecting this option, you can export the XML attributes and field names as elements so that the elements can be used during metadata mapping in the Content Collector configuration.

This option also preserves the extension of native file names when the binary content is exported. Although this configuration is required to enable Content Collector integration, you can use it for other reasons. For example, you can enable this option to preserve the extension of the binary content in the exported data.

### 10.4.3 Adding exported documents to the index

When you configure export options for crawled or analyzed documents, you can select the **Do not add the exported documents to the index** check box. If the purpose of using Content Analytics is to collect data or collect parsed information to be redirected to a different destination, you can save hardware resources by not building an index of the data in Content Analytics. When you select this option, the content analytics miner for the collection is not available.

### 10.4.4 Exporting information about deleted documents

By default, information about deleted documents is also exported when you export crawled or analyzed documents. To disable this option, when you configure export options for crawled or analyzed documents, select the **Do not export information about deleted documents** check box.

### 10.4.5 Scheduling

For search-result documents, export can be scheduled to start later. When scheduling an export request, you can specify when the export operation is to start and how often it must run. For example, you can schedule an export request to run at off-peak hours without impacting the production time search capability. Additionally, you can disable a specific export request or even delete it if you no longer need to export the data.

#### **Incremental export**

Incremental export means exporting only new documents that are added after the last export. When you have a dynamic collection where data is being added on a regular basis, you can export documents on an incremental basis to keep the data accurate and up-to-date.

## Custom schedule

You can configure the export request to run on a general schedule specified for all the requests, or you can customize a schedule for each discrete request to run at a specific time.

**Configuring and enabling scheduling:** After the export request is submitted through the content analytics miner, the administrator *must* configure and enable the schedule for the export request for Content Analytics to export documents at the scheduled time:

1. Enable the export feature for searched documents by using the administration console.
2. Using the content analytics miner, perform a search and export the search result. For the export option, select the schedulable option.
3. In the administration console, configure and enable the schedule for the export request in step 2.

## 10.5 Monitoring export requests

Content Analytics provides a monitoring capability to help you see the status of export requests for crawled, analyzed, and searched data. Figure 10-10 shows an example export monitor that provides a summary of all the crawled and analyzed documents.

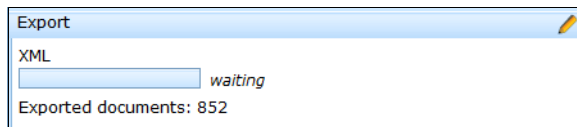


Figure 10-10 Export Monitor view

The Export Monitor view has a link to the summary of export requests for the searched document. Figure 10-11 on page 365 shows an example of the search result summary window.

Request ID	Export ID	Stop time	Number of documents	User Name
17	Ice Cream Compliant	10/21/13 10:52 PM	191 of 191	

Figure 10-11 Searched Document Export History window

## 10.6 Enabling export and sample configurations

Before you begin the configuration for exporting data, consider these questions:

- ▶ What data must be exported: crawled, analyzed, or search result?
- ▶ Where will you export the data: the file system or a relational database?
- ▶ What metadata or fields do you want to include in the exported result?
- ▶ Which application is using the exported data: Content Collector, Content Classification, or another product?
- ▶ Is this a one-time export or a recurring export task?

As a first step, you can create a collection with a sample set of data, configure the wanted export option, and then validate that the output to ensure that it is what you expect. For example, you can ensure that the metadata fields you want are present in the output.

This section takes you through the following export scenarios:

- ▶ Exporting crawled documents to a file system for IBM Content Collector
  - This scenario shows how to configure Content Analytics to export data to the file system for Content Collector usage.
- ▶ Exporting analyzed documents to a relational database
  - This scenario shows how to create a proper database mapping file and configuring Content Analytics to export data to a relational database.
- ▶ Exporting search results to CSV files
  - This scenario shows how to configure the export of the search results as CSV files.

These scenarios are configured by using the Sample Text Analytics Collection created with the Installation verification Option of Content Analytics Setup.

## 10.6.1 Exporting crawled documents to a file system for IBM Content Collector

Content Analytics supports the crawling of over 25 types of enterprise data sources. Many applications can use this robust crawling feature of Content Analytics by exporting crawled results for usage. Content Collector is one such IBM product that can use Content Analytics to crawl the enterprise and archive the crawled content.

This section explains how to configure Content Analytics to export metadata and content to the file system. Content Collector uses the format of the exported data. The first step is to configure and run the export, and the second step is to validate that the data is correctly exported.

### Configuring and running an export

To configure an export, follow these steps:

1. From the administration console, click **Collections** in the toolbar.
2. In the Collections view, locate the Crawl and Import option and click the **Arrow** icon (Figure 10-12).

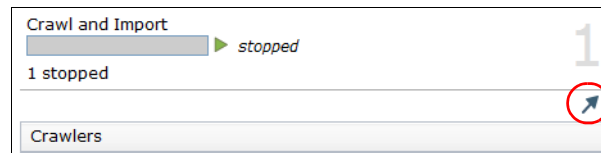


Figure 10-12 Collections view showing the editing and monitoring options

3. In the “Options for exporting crawled documents” section of the “Crawled document export options” window (Figure 10-13 on page 367), complete these steps:
4. Select **Export documents as XML files**.
  - a. Select the **Enable crawled document metadata export** and **Enable crawled document content export** check boxes.
  - b. For the Output file path fields, enter the paths of the *existing* directory. You can provide the same path for both content and metadata or separate paths for each field.

- c. Select the **Use field name or facet path as XML element** check box, which is *crucial* for exported documents to be consumable by Content Collector.
- d. Click **OK**.

**Crawled document export options:** You can export documents when they are crawled, and you can export metadata and content.

Options for exporting crawled documents

If you change these options, you must stop and restart the parse and index services.

Do not export documents  
 Export documents as XML files

Enable crawled document metadata export  
 Output file path:

Enable crawled document content export  
 Output file path:

Document URI pattern to export:

Do not pass any documents to document processing and do not add any documents to the index  
 Do not export information about deleted documents  
 Use field name or facet path as XML element  
 Export documents into a relational database  
 Export documents as CSV files  
 Export documents by using a custom plug-in

Figure 10-13 Crawled document export configuration

5. Click **Collections** in the toolbar.
6. Restart the document processor service. To restart the service:
  - a. Click **Details in Parse and Index**.
  - b. Click **Stop**.
  - c. When the service is stopped, click **Start**.

If you already built a collection, you must rebuild it after making this configuration change. To rebuild the index, click **Details**, and then click **Restart a full index build** (circled in Figure 10-14 on page 368).

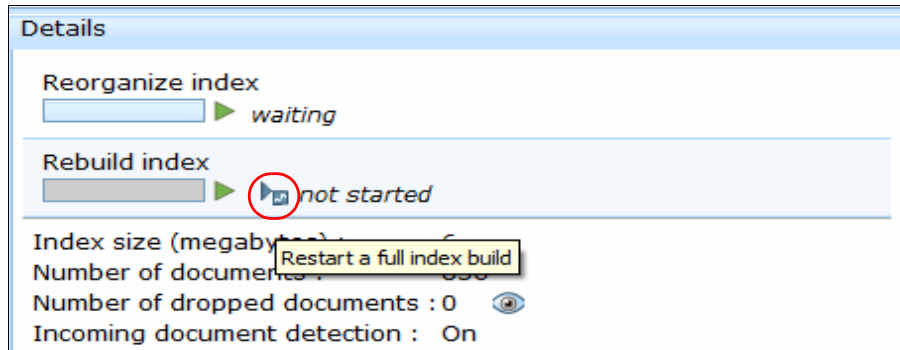


Figure 10-14 Building a full index

7. Wait for the rebuild index to complete (Figure 10-15).

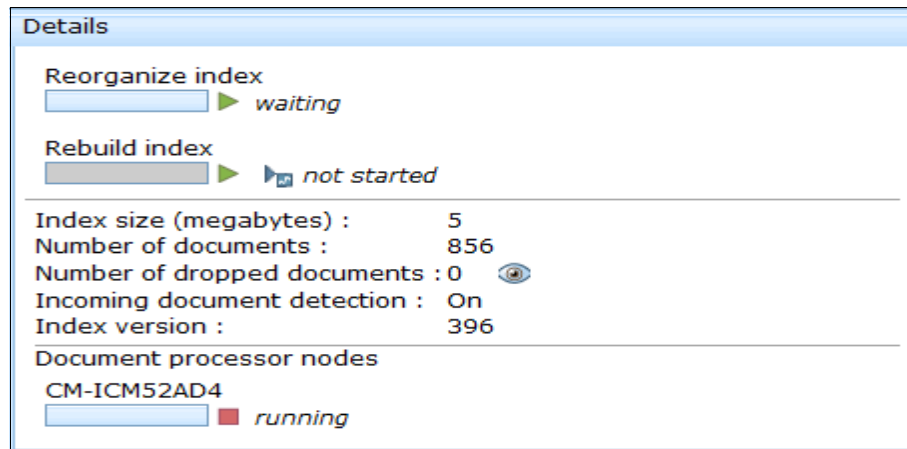


Figure 10-15 Completed rebuilding index process



## Validating the result crawled documents export

Using export monitoring in the administration console, validate that the number of documents exported (Figure 10-16) is the same as the number of documents crawled.

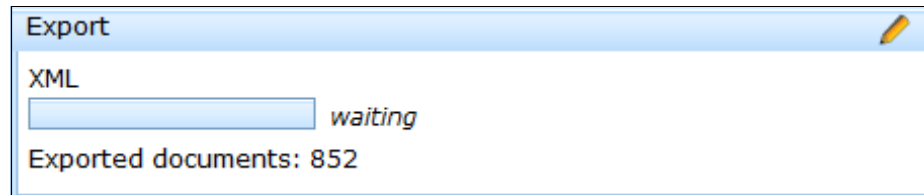


Figure 10-16 Crawled document export summary

Verify that the directory that you specified as the output path to verify that data is being exported. Example 10-3 shows the format of the output file for metadata.

Example 10-3 Metadata file exported to a file system for Content Collector integration

```
<?xml version="1.0" encoding="UTF-8"?>
<Document Id="file:///C:/IBM/es/samples/firststep/data/xml/xml-data/00000851.xml"
Type="NORMAL">
  <Content>
    <Path>c:\Export\CrawledData\AsXMLForICC\content\20100401025416\0\0000.xml</Path>
    <Directory>c:\Export\CrawledData\AsXMLForICC\content\20100401025416\0</Directory>
    <Name>0000.xml</Name>
    <Truncated>>false</Truncated>
  </Content>
  <Metadata>
    <Fields>

<Directory><![CDATA[C:\IBM\es\samples\firststep\data\xml\xml-data]]></Directory>
  <FileName>00000851.xml</FileName>
  <Extension>.xml</Extension>
  <ModifiedDate>1250055768000</ModifiedDate>
  <FileSize>453</FileSize>
  <Title>00000851.xml</Title>
  </Fields>
  <Facets></Facets>
  </Metadata>
</Document>
```

Example 10-4 shows a sample of the metadata file, which is also exported to the file system, *without* Content Collector integration enabled. Many of the attributes in Example 10-4 are transformed into XML elements, as shown in Example 10-3 on page 369. For example, look for the following line in Example 10-4:

```
<Field Name="__$FileName$__">00000851.xml</Field>
```

This line is transformed to the following lines in Example 10-3 on page 369 when Content Collector integration is enabled:

```
<Fields>
  <FileName>00000851.xml</FileName>
```

This switch allows seamless metadata mapping in Content Collector.

*Example 10-4 Metadata file exported to a file system when Content Collector is disabled*

---

```
<?xml version="1.0" encoding="UTF-8"?>
<Document Id="file:///C:/IBM/es/samples/firststep/data/xml/xml-data/00000851.xml"
Type="NORMAL">
  <Content Truncated="false"
Path="c:\Export\CrawledData\AsXML\content\20100401030225\0
C;0\0000.dat" Encoded="false"></Content>
  <Metadata>
    <Fields>
      <Field
Name="__$Directory$__">c:\IBM\es\samples\firststep\data\
;xml\xml-data</Field>
      <Field Name="__$FileName$__">00000851.xml</Field>
      <Field Name="__$Extension$__">.xml</Field>
      <Field Name="__$ModifiedDate$__">1250055768000</Field>
      <Field Name="__$FileSize$__">453</Field>
      <Field Name="__$Title$__">00000851.xml</Field>
    </Fields>
    <Facets></Facets>
  </Metadata>
</Document>
```

---

## 10.6.2 Exporting analyzed documents to a relational database

Before you begin exporting data to a relational database, you can export a small subset of data as XML to a file system for the following reasons:

- ▶ To see the type of data that is being exported as fields, facets, and metadata. For any native fields that you map to the search fields, the names of the search fields are displayed in the exported data.
- ▶ To determine the data that needs to be inserted into the database. For example, when you export analyzed documents, no binary data is exported. In this case, you must modify the default configuration. Similarly, you can modify the configuration to remove any fields or facets that you do not want to needlessly insert into the database.

Content Analytics uses the field configuration that you set for the export to automatically insert exported data into a relational database. Content Analytics inserts the data into star-schema tables.

Exporting analyzed documents to a relational database requires the following tasks, which are explained in the following sections:

1. Configuring the database export information
2. Running the export
3. Validating the results of the analyzed documents export

### Configuring the database export information

The first step in exporting data to a relational database is to configure the database export information. You must define how the metadata of a document is mapped to the columns of tables in the database. The database information includes connection and configuration information about the database.

To set up the database export configuration information for exporting analyzed documents, follow these steps:

1. From the administration console, click **Collections** in the toolbar.
2. In the Collections view, locate the collection that you want to edit and click **Edit**.
3. Select the **Export** Arrow (Figure 10-17 on page 372), and go to **Configure options to export crawled or analyzed documents**.

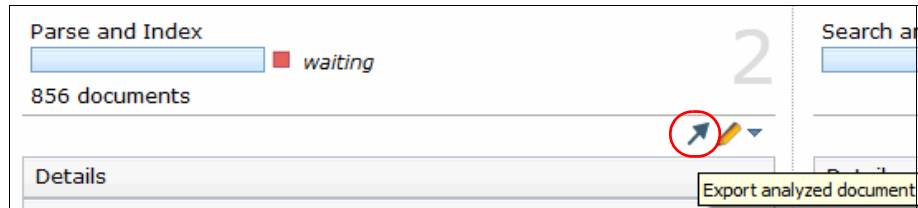


Figure 10-17 Export tab in edit mode

4. In the Export Searched Documents panel (Figure 10-18), follow these steps:
  - a. Under Analyzed document export options, select the **Export documents into a relational database** option.
  - b. Click **Configure**.

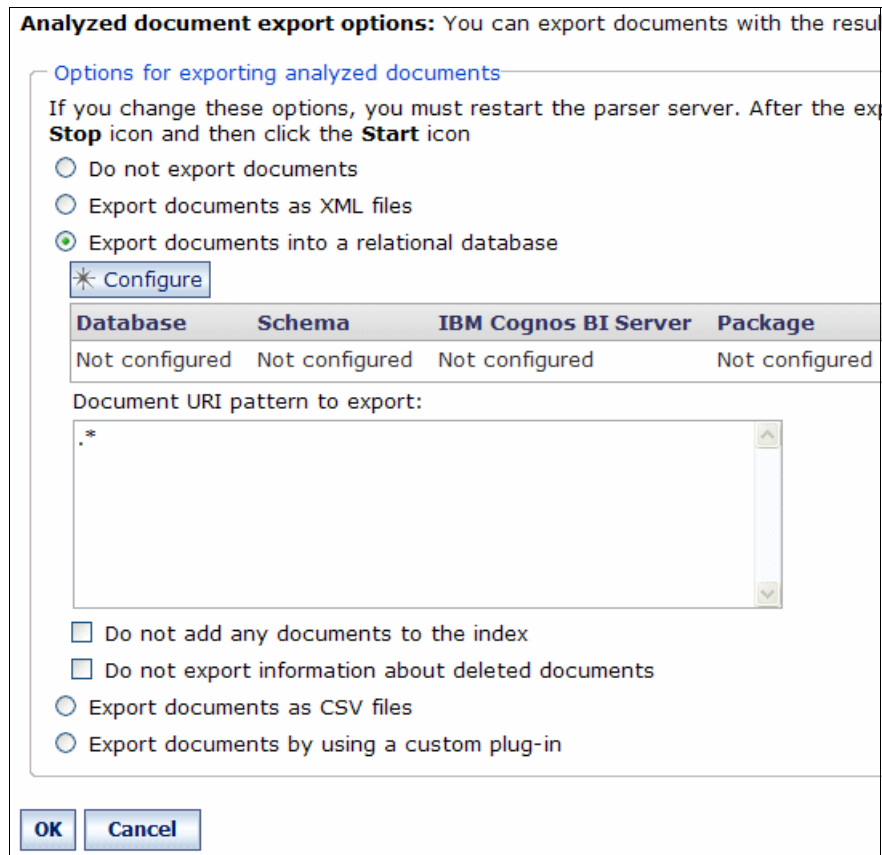


Figure 10-18 Export configuration into a relational database

- c. In the Content and Fields to Export to a Database panel (Figure 10-19), enter values for the database URL, user name, password, and class path variables according to your environment. Then, click **Next**.

**Table and database creation:** Content Analytics automatically creates the tables that you specified, but you *must* create a database or use an existing database when editing the database URL.

### Database Information for Exported Documents

[Help for this page](#)

Documents will be exported into star-schema tables.  
Specify the following database connection information.  
The database must exist and the user must have privileges to create a table and  
After you click Next, IBM Content Analytics tries to create tables under the spec

Tip: For Oracle, the schema name is usually the same as the user name.

\* JDBC database type:

\* JDBC driver name:

\* JDBC driver class path:

\* Database URL:  
Examples:  
- DB2 UDB: jdbc:db2://localhost:50000/sample  
- Oracle: jdbc:oracle:thin:@localhost:1521:sample  
- SQL Server 2008: jdbc:sqlserver://localhost:1433;DatabaseName=sample

\* User ID:

\* Password:

\* Schema of tables to store data for report generation:

Figure 10-19 Database Information for Exported Documents panel

- In the Content and Fields to Export to a Database panel, select the fields to export. Set a wanted column name, data type, and length for each exported field based on your data set. Figure 10-20 shows an example of setting a field to export. For this example, select the **A column for a document fact table** radio button in the doc\_category row. Then, click **Next**.

doc_category	<input type="radio"/> Do not export
	<input checked="" type="radio"/> A column for a document fact table
	Column name: DOC_CATEGORY
	Data type of keyword column: CHAR(50)
	<input type="radio"/> A table for a dimension

Figure 10-20 Setting a field to be exported

- In the Facets to Export to a Database panel (Figure 10-21), select the facets that you want to export. Set a wanted column name, data type, and length for each exported field based on your data set. Each selected facet to be exported results in a new table being added to the database. In this scenario, select **A table for a dimension** in the Category row. Then, click **Next**.

Category	<input type="radio"/> Do not export
	<input checked="" type="radio"/> A table for a dimension
	Table name: CATEGORY
	Data type of keyword column: CHAR(50)

Figure 10-21 Setting a facet to be exported to a database

7. In the “Continue or Finish the Wizard” panel (Figure 10-22), select the **Finish this wizard and save the current settings** radio button.

**Further analysis:** You can perform further analysis by using data warehouse applications and generating reports such as Cognos Business Intelligence. To use the wizard to configure IBM Cognos BI server reports, select the **Continue this wizard and configure the IBM Cognos BI server** radio button (Figure 10-22).

**Continue or Finish the Wizard**

[Help for this page](#)

If you click Finish, IBM Content Analytics tries to create tables under the specified database.

If you configured the IBM Cognos BI server and want to publish a package to the server, you must click Next. By using the package, the information exported to the database can be used for business intelligence. After you click Next, IBM Content Analytics tries to create tables under the specified database.

Continue this wizard and configure the IBM Cognos BI server

Finish this wizard and save the current settings.

**Back** **Next** **Finish** **Cancel**

Figure 10-22 Continue or Finish the Wizard window

Then, click **Finish**.

As a result, Content Analytics attempts to create tables under the specified database.

## Running the export

At this point, the export to a database is configured and the database tables are created. To export the analyzed documents, perform the following steps:

1. If your parse and index service is already started, restart the service.
2. If the collection does not have any crawled documents in the index, start the crawler, parse, and index components. If you already crawled and parsed documents, also perform a full build of index.

## Validating the results of the analyzed documents export

Using Export Monitoring in the administration console, validate that the export request has been queued by the Content Analytics server. Additionally, access the database, and check the number of records in the DOC\_FACT table.

For example, if you used the default database table and schema names, enter the following command in the DB2 command window:

```
SELECT COUNT(*) FROM COL_SAMPLE.DOC_FACT
```

The returned number is the same as the number of documents that were analyzed. Figure 10-23 shows the result of the exported data in a DB2 database.

Edits to these results are performed as positioned UPDATES and DELETES. Use the Tools Settings notebook to change the form of editing.

ID	URI	DOC_CATEGORY	DATE	DATE_FACET_ID	
1	file:///c:/program+fi...	Package / container ...	Jan 1, 2008 ...		1
2	file:///c:/program+fi...	Contamination / tamp...	Jan 2, 2008 ...		2
3	file:///c:/program+fi...	Number of pieces ...	Jan 2, 2008 ...		2
4	file:///c:/program+fi...	Package / container ...	Jan 2, 2008 ...		2
5	file:///c:/program+fi...	Ads ...	Jan 3, 2008 ...		3
6	file:///c:/program+fi...	Prank ...	Jan 3, 2008 ...		3
7	file:///c:/program+fi...	Taste / smell ...	Jan 3, 2008 ...		3
8	file:///c:/program+fi...	Number of pieces ...	Jan 4, 2008 ...		4
9	file:///c:/program+fi...	Package / container ...	Jan 4, 2008 ...		4
10	file:///c:/program+fi...	Change of properties...	Jan 5, 2008 ...		5
11	file:///c:/program+fi...	Prank ...	Jan 5, 2008 ...		5
12	file:///c:/program+fi...	Expiration date ...	Jan 7, 2008 ...		6
13	file:///c:/program+fi...	Number of pieces ...	Jan 7, 2008 ...		6
14	file:///c:/program+fi...	Number of pieces ...	Jan 8, 2008 ...		7
15	file:///c:/program+fi...	Package / container ...	Jan 8, 2008 ...		7
16	file:///c:/program+fi...	Package / container ...	Jan 8, 2008 ...		7
17	file:///c:/program+fi...	Taste / smell ...	Jan 8, 2008 ...		7
18	file:///c:/program+fi...	Ingredient ...	Jan 9, 2008 ...		8
19	file:///c:/program+fi...	Change of properties...	Jan 10, 200...		9
20	file:///c:/program+fi...	Package / container ...	Jan 10, 200...		9
21	file:///c:/program+fi...	Taste / smell ...	Jan 10, 200...		9
22	file:///c:/program+fi...	Empty ...	Jan 11, 200...		10
23	file:///c:/program+fi...	Ingredient ...	Jan 11, 200...		10
24	file:///c:/program+fi...	Ingredient ...	Jan 11, 200...		10
25	file:///c:/program+fi...	Number of pieces ...	Jan 11, 200...		10
26	file:///c:/program+fi...	Package / container ...	Jan 11, 200...		10

Figure 10-23 DOC\_FACT table in DB2 with exported analytics data

### 10.6.3 Exporting search result documents to the file system for IBM Content Classification

With Content Analytics, you can export search results (documents) from the content miner application for IBM Content Classification. You can export a limited set of documents for further analysis, monitoring, and reporting.



As shown in Figure 10-24, you can export the following items:

- ▶ **Crawled content and metadata**  
Exporting with this option yields similar output as exporting crawled documents. With this option, Content Analytics exports the native content and metadata as explained in 10.2.1, “Crawled documents” on page 355, for the search results. The content is exported as a .dat file or as native content.
- ▶ **Parsed content with analysis results**  
When you export search results with this option, Content Analytics exports metadata, facets, and extracted text. Facets and extracted text are included in the metadata file.
- ▶ **Crawled content and parsed content with analysis results**  
With this option, the exported output is a combination of the first two options.

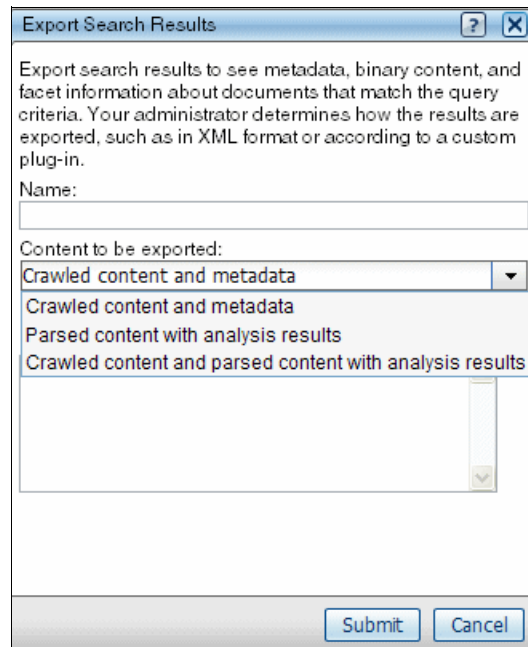


Figure 10-24 Export options for the search result documents

To export search results to the file system for Content Classification, use the following procedure:

1. Configure export for the search results by using the administration console.
2. Perform a search, and export the search results in the content analytics miner.

You can optionally schedule an export for a later time by using the administration console after the request is made. Use the steps in the following section to get started.

## Configuring export for the search results

To configure the export option for the search results in Content Analytics, follow these steps:

1. From the administration console, click **Collections** from the toolbar. Locate the collection that you want to edit, and click **Edit**.
2. Select the **Export** tab, and click the **Configure options to export searched documents** link.
3. Under “Options for searched document export” (Figure 10-25), select **Export documents as XML files for IBM Content Classification**, and enter the path of an existing directory for the output.

Options to Export Searched Documents

[Learn more](#)

**Searched document export options** Users can export documents after searching a collection.

Do not allow documents to be exported

Options for searched document export

Export documents as XML files

Export documents as XML files for IBM Content Classification

Output file path for searched document metadata:

Export documents into a relational database

Export documents as CSV files

Export documents by using a custom plug-in

Figure 10-25 Export search result documents

4. Click **OK**.

## Performing a search and exporting the search result

After you configure the export for the search results, search and export the search result. For illustration purposes, this example shows the steps to search for complaints about ice cream.

To search and export the search result, follow these steps:

1. Launch the content analytics miner, and click the **Expand this area** icon at the top of the panel to view the query text area.

2. Enter ice cream as the search term in the text area, and click **Search** (Figure 10-26).



Figure 10-26 Executing the search

3. Click the **Export** icon.
4. In the Export Search Results window (Figure 10-27), complete the following tasks to configure the export:
  - a. In the Name field, enter Ice Cream Complaints.
  - b. For Content to be exported, enter Crawled content and metadata.
  - c. For Schedulable, click **No** or **Yes**. If you select **Yes**, you must configure a schedule in the administration console. Select **No**.
  - d. Enter a description of Ice Cream.
  - e. Click **Submit**.

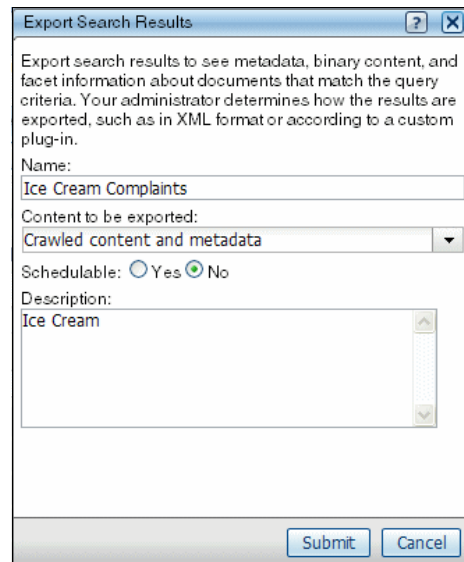


Figure 10-27 Export Search Results window

5. In the confirmation window (Figure 10-28), click **Close**.

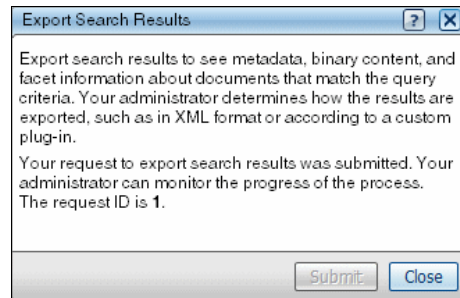


Figure 10-28 Export Search Results showing confirmation with a request ID

If you configured a scheduled export, continue to the next section. Otherwise, jump to “Validating the export” on page 382.

### Optional: Scheduling an export

Scheduling an export is an optional feature of Content Analytics. To schedule for an export, follow these steps:

1. Perform a search in the content analytics miner following steps similar to the steps in “Performing a search and exporting the search result” on page 378, except select **Yes** for the Schedulable radio button.
2. From the administration console, click **Collections** in the toolbar. Locate the collection that you want to edit, and click **Edit**.
3. Click the **Export** tab (Figure 10-29) and click the **Configure a schedule for exporting documents** link.

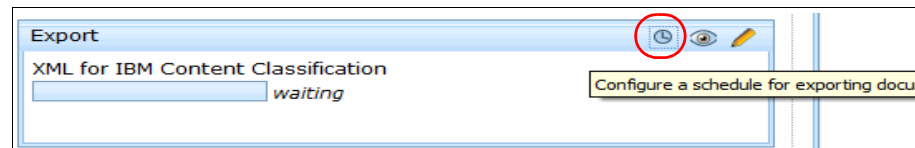


Figure 10-29 Scheduling an export for searched documents

4. Under “Specify a general schedule” (Figure 10-30 on page 381), follow these steps:
  - a. For the “Start on” field, select the appropriate values for hours, minutes, and time zone.
  - b. For the “Update interval” field, select the appropriate values for the specific days of the week, month, and hours.

- c. In the bottom table, locate your request ID for your exported scheduled search. Click the **Enable** icon to enable the export schedule.
- d. Optional: In the Schedule Type field, select **Custom**, and specify a custom schedule. You specify the custom schedule by clicking the **Configure** icon next to the schedule type field.
- e. Optional: Select the **Incremental Export** check box.
- f. Click **OK**.

**Specify a general schedule**

Start on: Hours: 11 pm, Minutes: 45, Time zone: Eastern Standard Time

Update interval:  Specific days of the week (hold the Ctrl key to select more than one day)

Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday

Specific days of the month (hold the Ctrl key to select more than one day)

1, 2, 3, 4, 5, 7

Specific hours of the day (hold the Ctrl key to select more than one hour)

0 am, 1 am, 2 am, 3 am, 4 am, 5 am, 6 am

Request ID	Export ID	User Name	Description	Next scheduled time	Enable or disable	Incr
2	Ice Cream Complaints		Ice Cream	12/5/10 5:00 AM		

OK Cancel

Figure 10-30 Configuring the schedule to export the search results

- 5. Click the **Configure a schedule to export search documents** link. In the “Next scheduled time” field (Figure 10-31), a time is set for the next scheduled export.

Request ID	Export ID	User Name	Description	Next scheduled time	Enable or disable	Incr
2	Ice Cream Complaints		Ice Cream	11/29/10 11:45 PM		

Figure 10-31 Scheduled export of searching the result documents

## Validating the export

Using Export Monitoring in the administration console, validate that the export request has been queued by the Content Analytics server. Check the output directory that you specified to verify that data is being exported.

For the request that we submitted, two files are created in the C:\Data2Export\ContentClassification directory:

- ▶ catalog.xml
- ▶ Ice Cream Complaints.xml

Example 10-5 shows a partial catalog.xml file.

### *Example 10-5 Partial catalog.xml file*

---

```
<?xml version="1.0" encoding="UTF-8"?>
<_Catalog entry_count="19">
  <Entry display_name="Body" type="string" nlp_usage="PlainText" is_viewed="true"
is_categories="false" is_link="false" is_matches="false" is_scores="false"
is_firedRules="false" is_changedNVPs="false"><![CDATA[Body]]></Entry>
  .....
</_Catalog>
```

---

Example 10-6 shows a partial Ice Cream Complaints.xml file.

### *Example 10-6 Partial Ice Cream Complaints.xml file*

---

```
<?xml version="1.0" encoding="UTF-8"?>
<Corpus_Bundle>
  <Corpus_Item>
    <ICM_NVP key="Body">
00000850
vanilla ice cream - Taste / smell
2008-12-30
1230635992343
Taste / smell
Strange odor
vanilla ice cream
I bought some ice cream today, but it had a strange odor.</ICM_NVP>
  <ICM_NVP key="date">1230635992343</ICM_NVP>
  <ICM_NVP
key="directory">C:&#x5C;IBM&#x5C;es&#x5C;samples&#x5C;firststep&#x5C;data&#x5C;xml&#x
5C;xml-data</ICM_NVP>
  <ICM_NVP key="doc_category">Taste / smell</ICM_NVP>
  <ICM_NVP key="doc_id">00000850</ICM_NVP>
  <ICM_NVP key="doc_product">vanilla ice cream</ICM_NVP>
```

```

    <ICM_NVP key="doc_subcategory">Strange odor</ICM_NVP>
    <ICM_NVP
key="docid">file:///C:/IBM/es/samples/firststep/data/xml/xml-data/00000850.xml</ICM_N
VP>
    <ICM_NVP key="extension">.xml</ICM_NVP>
    <ICM_NVP key="filename">00000850.xml</ICM_NVP>
    <ICM_NVP key="filesize">395</ICM_NVP>
    <ICM_NVP key="modifieddate">1250055768000</ICM_NVP>
    <ICM_NVP key="title">vanilla ice cream - Taste / smell</ICM_NVP>
    <ICM_NVP key="Category">Ice Cream</ICM_NVP>
  </Corpus_Item>
</Corpus_Item>
.....
</Corpus_Item>
</Corpus_Bundle>

```

## 10.6.4 Exporting search result documents to CSV files

With Content Analytics, you can export crawled documents, analyzed documents, or search results to CSV files on the file system. By following this approach, you can work with the data outside of Content Analytics. The CSV files conform to the RFC 4180 standard. The files are delimited by a comma to identify each column. Moreover, the generated CSV files are imitated star-schema tables that are similar to the star-schema created during the document export to relational database functionality.

### Configuring the search export to CSV files

To configure the export to CSV files in Content Analytics, follow these steps:

1. From the administration console, click **Collections** from the toolbar. Locate the collection that you want to edit, and click **Edit**.
2. Select the **Export** tab (Figure 10-32) and click the **Configure settings for exporting documents from search results** link.

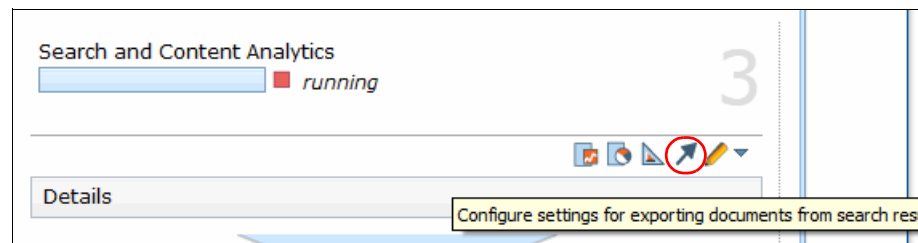


Figure 10-32 Export tab

- Under “Options for searched document export” (Figure 10-33), select **Export documents as CSV Files** and click **Configure**.

**Options to Export Searched Documents**

[Learn more \[?\]](#)

**Searched document export options** Users can export documents after searching a collection

Do not allow documents to be exported

Options for searched document export

Export documents as XML files

Export documents as XML files for IBM Content Classification

Export documents into a relational database

Export documents as CSV files

**Directory to save the CSV files**

Not configured

Document URI pattern to export:

.\*

Figure 10-33 Export documents as CSV files option

- Select the search fields that you want to export. You can export the field to a column in the document fact table by selecting the **A column for a document fact table** radio option associated to the field. Instead, you can select the **A table for a dimension** radio option to export the search field to its own CSV file that can be imported into a relational database as an individual table.

In this scenario, select the **A column for a document fact table** radio button for the doc\_category field, as shown in Figure 10-34. Then, click **Next**.

doc\_category

Do not export

A column for a document fact table

A table for a dimension

Figure 10-34 Fields to export to CSV files

- Select the facets that you want to export. You can select the **A table for a dimension** radio option to export the search field to its own CSV file so that it can be imported into a relational database as an individual table. For this scenario, select the **A table for a dimension** radio option for the **Product** facet, as shown in Figure 10-35 on page 385. Then, click **Next**.



Product	<input type="text"/>	<input type="radio"/> Do not export <input checked="" type="radio"/> A table for a dimension
		File name: <input type="text" value="product.csv"/>
<input type="button" value="Back"/> <input type="button" value="Next"/> <input type="button" value="Finish"/> <input type="button" value="Cancel"/>		

Figure 10-35 Facet to export to CSV files

- In the “Directory path to save the CSV files” panel (Figure 10-36), enter the directory path where the exported documents will be exported. For this scenario, type C:\Data2Export\csv in the “Save directory path” field. Create the directory path, C:\Data2Export\csv, on your machine because it must exist to configure the export. Then, click **Finish**.

### Directory path to save CSV files

[Learn more](#) ?

Specify the directory to save exported documents as CSV files. The directory must exist a

\* Save directory path:

Figure 10-36 Directory path to save the CSV files

7. Under “Options for searched document export” (Figure 10-37), if you want to limit the type of document that will be exported, add the document type to the Document URI pattern to export field. In this scenario, keep the default value of .\*, which exports all document types. Then, click **OK**.

**Options to Export Searched Documents**

[Learn more](#)

**Searched document export options** Users can export documents after searching a collection

Do not allow documents to be exported

Options for searched document export

- Export documents as XML files
- Export documents as XML files for IBM Content Classification
- Export documents into a relational database
- Export documents as CSV files

[Configure](#)

**Directory to save the CSV files**

C:\Data2Export\csv

**Document URI pattern to export:**

.\*

Figure 10-37 Search document export options

## Exporting search files to CSV files

After you configure the export to CSV files, perform a search, and export the search results to CSV files. After you perform the search and export steps, the crawled content and metadata for documents that contain the term, for example, ice cream, are exported to CSV files.

## Validating exported CSV files

Using Export Monitoring in the administration console, validate that the export request has been queued by the Content Analytics server. Check the output directory that you specified to verify that data is being exported.

For the request that we submitted, the following files are created in the C:\export\csv directory:

**date\_facet.csv** Contains the ID and Keyword fields as columns. Our document does not have any exported dates. Therefore, this file is empty.

<b>doc_fact.csv</b>	Contains the ID, URI, doc_category (search field that we previously configured for export), DATE, and DATE_FACET_ID.
<b>doc_flag.csv</b>	Contains the ID and Name for each document flag. Our example does not contain any exported document flags. Therefore, this file is empty.
<b>doc_flag_brg.csv</b>	Provides data to link the document flag with the specific document. It contains the ID for the file that matches the ID column value in the doc_fact.csv file and the ID of the document flag listed in the doc_flag.csv file.
<b>export_fact.csv</b>	Contains the document ID of the exported documents.
<b>export_metadata.csv</b>	Provides data related to the export. It contains the REQID, EXPORTID, DESCRIPTION, QUERY, and USER for the export request.
<b>product.csv</b>	Contains the ID and Product facet data.
<b>product_brg.csv</b>	Provides data to link the product facet with the specific document. It contains the ID for the file that matches the ID column value in the doc_fact.csv file and the ID of the product facet listed in the product.csv file.

## 10.7 Deep inspection

With the deep inspection feature of Content Analytics, you can export the entire analysis of a set of documents that match a query defined in the content analytics miner. The analysis and statistics generated by deep inspection are the same as those performed by the content analytics miner but without limits imposed on the number of facets and their values.

The content analytics miner is designed as an ad hoc text mining tool that supports rapid calculation in response to frequent changes in your query. If the analyzed data contains many facets and keywords, performance might be degraded. Consequently, limits are placed on the number of keywords (less than 500) that can be processed by the content analytics miner to prevent this kind of performance degradation.

In most cases, the limit does not affect your analysis. Discovery of trends, patterns, and high correlations usually surfaces in the most frequently occurring documents of your first 500 keywords (depending on your sort criteria). However, it is possible that, in certain scenarios, you want to view the entire set of calculations for all documents. You do that with deep inspection.

You enable the deep inspection function from the administration console. After enabling it, a deep inspection icon is enabled on each of the content analytics miner views, except for the Documents view. When you click the **Deep Inspection**, a batch submission is made for subsequent background processing of your query. The results of the deep inspection are stored in an XML file on the file system. The exported results from the deep inspection contain the selected keywords along with their frequency counts and correlation values. Deep inspection provides the same results as viewed from the content analytics miner but with deeper analysis.

You can schedule a deep inspection analysis and obtain reports on a periodic basis. Optionally, you can also submit the deep inspection request at any single point to view the analysis at an unscheduled time. The reports are exported as XML files.

The deep inspection feature is often confused with exporting search results because both features allow exporting the data of interest at any given time in the content analytics miner. When you export search results, you get metadata, binary content, and facet information about the individual documents that match the query criteria. The exported content is about individual documents. With the deep inspection feature, you get *analysis statistics*, such as the top keywords, trends, deviation, and correlation values, on the documents. With such statistics, a business analyst can create a custom application to compare deep inspection reports to detect any anomalies.

For example, a car manufacturing company that is analyzing incident reports can set up a query in the content analytics miner to find incidents on battery failure. The company can also run a deep inspection report based on the frequency of incidents on a weekly basis. By running the deep inspection reports on a weekly basis, the manufacturer determines that, on average, they encounter 10 battery failure incidents per week. However, the frequency of battery failure increased drastically to 20 for the last week and 17 the week before. This type of anomaly is detected by the trend and deviation analysis and can serve as an alert to the business analyst.

To obtain deep inspection reports, follow these steps:

1. Enable the deep inspection feature by using the administration console.
2. Issue deep inspection requests for analysis on a set of data.

You can monitor the request and validate the reports that are generated.

## 10.7.1 Location and format of the exported data

The location of the deep inspection report is similar to the location for exported documents as explained in 10.3.1, “Location of the exported data” on page 358. When you enable the deep inspection feature, you provide the path where the analysis statistics are to be exported. The output path of the analysis statistics generated by the deep inspection request depends on the name of the request and the time the request runs.

For example, if you create a deep inspection as XML request named `ProductFacetReport` on October 17, 2013 at 5:25pm, a new directory is created under the path. The name of the directory is based on the date and time at which the export occurs. The name of the created directory is in the *yyyymmddhhmm* format. The output files are created directly under the new directory. The report itself is in the XML format. Two files are created: One XML file that contains the analysis statistics and one XML file that contains information about the request.

The example has the following report output:

```
C:\Export\DeepInspectionReports\201310171725\  
ProductFacetReport.xml  
ProductFacetReport_log.xml
```

For an example of the generated reports, see 10.7.7, “Validating the deep inspection reports generation” on page 401.

## 10.7.2 Common configuration

To obtain meaningful analysis statistics from the deep inspection feature, you must understand the different configuration options and how they influence the analysis result. Figure 10-38 shows the configuration options when submitting a deep inspection request.

Create a Deep Inspection Report

Create a report to see detailed analysis statistics for all documents that match the query criteria, with no limits on the number of facet values that can be evaluated.

Name:

Maximum number of results:  
10

Maximum number of values (facet rows):  
100

Maximum number of values (facet columns):  
100

Sort results by:  
 Frequency  Index or correlation

Exclude results below this threshold value:  
5

Schedulable:  
 Yes  No

Description:

Select an output format:  
 XML  CSV

Submit Cancel

Figure 10-38 Deep inspection request configuration

Set the following options for a deep inspection request:

► Name

This field refers to the name for the deep inspection report. You can enter your own name or accept the default value.

- ▶ Maximum number of results

This field refers to the maximum number of analysis results to include in the generated report. You can select a value from the drop-down menu or enter your own value. For example, you can select the top 10 facets or the top 50 most correlated pairs of keywords for the two selected facets.

- ▶ Maximum number of keywords (facet rows)

This option refers to the maximum number of the most frequent keywords to include when calculating analysis statistics result. For example, if the maximum number of keywords is set to 1000 and the maximum number of results is set to 10, the deep inspection report contains the top 10 most frequent or correlated results out of top 1000 results.

Figure 10-39 shows an example of a Facet Pair view in the content analytics miner. In this view, the keywords “vanilla ice cream,” “chocolate ice cream,” “strawberry ice cream,” and “fruit jelly” are top keywords for the Product facet that will be considered when calculating the frequency or correlation among the documents. If the maximum number of keywords is set to 3, only the facet values “vanilla ice cream,” “chocolate ice cream,” and “strawberry ice cream” will be used. The reason is that they are the top three most frequent keywords for the Product facet.

Subfacets/ Keywords	Shortage	Leak (top)	Leak (bottom)	Lowing prices
vanilla ice cre... 101	21 1.0	0 0.0	0 0.0	2 0.0
orange juice 86	0 0.0	37 5.5	18 2.1	1 0.0
chocolate ice ... 85	24 1.5	0 0.0	0 0.0	1 0.0

Figure 10-39 Facet pair view in table format

- ▶ Maximum number of keywords (facet columns)

This option is enabled for the Facet Pairs view where the configuration for the second facet is necessary to perform facet pair analysis. When analyzing facet pairs, you also select the maximum number of most frequent keywords for the second facet. For example, in Figure 10-39, the values “Shortage,”

“Dirt (inside),” and “Allergy” are the top keywords for the facet subcategory, and they are displayed in the maximum number of keywords as facet columns.

► Sort result by

With this option, you can choose to sort the results by frequency or correlation.

**Deep inspection calculation:** Even though you can specify sorting by index or correlation, Content Analytics always finds the most frequent keywords first for the selected facet or facet pair. Content Analytics then calculates the top correlation or frequency among the most frequent keywords or facet pair values.

For example, as shown in Figure 10-39 on page 391, you select Product as a facet row and Subcategory as a facet column in the Facet Pair view. When you submit a deep inspection request, you set 1000 for Maximum number of keywords (facet rows) field and 1000 for Maximum number of keywords (facet columns) field. In this case, Content Analytics selects the 1000 most frequently used keywords for both the Product and Subcategory facets. It also computes a correlation and frequency (up to 1,000,000 correlation or frequency values). Depending on whether you select frequency or correlation for the Sort results by option, the list is generated and exported in the report.

Lastly, the deep inspection report contains results based on the value set for the Maximum number of results. For example, if the value for Maximum number of results is set to 100, the deep inspection report contains 100 out of 1,000,000 correlation or frequency values.

To find the most correlated data in less frequent keywords, you can select a larger number for Maximum number of keywords for facet rows and columns. By using this setting, deep inspection takes longer to generate the report.

► Exclude results below this threshold value

This option refers to the cutoff threshold of the result set. For example, for the result in Figure 10-40 on page 393, if you set a threshold of 80 and sort by frequency, only vanilla ice cream and chocolate ice cream are included in the report.







<input type="checkbox"/>	Keywords	Frequency	
<input type="checkbox"/>	vanilla ice cream	101	
<input type="checkbox"/>	chocolate ice cream	85	
<input type="checkbox"/>	fruit jelly	53	
<input type="checkbox"/>	strawberry ice cream	4	

Figure 10-40 Sample analysis result

If you set the threshold value to 5 and sort by correlation, the deep inspection report does not contain any results because the highest correlation value of 3.7 is less than 5.

► **Schedulable**

You can define the request as schedulable or as a one-time request. If you select the request as schedulable, see 10.7.5, “Optional: Scheduling a deep inspection run” on page 397, to configure a schedule. You must configure a schedule to obtain a deep inspection report.

► **Description**

Use this field to specify a description of the request. The description can be any string value.

### 10.7.3 Enabling deep inspection

You enable the deep inspection feature from the administration console. The icon for invoking a deep inspection request is enabled in the content analytics miner *only* if you enable it first from the administration console for the collection that you work on.

To enable deep inspection, follow these steps:

1. From the administration console, click **Collections** in the toolbar.
2. In the Collections view, locate the collection that you want to edit, and click **Edit**.
3. Click the **Text Analytics** tab (Figure 10-41 on page 394), and click the **Configure deep inspection settings** link.

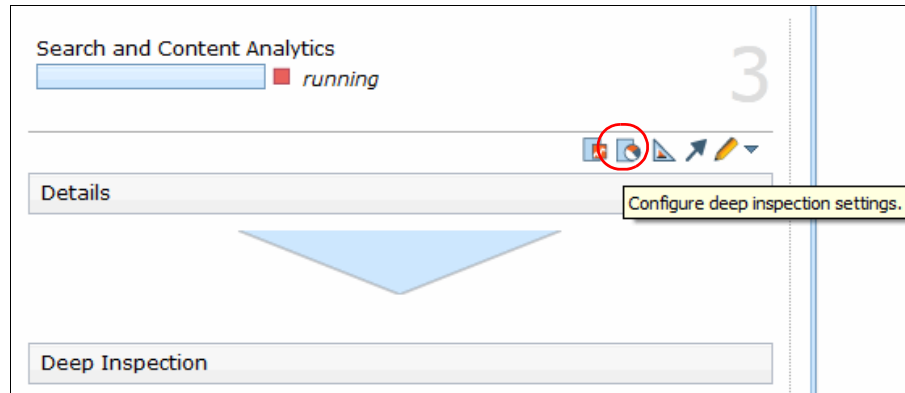


Figure 10-41 Selecting to configure the deep inspection options

4. Under Options for Deep Inspection (Figure 10-42), select **Export documents as XML files or CSV files**. In the “Output file path for inspection results” field, enter an existing directory name. Then, click **OK**.

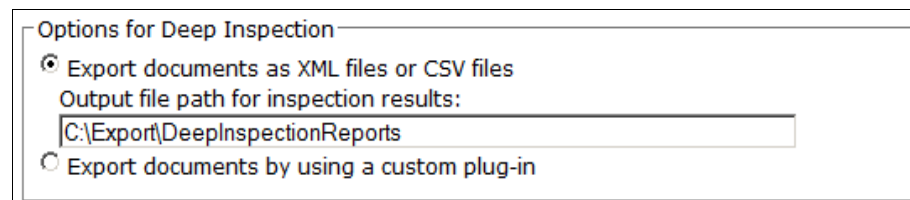


Figure 10-42 Configuring the options for deep inspection

## 10.7.4 Generating deep inspection reports

To generate a deep inspection report, submit a request using the content analytics miner. In this section, you are asked to click the **Deep Inspection** to issue a deep inspection request.

To submit a request for a deep inspection report (on facets), follow these steps:

1. Access the content analytics miner.
2. Click the **Show query input area** link.
3. Enter search text (for example, ice cream), and click **Search** (Figure 10-43 on page 395).

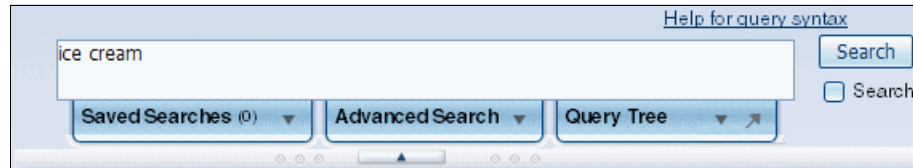


Figure 10-43 Executing a search

4. Go to the Facets view, and click the **Product** facet. Four different flavors of ice creams are displayed (Figure 10-44).

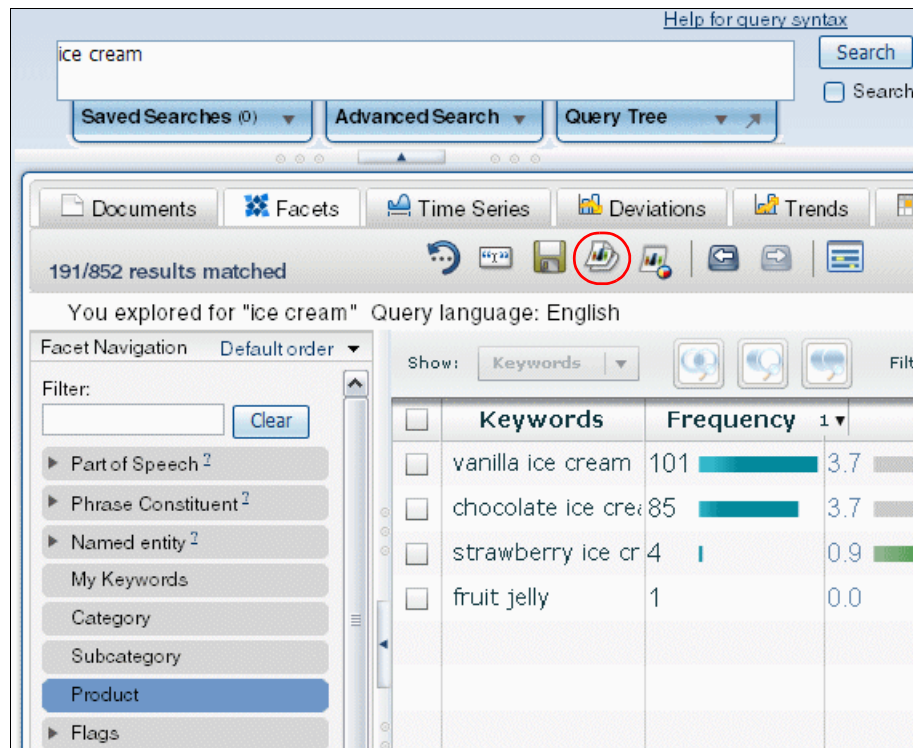


Figure 10-44 Facet Pair view

5. Click the **Deep inspection** icon.
6. In the Create a Deep Inspection Report window (Figure 10-45 on page 396), complete the following steps:
  - a. Enter a name.
  - b. For Maximum number of results, select **10**.
  - c. For Maximum number of keywords (facet rows), select **100**.
  - d. For Sort results by, select **Frequency**.

- e. For Exclude results below this threshold value, select 5.
- f. For Schedulable, select **No**.

**Schedulable field:** If you select **Yes** for the Schedulable field, you must configure a schedule as explained in 10.7.5, “Optional: Scheduling a deep inspection run” on page 397.

- g. Enter a description.
- h. For Select an output format, select **XML**.

**Select an output format field:** You can export the deep inspection results as XML or CSV files. If you select the **CSV** option, the deep inspection report is displayed in the CSV file format.

- i. Click **Submit**.

The screenshot shows a dialog box titled "Create a Deep Inspection Report". It contains the following fields and options:

- Name:** ProductFacetReport
- Maximum number of results:** 10
- Maximum number of keywords (facet rows):** 100
- Maximum number of keywords (facet columns):** 100
- Sort results by:**  Frequency  Index or correlation
- Exclude results below this threshold value:** 5
- Schedulable:**  Yes  No
- Description:** Deep Inspection report on the Product facet
- Select an output format:**  XML  CSV

Buttons for "Submit" and "Cancel" are located at the bottom right of the dialog.

Figure 10-45 Submitting a deep inspection report request

7. In the Export Search Results confirmation window (Figure 10-46), click **Close**.

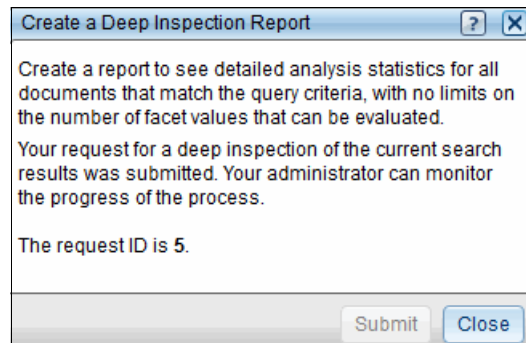


Figure 10-46 Deep inspection request submitted

If you configured a scheduled export, continue to the next section. Otherwise, jump to 10.7.6, “Monitoring the deep inspection requests” on page 400.

### 10.7.5 Optional: Scheduling a deep inspection run

With Content Analytics, you can schedule deep inspection analysis. The scheduling capability is similar to the capability for exporting the search results mentioned in 10.4.5, “Scheduling” on page 363, except for incremental export. That is, you cannot run deep inspection analysis on an incremental basis.

To schedule for a deep inspection run, follow these steps:

1. Generate a deep inspection report within the content analytics miner by following steps similar to the steps in 10.7.4, “Generating deep inspection reports” on page 394. The difference is that you must select **Yes** for the Schedulable radio button.
2. From the administration console, click **Collections** in the toolbar.
3. In the Collections view, locate the collection that you want to edit, and click **Edit**.

4. Click the **Text Analytics** tab (Figure 10-47), and click the **Configure deep inspection settings** link.

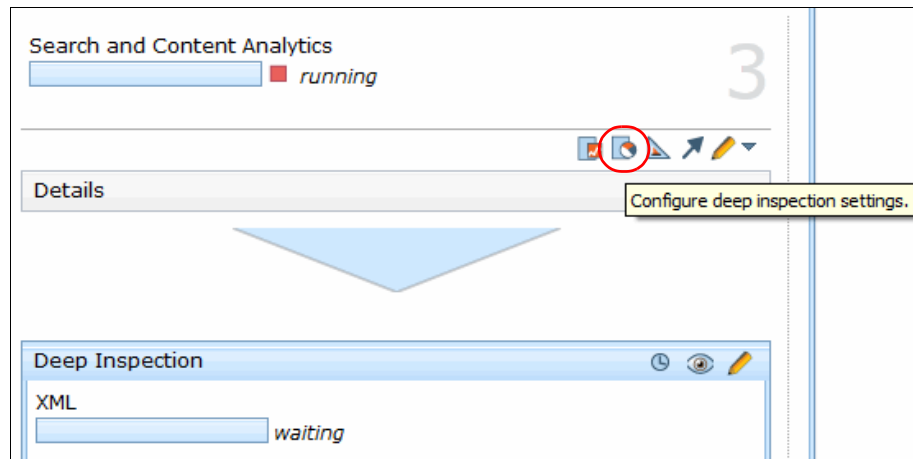


Figure 10-47 Selecting the Configure deep inspection schedules link

5. Under “Specify a general schedule” (Figure 10-48 on page 399), follow these steps:
  - a. For the “Start on” field, select the appropriate values for hours, minutes, and time zone.
  - b. For the “Update interval” field, select the appropriate values for specific days of the week, month, and hours of the days.

- c. Click the **Enable** icon to enable the export, as shown in Figure 10-48.
- d. Optional: for the Schedule Type field, select **Custom**, and specify a custom schedule.
- e. Click **OK**.

**Specify a general schedule**

Start on      Hours      5 am

                 Minutes      0

                 Time zone:      Pacific Daylight Time

Update interval  Specific days of the week (hold the Ctrl key to select more than one d

                 Sunday

                 Monday

                 Tuesday

                 Wednesday

                 Thursday

                 Friday

                 Saturday

Specific days of the month (hold the Ctrl key to select more than one

                 1

                 2

                 3

                 4

                 5

                 6

                 7

Specific hours of the day (hold the Ctrl key to select more than one ho

                 0 am

                 1 am

                 2 am

                 3 am

                 4 am

                 5 am

                 6 am

Request ID	Export ID	User Name	Description
7	ProductFacetReport		Deep Inspection Re

OK      Cancel

Figure 10-48 Configure schedule for deep inspection request

6. Click the **Configure deep inspection schedules** link. You see a time for the next scheduled run for deep inspection (Figure 10-49).

Request ID	Export ID	User Name	Description
7	ProductFacetReport		Deep Inspection Re

OK Cancel

Figure 10-49 Scheduled deep inspection request

## 10.7.6 Monitoring the deep inspection requests

Content Analytics provides a monitoring capability from the administration console to help you see the status of deep inspection analysis requests. You can validate that requests are created by using the following steps:

1. From the administration console, click **Collections** in the toolbar.
2. In the Collections view, locate the collection that you want to edit, and click the **Monitor** icon (Figure 10-50).

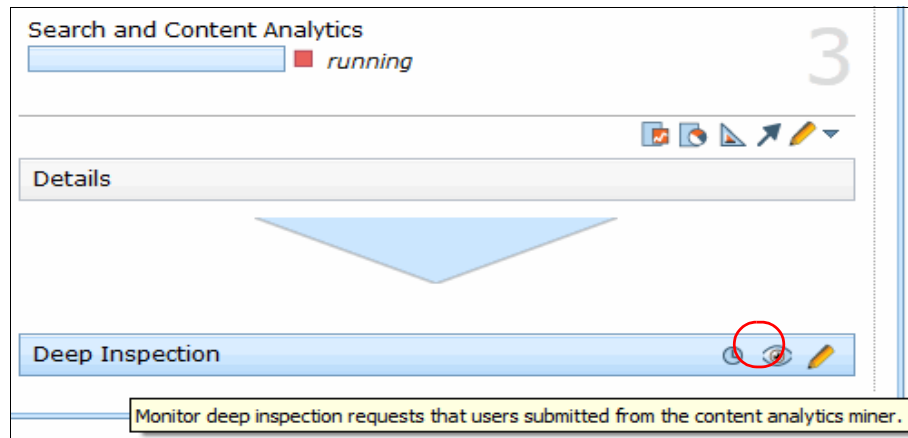


Figure 10-50 Collections View with editing and monitoring options

3. Click the **Text Analytics** tab (Figure 10-51 on page 401), and click the **View a history of deep inspection requests** link to see all the requests made for deep inspections.



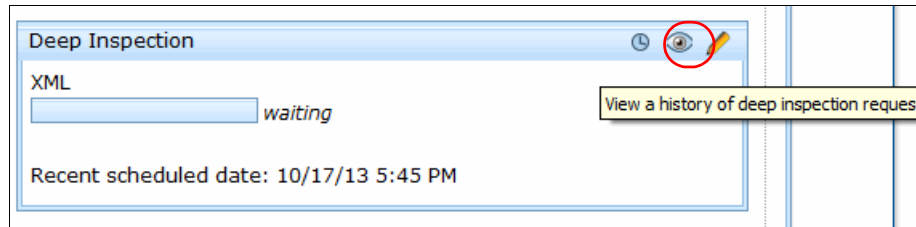


Figure 10-51 Viewing the deep inspection requests

Figure 10-52 shows all deep inspection requests that have been processed.

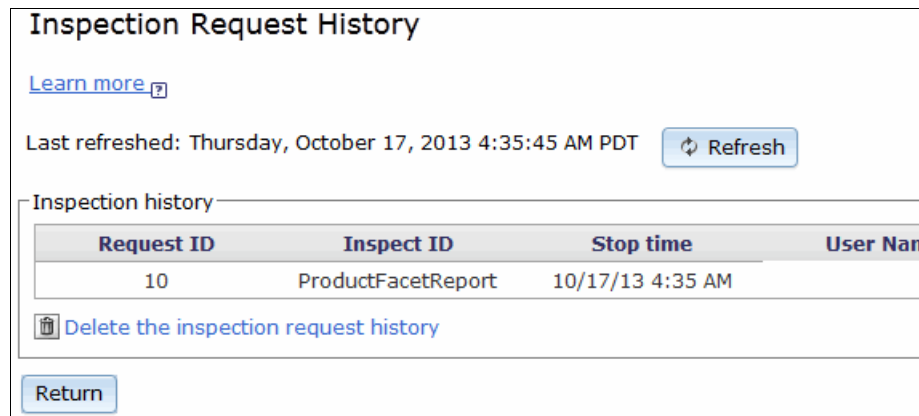


Figure 10-52 Deep Inspection Request History window

## 10.7.7 Validating the deep inspection reports generation

You can check the directory for the deep inspection output to verify that the reports are being generated. The output contains two files in the XML format. The file name ending with `_log.xml` is a metadata file that contains information about the deep inspection request, such as the name of the report, and the time it is requested. Example 10-7 shows a metadata file.

Example 10-7 Metadata file about deep inspection request

```
<?xml version="1.0" encoding="UTF-8"?>
<Document Id="ProductFacetReport" Type="NORMAL">
  <Content Truncated="false"
  Path="C:\#\x5C;Data2Export#\x5C;DeepInspectionReports#\x5C;20131017_11_ProductFacetReport#\x5C;ProductFacetReport.xml" Encoded="false"></Content>
  <Metadata>
    <Fields>
      <Field Name="QueryText">*. *</Field>
```

```

<Field Name="Facet.1">Product</Field>
<Field Name="Facet.1.Id">$.product</Field>
<Field Name="Facet.1.TaxonomyType">keywords</Field>
<Field Name="SortKey">frequency</Field>
<Field Name="ViewName">Facets</Field>
<Field Name="Description">Deep Inspection Report on Product Facet</Field>
<Field Name="Facet.1.MaxKeywords">100</Field>
<Field Name="Facet.2.MaxKeywords">100</Field>
<Field Name="MaxResults">10</Field>
<Field Name="ThresholdValue">5.0</Field>
<Field Name="StartedDateTime">2013.10.17 04:45:30 PDT</Field>
<Field Name="CompletionDateTime">2013.10.17 04:45:30 PDT</Field>
<Field Name="ResultCode">0</Field>
<Field Name="NumberOfRecords">10</Field>
</Fields>
<Facets></Facets>
</Metadata>
</Document>

```

---

The second XML file is a deep inspection report that contains details about facets, counts, deviations, correlations, and other information. Example 10-8 shows a deep inspection report that contains the top 10 keywords for the Product facet. In this report, the Count attribute represents frequency. Because the deep inspection report is generated by using the Facets view, the Index attribute represents the correlation value. If the deep inspection report is generated from the Trends or Deviations views, the Index value represents the index value.

*Example 10-8 A deep inspection report on the Product facet*

---

```

<?xml version="1.0" encoding="UTF-8"?>
<Report Id="ProductFacetReport">
  <Record Rank="1" Count="91" Index="0.8170133721666816">
    <Facet dimension="Facet.1">vanilla ice cream</Facet>
  </Record>
  <Record Rank="2" Count="83" Index="0.8723878024433697">
    <Facet dimension="Facet.1">orange juice</Facet>
  </Record>
  <Record Rank="3" Count="79" Index="0.8306349591676936">
    <Facet dimension="Facet.1">chocolate ice cream</Facet>
  </Record>
  <Record Rank="4" Count="57" Index="0.8549133848488474">
    <Facet dimension="Facet.1">mint jelly</Facet>
  </Record>
  <Record Rank="5" Count="49" Index="0.7705943389693553">
    <Facet dimension="Facet.1">fruit jelly</Facet>
  </Record>

```

```
</Record>
<Record Rank="6" Count="49" Index="0.8420716507091434">
  <Facet dimension="Facet.1">apple juice</Facet>
</Record>
<Record Rank="7" Count="46" Index="0.6443908369977036">
  <Facet dimension="Facet.1">pastry</Facet>
</Record>
<Record Rank="8" Count="40" Index="0.7444404373371146">
  <Facet dimension="Facet.1">pine juice</Facet>
</Record>
<Record Rank="9" Count="37" Index="0.7084494287371701">
  <Facet dimension="Facet.1">chocolate</Facet>
</Record>
<Record Rank="10" Count="35" Index="0.6767251666114928">
  <Facet dimension="Facet.1">lemon tea</Facet>
</Record>
</Report>
```

---

## 10.8 Creating and deploying a custom plug-in

You can customize both the export and deep inspection features by using the plug-in capability. A custom plug-in is a Java program that you write that is applicable to your business rules and processing of the exported data. Customizing can be done at each of three stages of export and for deep inspection. Figure 10-53 on page 404 shows an example of how you can select to configure a custom export plug-in for crawled and analyzed documents.

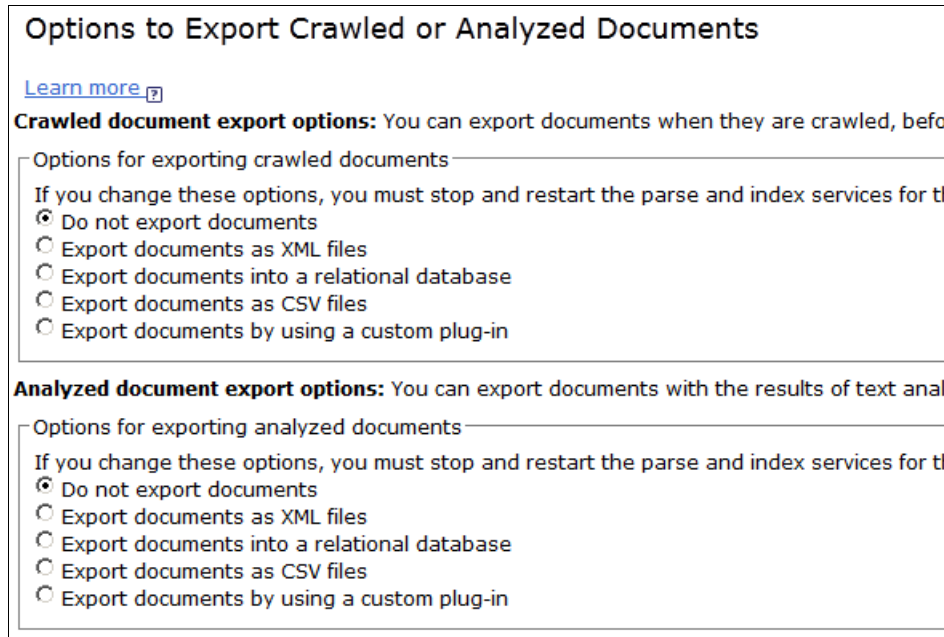


Figure 10-53 Option to configure a custom export plug-in

You can use this option to export data to different relational databases or in a different format. For example, on social websites or forums, each page contains multiple entries from different users. When Content Analytics crawls the content, a single web page is treated as a single HTML document, and thus the page is exported as a single crawled document. If you want to export multiple entries (from different users) as multiple documents, you can create a custom export plug-in at the crawled document stage to separate multiple entries as individual documents and export them individually.

You can also use the custom plug-in option of the deep inspection feature to export data to a relational database instead of to the file system. Then, you can use the Cognos 8 Business Intelligence tool to generate reports or to do additional analysis. By default, the deep inspection feature exports the results to the file system. Similarly, you can create a custom plug-in to analyze and deliver daily deep inspection reports by email.

For information about creating and deploying a custom export or deep inspection plug-in, go to the IBM Content Analytics Information Center at the following address, and search on *creating and deploying a plug-in for exporting documents or deep inspection results*:

<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/index.jsp>



# Customizing content analytics with IBM Content Analytics Studio

IBM Watson Content Analytics (Content Analytics) enables you to find insight from your content. With the addition of IBM Content Analytics Studio (ICA Studio), you can further customize your analysis based on a particular domain. Content Analytics populates facets by processing your content through analysis engines (annotators) that conform to the Unstructured Information Management Architecture (UIMA). You can extend the analysis capability by developing your own UIMA-compliant analysis engines (custom annotators) in ICA Studio.

This chapter explains how to compose and configure a UIMA pipeline (a sequence of UIMA annotators) in ICA Studio. We also show you how to export the UIMA pipeline to Content Analytics and associate its analysis results with facets.

This chapter includes the following sections:

- ▶ ICA Studio overview
- ▶ The building process of UIMA pipeline
- ▶ Use case: Building a UIMA pipeline for analyzing customers complaints
- ▶ Exporting annotators

## 11.1 ICA Studio overview

Content Analytics has rich capabilities to analyze unstructured text. This analysis becomes much stronger when it is customized to the relevant domain with the addition of ICA Studio. Each domain typically has its own set of frequently used and relevant terms, and some of these terms have meanings that change over domains.

To perform custom analysis that is based on a specific domain, you can use ICA Studio to define domain-specific custom vocabularies with richly defined entries and further process the resulting annotations with custom rules that are domain-specific.

In some technical domains such as medicine, vocabularies are highly controlled and it is easy to predict what terms appear in a document. When more informal language is used, you see similarity of words or near-synonyms. For example, if you are looking for references to packaging among customer comments, you need to know that these various terms might be relevant: pack, box, container, bottle, jar, and so forth. Extracting information from documents often means that you need to look for specific syntaxes such as phone numbers, street addresses, and license numbers. These too can vary from country to country, or even among different organizations. Also, the meaning of token depends on the context of appearance and requires checking their validity. For instance “Dr.” in the address “108 Glen Cove Dr.” could be picked up as a personal title (“doctor”) if the word context is not considered.

To capture these kinds of domain-specific expressions, ICA Studio provides a UIMA pipeline, which consists of four UIMA annotators (Figure 11-1 on page 407). Each annotator is designed to be data driven so that you can fit it to the target domain by creating domain-specific data resources (for example, dictionaries and rules).

There are four stages in the pipeline:

1. Document Language: The first stage identifies the language in which the document is written.
2. Lexical Analysis: The second stage splits the document into sentences and identifies tokens and their grammatical attributes such as lemma (base form) and part of speech. This is accomplished by looking up dictionaries and recognizing tokens according to their character makeup (such as telephone numbers and dates).
3. Parsing Rules: The third stage takes annotations sequence created by preceding stage and picks out token combinations as parts of defined contexts to identify meaningful, domain-specific expressions.

4. Clean Up: This fourth stage is an optional stage that filters out possible false hits and intermediate analysis results from final outputs.

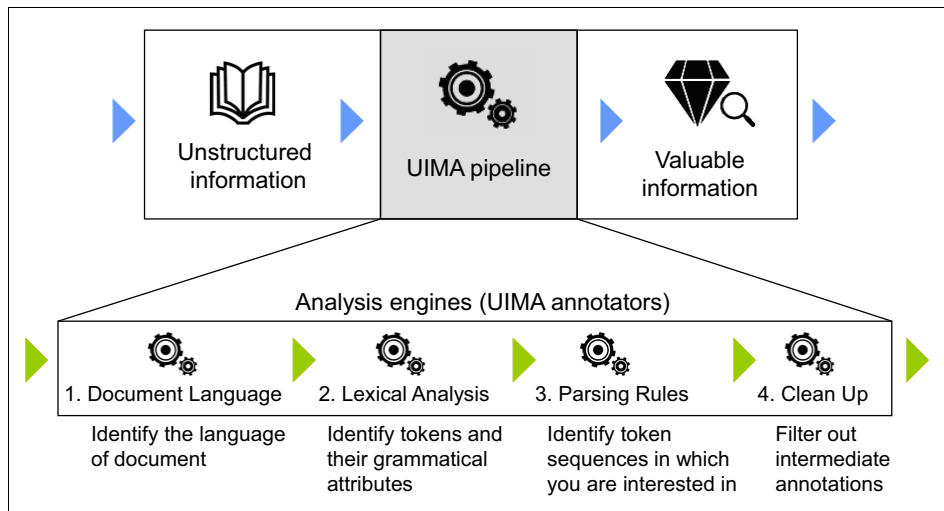


Figure 11-1 UIMA pipeline

ICA Studio provides tools for extracting annotations at different levels. Starting near the sentence surface, it extracts tokens word by word. You can add information about these tokens that are used by higher-level annotations. By checking the character makeup of the word, or looking it up from a dictionary, you can attach features to the words. At a higher level, parsing rules can identify word groupings according to constraints imposed on lower-level properties.

For example, if you are given a call center complaint, such as “I tried running ACME Database Search on my T100 Laptop and it crashed”, you can use ICA Studio to develop domain-specific resources to identify “T100 Laptop” as a piece of hardware, with a hardware problem “crash”. Figure 11-2 on page 408 shows various methods that are used to identify tokens and add meanings to them.

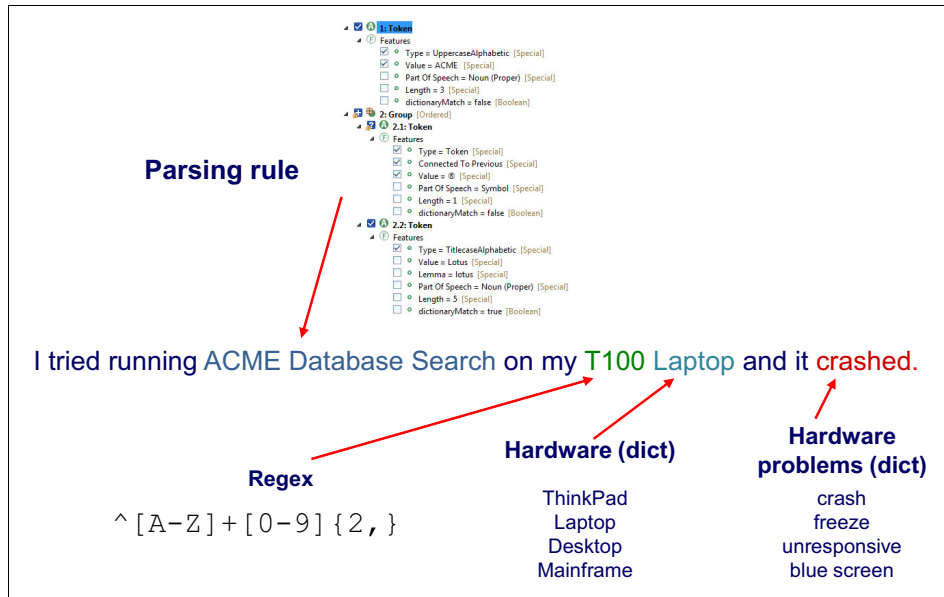


Figure 11-2 Various methods are used to identify tokens and add meaning to them

In Figure 11-2, two custom dictionaries (Hardware and Hardware problems) and one-character rule (in Regex) are created for the lexical analysis stage to identify domain-specific terms, and one parsing rule is created for parsing rules stage to capture customer complaint expression. This might look trivial, but when Content Analytics with ICA Studio processes thousands of call-center complaints such as this, a company might see a pattern with their hardware or a component of their hardware, that causes a particular type of problem. With this type of analysis, it helps a company to focus on a problem pattern. The company can then look into that hardware or a component to identify potential defect. If it is a defect, the company can resolve the defect proactively before more complaints come in due to this defect, or before they lose customer base because of dissatisfaction of their product due to this defect, or worse yet, before the defect, which causes serious safety problems among the customers.

## 11.2 The building process of UIMA pipeline

The UIMA pipeline organizes all the annotators and resources that you create in ICA Studio. Creating UIMA pipeline is an iterative process. In the planning phase, you usually start from a somewhat developed idea of what types of information you are interested in extracting from your content, which is based on business needs. For example, you want to discover what types of problems are the most



commonly reported ones to the IT department and which workers are reporting what kinds of problems. As you build your UIMA pipeline in ICA Studio and implement it within a solution, you get to know your content. As we show you in later sections, you create rules and dictionaries for a specific domain or purpose. When you implement these resources, the results might not exactly match your expectations. Trying out your annotator on as wide a range of sample texts as possible is the best way to learn exactly what kind of content you really have. When you try out your UIMA pipeline, you probably discover words that need to be added to your dictionaries. You might also find false hits that need to be formalized into parsing rules that can validate annotations according to their context. When your solution enters the production phase, the annotator results need to be monitored and evaluated, especially when the source or nature of the content changes.

One of the preferred practices for resource development is to repeat small steps iteratively (see Figure 11-3):

1. Start with a small document set.
2. Create a small set of resources.
3. Analyze and validate the results. Check to ensure that the results are as expected. If not, make appropriate changes.
4. Continue to build and enhance the engine by extending the resources.
5. Repeat these steps.

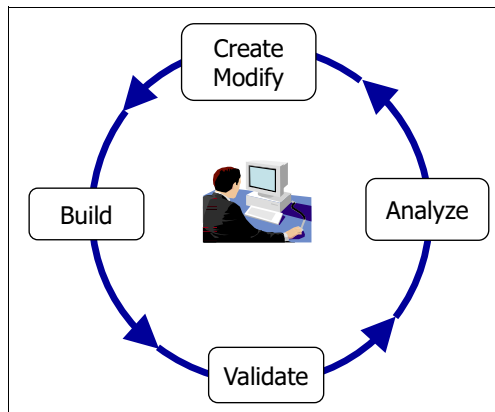


Figure 11-3 Iterative process

ICA Studio supports the iterative process:

1. Annotate a collection of documents.
2. Save annotation results.

3. Compare annotation results.
4. Check performance numbers.

This iterative process continues throughout the lifetime of the solution, adapting it to the changing business needs. The iterative process takes place not only within the ICA Studio, but also over time, when the annotator is being tested on the server and even when it is in production. By examining the successes and failures of the annotator, you can formulate changes in the annotator strategy, implement them in ICA Studio and then test them, within ICA Studio and also on the server.

The later sections provide the step-by-step procedure for configuring a UIMA pipeline `CustomerComments` for finding customer complaints among comments posted on the public website of a supermarket chain.

## 11.3 Use case: Building a UIMA pipeline for analyzing customers complaints

In this section, we create a UIMA pipeline, `CustomerComments`, which looks for complaints among customer comments. Within the domain of food retail, there are various types of complaints. For this use case, we create resources for detecting packaging defects such as leaking, broken packages, and product defaults such as moldy, spoiled cakes or cheeses. By identifying packaging defects and product problems, the goal is to make improvements on specific areas, such as packaging and products, to improve business operation and gain better customer satisfaction.

In addition to the basic tokenization supplied by lexical analysis, we want to identify various words that are relevant to our use case. For example, we want to find words that identify products, packages, and defects using custom dictionaries.

At the first stage of processing, Content Analytics provides lexical analysis for each supported language based on standard dictionaries. In order to extract meaning relevant to our domain (food retail), we need to identify words that refer to the concepts we want to identify, such as:

- ▶ Products, such as cakes, cheese, snacks, beverages
- ▶ Packaging, such as packet, jar, container, tub
- ▶ Relevant defects for products, such as stink, smell, mold
- ▶ Packages, such as leaking, crushed, open

In the following subsections, we provide a step-by-step procedure for creating the resources for finding customer complaints among comments posted on the public website of a large supermarket chain. The procedure is summarized as follows:

1. Create the ICA Studio project.
2. Create the UIMA pipeline.
3. Configure the basic UIMA pipeline.
4. Test the UIMA pipeline and review output.
5. Create custom dictionaries.
6. Create parsing rules.

After this end-to-end description, we examine some more advance rule parsers based on more complex texts from various domains.

This iterative process continues throughout the lifetime of the solution, adapting it to the changing business needs. The iterative process takes place not only within the ICA Studio, but also over time, when the annotator is being tested on the server and even when it is in production. By examining the successes and failures of the annotator, you can formulate changes in the annotator strategy, implement them in ICA Studio and then test them, within ICA Studio and also on the server.

The later sections provide the step-by-step procedure for configuring an UIMA pipeline CustomerComments for finding customer complaints among comments posted on the public website of a supermarket chain.

### 11.3.1 Creating the ICA Studio project

The ICA Studio project organizes all the resources needed to build and test a UIMA pipeline, with dictionaries, rule databases, documents, annotators, and so on. Before creating any resources, you need to first create an ICA Studio project.

To create the ICA Studio project, follow these steps:

1. From the **File** menu, choose **New** → **Content Analytics Studio Project**. The New Content Analytics Studio Project dialog appears, as shown in Figure 11-4 on page 412.

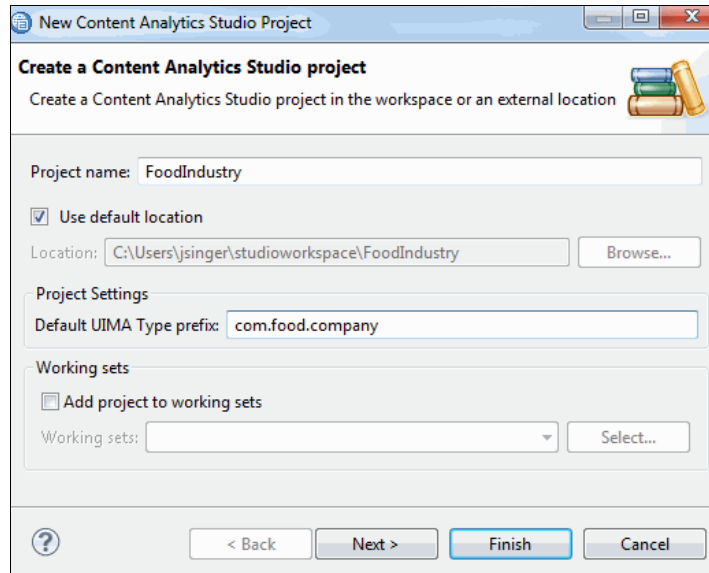


Figure 11-4 Creating an ICA Studio project

2. Enter the Project name and the Default UIMA Type prefix. This prefix is used as the default namespace of output annotations.
3. Click **Finish**. ICA Studio sets up a project with the default resource (such as dictionaries and parsers) folders.

### 11.3.2 Creating the UIMA pipeline

After the ICA Studio project is created, we need to create a UIMA pipeline, which is the container for organizing annotators and their resources. Initially, the UIMA pipeline contains only skeleton resources provided by ICA Studio.

To create the initial UIMA pipeline that will be the container for other resources, follow these steps:

1. In the Studio Explorer panel, go to the new ICA Studio project, then select the **Configuration** → **Annotators** folder, as shown in Figure 11-5 on page 413. This folder is where the new UIMA pipeline configuration will reside.

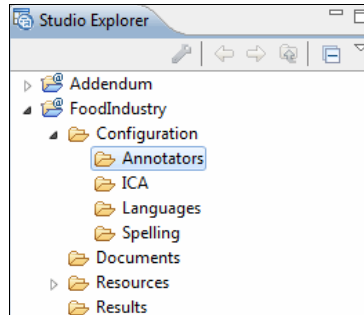


Figure 11-5 Navigating to the default annotators folder of the newly created project

2. Right-click the **Annotators** folder and select **New** → **UIMA Pipeline Configuration**.
3. Name the annotator CustomerComments.

The initial pipeline configuration contains a number of default stages. Notice that the new pipeline reports errors, as shown in Figure 11-6. These errors report missing resources within these stages. We configure the pipeline in the following sections.

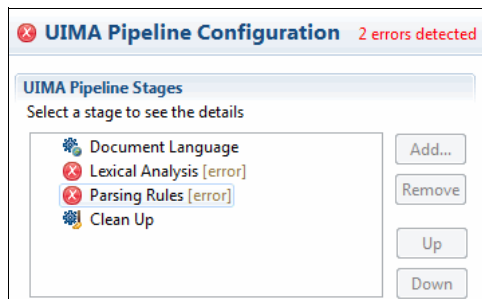


Figure 11-6 The new annotator (pipeline) contains default stages and errors

### 11.3.3 Configuring the basic UIMA pipeline

The UIMA pipeline contains Document Language, Lexical Analysis, Parking Rules, and Clean Up stages. For this use case, we simply set up Document Language and Lexical Analysis stages so that we can quick run and test the UIMA pipeline and examine the lexical analysis output with built-in resources.

## Setting up the Document Language stage

All UIMA pipelines begin with a document language analysis. You can choose to manually set the text language, or to automatically identify texts in the case where texts in more than one language exist:

- ▶ **Manually:** You can help the document language analysis by telling it the language of the input text. This removes any ambiguities created by words that are not identified by the language dictionaries. If the text contains a high percentage of nonstandard words, manually identifying the language avoids mistakes in automatic language identification.
- ▶ **Automatically:** Document language analysis can automatically identify languages.

For this use case, we configure the Document Language stage and remove the Parsing Rules stage as follows:

1. Select the **Document Language** stage. The Document Language configuration appears as shown in Figure 11-7.

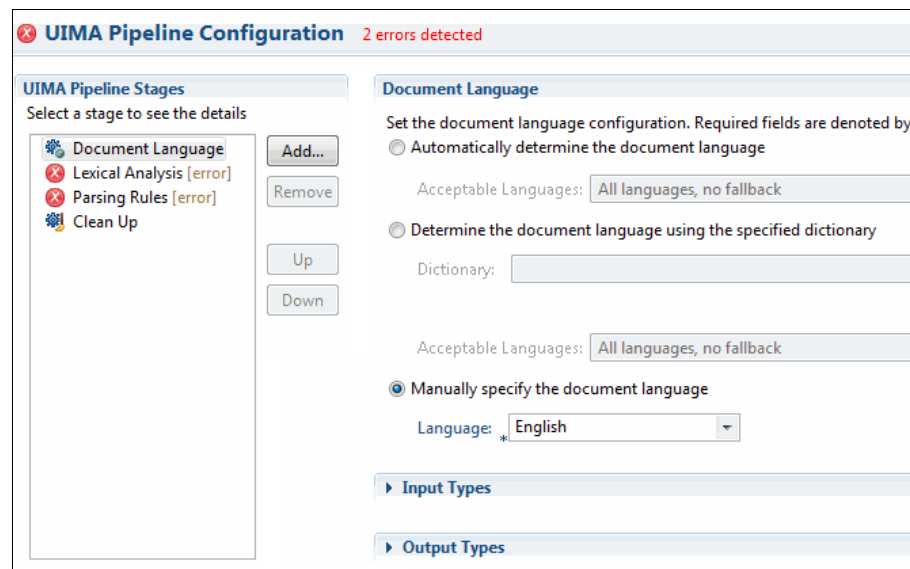


Figure 11-7 Setting up a document that will be used for dictionaries lookup

2. Select **Manually specify the document language**.
3. Select **English** as the language option.

## Setting up the Lexical Analysis stage

At the Lexical Analysis stage, you configure the dictionaries to use. ICA Studio is shipped with a set of dictionaries for annotating basic tokens, and to identify its part of speech (for example, noun, verb) according to the context of tokens.

For the use case, we set up the Lexical Analysis stage as follows:

1. Select the **Lexical Analysis** stage. The **Lexical Analysis** configuration pane appears (Figure 11-8).

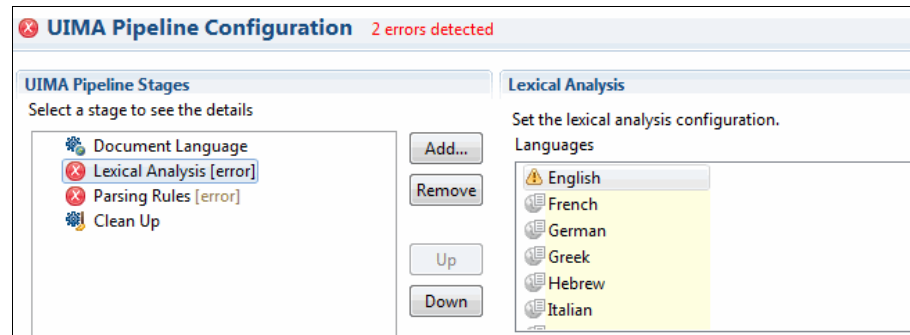


Figure 11-8 Setting up Lexical Analysis stage

2. In the **Language** pane choose **English**.
3. In the lower pane (not shown in Figure 11-8), select **Built In**. This inserts the default language dictionaries for the selected language.
4. Save the annotator by clicking the **Save** icon in the main toolbar. The error messages disappear.

## Removing the Parsing Rules stage

To examine the lexical analysis results, you can first run the UIMA pipeline without the Parsing Rules stage.

To remove the Parsing Rules stage:

1. Select the **Parsing Rules** stage from the UIMA Pipeline Configuration window.
2. Click **Remove**.

You have now created a UIMA pipeline that contains only Document Language and Lexical Analysis stages. You are ready to run the UIMA pipeline on an example text.

### 11.3.4 Testing the UIMA pipeline and reviewing output

The UIMA pipeline that we created contains the basic annotators. By looking at this output, we can formulate how to use these annotations at the Parsing Rules stage.

The procedure described here will be reused throughout the chapter to run the UIMA pipeline after adding successive stages and resources.

**Removing parsing rules:** If you have not removed the Parsing Rules stage, do so now before you run and test the UIMA pipeline. We show you how to add the Parsing Rules stage later in the chapter.

#### Creating a sample text file

To create a test document for testing the UIMA pipeline, follow these steps:

1. In the Explorer pane, select the **Documents** folder.
2. Right-click and choose **New** → **File**.
3. Name the file `CustomerComments.txt`.
4. Enter the following text and save the file:

It is before the expiration date, but the pastry was moldy. Isn't this a problem?

**Importing text files from a collection on Content Analytics:** Instead of creating a new text file on ICA Studio manually, you can import a set of flagged documents from a collection on Content Analytics instead. To import the flagged documents, select **File** → **Import** → **Documents from IBM Content Analytics with Enterprise Search Server**. See 5.7, “Document flagging” on page 148 for the details of document flagging.

**Text file:** Do not forget to add file extension “.txt” when creating a text file. ICA Studio relies on the file extension to choose files to display within the Studio Explorer.

#### Testing the UIMA pipeline

When the example text is created and opened in the text editor, you can run and test the UIMA pipeline as follows:

1. Right-click anywhere on the text editor.
2. Click **Analyze Document**.
3. Select the **CustomerComments.annoconfig** pipeline.



The document analysis results appear in the Outline tab in the upper right pane. This tab, as shown in Figure 11-9, lists all annotations that are detected by the chosen UIMA pipeline.

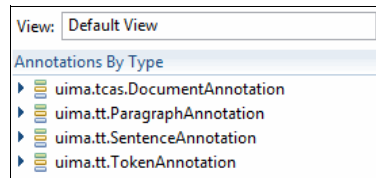


Figure 11-9 Results of the document analysis

4. Click any line, and the Properties tab appears in the lower pane listing all the annotation properties.

## Reviewing the Lexical Analysis output

The Lexical Analysis stage assigns parts of speech to tokens and also provides further information about each word. These attributes are available as constraints for parsing rules later.

### ***Parts-of-speech (nouns, verbs, and so on)***

Lexical Analysis parses sentence structure and tags words according to their grammatical role in the sentence (verb, noun, and so on) and adds this information as an attribute. This information can be useful to you for writing a parsing rule.

### ***Sentence, paragraph, and document level annotations***

Lexical Analysis identifies sentences according to a number of configurable parameters. At the most basic level, sentences are delimited by periods. Each of these annotations has associated properties. After running the analysis described above, the Outline tab in the upper-right corner displays each line in an expandable form. The text is analyzed on a few levels. At the surface level, each word and punctuation mark becomes a `uima.tt.TokenAnnotation`, each sentence becomes a `uima.tt.SentenceAnnotation`, and each paragraph becomes a `uima.tt.ParagraphAnnotation`.

Figure 11-10 on page 418 shows the different levels of annotations: Token, sentence, paragraph, and document.

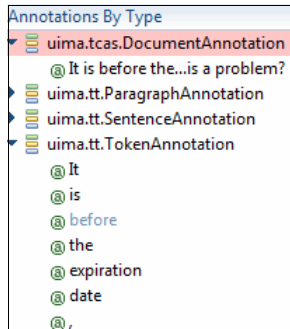


Figure 11-10 The default Lexical Analysis provides several levels of annotation

Click an annotation and details appear in the Properties tab (lower pane), the first uima.tt.TokenAnnotation listed is “It”. See Figure 11-11 that shows its property.

Property	Value
Covered text	@ It
Lemma	@ it
Part of speech	@ Pronoun
Part of speech (Disambiguated)	@ PRP - Personal pronoun
Token found in dictionary	@ true
Type	com.ibm.langware.uimatypes.TitlecaseAlphabetic

Figure 11-11 Properties of the token annotation

Review the annotations. Many of the properties are useful for writing parsing rules.

The UIMA pipeline also recognizes contractions. For instance the character sequence `isn't` is parsed into two tokens `is` and `not`. The first token of `isn't` is in the form of the verb `to be`. See Figure 11-12. The second token of `isn't` is annotated as `not`. See Figure 11-13 on page 419.

Property	Value
Covered text	@ Is
Lemma	@ be
Part of speech	@ Verb
Part of speech (Disambiguated)	@ VBZ - Verb, 3rd person singular present
Token found in dictionary	@ true
Type	com.ibm.langware.uimatypes.TitlecaseAlphabetic

Figure 11-12 The first token of “isn't” is in the form of the verb “to be”

Property	Value
Covered text	@ n't
Lemma	@ not
Part of speech	@ Adverb
Part of speech (Disambiguated)	@ RB - Adverb
Token found in dictionary	@ true
Type	uima.tt.TokenAnnotation

Figure 11-13 The second token of “isn’t” is annotated as “not”

### 11.3.5 Creating custom dictionaries for Lexical Analysis

Now that we have a basic Lexical Analysis and have identified syntax (sentences and parts of speech), we need to add customer dictionaries for identifying the tokens that are necessary for identifying customer complaints: Products, packages, and their attributes.

ICA Studio provides tools for creating custom dictionary databases, adding entries and compiling into binary dictionaries that Lexical Analysis uses at runtime.

**Preferred practice:** The best resource for finding words and patterns is your collection data. It is always better to use dictionaries rather than using rules. Dictionaries are easier to maintain and expand, and also better for performance.

#### Creating an empty dictionary database

The source of customer dictionaries is managed by a local database in the ICA Studio project. To create an empty dictionary database, follow these steps:

1. In Studio Explorer pane, right-click the **Resources** → **Dictionaries** folder. Choose **New** → **Dictionary Database**.
2. In the New Dictionary dialog, enter the dictionary name: Products.
3. Click **Next** twice. The Create Additional Database Columns dialog appears.
4. Click **Add** and the Database Column Definition dialog appears, as shown in Figure 11-14 on page 420.

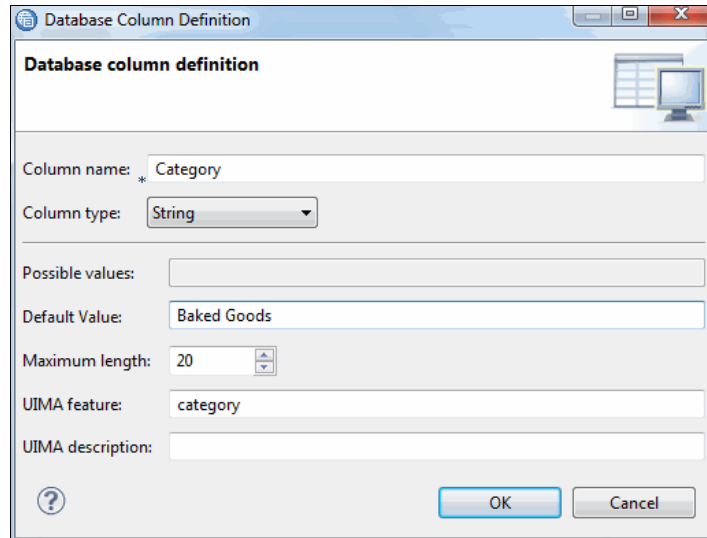


Figure 11-14 Creating database column

Dictionary entries can include any number of properties that will be added to the annotations resulting from a lookup match.

The various products that are sold in the supermarket fall into broad categories that might be relevant to the analysis.

5. Add the column name. For our use case, we enter Category.
6. Add a default value. For our use case, we enter Baked Goods.
7. Click **OK**.
8. Click **Next** and then click **Finish**.

### **Adding entries to the new dictionary database**

We have now created an empty dictionary database. To add new entries to the dictionary database, follow these steps for our use case:

1. On the Studio Explorer pane, right-click the newly created **Products** dictionary database, in the Dictionaries folder. Be careful to choose the database that has the [local database] label, not the dictionary itself.
2. In the main workspace, a Products tab is added. Right-click anywhere within the Products pane and select **Add Entry to Dictionary Products**, as shown in Figure 11-15 on page 421.

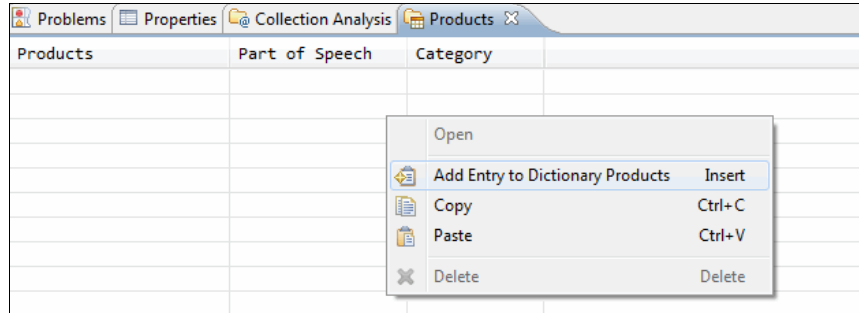


Figure 11-15 Adding new entries to the newly created dictionary database

3. Add the entry **bread**, as shown in Figure 11-16. In this instance, we can leave the default category **Baked Goods** and part of speech as **Noun**.

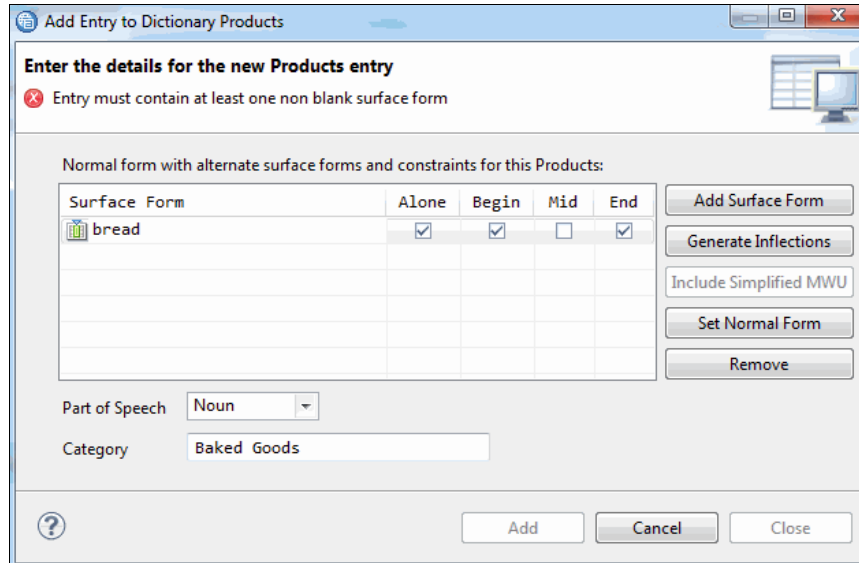


Figure 11-16 Adding entry to the Products dictionary

4. Leave the cursor on the entry **bread** and click **Set Normal Form**.  
Each dictionary entry must contain one normal form. It can contain any number of surface forms. Surface forms describe different ways of writing the surface value, such as contractions, abbreviations, and acronyms.
5. Click **Add**. The new entry is added and the dialog is reset.
6. Add additional entries for the Products dictionary (for example, ice cream, yogurt, potato chips), including values for the categories: Dairy and Snacks.

The annotator created in this chapter's use case includes two more dictionaries: Brands and Packaging. These can be created similarly, using Noun as the only part of speech. No Category column is necessary.

Here are some suggestions for the Packaging entries:

bottle, box, jar, package, pack, packet, top, cap, tub, container

### Defining Parts of Speech for dictionary entries

The Products dictionary contains a list of grocery items and therefore we do not need to change the default **Part of Speech** noun defined when we created the dictionary.

The Defects dictionary needs to list various types of words with negative connotations, all relevant to different aspects of the supermarket industry: products, packaging, and services. These words include:

- ▶ Adjectives such as “dirty”, “smelly”
- ▶ Verbs such as “leaks”, “stinks”
- ▶ Nouns such as “problem” and “mold”

Create a dictionary called *ProductDefects* and define a new column “Category”, similar to the Products dictionary above.

Add a new entry and its subforms to the dictionary:

1. From the Add Entry to Dictionary ProductDefects dialog, add the surface form leak.
2. For Category field, enter Packaging.
3. For Part of Speech, select **Verb**.
4. With the cursor on the new entry, click **Generate Inflections**.

This adds new surface forms, which are based on the detected conjunctions of the entry as a verb. This allows the annotator to match the word in different forms, as shown in Figure 11-17 on page 423.

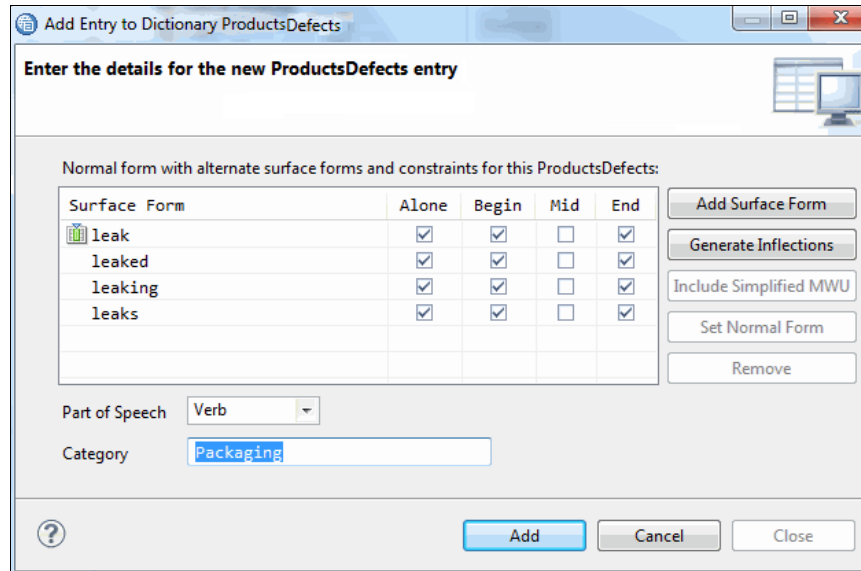


Figure 11-17 Creating new surface forms using the **Generate Inflections** button

In general, surface forms can be any variations of the word that need to be associated with the normal form. For our example, “leak” is the normal form. “leaked”, “leaking”, and “leaks” are the surface forms.

5. Save all the entries.

When all the entries are saved, compile the contents of the dictionary database:

1. In the explorer, find the dictionary database.
2. Right-click the database and select **Build Studio Resource**.

**Note:** Be careful to rebuild the dictionary after adding new entries to the dictionary database.

Alternatively, you can right-click the **Resources/Dictionaries** folder, and select **Build Studio Resource** to build all resources in this folder.

### Adding the custom dictionaries to the UIMA pipeline

To continue with the examples below, ensure that you have created the ProductDefects and Products dictionaries.

To add these dictionaries to the UIMA pipeline, follow these steps:

1. Open the CustomerComments annotator by selecting **FoodIndustry** → **Configurations** → **Annotators** → **CustomerComments** in the browser.
1. Select the **Lexical Analysis** stage.
2. In the **Dictionaries** pane, click **Select**.
3. Find the dictionaries created earlier and select them.
4. Click **OK**.
5. **Save** the changes.

### Testing the UIMA pipeline with custom dictionaries

After adding the custom dictionaries to the UIMA pipeline, you can test the new output by analyzing a test document. See 11.3.4, “Testing the UIMA pipeline and reviewing output” on page 416 for step-by-step instruction.

**Tip:** Try adding different words from your dictionary to your test text and check the analysis results.

## 11.3.6 Creating parsing rules

A parsing rule defines a sequence of annotations that indicates something of interest in the text. The Parsing Rules stage examines annotations created by earlier stages and checks their validity. It can also add additional meaning to the combinations of annotations. For instance, by looking for products and product defects in a sentence, it can identify a complaint. It can similarly identify products and positive attributes (tasty, favorite, delicious) to identify customer satisfaction.

Parsing rules can:

- ▶ Create new annotations.
- ▶ Remove invalid annotations.

The matching criteria of a parsing rule is defined for annotations that are created by earlier stages and can be:

- ▶ Tokens such as words, punctuation, and numbers
- ▶ Terms that you have defined in a custom dictionary
- ▶ Annotations that are created by another parsing rule

**Preferred practice:** It is always better to use dictionaries because they are easier to maintain and expand, and also better for performance. Do not try to stretch a rule to make it solve every problem. Complicated rules make maintenance difficult. It is important to keep rules short and simple.



## Creating an empty Parsing Rules database

Rules are stored in a Parsing Rules database. Each database contains multiple rules.

To create a new Parsing Rules database:

1. In the explorer, right-click **Resources** and select **Parsing Rules** folder icon.
2. Select **New** → **Parsing Rules Database**.
3. Name the database `MatchComplaint`.

**Note:** As a good practice, we usually do not recommend changing the UIMA type prefix after creating the project.

4. Accept all the default values and save the database.

## Adding the parsing rules to the UIMA pipeline

You can add multiple rule files to a single stage in the UIMA pipeline, or have multiple Parsing Rule stages.

To add the parsing rule file to the UIMA pipeline, follow these steps:

1. Under the **Configuration** → **Annotators** folder, double-click the **CustomerComments.annoconfig** file. The UIMA Pipeline configuration editor opens in the Workspace pane.
2. Click **Add** and select the **Parsing Rules** stage.
3. Click **OK**. The stage is added with error icons. Confirm that the Parsing Rules stage is selected.
4. Click **Select**. Select the `matchComplaint.jar` in the **Resource** → **Parsing Rules** folder and click **OK**.

## Writing the first parsing rule for matching a single token

Here we create a parse rule that creates a new annotation from a single annotation (token) based on its properties. We are looking for possibly new products or brand names that have not yet been added to our custom dictionaries.

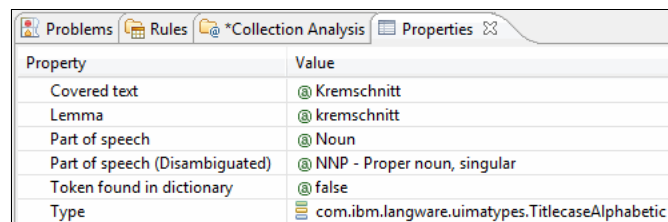
Words that are not recognized as belonging to the identified language, or are not included in a custom dictionary, can be tagged as such. Unrecognized words can then be tagged either as “words of interest” or become candidates for spelling correction.

In the Lexical Analysis output, basic tokens are identified, each having a set of properties. Here we explain how the annotator can be configured to check these properties, by itself or in combination. By checking these properties, we are able to assign new meaning to the resulting annotations. The process can be repeated at subsequent levels, while combining lower-level annotations, to create higher-level annotations.

Analyze the following text with the present version of the CustomerComments annotator using the procedure described in 11.3.4, “Testing the UIMA pipeline and reviewing output” on page 416):

It is before the expiration date, but the Kreamschnitt was moldy. Isn't this a problem?

To display the properties for the non-English word, find the line with the `uima.tt.TokenAnnotation` for the word `Kreamschnitt` in the Outline view and click this line. The properties of this annotation that is assigned will appear as shown in Figure 11-18.



Property	Value
Covered text	@ Kreamschnitt
Lemma	@ kreamschnitt
Part of speech	@ Noun
Part of speech (Disambiguated)	@ NNP - Proper noun, singular
Token found in dictionary	@ false
Type	com.ibm.langware.uimatypes.TitlecaseAlphabetic

Figure 11-18 The properties assigned by the initial Lexical Analysis

## Specifying the matching criteria of parsing rules

Now that we have looked at the available properties for the `Kreamschnitt` annotation, we open this annotation in the rule editor and use them as follows:

1. In the test document pane, right-click **Analyze Document** and select the **CustomerComments** annotator.
2. Double-click the **MatchComplaint** Parsing Rules database with the label [local database] in the **Resources** → **Parsing Rules** folder. The database opens and shows the Rule view in the bottom pane.
3. In the upper-right pane, select the **Create Parsing Rules** tab to open the rules editor.
4. In the text editor, select the word **Kreamschnitt**.
5. Drag-and-drop the selected word to the rule editor pane. This pastes the word `Kreamschnitt` into the text box and displays the annotations found in the text, according to the CustomerComments annotator.

6. In the list of properties, choose the properties that will become constraints for matching the new annotation. Check **dictionaryMatch=false** and clear **Type=TitlecaseAlphabetic**, as shown Figure 11-19.

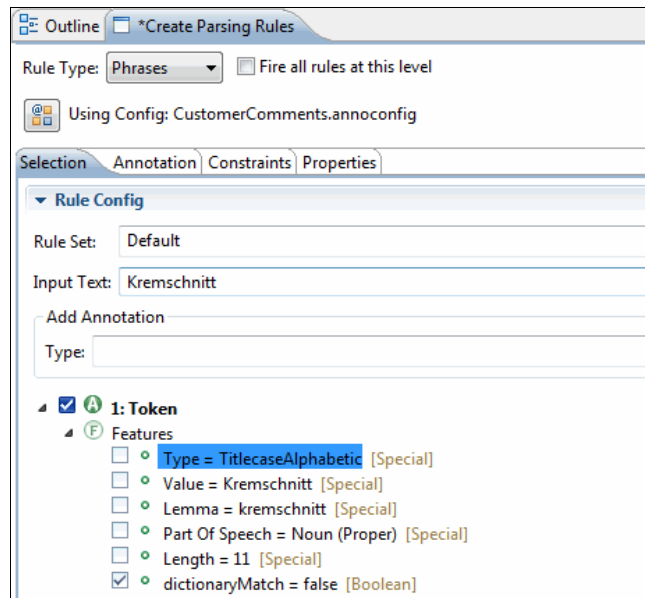


Figure 11-19 Creating Parsing Rules

By selecting **dictionaryMatch=false** in the Creating Parsing Rules window, we can collect words of interest that are not found in the default language dictionaries, or in our customer dictionaries.

In this tab, you can choose required properties that will be used as rule matching conditions. These properties are chosen according to the text that you selected in the test document.

**Note:** If you do not find the expected properties in the editor, you can change the available properties by changing the selected text, saving the document, and pasting the new text into the text box in the Selection tab. If the wanted results are still not produced, you need to check the relevant dictionary entry or create a new one.

### Defining an annotation created by a parsing rule

To create a new annotation for the token that satisfies the matching criteria, follow these steps:

1. Select the **Annotations** tab in the Parsing Rule editor.

In this tab, you can create new annotations by combining annotations created by the previous Lexical Analysis annotator. For our use case, we have only one annotation.

2. Right-click the root node (the token) and select **Insert Annotation**, as shown in Figure 11-20.

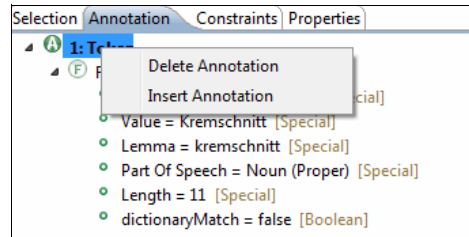




Figure 11-20 Select an annotation

3. Enter the name of the new annotation: `com.food.company.WordOfInterest`

**Note:** The default UIMA type prefix is created together with the project. To change the prefix, right-click the project in the explorer and select **Properties** → **Studio Content Analytics Project**. However, we do not recommend changing the prefix after the ICA Studio project is created.

4. Click **OK**.
5. Click **Add and Save** using the  button.
6. Build the rules JAR files from the database button  to build a binary resource.

After the build process is complete, ICA Studio automatically reruns your annotators and shows the new annotation, `com.food.company.en.WordOfInterest` in the Outline view.

## Writing more complex parsing rules to match token sequence

After you have analyzed the surface tokens, added annotations based on dictionary lookups, and discovered additional properties by Lexical Analysis, you can consider meanings that are based on combinations of these low-level annotations.

Now that we have identified packages, products, and defects, we want to further detect which products and packages are the most problematic. Specifically, we want to detect when a product or package is associated with a defect.

The sample content set contains comments submitted by customers to a supermarket website. Among the comments are complaints about different aspects of the products and services. The following are some aspects that we might find relevant to finding and describing these complaints:

- ▶ Comments that are negative
- ▶ Product, service, or brand that is associated with complaints
- ▶ Positive remarks about a competitor

Now that we have added various classes and properties to each word and token appearing in the raw text, we need to look at the relationships between the words and try to detect when a complaint is being expressed.

*Sentiment analysis* is a technique for identifying when a writer is expressing a negative opinion about the subject. Negativity is often expressed in general terms, using words such as “bad” and “disappointed”. Content Analytics provides sentiment analysis, which is ready for immediate use, in the content analytics miner. For information, see 6.10, “Sentiment view” on page 222. Here we take a more domain-specific approach and limit ourselves to those words that might reference the limited range of words in our custom dictionaries for the particular domain. Some words have negative connotations only within a specific domain. For example, the word “hair” in the beauty products domain is an important word with no negativity. However, the same word in regard to food and food packaging domain, it is almost exclusively negative.

**Note:** To walk through the example below, you need to have at least two dictionaries compiled and added to the CustomerComments annotator: Products and ProductDefects.

If you do not have the entry cake (in the Products dictionary) and moldy (the ProductDefects dictionary), substitute with your own entries.

To find additional annotations, we start by doing the following analysis:

1. In the test document, enter the following text:  
It is before the expiration date, but the cake was moldy.
2. Analyze this with the CustomerComments annotator. Among the annotations are the two as shown in Figure 11-21.

@com.food.company.ProductDefects	51	ration date, but the cake was moldy.	English Dates.txt	/IBM/Documents/Dates
@com.food.company.Products	42	the expiration date, but the cake was moldy.	English Dates.txt	/IBM/Documents/Dates

Figure 11-21 Two of the resulting annotations

3. Select the **Create Parsing Rule** tab.

4. Select the text **cake was moldy** and drag it to the rules editor pane.

The template of rule conditions that is based on analyzed annotations is created. See the collapsed view as shown in Figure 11-22. See the assigned annotations by our annotator containing only dictionaries

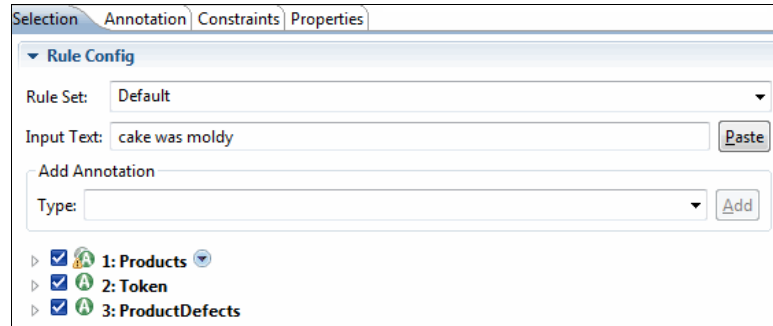


Figure 11-22 Assigned annotations by our annotator containing only dictionaries

### Specifying the matching criteria of parsing rule

There are many possibilities for constraint combinations depending on how precisely we want to describe the annotation. One possibility is to designate a noun-verb-adjective word group. In the Products dictionary, only nouns appear. However, we might want to constrain the ProductDefects annotation to an adjective, since that dictionary contains different parts of speech.

In the Selection tab, we can choose which annotations and properties we want to include in our constraints.

For our use case, perform these steps to choose which annotations and properties to include in the constraints:

1. Do nothing for the first annotation, Products. This annotation does not need any further constraints because all entries in that dictionary are nouns.
2. Configure the second annotation, Token:
  - a. Expand the Token annotation as shown in Figure 11-23.

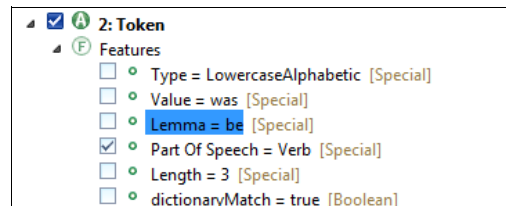


Figure 11-23 The properties for the token was

- b. Select the **Part Of Speech = Verb** feature as the only constraint.

**Note:** We can specify that there needs to be a conjugated version of the verb “to be” (by selecting **Lemma = be**), but this is probably too constraining. It would be helpful to create a dictionary of linking verbs such as seems and appears, and their expanded forms and use that annotation as a constraint.

3. Configure the third annotation, ProductDefect:
  - a. Expand the third annotation, ProductDefects, as shown in Figure 11-24.

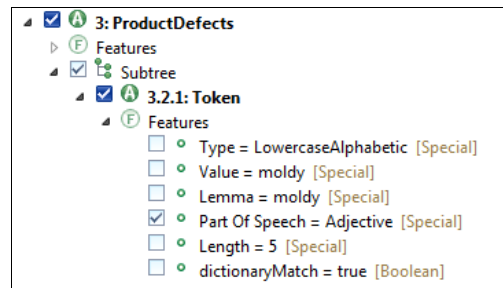


Figure 11-24 The ProductDefects annotation

- b. Select the **Part Of Speech = Adjective** feature in the Token subtree because there are both nouns and adjectives in that dictionary.

### ***Adding annotations to rule conditions manually***

The select text that is used as an example is simple: cake was moldy.

To expand the range of texts that this rule identifies, it needs to be more flexible. For example, we want it to pick up these phrases:

the cake was very moldy  
the cake really seemed moldy

We can add the Token class to accept any intervening words, and add properties to match our needs:

1. In the Selection tab, right-click the first annotation, **Product**.
2. Select **Add Annotation** → **uima.tt.TokenAnnotation** → **After selection**. See Figure 11-25 on page 432.

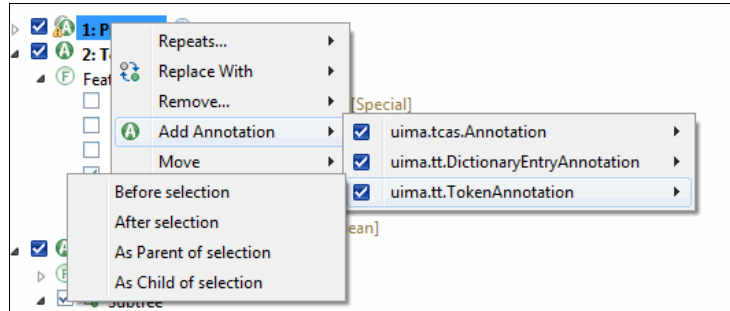


Figure 11-25 Add a Token annotation after the Product annotation

3. We want this annotation to be optional, so specify how many times this annotation can occur:
  - a. Right-click the newly added Token annotation.
  - b. Select **Repeats** → **Advanced**.
  - c. In the **Advanced Repeat** options window, select the **to** option and specify the range as 0 - 2.
4. Similarly, add an annotation after the Token annotation that has the Part of Speech = Verb feature.

Now, the annotation also identifies this sentence as a customer complaint:

The expiration date was not exceeded, but the cake already was very moldy.

We have now defined the relevant properties for our annotation. To create the annotation, we need to define the group of annotations.

### ***Defining annotations that are created by the parsing rule***

In the second tab, Annotations, the chosen attributes are displayed. This tab allows you to select annotations for grouping, adding and removing (Figure 11-26 on page 433). It is here that we can associate a product or package with a defect.



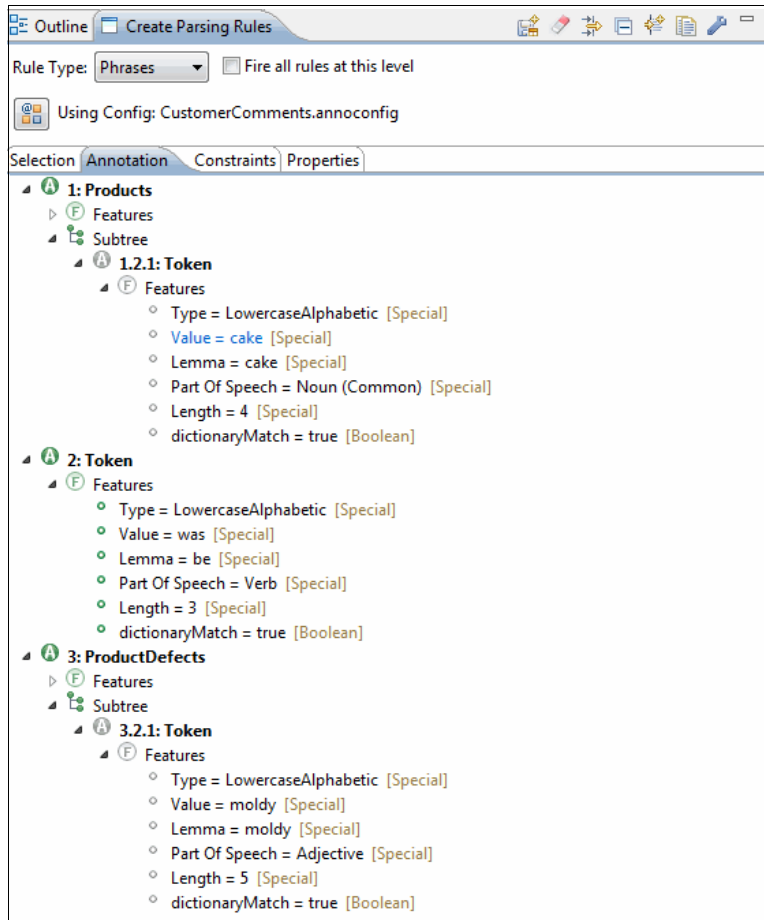


Figure 11-26 Grouping, adding, and removing annotations

After you define rule conditions in the Selection tab, you can create a new annotation.

To add a new annotation:

1. Select all the relevant annotations (press Ctrl while you click).
2. Right-click any of the selected annotations.
3. Select **Insert Annotations**. A dialog appears for naming the new annotation class.
4. Enter the name `com.food.company.ProductComplaint`.

## ***Using dictionaries and parsing rules to recognize positive comments about competitors***

Sometimes, sentiment analysis might find positive comments about a competitor. These types of comments actually need to be identified as negative comments. For example, the following comment should be identified as a negative comment:

The service in Cheapermarket is much better.

To identify positive comments for competitors (thus, then identifying them as negative comments), we need two dictionaries for our use case: The Competitors dictionary, and the IndustryAttributes dictionary.

Table 11-1 shows the Competitors dictionary. In the dictionary, we include a list of competitor names.

*Table 11-1 Competitors dictionary*

<b>Competitor</b>
Cheapermarket
Wilson's market
Super Cheap
Superfood
Food World
Willoby's

Table 11-2 shows the IndustryAttributes dictionary. It contains both negative and positive attributes. Attributes are assigned to annotations by the dictionary lookup to be later checked by parsing rules.

*Table 11-2 IndustryAttributes dictionary*

<b>Attribute</b>	<b>Negative/positive</b>
great	positive
good/better/best	positive
cheap/cheaper/cheapest	positive
wonderful	positive
like	positive
fresh/fresher/freshest	positive

After saving and compiling the dictionaries, create the Phrase Rule. Save and compile the dictionaries and add them to the CustomerComments annotator.

Create the parse rule for a new annotation, similar to the ProductComplaint annotation that you created above (see “Adding annotations to rule conditions manually” on page 431). See the preceding examples for more details.

You can add the rule to the MatchComplaint Parsing Rules database:

1. Add this text to a new test document that we will use to analyze.  
Supercheap is better.
2. Drag “Supercheap is better” in the text editor and drop it into the Parsing Rules editor. See Figure 11-27.

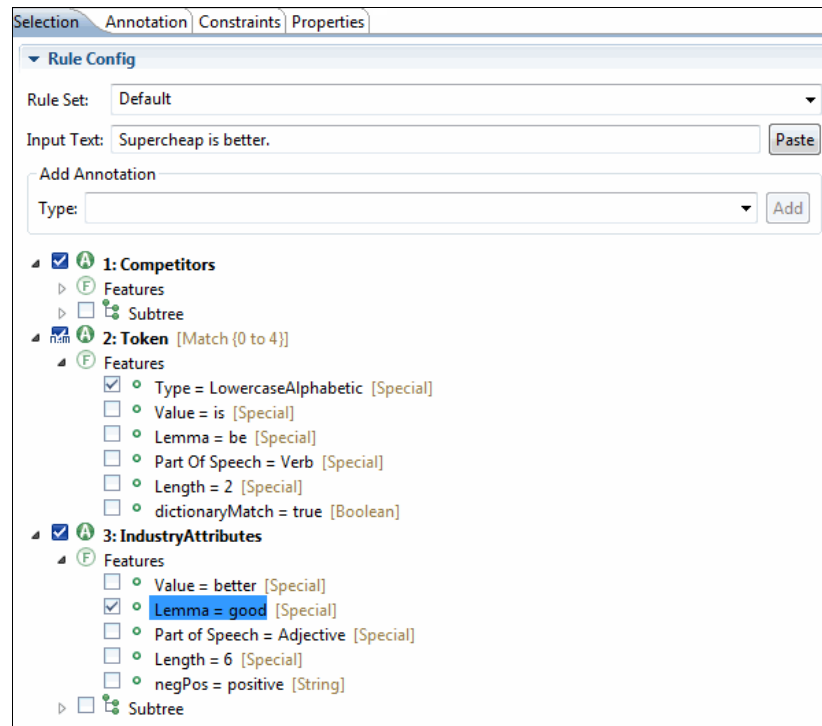


Figure 11-27 The annotator results before creating new annotations

3. Select the **IndustryAttributes** annotation and check the feature **Lemma = good**.
4. Select the **Token** and change the repetitions to 0 to 4.

- In the Annotation tab, select all the tokens and create a CompetitorCompliment annotation as shown in Figure 11-28.

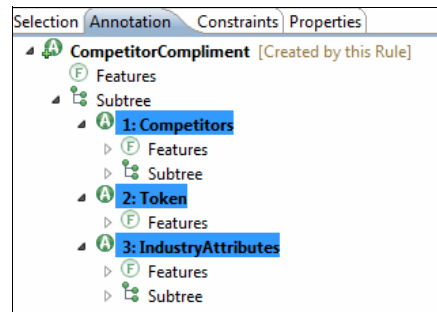


Figure 11-28 Group the existing annotations into new ones

### Testing the UIMA pipeline with the newly created resources

To test, follow these steps:

- Open the test document and enter some phrases.

For our use case, we use the following text:

The expiration date was not exceeded, but the cake already was very moldy. Cheapermarket has much fresher cakes.

- Instead of running Analyze Document, try running **Analyze Collection**:
  - Right-click the document folder in the Explorer tab.
  - Click **Analyze Collection**.
  - Select the relevant annotator.

This process analyzes *all documents in the selected folder*, and presents a different display (see Figure 11-29) in the lower pane. This method can be used for analyzing multiple documents.

Type	Offset	Context
@com.food.company.Competitors	75	cake already was very moldy. Cheapermarket is better
@com.food.company.en.CompetitorCompliment	75	cake already was very moldy. Cheapermarket is better
@com.food.company.en.ProductComplaint	46	ate was not exceeded, but the cake already was very moldy. Cheap
@com.food.company.IndustryAttributes	92	very moldy. Cheapermarket is better

Figure 11-29 Using Analyze Collection view when working with multiple annotations

The analysis now recognizes this as a customer complaint.

When there is more than one rule in a rule database, you might want to choose exactly which rules you are testing. In the Rules pane, you can select rules to be eliminated from the compilation. Figure 11-30 shows that the first rule is eliminated from the compilation. You need to recompile to eliminate the selected rules.

X	Set	Type	Label	Original Text
Rules 1 - 2				
<input checked="" type="checkbox"/>	Default	Phrases		[Products Token Token Token ProductDefects]
<input type="checkbox"/>	Default	Phrases		Supercheap is better.

Figure 11-30 Selecting rules for exclusion from the compiled rule database

**Note:** When there is more than one rule in a dictionary, the user has no control over the order of the rule execution. Therefore, if a rule depends on annotations that are created by other rules, you need to control the order by creating separate rule databases and setting the order in the annotation configuration.

### Creating more resources

The MatchComplaint Parse Rules database now includes a few rules for annotating some types of customer complaints, such as *the cake already was very moldy* (noun-verb-adjective), by using two custom dictionaries. The above workflow can be used as a basis for additional rules:

- ▶ Complaints about packaging, brands, or any aspect that is required by the business need
- ▶ Additional phrasal structures, such as adjective-noun (moldy bread) noun-verb (the package leaked)

In this section, we created an annotator from end to end, including lexical analysis, dictionary lookups, character rules, and parsing rules.

## 11.4 Exporting annotators

After we have created and tested the UIMA pipeline, you can deploy it to Content Analytics Search or Analytics server as a UIMA PEAR file.

The exported pear file is a fully compliant UIMA annotator. While exporting to Content Analytics server, you can map the annotations to fields or facets on the search server. During this process, you can create new fields or facets relevant to the new annotations.

Chapter 12, “Enterprise search” on page 445 includes details about how to configure the mapping and how to use the annotation output within the enterprise search application. In this chapter, we create the annotator used for the enterprise search collection. This annotator finds IBM brand names.

### 11.4.1 Creating the IBM Brands annotator

The custom annotator in Chapter 12, “Enterprise search” on page 445 creates an annotation for IBM brand names and is mapped to a similarly named facet. We include the instructions for building this annotator in here.

The IBM Brands annotator is based on patterns, such as ‘IBM Lotus Notes’:

- ▶ The first token ‘IBM’ is constrained by its literal value.
- ▶ The next token is a titlecase token.
- ▶ The registered trademark symbol is optional.
- ▶ The second and third tokens (titlecase and trademark) are grouped and marked as repeating.

Create a new annotator IBMBrands:

1. Create a new Parsing Rules database: FindIBMProducts.
2. Create a new UIMA pipeline using the default Lexical Analysis: IBMBrands
3. Add the empty FindIBMProducts rule database to the IBMBrands pipeline.

Create a rule for finding the IBM brand names:

1. Copy the text “IBM Lotus Notes” into the text pane and run **Analyze Document**, choosing the **IBMBrands** pipeline.
2. Copy and past the text into the **Selection** tab text box of the rule editor.
3. Delete all tokens except for the first three.
4. Adjust the resulting token properties, according to the example in Figure 11-31 on page 439.
5. Right-click the registered trademark token and select **Repeats**, and enter 0 or 1 time.
6. Select the Titlecase token and registered trademark token and group them by right-clicking and select **Group** → **Ordered**.
7. Right-click the new group and set **Repeats** and select **Repeating one or more times**.

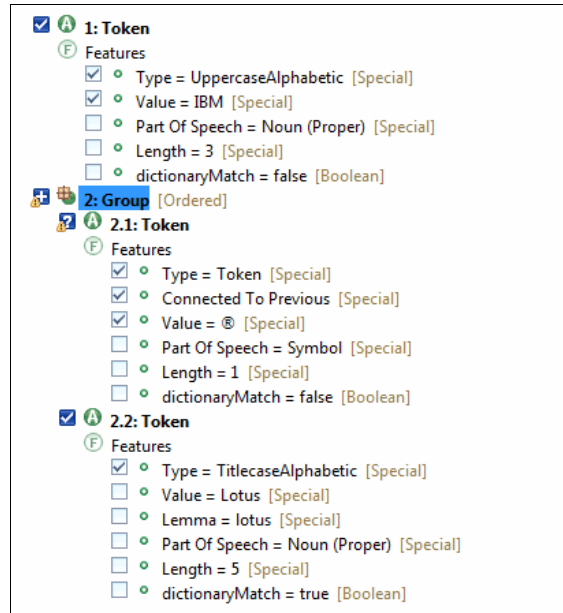


Figure 11-31 Add grouping constraints

- In the Annotation tab, select all the tokens and create a new IBMBrand annotation, as shown in Figure 11-32.

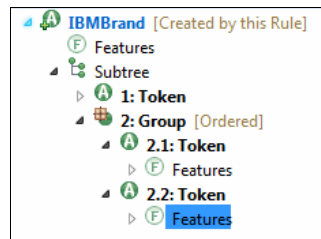


Figure 11-32 Select all tokens and create new annotation

- Save the rule and compile the rule database.

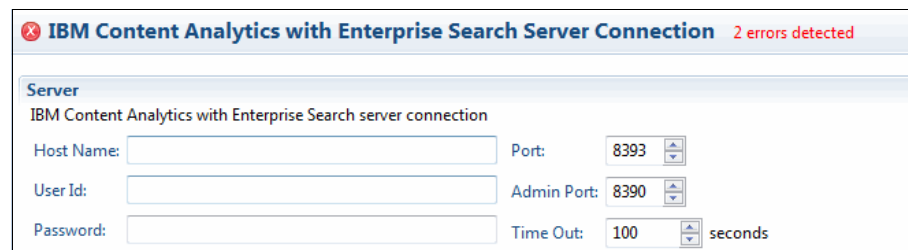
Test the new UIMA pipeline on the example text by running **Analyze Document** by using the IBMBrands annotator.

To test within the enterprise search application, the annotator needs to be exported.

## 11.4.2 Exporting the annotator

To export the pipeline to a Content Analytics installation, first create a connection configuration file to that server:

1. Right-click the **Configuration** → **ICA** folder within the Studio Explorer pane.
2. Select **New** → **IBM Content Analytics with Enterprise Search Connection**.
3. In the next window, enter a file name for the connection configuration file, for example: `ICAConnection`.
4. In the Server Connection window (Figure 11-33), enter the URL for the host name, user ID, and password information.



The screenshot shows a dialog box titled "IBM Content Analytics with Enterprise Search Server Connection" with a red "x" icon and "2 errors detected" in the top right corner. The dialog is divided into a "Server" section. Below the title, it says "IBM Content Analytics with Enterprise Search server connection". There are six input fields: "Host Name:" (text box), "Port:" (spin box with value 8393), "User Id:" (text box), "Admin Port:" (spin box with value 8390), "Password:" (text box), and "Time Out:" (spin box with value 100 and "seconds" label).

Figure 11-33 Enter connection credentials


**Note:** The error messages disappear when you enter the values.

5. To test the connection, click **Update**.  
If the connection is working, the list of collections refreshes.

To export the UIMA pipeline, follow these steps:

1. Select **File** → **Export**.
2. In the Export dialog, choose **Studio UIMA Pipeline to Content Analytics with Enterprise Search Server**.
3. Click **Next**.
4. Select the **FindIBMProducts.annoconfig** entry and click **Next**.
5. Select the connection, **ICAConnection**, created earlier, and click **Next**.
6. Select the relevant enterprise search collection and click **Next**.
7. Click **Add**. A list of all annotations that are created by the selected annotator appears.
8. Choose **IBMBrand** annotation.



9. In the next window, a list of features of the selection annotation type is shown. Choose the value that you want to associate. The following features are typically used:
  - Covered text: The annotated text itself.
  - Literal value: A constant value (value configured in the next panel).
  - lemma:key: The lemma (normalized form) of the annotated word. It is usually used for dictionary-based annotations.
10. Click **Next**. The dialog displays existing facets. You should choose one or add a new one. Figure 11-34 shows the dialog for mapping the annotation value with a field and facet:
  - Index Field Name: Choose an existing field.
  - Facets: Choose an existing facet, or click the  icon to add a facet.

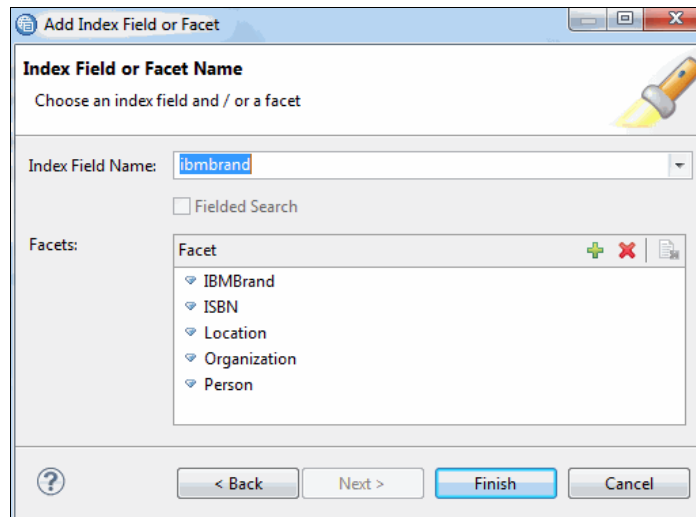


Figure 11-34 Mapping index field with a facet

11. Click **Finish**.
12. Enter the appropriate information in the dialog window (as shown in Figure 11-35 on page 442). This is the last stage before export.
  - Enter a name for the Text Analysis Engine Name field. This name must be unique on the server.
  - Select the first two options to fully automate the installation and facet mapping.

- Select the “Re-index existing collection documents” box when the collection has already been crawled. Without reindexing the collection, your PEAR output will not be included in the index.
- Select the **Restart the collection’s index service** box to populate the facets and fields in the collection.

Text Analysis Engine Name: \* FindIBMBrands

Upload PEAR file to the Content Analytics with Enterprise Search Server

Install in the custom stage of the IBM Content Analytics with Enterprise Search collection

Define required Fields and Facets on the Content Analytics with Enterprise Search collection

Restart the collection's index service

Re-index existing collection documents

Figure 11-35 Last stage configuration before import

To check that the new annotator appears in your search server, follow these steps:

1. In the administration console, go to the System tab on the top of the console window. The annotators install globally and are accessible to any collection.
2. Select **System** → **Parse** → **Configure text analysis engines**.
3. Find the annotator name in the list of text processors.

Chapter 12, “Enterprise search” on page 445 describes the installed annotator and shows how to use the new facet in the enterprise search application.

## 11.5 Conclusion

In this chapter, we build a number of resources: dictionaries, rule databases, and annotators. The goal is to show the range of resources that are available for extracting meaningful concepts from your free text content. This chapter helps you get started with ICA Studio.

We provide detailed examples of how to build simple resources. These examples can be adapted to solve your business problems according to your business domain.

The best resource for finding words and patterns is your collection data. It is always better to use dictionaries because they are easier to maintain and expand, and they are better for performance. Do not try to stretch a rule to make

it solve every problem. Complicated rules make maintenance difficult. It is important to keep rules short and simple.

Building extensive resources might not be trivial. It requires a business user who understands the business scenarios to work and improve the system by adding dictionary entries and enriching rules with added constraints. Improving these resources improves the performance of the annotators.

To re-emphasize one of the preferred practices for resource development that we mentioned earlier in the chapter, remember to repeat small steps iteratively. Start with a small document set. Create a small set of resources. Analyze and validate the results. Check to ensure that the results are as expected. If not, make appropriate changes. Continue to build and enhance the engine by extending the resources. Repeat these steps.

This iterative process continues throughout the lifetime of the solution, adapting it to the changing business needs. The iterative process takes place not only within the ICA Studio, but also over time, when the annotator is being tested on the server and even when it is in production. By examining the successes and failures of the annotator, you can formulate changes in the annotator strategy, implement them in ICA Studio, and then test them within ICA Studio and also on the server.





## Enterprise search

IBM Watson Content Analytics (Content Analytics) addresses two categories of use cases, content analytics and enterprise search. Content analytics focuses on the analysis of a set of content to find patterns, trends, and anomalies in that set. Enterprise search focuses on the discovery and retrieval of documents using various query and visual navigation techniques. These use cases share a common technology infrastructure for indexing and presenting content collections, and you can configure both solution types from within a single administration console. However, there are differences between how a user works with the information in each case. Content Analytics solutions are for a relatively small number of analysts in a typical organization who would use a number of sophisticated visualizations to find insights, patterns, and trends in the content. In contrast, enterprise search solutions are for a broad set of workers who expect a simple user interface. Content Analytics offers separate collection types and user interfaces that are optimized for both use cases.

This chapter focuses on the Content Analytics features supporting the enterprise search use case and includes the following sections:

- ▶ Overview of enterprise search capability in Content Analytics
- ▶ Use case overview
- ▶ Customizing crawling, parsing, and indexing
- ▶ Customizing runtime search
- ▶ Search Customizer
- ▶ Performing search
- ▶ Security

## 12.1 Overview of enterprise search capability in Content Analytics

The intent of enterprise search capability allows users to find and retrieve documents that contain needed information, using various query and visual navigation techniques. One of the primary challenges to achieving this goal is providing a seamless view of a broad variety of enterprise content repositories that might have this information.

Content Analytics provides a unified interface to diverse sources, structured and unstructured, by creating indexes optimized for real-time querying. Such an approach reduces the investment in restructuring, cleaning up, and maintaining multiple repositories. Search can link fields and concepts from different sources to a single search field, screen out inappropriate content and outdated records, all without requiring any changes to the original repositories.

Content Analytics also provides a wide range of techniques for adding metadata and facets for more advanced enterprise search experience. A pipeline is available for running various annotators: identifying language, basic linguistic analysis, extracting named entities (such as People, Locations, or Organizations), and for performing statistical classification according to your custom category set. Multiple repositories can be added to the index, making them accessible within a single user interface. You can also create custom applications to satisfy specific business requirements using search and analytics application programming interfaces (APIs).

Content Analytics enterprise search capability provides the following functions:

- ▶ Ability to create a single logical view (via a single enterprise search collection) over multiple physical repositories, even repositories of different types.
- ▶ Increase the accessibility of documents in repositories that have poor organization and poor metadata quality.
- ▶ Offer search term suggestions for suspected spelling errors in a user query and also provide type-ahead assistance when user enters query.
- ▶ Allow temporal searching using dates that appear in document metadata, or the document text itself. The date filtering can be accomplished via selection of specific years or months, or visual selection of a date range.
- ▶ Allow a meaning-based search by allowing users to filter based on concepts and entities extracted from documents.
- ▶ Allow users to see a summarized or detailed (including metadata) results set, and retrieve the original document from a link in the results set.

- ▶ Expand a user's search query to include more effective search terms, according to configurable triggers and actions.

Searches performed by Content Analytics enterprise search depend on the following factors:

- ▶ The fields and facets defined in the index.
- ▶ The search runtime configuration.
- ▶ The enterprise search application interface configuration.

### 12.1.1 Adding values with Content Analytics features

Content Analytics provides both content analytics use cases and enterprise search use cases. Although they are different types of solutions, there is functionality shared between the use-cases. A content analytics user might benefit by search and query features to filter a content set, and an enterprise search user might benefit by text analytics and visualizations for more sophisticated filtering.

Content Analytics features are available for both enterprise search and content analytics collections. These features include out-of-the-box annotators for extracting dates, names, and places from the free-text part of your content. Dictionaries allow you to flag synonymous groups of words. The IBM Content Analytics Studio (ICA Studio) developer environment allows you to build custom annotators (based on the UIMA standard) for generating new metadata based on word and phrase combinations found in each document. In addition, the use of built-in clustering or integration with the IBM Content Classification product allows you to assign categories to documents based on statistical analysis.

The output of these text analytics techniques in a search solution is new metadata, available in facets for querying and visual filtering. These new attributes provide novel ways to find “the needle in the haystack” beyond what a basic search solution might provide. As needed, new annotators can also be implemented to extract additional content features, and provide more refined filtering.

For example, how to answer this question: “Where is that presentation that Joe delivered to a banking customer last year regarding new features of DB2?” Standard search capabilities get us close to our goal. We can search for the word “DB2”, and look for docs of type “PPT”. However, this might produce a results set too large to be practical.

How would we further refine the search to include only documents relevant to a “banking customer”? We could manually search for every bank name and banking term we could think of. A better approach would be for an ICA Studio

annotator or Content Classification Knowledge Base (KB) to determine which *Industry* is relevant to a document. It could do that using dictionaries of company names tied to different industries, or industry-specific terms. The *Industry* category could be added as a facet, which a user could easily select to filter out documents for “Banking” or “Utilities” or “Retail”, and so on.

Finally, how do we narrow down our results to include only documents from “last year”. Some search tools could do this based on document metadata. But we might also want to look for date information about the first slide of the PPT document, which might be a more reliable indicator of when the presentation took place. We could extract that data using another ICA Studio annotator. Then, this date could be displayed within the new timeline view widget in the enterprise search application interface, where a user could select and filter on documents from the last year, or last 18 months.

In this way, Content Analytics enables searching using document features beyond the basic text and metadata. Concepts and entities can be surfaced automatically, and provided to the user for filtering and querying.

### 12.1.2 Enterprise search application user interface

Content Analytics provides an enterprise search application for submitting queries to the search server. The most useful and important capabilities of this application are the various widgets displayed once a search is made and the search results are displayed (see Figure 12-1 on page 449). Once a user with little experience or knowledge of the content submits a simple search, consisting even of a single word, these widgets display a detailed view of different features represented within the results set: Distribution by date, category, and any other defined facets.

These components not only display a meaningful overview of the set, they are also interactive devices for drilling down and up into the set, refining or generalizing the results according to the properties chosen by the user. Each component lists the document count per element and allows you to add terms to the drill-down search.

By exposing to the user lists and trees of existing facets and values, it allows users who are not familiar with the content or defined fields, to access the content according to the facet design and available content.

The widgets that we configure in this chapter are:

- ▶ **Facet Tree:** Provides a list of elements that are derived from document metadata values, or text analytics annotators.
- ▶ **Category Tree:** Displays categories that are defined by configurable rules.



- ▶ Dynamic Facet Chart: Displays date ranges.
- ▶ Time Series Chart: Provides a timeline view that can be expanded and contracted: by year, month, or day.

Figure 12-1 shows some sample views of these components.

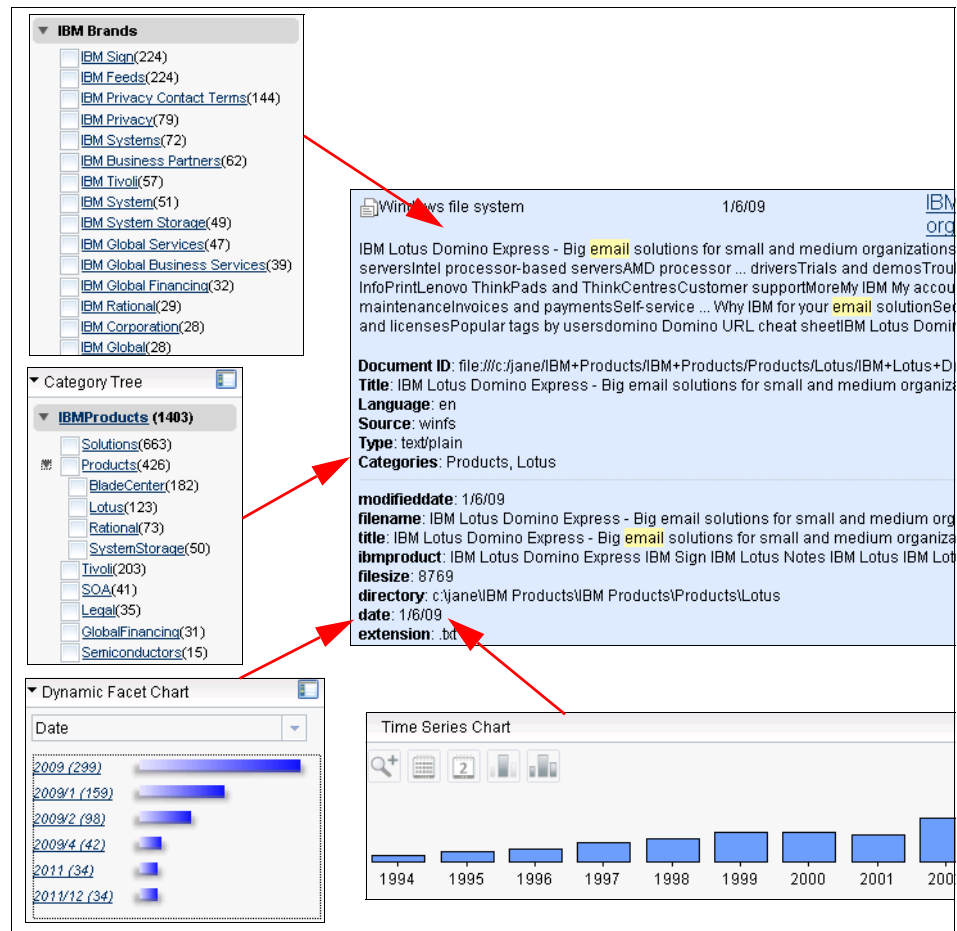


Figure 12-1 Some examples of the available widgets

### 12.1.3 Components supporting Content Analytics enterprise search capability

Content Analytics enterprise search capability depends on three components:

- ▶ Enterprise search application: Handles the search queries from users and sends them to the search server.



Figure 12-4 shows how the administration console works with other components. The administration console accesses both the controller server (to configure the crawler and index configuration) and the search server (to access and update the search configuration). The controller server runs the crawler, document processing, and builds the index. The search server accesses the index to process search queries. The enterprise search application sends and receives requests from the search server. The enterprise search application can be Content Analytics out-of-box default enterprise search application or custom search application.

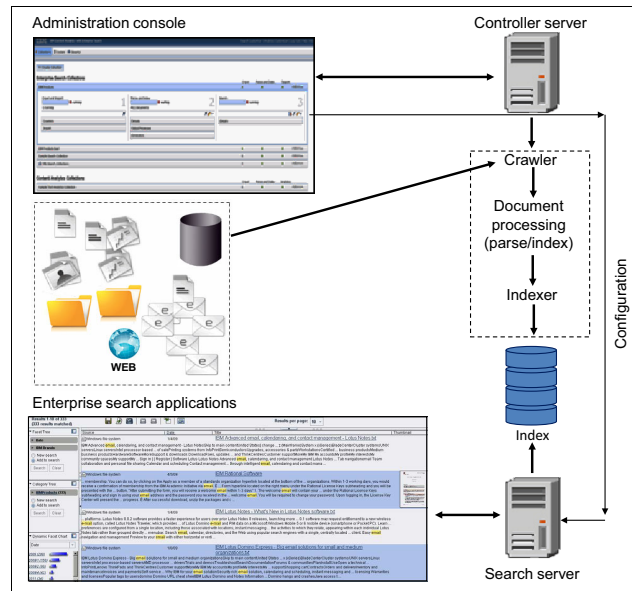


Figure 12-4 Administration console working with controller and search servers

## 12.1.4 REST Search API

Content Analytics provides Representational State Transfer (REST) Search API that you can use to create a custom enterprise search application. The REST Search API supports all the functions provided by the Content Analytics enterprise search application including search query expansion, document ranking, and facets.

For information about using the API, see the following references:

- *IBM Content Analytics with Enterprise Search Version 3.0, Programming Guide, SC19-3348*

- ▶ Sample programs in the `<installation directory>\samples\rest` directory
- ▶ Javadoc information in the `<installation directory>\docs\api\rest` directory

## 12.2 Use case overview

The use case used in this chapter describes how to build an enterprise search solution. Specifically, the use case is used to show you how to perform the following tasks:

- ▶ Customizing crawling to crawl content from multiple sources.
- ▶ Customizing parsing and indexing to map multiple content sources to common indexes, to add additional facets through custom annotators, to use category tree, and to alter field values to conform to uniform standard.
- ▶ Tuning queries and search results using boost factors, document importance field, document ranking filter, query expansion, and aggregate rules.
- ▶ Customize the search interface to include Person facet, Timeline, and Category Tree widgets.

For the use case, we use a content set that consists of two content sources of different types:

- ▶ Windows file system folder containing around 600 files existing within 10 file system folders.
- ▶ The website containing approximately 10k pages from the IBM Redbooks website.

We combine these two sources by crawling each with a relevant crawler (one for the file system and one for processing the HTML request to the website). The two sources are then combined in a single index.

The goal is to extract information from these files that are not directly available, use some of the fields to link the content logically, and use these features to guide search strategies that are designed specifically for this content.

Using this combined collection, we demonstrate the capability of enterprise search and show how to configure the enterprise search application to get the results that you want.

In the use case example, we performed the following steps:

1. Customize unified date and title fields and display it in the enterprise search application.

2. Combine these two content resources into a single category tree by using rules that are appropriate to each content source.
3. Add a custom annotator for extracting IBM Brand names.

The following sections show the detailed steps that are involved.

## 12.3 Customizing crawling, parsing, and indexing

The three basic stages of the enterprise search application definitions are:

- ▶ Crawling
- ▶ Parsing and indexing
- ▶ Search

Content Analytics is capable of integrating multiple repositories into a single enterprise search solution. These sources are crawled and then included within a single index. Depending on your repository type, various fields can be mapped directly from the source repository. During the parsing phase, annotators can add additional fields and facets. The enterprise search application then processes the user's query, expands it, and ranks the search results.

This section focuses on crawling, parsing, and indexing. In 12.4, "Customizing runtime search" on page 462, it focuses on customizing run time search configuration.

### 12.3.1 Setting up multiple content sources for crawling

Each repository type has its own crawler type. The number and nature of the extracted fields depend on the degree of structure in the repository. Each crawler type is adapted to its repository type and extracts the maximal amount of structured data, in addition to any document content, in order to make it available to the parser and indexer.

For our use case, we set up two crawler types: File system and web crawlers.

#### Setting up a web crawler

For an explanation about how to set up a web crawler, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iiysacweb.htm>

For our use case, we crawl the IBM Redbooks site. To set up the crawl space for a web crawler, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/index.jsp?topic=%2Fcom.ibm.discovery.es.ad.doc%2Fiysacweb.htm>

After the crawler is created, you can access the crawl space configuration:

1. From the Crawler pane, click the **Pencil** icon, and select **Edit Crawl Space**.
2. In the text box, enter the starting URL for the crawler to crawl (or any relevant and available URL):

<http://www.redbooks.ibm.com/cgi-bin/searchsite.cgi?query=analytics>

### Setting up Windows file system crawlers

For an explanation about how to set up a Windows file system crawler, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iiysacwindist.htm>

To set up the crawl space for the sample data, use the following steps:

1. In the “Select Windows Subdirectories to Crawl” dialog box, add the root folder of the sample data.
2. Click **Next** and **Finish**.

## 12.3.2 Mapping multiple content sources to the index

After the content is crawled, the parsing and indexing stage detects fields in the crawled data. These include fields that existed in the source repository, such as file names, database fields, and HTML metadata. These fields can then be mapped to facets, or joined to form common fields, such as the Title metadata fields from Microsoft Word and FileNet title field.

If you crawled against multiple content sources, you want to map multiple sources to a common index for unified search among multiple sources.

Depending on the crawler type, a number of metadata becomes available for mapping to the search index fields. The Windows file system crawler, for example, can provide basic information about the file including folder name, file name, creation date, modification date, and possibly user access information. Some document formats include metadata, such as Microsoft Word documents and PDF files. These may include Title and Author fields.

As more documents from multiple repository types are mapped to common fields, users can access the content more freely. This allows the organization to view content by more logical parameters such as content similarity.

By default, Content Analytics maps a few basic content elements to index fields. By coordinating these field mappings, we can better organize multiple sources.

In the following two subsections, we discuss mapping Windows file system properties and HTML metadata to index fields.

## Mapping Windows file system properties to index fields

Look at the field mapping for the Windows file system crawler that you created:

1. In the Crawlers pane of the collection, select the **Pencil** icon in the Windows file system crawler that you created.
2. Select **Edit Crawl Space**.
3. Click **Edit Metadata**. Figure 12-5 shows the list of Windows file system mappings.

Crawl Name	Index field name
<input checked="" type="checkbox"/> __\$ArchiveCharset\$__	[default]
<input checked="" type="checkbox"/> __\$ArchiveType\$__	[default]
<input checked="" type="checkbox"/> __\$Date\$__	date
<input checked="" type="checkbox"/> __\$Directory\$__	directory
<input checked="" type="checkbox"/> __\$EntryDate\$__	[default]
<input checked="" type="checkbox"/> __\$EntryName\$__	[default]
<input checked="" type="checkbox"/> __\$EntrySize\$__	[default]
<input checked="" type="checkbox"/> __\$Extension\$__	extension
<input checked="" type="checkbox"/> __\$FileName\$__	filename
<input checked="" type="checkbox"/> __\$FileSize\$__	filesize
<input checked="" type="checkbox"/> __\$ModifiedDate\$__	modifieddate
<input checked="" type="checkbox"/> __\$Title\$__	title

Figure 12-5 Assigning the field names for the available file system properties

As you can see from Figure 12-5, not all crawled documents provide values for all the fields. The crawler extracts metadata from Microsoft Word files and PDF files with the \_\_\$Title\_\_ field, but no such metadata is available for text files (.txt), for example. In the case of the title field, the crawler can provide the file name value for this field where no other appropriate metadata is available.

To set the file name of any file as the \_\_\$Title\_\_ field, follow these steps:

1. Click the **Pencil** icon for the relevant crawler in the Crawler pane, and select **Edit crawl space**.

- In the list of subdirectories, select a relevant directory and select **Edit Options**.
- Select the **Use the file name as the document title** check box.

## Mapping HTML metadata to index fields

The HTML <meta> tag provides customizable metadata that can be added to any HTML document.

In the sample HTML data for our use case, Figure 12-6 shows the tags that are available. The HTML metadata elements can be crawled and mapped to the search index.

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
<link rel="schema.DC" href="http://purl.org/dc/elements/1.0/" />
<link rel="SHORTCUT ICON" href="http://www.redbooks.ibm.com/favicon.ico"/>
<meta name="DC.Rights" content="© Copyright IBM Corp. 2011"/>
<meta name="Source" content="v17 Template Generator, Template 17.02"/>
<meta name="Security" content="Public"/>
<meta name="DC.Language" scheme="rfc1766" content="en"/>
<meta name="IBM.Country" content="ZZ"/>
<meta name="Owner" content="Redbooks at IBM/Raleigh/IBM"/>
<meta name="Keywords" content="IBM Redbooks discussions, comments">
<meta name="DC.Date" scheme="iso8601" content="2011-02-24">
<meta name="IBM.Effective" scheme="W3CDTF" content="2011-02-24">
<meta name="DC.Subject" scheme="IBM_SubjectTaxonomy" content="SS6MTS, SSQH9M, SSRM8N, SS...
<meta name="DC.Type" scheme="IBM_ContentClassTaxonomy" content="CT300">
<meta name="Robots" content="index, nofollow">
<meta name="Description" content="This IBM® Redbooks® publication describes how the IBM
<meta name="ISBN" content="073843518X">
```

Figure 12-6 HTML tags that are available in the HTML documents for our use case

In this example, we add the following metadata to the index:

DC.Date  
Description  
ISBN

To map the HTML metadata to a search field, follow these steps:

- On the Parse and Index pane of your collection, click the **Pencil** icon.
- Select **More** → **Index Field Mappings for HTML metadata elements**. Figure 12-7 shows the mapping possibilities.

HTML metadata element name	Field name
DC.Date	date
Description	description
ISBN	isbn

Figure 12-7 Mapping of HTML metadata to indexes

- Map the relevant HTML metadata to their corresponding index fields.



For information about adding index fields, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iysasrchflds.htm>

In addition, the web crawler looks for one more element to add to the index: the `<title>` element. For example, a document with the following element will have the value appear in the `title` index field:

```
<title>IBM Redbooks | Reader Discussion of Patterns: Integrating  
WebSphere ILOG JRules with IBM Software</title>
```

This is a default behavior of the crawler and cannot be configured.

We now have a number of common fields to link the two collections.

### 12.3.3 Adding additional facets and fields with a custom annotator

The success of your enterprise search solution depends on the richness and appropriateness of your index entries and structure. At the parsing and indexing stage, not only fields that existed in the source repositories can be mapped to facets, or joined to form common fields, you can also extract additional facets and fields through the parsing of the unstructured parts of the content.

In the Document Processing pipeline, you can add a custom annotator that extracts new facets and fields from your enterprise search collections.

The set of available annotators can be configured by accessing the Parse and Index pane in the collection.

Content Analytics provides the following annotators for the document processing pipeline to support the enterprise search capability:

1. Language Identification annotator
2. Linguistic Analysis annotator
3. Named Entity Recognition annotator
4. Content Classification annotator
5. Custom annotator

Content Analytics provides similar annotators in the document processing pipeline to support the content analytics miner:

1. Language Identification annotator
2. Linguistic Analysis annotator
3. Dictionary Lookup annotator (unique for content analytics miner)
4. Named Entity Recognition annotator
5. Pattern Matcher annotator (unique for content analytics miner)
6. Content Classification annotator

## 7. Custom annotator


The Language Identification and Linguistic Analysis annotators are default annotators that cannot be changed or configured. The output is used internally.

The Name Entity annotator extracts personal, place, and organization names. For information about how to customize this annotator, see Chapter 7, “Performing content analysis” on page 231. Select this annotator if you want to add name, place, and organization annotations to your enterprise search collection.

The Content Classification annotator uses IBM Content Classification to categorize unstructured data based on statistical methods. See Chapter 9, “Content analysis with IBM Content Classification and document clustering” on page 305 for information about the annotator and how to use it for content analysis.

The Custom annotator can be used to extract additional fields and facets information such as those for domain-specific content. See Chapter 11, “Customizing content analytics with IBM Content Analytics Studio” on page 405 for information about how to build a custom annotator (a UIMA annotator) with ICA Studio and install it to the search server.

Follow these steps to check what facets are available:

1. From the Parse and Index pane, click the **Pencil** icon and choose **Facets**. You can see the new facet being listed.
2. To check the field linked to the facet, click the  icon.

These facets are added automatically to the enterprise search application. For more information, see 12.5, “Search Customizer” on page 476.

### 12.3.4 Adding a category tree

The category tree is a component that can be displayed in the enterprise search application. The tree node values can be based on a range of values defined in category rules. When a search result set is returned, the nodes represented in the set are displayed. These nodes can be used to add additional parameters to the search query, guiding the user through a drill-down search.

The category tree usually represents an overall view of the content. Ideally, each entry in the content set should be represented by at least one node in the tree.

For our use case, the file system content in the collection resides in a file system folder hierarchy. This hierarchy is used as the basis for the category tree.

Figure 12-8 shows the Category Tree widget as it displays in the enterprise search application.

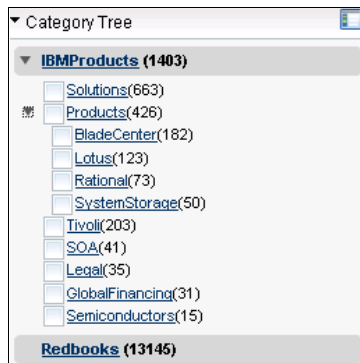


Figure 12-8 The Category Tree widget

To create the category tree:

1. In the Parse and Index pane, click the **Pencil** icon.
2. Select **Global Processing** → **Rule Based Categories**.
3. Select the root node in the left pane and enter the node name **IBMProducts** in the Category Path and Category Name boxes.

**Note:** Most categories contain at least one inclusion rule. Since this node contains only subnodes and no documents, it does not require a rule.

4. Create the first subnode:
  - a. Select the **IBMProducts** node and enter the first subnode category path and name: `Global Financing`.
  - b. Click **Add**.
  - c. Add the rule for this node by clicking the **Pencil** icon next to the Rules box.
  - d. Add the rule name: `Global Financing`.
  - e. Add the URI: `IBM Global Financing`.

You can choose to add a full or partial file path. If you use only a partial path, such as 'Financing', a document may match anywhere in the file path and therefore be assigned more than one category. This value is used to match a substring in the full path, including the file name.

Figure 12-9 on page 460 shows how to add a URI pattern to a category rule.

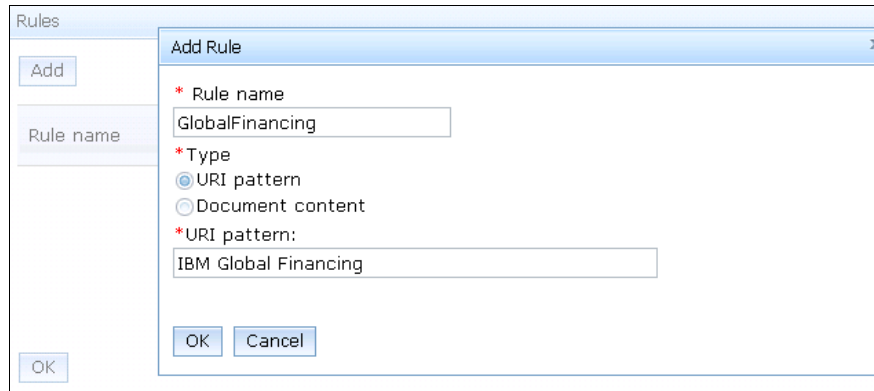


Figure 12-9 Configuring a category based on a file system path substring

5. Click **OK**.

Similarly, add all category nodes, which are based on the folder name of the source file.

**Note:** When you add a node, ensure that the relevant parent mode is selected in the left pane.

To add the crawled web content to an additional node, create a new category directly underneath the root. The rule for this node is based on a substring of the webpage URI. You can add all documents under a single node called Redbooks. See Figure 12-10.

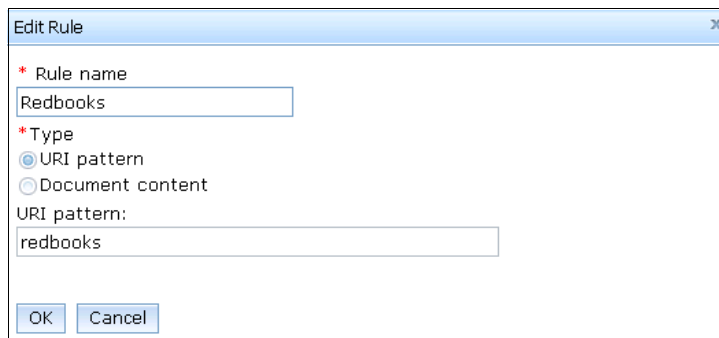


Figure 12-10 Configuring a rule-based category based on a URL substring

Multiple rules can be added to a single category. If a document triggers at least one rule, it is included in the category.

Figure 12-11 shows the complete category tree configuration.

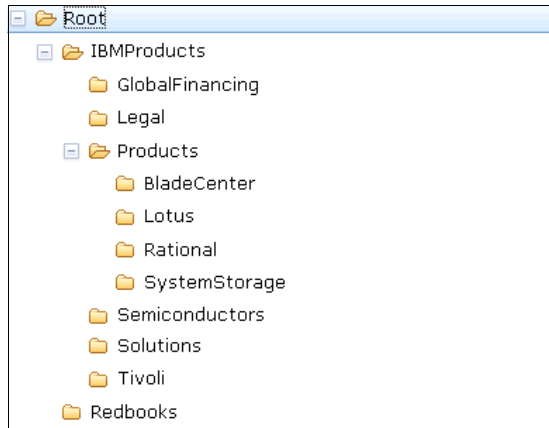


Figure 12-11 The category tree displayed in the tree editor

### 12.3.5 Altering field values to conform to uniform standard

The Parse and Index pane includes a helpful set of tools for changing field values in order to conform to a uniform standard, such as remove unwanted characters, copy fields, and split fields.

You can alter the field content by using field filters:

1. In the Parse and Index pane, select the **Pencil** icon to edit.
2. Click **More**.
3. Click **Field Filters**.

The filters include a number of options for manipulating fields and their values, as shown in Figure 12-12.

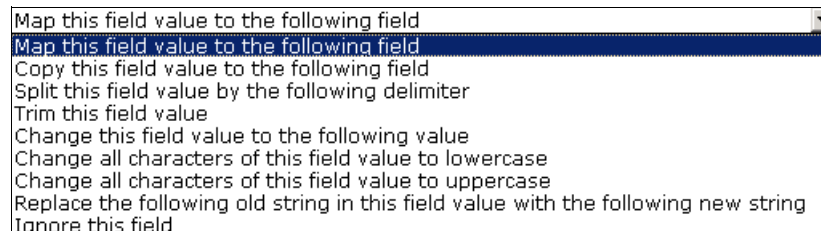


Figure 12-12 Options for altering field values

4. Select the option that you want for the field filter from Figure 12-12.

For more information about field filters, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iiysafilter.htm>

For our use case, we modified and added a number of fields and facets to the enterprise search collection.

The administration console contains one addition pane: Search. This pane configures how the search server processes submitted queries. Later, we complete the design by configuring how the fields and facets added in this section are to be in the enterprise search application.

## 12.4 Customizing runtime search

The Search panel of the administration console provides configurations that control the runtime behavior. These are relevant to both the enterprise search application and to any custom application created with the Content Analytics search REST API. For information about the REST API, see Chapter 14, “Customizing and extending the content analytics miner” on page 535.

### 12.4.1 Tuning queries and results

Good search results are measured not just by what is returned in the results set, but it is also measured by how easily a user can find items of interest in the result set. With a typical keyword search, the results set may contain hundreds to thousands of items. When you use rules to effectively configure search query behavior and organize search results, you help users to better formulate their search and browse through the results.

There are several ways that you can use rules to tune queries and results:

- ▶ **Ranking:** Controls the order in which items are listed in the results set. This helps to bring more relevant search results to the users’ attention.
- ▶ **Query expansion:** Enables additional terms to be added to the search query, thus expanding the results set. In this way, words (synonyms) that are not chosen by users, but occur in the text, can be included in the query.
- ▶ **Aggregate:** Enables the logical grouping of the search results.

Query expansion, ranking, and aggregation are all interrelated and the proper combination of them can be very powerful. For example, the ranking rules decide the order of the returned items based on which fields contain the search terms. The query expansion rules can also influence ranking. The aggregate rules

decide what grouping includes which results. These rules directly effect how the search results are presented to users and therefore they influence how easy or difficult it is for users to navigate through a result set especially if it is a large one.

The enterprise search application uses these tuning rules to add additional search query information and display relevant search results. Custom applications can use API to get and configure this information based on the application design.

Factors influencing the sorting order of a search result set include boost factor for a field, for a URI pattern, document importance setting, document ranking filter, query expansion rules, and aggregate rules.

### **Configuring the boost factor for fields**

Each document field can be configured to add a relative boost factor. This affects the search result that is based on these fields being matched.

To set the boost factor of a specific field:

1. In the Parse and Index pane of the administration console, choose the **Pencil** icon.
2. Select **Index Fields** and click the **Pencil** icon for the relevant field.
3. Select the **Fielded Search** check box and set the boost factor.

Figure 12-13 on page 464 shows the window where you edit an index field setting.

### Edit an Index Field

[Learn more](#)

You can change how this index field is defined or view which crawlers and facets are using this index field.

\* Field name:

filename

Returnable  
Shows the value of this field in the search results.

Free text search  
Enables this field to be searched with a free text query.

Document summary  
Shows the value of this field in the search result summary.

Fielded search  
Enables this field to be searched by field name.

Exact match  
Enables this field to be returned only when the query terms exactly match the field value.

Case-sensitive  
Preserve uppercase and lowercase characters in the query terms.

Boost factor  
Boost the ranking of documents that match this field. The valid range of values is 0 to 100.

Boost factor:

1.0

Return spelling suggestions from this field  
Show spelling suggestions from this field in the search results.

Fielded Search Expansion  
Expand free text terms to fielded terms automatically. To enable this option, select Fielded search and Free

Figure 12-13 Edit an index field

This boost factor will now be considered among the other factors for determining sort order of the search results.

## Configuring the boost factor by URI pattern

The boost factor can also be based on a URI pattern. To set the boost factor by URI pattern:

1. In the Search pane of the administration console, choose the **Pencil** icon.
2. Select **URI patterns for influencing scores**.
3. Click **Add URI pattern**.
4. Add the pattern and a negative or positive boost score percentage.
5. Click **OK**.

## Setting the document importance ranking method

Another way to affect search result order is the document Importance setting. This is a content collection-wide setting that is accessible from the Search pane. To set the document importance setting:

1. In the Search pane of the administration console, choose the **Pencil** icon.



2. Select **Document importance**.
3. In the Document importance box, select the ranking method. Figure 12-14 shows the options that are available for ranking document importance.

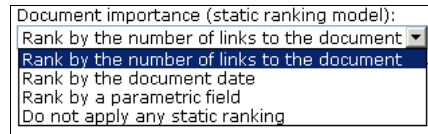


Figure 12-14 Choosing the document ranking factor

4. Save the configuration.

## Setting document ranking filters

Document ranking filters are based on the varying importance of fields. The shorter title or subject fields typically include highly relevant words, whereas the body or abstract fields might contain a large range of less relevant words. If a word appears in the title field, it is probably very relevant. The ranking filters allow you to define the relative importance of the fields.

Figure 12-15 shows an administrator configuring ranking filters used at run time. In this case, the documents are ranked by the matching fields.

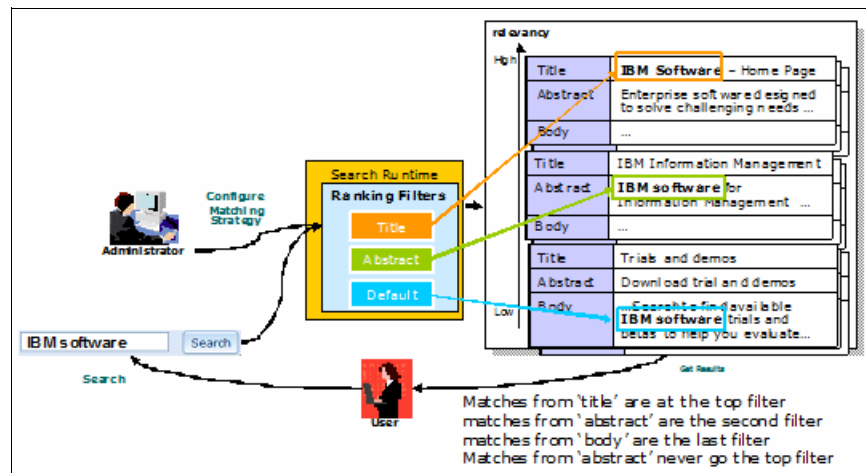


Figure 12-15 The administrator configures ranking filters used at runtime

We cover document ranking in more details in 12.4.5, “Ranking the search results” on page 475.

## Configuring query expansion to influence document ranking

Document ranking filters are triggered by query expansions. The query expansion rules add similar terms or synonyms to the search query to expand the range of results. Result items added by the expansion can be ranked as being higher such as the case of preferred terms. They can also be ranked lower than the results that would be returned by the original query.

We cover query expansion in more details in 12.4.3, “Expanding the search query” on page 466.

## Aggregating rules to influence search results orders

Aggregate rules by creating groupings that can be prioritized. Prioritized groups are listed first in the search results, thus affect the search result orders.

We cover aggregating rules in more details in 12.4.4, “Grouping results” on page 473.

## 12.4.2 Enhancing free text search to use field search

Users commonly use free text searches, especially when they are not familiar with the index fields. It is therefore helpful to define fields that will be searched when a free text search is submitted. A recommended practice is to search the most meaningful free text fields (such as the document content and title field) when a free text search is made.

To add a field to the free text search:

1. In the Parse and Index pane of the Administration console, click the **Pencil** icon.
2. Select **Index Fields**.
3. Click the **Pencil** icon for the relevant field that you want to use in the free text search.
4. Select the **Fielded Search Expansion** check box.
5. Save the configuration.
6. Reindex the collection so that this setting can take effect.

## 12.4.3 Expanding the search query

Users perform searches by using terms that are familiar to them. Yet these query terms might not be the same terms appearing in documents. To ensure that documents with similar terms are included in search results and that users get relevant hits, the enterprise search application provides a number of options for

defining synonyms or abbreviations that can be used to expand the search queries and thus helping users to get better search results.

To expand the search query, you can create rules to add or replace original search terms with preferred terms. Each rule has two parts:

- ▶ **Matching rule:** A trigger that looks for keywords in a user's search query, according to different methods.
- ▶ **Expansion rule:** An action that replaces or supplements the matched words.

## Adding expansion rules

To add expansion rules, follow these steps:

1. From the Search pane of the administration console, click the **Pencil** icon and select **Rules to Tune Queries and Results**.

The first pane is Rule-Based Query Expansion, as shown in Figure 12-16.

The Rule-Based Query Expansion dialog box allows you to configure query expansion. Rules can include keywords, or refer to dictionaries for lookup.

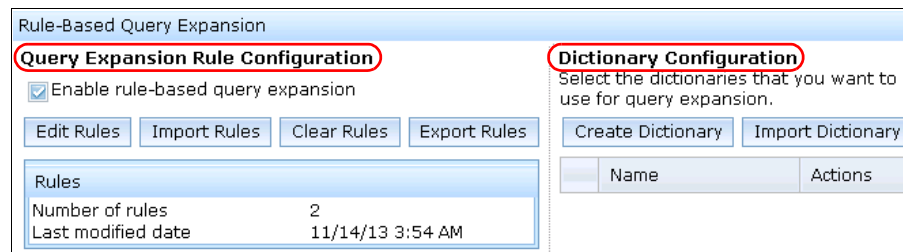


Figure 12-16 The Rule-Based Query Expansion dialog box

2. Click **Edit Rules**.

Figure 12-17 on page 468 shows the two parts of the rule edition dialog.

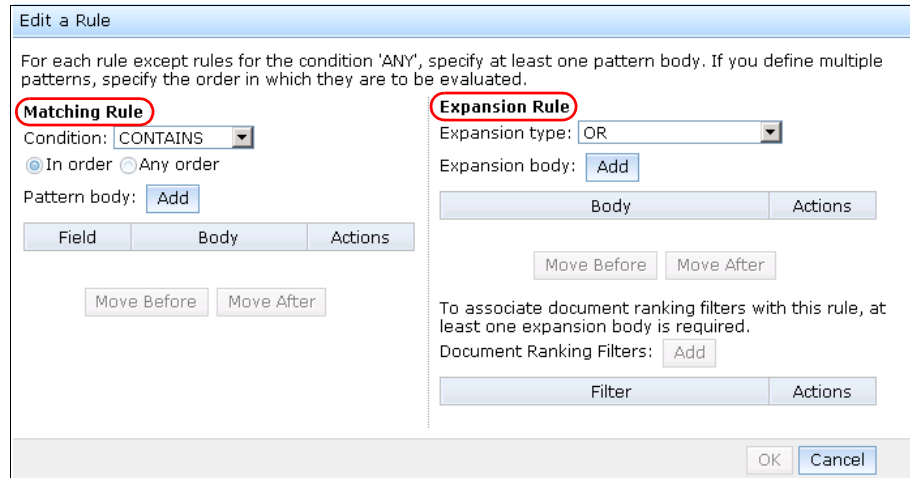


Figure 12-17 Editing the match (trigger) and rule (action)

3. To add a matching rule, click **Add** from the Matching Rule pane.
4. To add an expansion rule, click **Add** from the Expansion Rule pane.

For details of editing the matching rules and expansion rules, see the following sections.

## Editing matching rules

You can edit the matching rules that specify the condition of the matching rule, the matching order, match pattern type, and pattern body.

To edit a matching rule, follow these steps:

1. Select the condition for the matching rule.

When a rule attempts to match words in a query, it can apply strict or lenient condition constraints. Figure 12-18 shows different matching conditions that you can use for a matching rule.

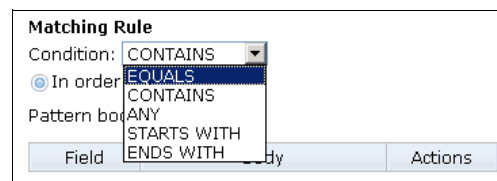


Figure 12-18 Editing the matching rule condition

2. Select the pattern body type.

At run time, Content Analytics finds alternative search terms by using dictionary lookup, regular expressions, or keywords that are defined directly in the rule. To specify what type to use at run time for the matching rule, select one of the pattern body types as shown in Figure 12-19.

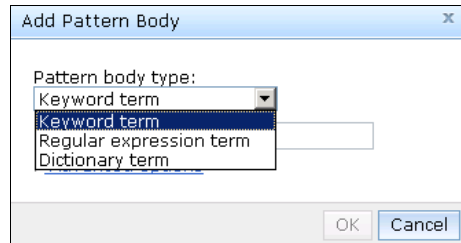


Figure 12-19 Choosing the pattern type for matching

3. If you select **Keyword term** in the previous step, enter the keywords for matching.
4. If you select **Dictionary term** in the previous step, select the dictionary for synonyms and abbreviation lookup.

In the Rules to Tune Queries and Results pane (Figure 12-16 on page 467), the Dictionary Configuration pane lists the dictionaries that are available for query expansion. Each term that is listed in a dictionary is associated with a number of synonyms, abbreviations, or acronyms. Each dictionary entry that appears in the search query is supplemented by all the aliases for that entry.

The dictionary lookup also determines the expansion values.

Figure 12-20 shows the panel where you create a dictionary. Within the panel, you can add a word and its alias.

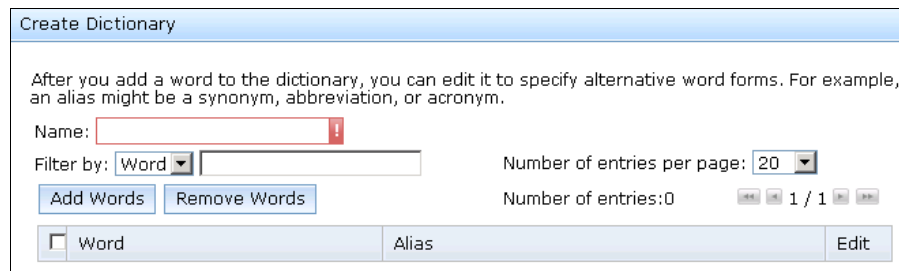


Figure 12-20 Creating a dictionary where you can create an alias for words

5. If you select **Regular expression term** from the previous step (in Figure 12-19), enter the regular expression term for the matching rule. The rule attempts to make a match according to the defined regular expression. If a match is made, the expansion rule is fired.

## Editing expansion rules

After a rule is triggered by a matching rule, the expansion rule controls the behavior of the search query expansion.

To edit an expansion rule, use the following steps:

1. Select the Expansion type in the Expansion Rule pane.

The expansion type determines how rules are expanded. You can select from one of the following types:

- Replace: The matched string is replaced by the expansion string.
- Or: The matching string is added to the expansion string.
- New over original: The matching string is added to the expansion string and scores higher than the original search string in the results order.
- Original over new: The matching string is added to the expansion string, but is scored lower than the original search string in the results order.

2. Click **Add** next to Expansion body text in the Expansion Rule pane.

3. Choose an expansion value.

The expansion value can be one of the following:

- Dictionary lookup: From the dictionary named in the match rule.
- Keyword: Defined in the expansion rule.
- Rewrite: Specific terms are replaced by keywords

If you select **Keyword**, enter a keyword value. Figure 12-21 on page 471 shows a keyword expansion: email → lotus.

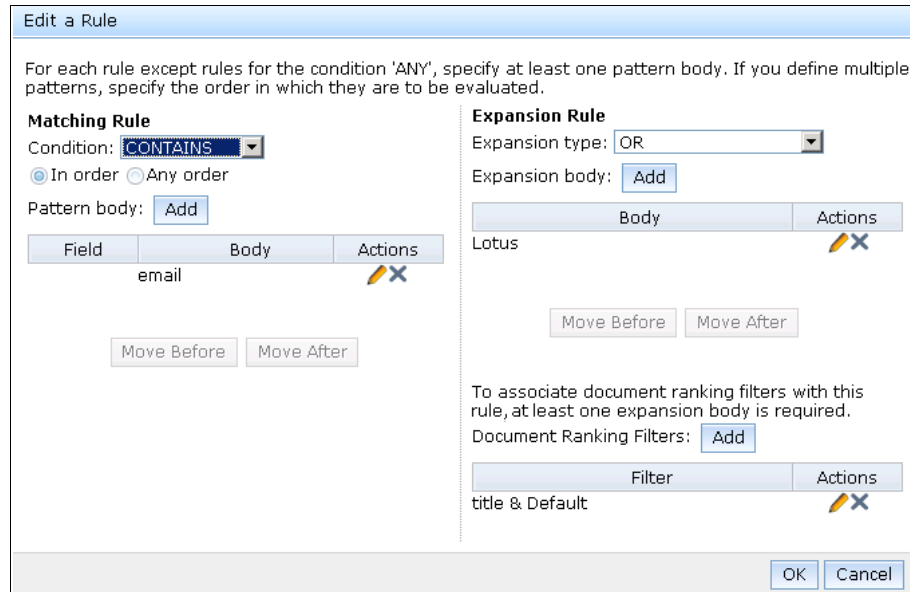


Figure 12-21 Keyword expansion

## Rewriting rules

The rewrite rules can replace parts of a query string according to regular expression matches. For example, if we know that there are websites that are called 'java technicalpage' and 'dojo technicalpage', in the document, but users commonly look for 'java site' or 'dojo site', we can replace 'site' with 'technicalpages' when they appear in these contexts.

Figure 12-22 shows the rewrite for this example.

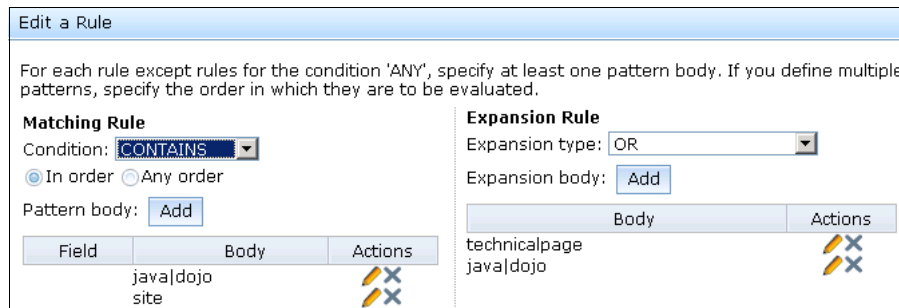


Figure 12-22 Rewriting java or dojo site to java or dojo technicalpages

To create the rewrite expression rule, follow these steps:

1. Create a regular expression, `java|dojo`, for the matching rule as shown in Figure 12-22 on page 471:
  - a. From the Matching Rule pane, click **Add**. The Add Pattern Body dialog box appears as shown in Figure 12-23.

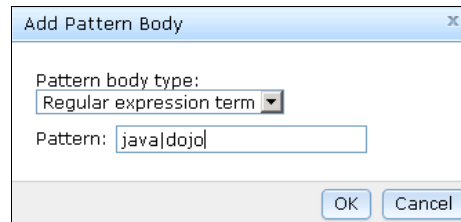


Figure 12-23 Defining the match pattern type

- b. For the Pattern body type field, select **Regular expression term**.
  - c. For the Pattern field, enter a value, `java|dojo`
  - d. Click **OK**.
2. From the Matching Rule pane, create a matching rule for the keyword 'site'.
  3. In the Expansion Rule pane, add `technicalpage` as a rewrite rule:
    - a. Click **Add**. The Add Expansion Body dialog box appears as shown in Figure 12-24.

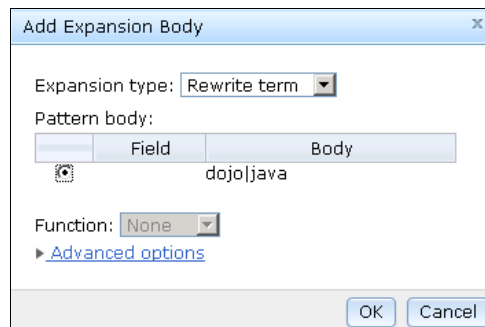


Figure 12-24 The rewrite rule

- b. For the Expansion type field, select **Rewrite term**.
- c. For the Pattern body, select the **dojo|java** pattern.
- d. Click **OK**.
- e. Create the keyword expansion for `technicalpage`.



4. Save the rule.

If the rule expansion type is REPLACE, the rule now replaces all instances of `java site` with `java technicalpage`. If the rule expansion type is OR, the rule now supplements `java site`: `'java site'` or `'java technicalpage'`.

### Adding ranking for the expansion rules

To associate an expansion rule with a Document Ranking Filter, follow these steps:

1. Click **Add** on the lower portion of the Expansion Rule pane.
2. Choose a previously defined filter (see 12.4.5, “Ranking the search results” on page 475).

## 12.4.4 Grouping results

Aggregation rules group your results. Each rule has two parts:

- ▶ Trigger: Based on the user’s query, the terms, and fields used.
- ▶ Action: Based on words in the result document body or fields.

**Note:** In order for the grouping to be displayed, the results need to be collapsed. To do so, in the enterprise search application, select **Preferences** → **Results** → **Collapse results from same source**.

To add groupings, use the following steps:

1. From the Results Aggregation pane, select **Edit Aggregation**.
2. Click **Add Entry**.
3. Choose from one of the conditions: Equals, Contains, Any, Starts With, Ends With. These options describe the matching conditions for the query string.  
If a match is made, the aggregations are created according to the Aggregation body definitions (right pane), as shown in Figure 12-25 on page 474.
4. Click **Add** from the right pane of Figure 12-25 on page 474.
5. Choose a field and enter an exact matching value (not a substring).

**Note:** The Aggregation body rules look for exact matches in the defined fields. The only fields that are listed in the drop-down box are those that are defined as returnable and limited to exact match. To configure fields to appear in the list, click the **Pencil** icon in the Parse and Index panel in the administration console. Click the **Pencil** icon next to the relevant field in the Index Fields Definition page. Ensure that **Returnable**, **Field Search**, and **Exact Match** are selected.

6. Select the **Prioritize** check box if you want the groups to be listed first in the search result.

Figure 12-25 shows the setup for prioritizing PDF files in a search for 'Redbk' and 'Redbooks'.

For each rule except rules for the condition 'ANY', specify at least one pattern body. If you define multiple patterns, specify the order in which they are to be evaluated.

**Matching Rule**

Condition: **CONTAINS**

In order  Any order

Pattern body: **Add**

Field	Body	Actions
Redbk		
Redbooks		

**Aggregation body** **Add**

Prioritize these content source groups higher

Body	Show	Actions
extension:.pdf		

**Move Before** **Move After**

**OK** **Cancel**

Figure 12-25 Configuration for prioritizing PDF files in the Redbooks search

7. Select **OK**.

Each aggregation rule defines one grouping. Typically more than one group is defined. For example, if someone searches for software, it would be logical to group results according to the product name field. Each search type can be associated with its own grouping.

In this example, the PDF files are listed first in the search result. After the first listed item, each group is marked by an indentation after the first listed item.

## 12.4.5 Ranking the search results

There are many factors that determine the display order of the items in your search results. The Ranking Filters set scores for ranking based on the field where the match is found. These filters can be added to query expansion rules and are applied when the rule is fired.

To set document ranking rules:

1. In the Search panel of the administration console, click the **Pencil** icon.
2. Select **Rules to Tune Queries and Results**.
3. Click **Edit Document Ranking Filters**. The Document Ranking Filters dialog appears and displays existing filters.
4. Click **Add Filters**. The Add **Ranking Filter** dialog appears as shown in Figure 12-26.
5. On the left are three vertical tabs: Top-most, Top, and Bottom. These define three buckets for document sorting. Click the appropriate bucket and select **Add Filters**.
6. Choose a field from the **Add Filters** window as shown in Figure 12-26.

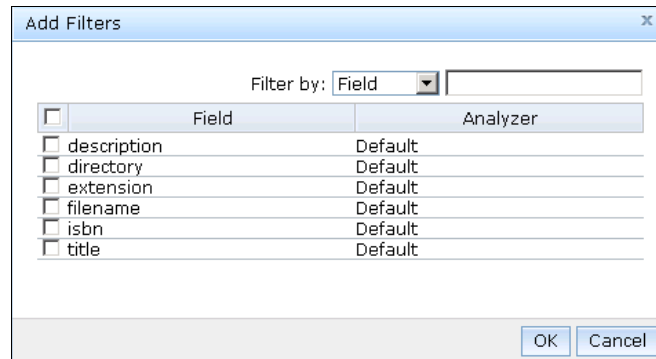


Figure 12-26 Adding fields to a priority bucket

7. Select **OK**.

A document that matches the search terms in this field now receives a ranking according to the chosen bucket. Other factors influence the ranking within the bucket.

## 12.5 Search Customizer

After the index is set up, and the facets and category tree are available, additional components can be added to the enterprise search application using Search Customizer.

Search Customizer is accessed by a button in the upper right corner of the administration console. To work with Search Customizer, use the following steps:

1. Click **Search Customizer**. A small option bar appears in the lower right corner.
2. Select **Open Customizer**. The main Customizer window opens.
3. Make your changes.
4. After you finish making your changes, click **Close** to close the main window.
5. In the small option bar, select **Save changes**.

**Important:** Wait until the Alert box appears, as shown in Figure 12-27, before proceeding.

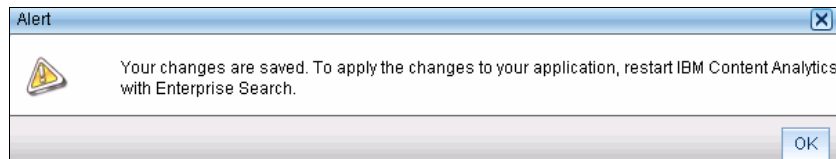


Figure 12-27 The Alert box appears when the changes are saved

6. Click **OK**.

In the following subsections, we add a few components that are based on the new fields and facets that we customized in the administration console Crawler, Parsing, and Indexing panes.

### 12.5.1 Adding the Person facet

The Name Entity annotator extracts personal, place, and organization names. To add the Personal facet, make sure that this annotator is selected in the document processing pipeline. All defined facets appear in the enterprise search application by default.

For information about this annotator, see Chapter 7, “Performing content analysis” on page 231.

## 12.5.2 Adding the timeline

In the main window of the Search Customizer, perform the following steps to add the timeline:

1. Select the **Other Panes** tab. The pane defined as Time Series Chart - pane5 appears.
2. Select **Enable Time Series Chart** and **Expand series chart pane**.
3. Close the window.
4. Select **Save Changes** from the Customizer Controls box.
5. Click **Open Customizer**.
6. Select the **Layout** tab. The Time Series Chart appears in the Top Container.
7. Drag the **Time Series Chart** to any available container.
8. Restart the search service in the Search pane of the administration console.
9. Perform a search.

For our example, we search for IBM. The Time Series Chart appears as shown in Figure 12-28.

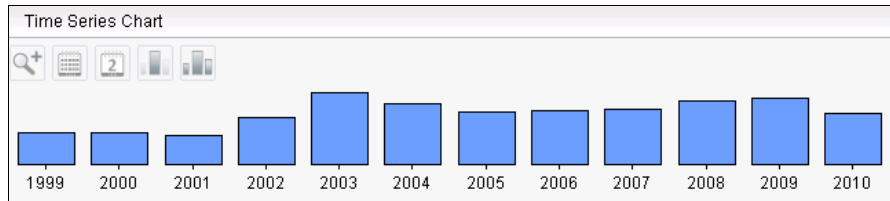


Figure 12-28 The Time Series Chart widget

## 12.5.3 Adding the Category Tree

After the category tree has been set up in the administration console (see 12.3.4, “Adding a category tree” on page 458), you can add it to the enterprise search application.

To add the Category Tree, follow these steps:

1. In the Search Customizer, select the **Facets** tab. The Category tree is listed as “Category Tree - pane 2”.
2. Select the options as shown in Figure 12-29 on page 478.

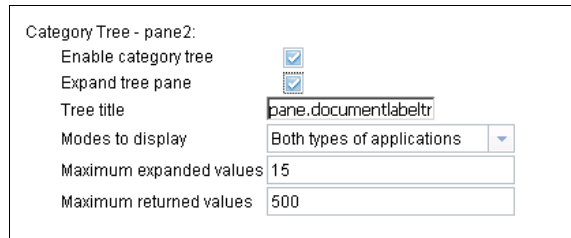


Figure 12-29 The Category Tree widget configuration

3. Close the window.
4. Select **Save Changes** from the Customizer Controls box.
5. Select **Open Customizer**.
6. Go to the **Layout** tab. The Category Tree pane appears in one of the defined panes.
7. Drag the Category Tree to any pane that you want.
8. Restart the Search service in the administration console. The Category Tree now appears in the chosen pane and can be browsed by clicking the arrow icons.

Figure 12-30 shows the Category Tree in our example.

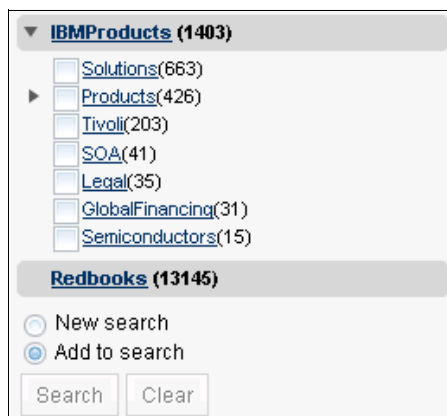


Figure 12-30 The Category Tree widget

## 12.6 Performing search

By now, the content is crawled and indexed with the new configuration, the search configuration is set up, and the new interface components have been added. It is time to perform the search. In this section, we show you how to perform search efficiently by using the enterprise search application and the various configuration and components that we set up earlier.

### 12.6.1 Search strategies

A good search strategy is to make an initial search that returns a broad range of items. Then, examine the results and narrow down the search results based on your business requirements.

In the previous sections, we show how to add components to the enterprise search application for viewing distributions of the results over a range of parameters. These components are interactive, that is, they enable you to drill down and narrow the search result content. As you drill down, also look for the additional terms that are added through query expansion in each subsequent search query.

To best demonstrate the search strategies that take advantage of the custom fields, facets, and search enhancements created earlier in this chapter, run a few search examples.

### 12.6.2 Search example

We begin with a search goal: Look for the most relevant *Redbooks* that describe email products.

Let us perform some searches:

1. Initially, we enter a simple search query as follows:

```
email
```

Based on the expansion rule that is defined in Figure 12-21 on page 471, the search was expanded to include the term, Lotus. Figure 12-31 on page 480 shows the search result for our example.

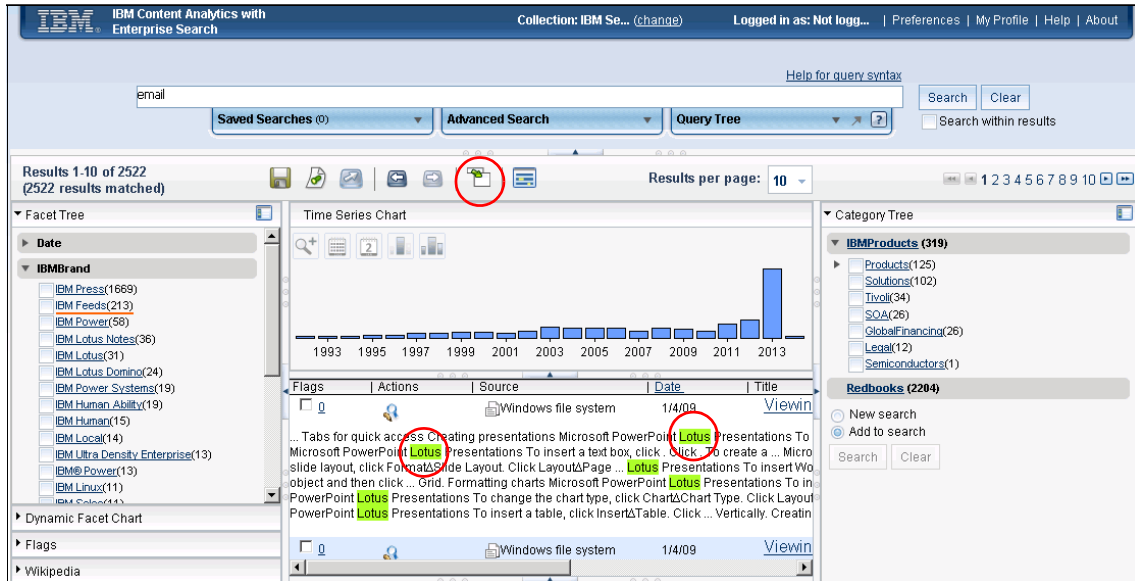


Figure 12-31 The customized enterprise search application

Various components display the distribution of the results by different parameters such as date, IBM Brands, and Categories.

**Note:** To see a detailed view of all displayable fields, click the icon that is shown in red in Figure 12-31.

2. We now want to restrict the search results to only recent Redbooks publications.

This can easily be done by adding date ranges for the search results by using the Time Series Chart. The Time Series Chart includes tools that allow you to increase and decrease the scale of date display that is based on years, months, and days so that you can choose ranges that are based on this date period, and zoom in to a selected date range.

For our example, we select the beginning year on the Time Series Chart, holding and dragging the mouse, to the ending year on the chart to add a date range to the initial search query. Now the query becomes the following:

```
email AND (date>="2000-01-01" date<="2013-12-31")
```



3. We further narrow down the search result based on a specific category.

For this example, we select **Redbooks** from the category tree. The resulting query is as follows:

```
email AND (date>="2000-01-01" date<="2013-12-31") AND
rulebased:./"Redbooks"
```

4. Because we want Redbooks that deal with Lotus products (that deal with emails), we can further narrow down the search results by using a Facet tree.

For this example, in the Facet pane, select all the Lotus brands that are listed in the IBMBrands facet tree and click **Search** in the same pane. The new search query becomes:

```
email AND (date>="2000-01-01" date<="2013-12-31") AND
rulebased:./"Redbooks" AND (/"IBMBrand"/"IBM Lotus" OR
/"IBMBrand"/"IBM Lotus Domino" OR /"IBMBrand"/"IBM Lotus Notes")
```

Figure 12-32 shows the resulting display of these search activities.

Based on your business requirements and the type of content you have for your organizations, you can create relevant categories. Possible categories can include: Facets, synonyms from your content, create query expansion rules using keywords, dictionary lookups, regular expansion rules, and use default tools such as timeline charts, person, location, and organization annotations and custom annotators, to help users to perform searches more efficiently and effectively.

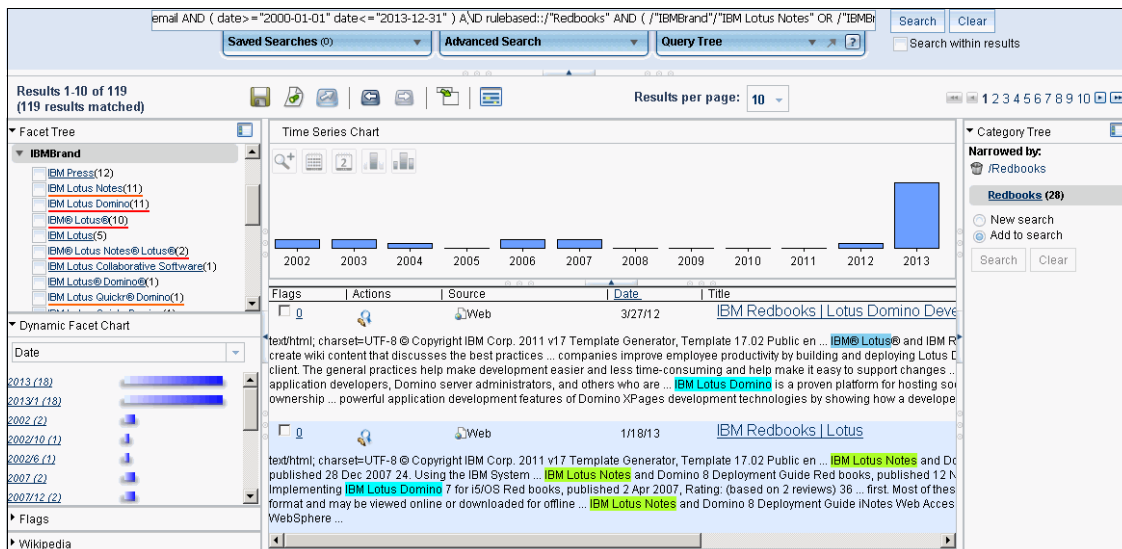


Figure 12-32 The search results after refinement

## 12.7 Security

An important topic that needs to be addressed in any enterprise-level search application is security. Security is a broad term that includes many aspects. In this section, we cover the most relevant concepts including authentication and authorization. This section provides only overview information. For detailed information, follow the references that are provided in the applicable section.

*Authentication* checks users credentials against different possible sources. *Authorization* restricts the access to the application and to the content. In the case of the enterprise search application, the limiting of document access is inherited from the crawled content sources. Depending on the permissions set for file system files, database records, and content management documents, they limit what users can search and see.

### 12.7.1 Authentication

Authentication is the process by which the system verifies the identity of a user. The first level of security is the web application server, either through the embedded web application server or through WebSphere Application Server global security settings. You can configure the system to use an LDAP registry and allow only registered users to log in to applications or the administration console. You can also configure the system to use an LTPA key file to provide single sign-on (SSO) authentication support to application users. For more information about single sign-on, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iisasecjetty.htm>

Enterprises generally perform user authentication during the login process. The user enters a user ID and an associated password. The authentication process is a part of the IT infrastructure. It is typically performed by the web server or application server with a user registry (for example, LDAP).

The enterprise search application is designed to work with the existing authentication components. It does not require a separate login process for the users. When the enterprise search application requires the identity of the user who is logged in, it interacts with the host environment (in this case, the Jetty web server or WebSphere Application Server) to obtain the user's credentials. With this approach, the enterprise search application can be smoothly integrated with the existing authentication policies of an enterprise without requiring a separately maintained user registry.

To enable authentication, you need to configure the web server that hosts the Content Analytics server.

For the embedded Jetty server, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iiysasecjettyproc.htm>

For WebSphere Application Server, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iiysawassec.htm>

## 12.7.2 Authorization (access control)

Authorization is the mechanism by which a system grants or revokes the right to access specific data or perform certain actions. Normally, a user must first log in to a system, using an authentication system as described previously. Next, the authorization mechanism controls the operations that the user might perform by comparing the user ID to an access control list (ACL). Content Analytics employs several levels of access control that can be used independently or together to provide increasing levels of authorization.

Content Analytics offers the following levels of access control to support the enterprise search capability:

- ▶ Administrative: Controls who can set up and maintain collections.
- ▶ Collection: Controls who has access to enterprise search collections. Collection security is enabled when the collection is created. It cannot be added or removed after the collection is created.
- ▶ Document: Controls who has access to which documents. These permissions need to be coordinated with the content source.

**Note:** Document security is only possible when collection security is enabled.

- ▶ Encryption: Encrypts sensitive data, such as passwords, which are specified in the administration console.

### Administrative access control

Within an enterprise, multiple enterprise search collections are likely to be created for different applications. Collections can contain documents from multiple data sources. Each application and its associated collection can require the expertise of different individuals in the organization to aid in the setup and configuration of the collection.

Content Analytics supports this kind of differentiation by allowing the administrator to assign individual administrator user IDs to one or more specific collections. The administrator can indicate that a particular user ID with a given role can access all collections or selected collections. Roles are an abstract logical grouping of users that are predefined by the Content Analytics product.

For administration, some of the predefined roles to support enterprise search capability are listed:

- ▶ Master administrator
- ▶ Collection administrator
- ▶ Operator
- ▶ Monitor

### ***Master administrator***

These users create collections and have the authority to administer all aspects of your system, including adding administrative users. Until the default administrator assigns other users to the master administrator role, only the default administrator can create collections or add administrative users.

### ***Collection administrator***

These users can edit, monitor, and control the operation of collections that they are authorized to administer. These users cannot create collections. Collection administrators can monitor and operate system-level activities only if that authority is granted to them by a master administrator.

### ***Operator***

These users can monitor and control the operation of collections that they are authorized to administer. These users can start and stop collection activities, for example, but they cannot create collections or edit collections. An operator can monitor and operate system-level activities only if that authority is granted to the operator by a master administrator.

### ***Monitor***

These users can monitor collections that they are authorized to administer. These users cannot control operations (such as starting and stopping servers), create collections, or edit collections. A monitor can observe, but not operate, system-level activities only if that authority is granted to the monitor by a master administrator.

For more administrative roles, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iiysatrolecf.htm>

## Runtime access control (document level security)

At run time, users submit queries and receive results sets. Each result item displays metadata for a document and a link to the document source.

Since not all users are allowed to access all resource items (controlled by the resource when the user attempts to access the provided link), only items that are accessible to the user are displayed.

You can control the result sets by using the prefiltering or postfiltering approach.

The *prefiltering* result set requires the crawler to collect all necessary ACL token information and include it in the index. Since the crawler collects all data permissible to the defined crawler user, information is sent to the index to allow the search to filter out any items not accessible to the search user.

For more information about how to set up crawlers to work with document security, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iiysaseccrawlreq.htm>

This approach is not responsive to any changes that might occur in the source ACLs. Also, it is not always possible to correctly evaluate all security policies of the content resource.

With the *postfiltering result set* approach, at run time, the search results are checked against the source to ensure that only permissible items are included. This method degrades the search performance but ensures that all access information is up to date.

When including facets in a search result by using postfiltering, the facets need to be counted for each search. This might further degrade performance.

For more information about how facets effect document security, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iiysasecfacets.htm>

Prefiltering and postfiltering can be combined. This method involves using the high-level, indexed ACL information to extract initial information about the access rights to retrieve an initial group and then to post-filter the group to eliminate additional items. The assumption is that if the user has access to the repository that owns the document, chances are that the user also has access to the document. In this case, postfiltering is used to validate this assumption.

The access control data that is stored in the index varies with the crawler type. For more information, see the following web page:

<http://pic.dhe.ibm.com/infocenter/analytic/v3r0m0/topic/com.ibm.discovery.es.ad.doc/iysasecdocfilter.htm>

## 12.8 Summary

In this chapter, we described various enterprise search capabilities of Content Analytics, and showed how to configure them through the administration console. We walked through the processes of configuring back-end functionality, such as the crawlers that retrieve content from source repositories, and parsing and indexing options to create an optimized index for a particular content set and business need. Finally, we discussed steps to enhance runtime search performance, including ranking and query expansion to improve query results, and various available modifications to the user interface.

Designing the search solution with Content Analytics benefits from insight into both the customer needs and the nature of the content. Your goal is to provide the most effective and intuitive filters for users to navigate and filter content in order to target the needed documents and information. These can be periodically evaluated and modified according to changing content and changing business needs. This iterative process helps to maintain the quality and relevance of the solution.

One recurring theme in this chapter has been the complementary nature of Content Analytics functionality within an enterprise search solution. Some of the more powerful features in an Content Analytics enterprise search application are the features that implement some manner of text analytics or visualization to provide users a refined and effective way to filter a content set. Even if you are only interested in the enterprise search use case and do not plan to implement Content Analytics solutions, it would be valuable to review other chapters of this book to understand the functions and capabilities of what Content Analytics can do. Several chapters, including chapters pertaining to ICA Studio, IBM Content Classification product integration, and others, provide a deeper understanding of how these features and components can enhance an enterprise search solution.



# Adding value to Cognos Business Intelligence

IBM Cognos Business Intelligence (BI) delivers business intelligence that is based on structured content. IBM Watson Content Analytics (Content Analytics) derives business insights from unstructured content. This chapter explains how Content Analytics can add values to Cognos BI reports. It describes the architecture behind the integration of Content Analytics and Cognos BI, and the initial configuration steps that are required to enable and configure connectivity between the two products. In addition, this chapter guides you through the process of generating Cognos BI reports from the data derived from Content Analytics.

This chapter includes the following sections:

- ▶ Integration overview
- ▶ Integration architecture
- ▶ Initial setup
- ▶ Generating Cognos BI reports
- ▶ Creating custom Cognos BI reports

## 13.1 Integration overview

IBM Cognos BI delivers a complete range of business intelligence capabilities, including reporting, analysis, OLAP Cubes, dashboards, and scorecards mostly by processing structured data within an organization. Content Analytics adds value to the Cognos BI analytical product suite by providing insights that are derived and extracted from unstructured content by using its advanced content analytics capabilities.

Information discovered by Content Analytics along with content already available from structured content can be used to automatically generate predesigned Cognos reports from Content Analytics. Alternatively, the extracted data from Content Analytics can be exported into a relational database, which can then be used by Cognos BI to generate more predesigned reports. Using these two approaches, you can design your own customized Cognos BI reports by using the Cognos Framework Manager and the Cognos Advanced Report Studio.

## 13.2 Integration architecture

Traditionally, IBM Cognos BI helps business users to understand and analyze structured data. Cognos BI consists of the following query and reporting tools:

- ▶ Query Studio: Query data and performs simple data retrieve.
- ▶ Analysis Studio: Analyze data and perform OLAP analysis using drilling, dicing, and slicing.
- ▶ Report Studio: Create professional reports.
- ▶ Event Studio: Manage events. Configure triggers such as send an email when a value exceeds a threshold.

With Content Analytics, business users now have a powerful mechanism of converting unstructured content into relevant structured data and have the ability to pass this information to Cognos BI to be processed. This mechanism enables business users to understand and analyze both structured data from traditional structured sources and the converted structured data from unstructured content sources in a more meaningful way, thus providing a more complete reporting and analysis framework.

Figure 13-1 on page 489 shows how Content Analytics passes processed data to Cognos BI. When the data is available in the relational database, the Cognos BI Engine running inside of the Cognos BI Server can expose this information to be utilized and consumed appropriately by Query Studio, Analysis Studio, Report Studio, and Event Studio.



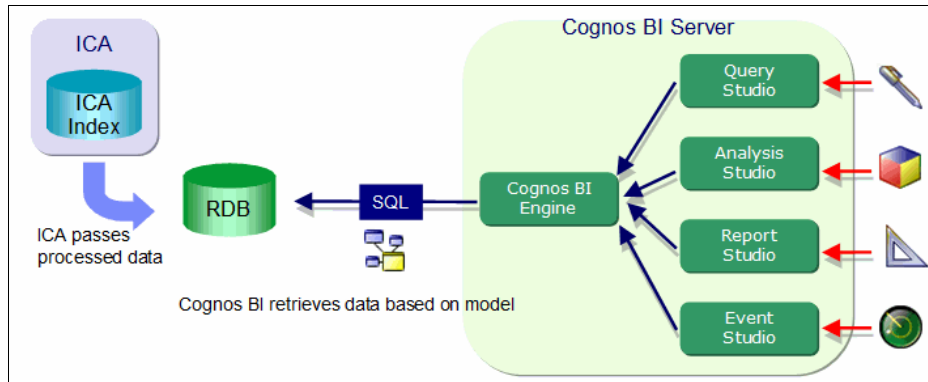


Figure 13-1 Content Analytics passing processed data to IBM Cognos BI

There are primarily two types of information that can be exported from Content Analytics for its use within Cognos BI:

- ▶ Text analysis results such as fields and facets
- ▶ Text mining results such as keywords, frequency, correlations

An example of these two types of information that use the Sample Collection is shown in Figure 13-2 on page 490.

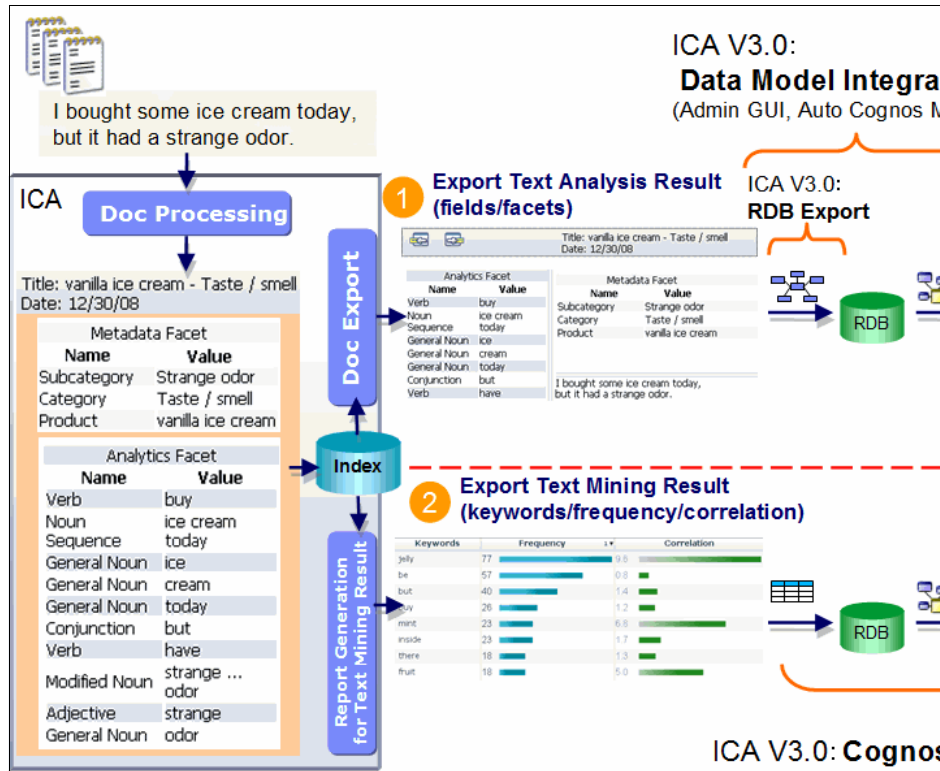


Figure 13-2 Two types of information exported from Content Analytics to IBM Cognos BI

Content Analytics provides two methods of Integration with IBM Cognos BI. In the next two sections, we examine these two methods in more detail:

- ▶ Data model integration
- ▶ Cognos report generation

### 13.2.1 Data model integration

The data model integration is for the analysis of a mixture of unstructured and structured data sets. Business users can analyze the data exported from Content Analytics with data stored in another data source (for example, relational database (RDB)) together by using Cognos BI.

To perform this method of integration, Cognos BI data model should be created with IBM Cognos Framework Manager. Data model integration creates a Cognos BI data model automatically based on the relational database export configuration. Therefore, users do not need to create the Cognos data model

from scratch. The step-by-step instructions for data model integration method are described in 13.5, “Creating custom Cognos BI reports” on page 521.

Figure 13-3 illustrates the data model integration process flow in a step-by-step fashion. The A-x are actions performed by an Content Analytics administrator, and the T-x are actions performed by a content analytics miner user.

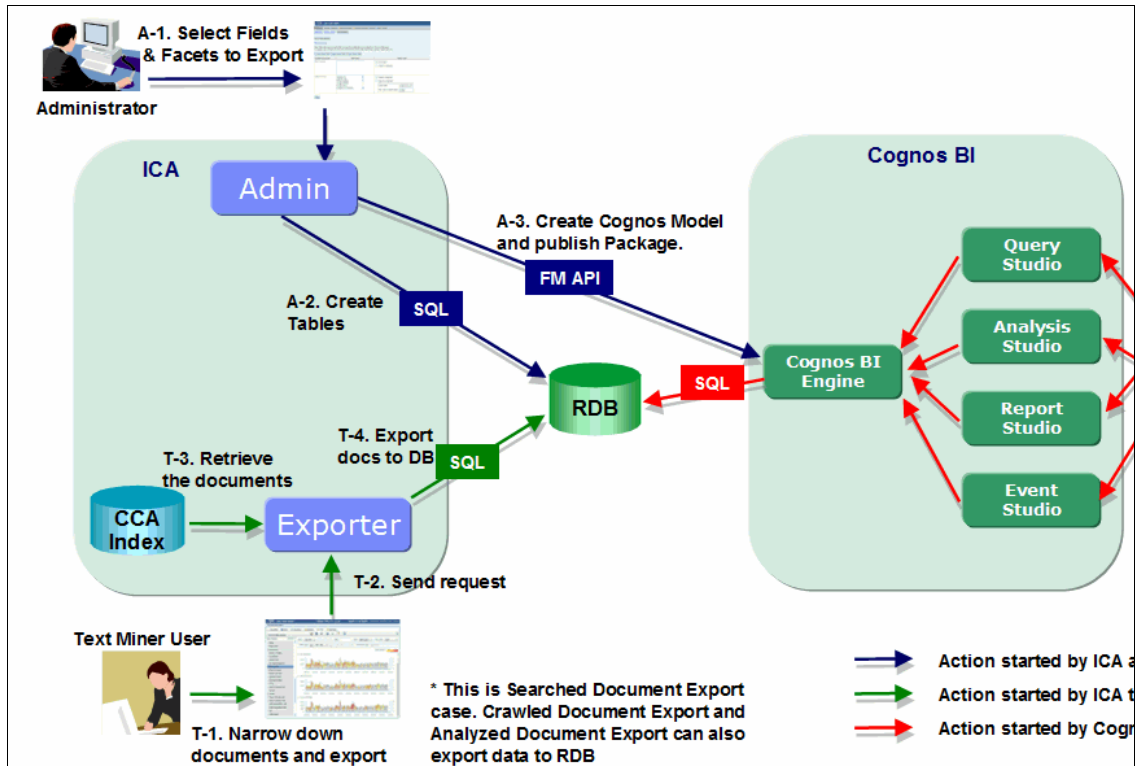


Figure 13-3 Data model integration business process flow

The steps can be summarized as follows:

1. From Content Analytics, the administrator selects fields and facets to export (A-1).
2. From Content Analytics, the administrator creates the tables and exports the Content Analytics information to a relational database (RDB) (A-2).
3. From Content Analytics, the administrator creates a Cognos model and publishes the package to Cognos BI (A-3).
4. From content analytics miner, the user narrows down the documents and exports them (T-1).

5. From Text Minor application, the user sends requests to the exporter (T-2).
6. Content Analytics retrieves the documents (T-3).
7. Content Analytics then exports the documents to the relational database (T-4).
8. Cognos BI user uses information from both Content Analytics and content analytics miner that are all in the relational database and perform appropriate query and reporting.

## 13.2.2 Cognos report generation

The Cognos report generation integration method is primarily used for reporting the content mining results quickly. Business users can create a Cognos BI report about the information that business users found on the content analytics miner user interface with few clicks quickly, thereby providing quick initial insights. This feature exports frequency and correlation already calculated by Content Analytics to RDB. As a result, there is no need to calculate any numbers on Cognos BI again (the calculation may take time for large data). Also, pre-configured report templates are bundled, which eliminates the need of the business user to create the Cognos BI report from scratch. Bundled report templates can be modified for the customization. The step-by-step instructions for the Data Model Integration method are described in 13.4, “Generating Cognos BI reports” on page 515.

Figure 13-4 on page 493 illustrates the Cognos report generation process flow in a step-by-step fashion. The A-x are actions performed by an administrator and the T-x are actions performed by a content analytics miner user.

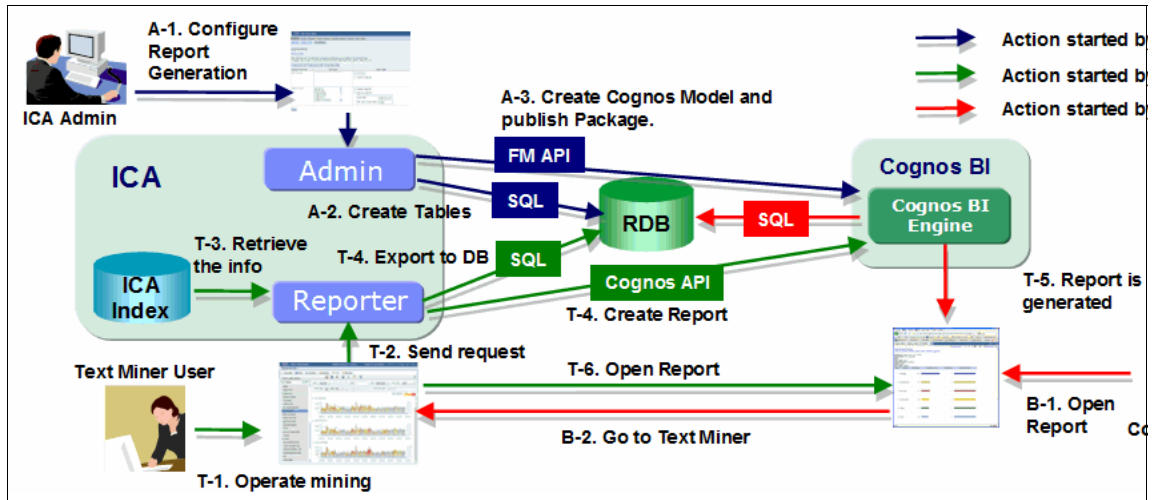


Figure 13-4 Cognos report generation process flow

## 13.3 Initial setup

To integrate with Cognos BI, you must enable Cognos BI and provide the parameters that are necessary for communication. The following sections provide information about the initial setup steps:

- ▶ Verifying IBM Cognos BI
- ▶ Creating Data Source connection by using IBM Cognos BI Administration
- ▶ Configuring default application user roles by using Content Analytics administration console
- ▶ Configuring export to RDB by using Content Analytics administration console
- ▶ Configuring Cognos BI server for reporting by using Content Analytics administration console

JACKIE STARTS GRAPHICS WORK HERE.

### 13.3.1 Verifying IBM Cognos BI

Verify that the IBM Cognos Connections environment, the IBM Cognos Connections Content Store, the Cognos Connections gateway URI, and the Dispatch URI for the gateway are reachable and accessible without any issues.

Figure 13-5 shows the Cognos Connections configuration environment that is being used for this chapter.

**Note:** The Report server execution mode should be set to 32-bit.

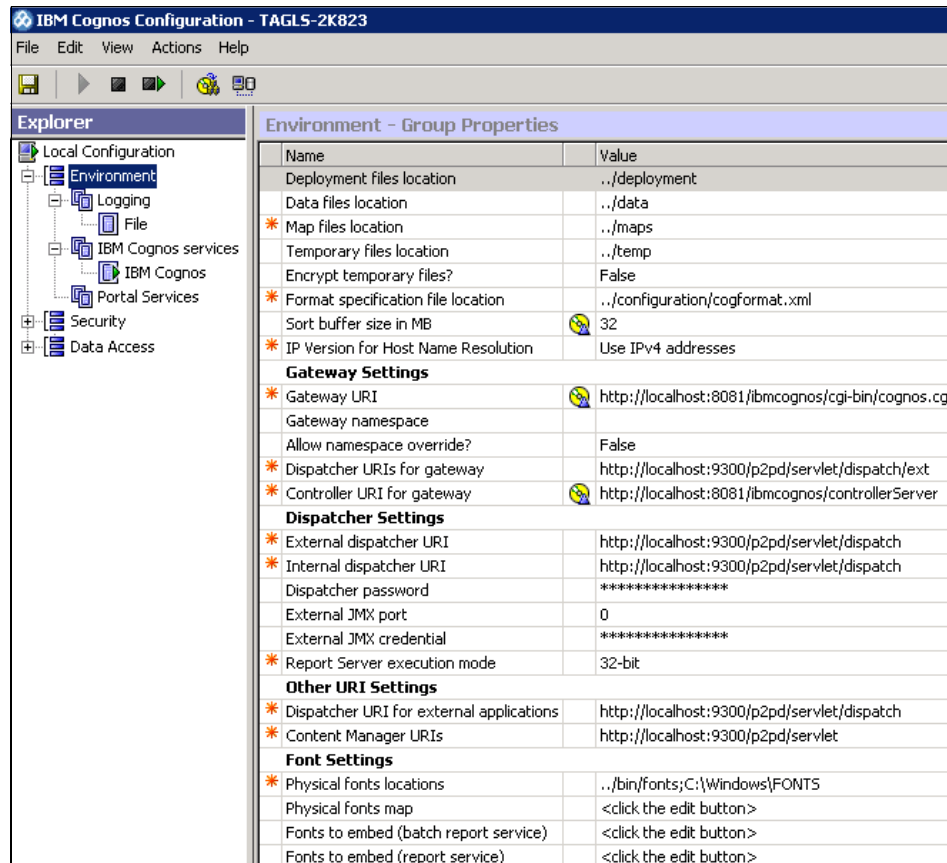


Figure 13-5 IBM Cognos Connections Environment tab

**Note:** The example uses localhost, but in a real environment, use FQDN. Even if both Content Analytics and Cognos are installed on one machine, the Cognos BI setting with localhost sometimes has a problem.

Figure 13-6 on page 495, shows the Cognos Connections Content Store that is being used for this chapter.

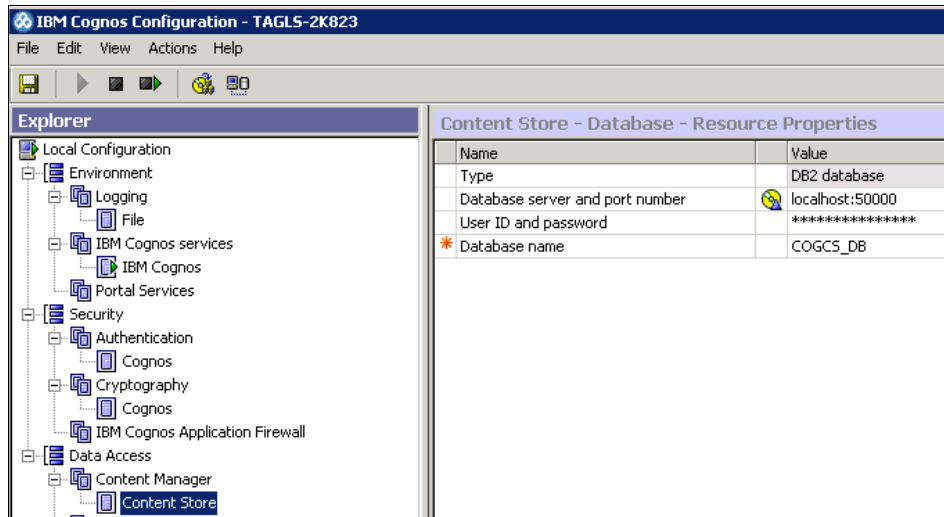


Figure 13-6 IBM Cognos Connections Content Store tab

Ensure that the IBM Cognos Gateway URI is accessible.

The Launch menu from the home page should display the IBM Cognos Administration link, as shown in Figure 13-7.

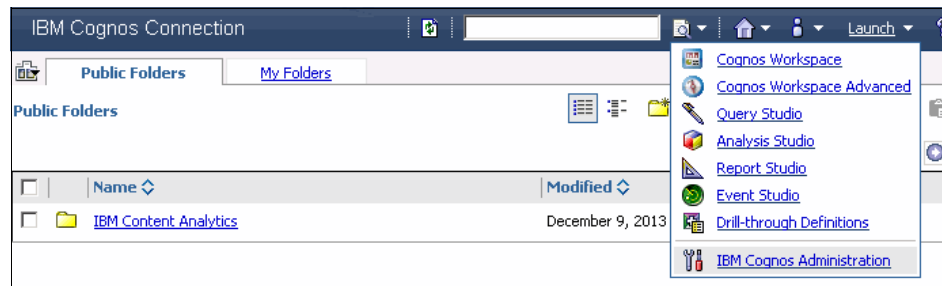


Figure 13-7 Launch IBM Cognos Administration link

After launching the IBM Cognos Administration link, the Cognos Administration home page is displayed, as shown in Figure 13-8 on page 496.

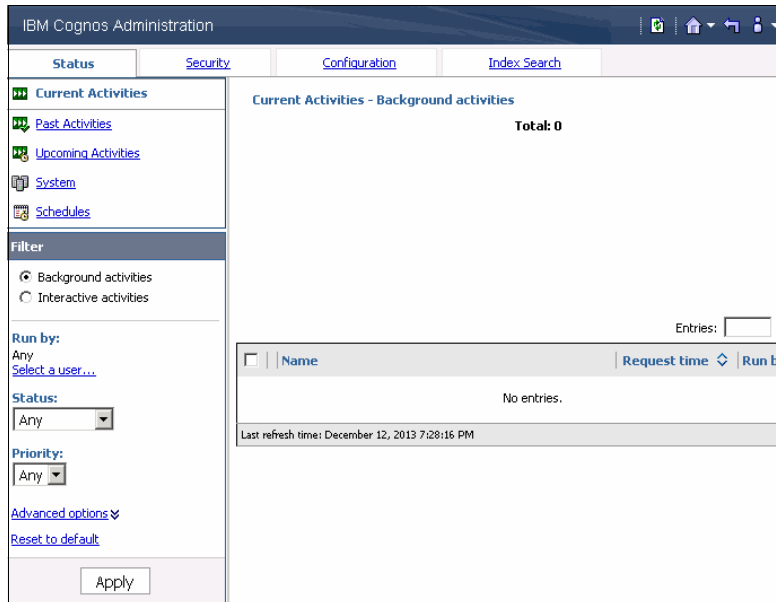


Figure 13-8 IBM Cognos Administration home page

**Note:** At the time of writing this book, Content Analytics V3.0 only supports IBM Cognos BI V10.1.1. We were able to use Cognos BI V10.2 to integrate with Content Analytics V3.0 by performing the following steps, where xxx denotes the collection ID of the collection that is being configured for exports to IBM Cognos BI:

1. Copy XML files under  
 $\$ES\_NODE\_ROOT/master\_config/xxx.exporter/cognos/report/templates/8.0/$  to  
 $\$ES\_NODE\_ROOT/master\_config/xxx.exporter/cognos/report/templates$
2. Open each XML file with a text editor and update this line.

[Original]

```
<report expressionLocale="en" xmlns="
http://developer.cognos.com/schemas/report/8.0/";
useStyleVersion="10">
```

[Modified]

```
<report expressionLocale="en" xmlns="
http://developer.cognos.com/schemas/report/9.0/";
useStyleVersion="10">
```



## 13.3.2 Creating a data source connection by using Cognos BI Administration

To integrate Content Analytics to IBM Cognos BI, a new data source connection is required to be created on Cognos BI. Each Content Analytics analytics collection requiring IBM Cognos BI integration requires a new data source connection to be created on Cognos BI. Follow these steps to create a new data source connection using IBM Cognos BI Administration:

1. From the IBM Cognos BI Administration home page, go to the **Configuration** tab and click the **New Data Source** icon, as shown in Figure 13-9.

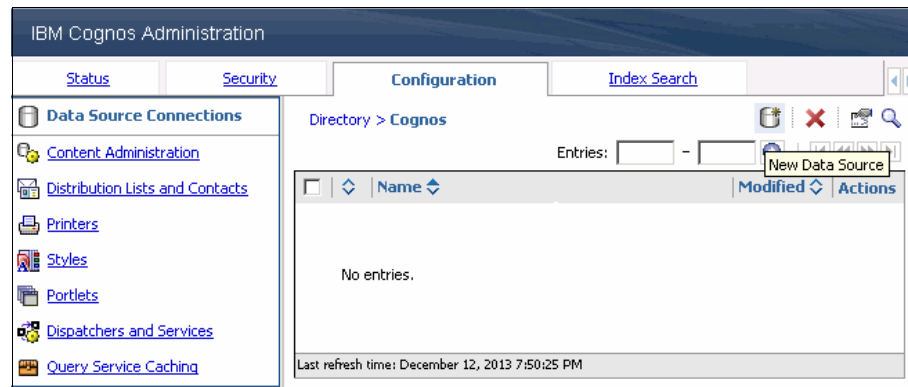


Figure 13-9 IBM Cognos BI Administration: Creating a new data source

2. Enter the name, description, and screen tip for the data source connection. For our example, we enter Sample for all the fields as shown in Figure 13-10 on page 498.

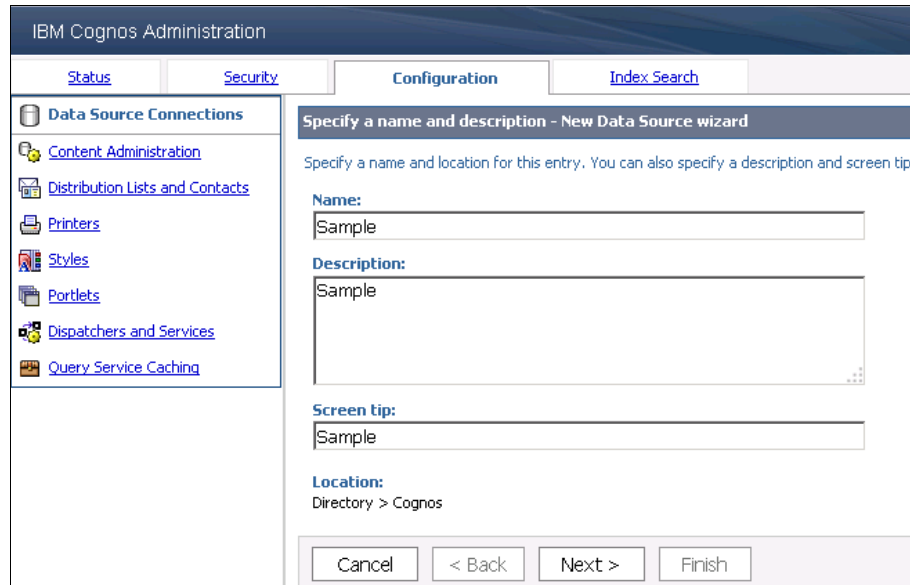


Figure 13-10 New data source configuration: Entering name and description

3. Choose the type of the data source connection. For our example, we select **IBM DB2** for the **Use the default object gateway** option, and the **Configure JDBC connection** option, as shown in Figure 13-11.

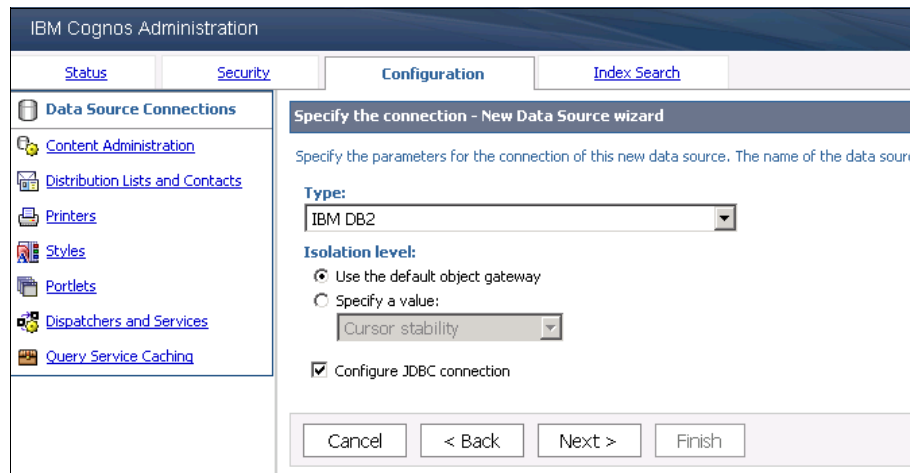


Figure 13-11 New data source configuration: Selecting the data source type

4. Enter the data source connection details. For our example, see Figure 13-12.

IBM Cognos Administration

Status Security Configuration Index Search

Data Source Connections

- Content Administration
- Distribution Lists and Contacts
- Printers
- Styles
- Portlets
- Dispatchers and Services
- Query Service Caching

### Specify the IBM DB2 connection string - New Data Source wizard

Edit the parameters to build a DB2 connection string.

**DB2 database name:**  
sampledb

**DB2 connect string:**  
jdbc:db2://localhost:50000/SAMPLEDB

**Collation sequence:**

Open asynchronously

**Timeouts**  
Specify the time in seconds, in which you want the database to connect or wait for your reply

**Connect time:**  
0

**Reply time:**  
0

**Signon**  
Select whether or not authentication is needed, and if so, the type of authentication to use, w

No authentication  
 An external namespace:  
[Dropdown]  
 Signons

Password  
 Create a signon that the Everyone group can use:

**User ID:**  
db2admin

**Password:**  
●●●●●●●●

**Confirm password:**  
●●●●●●●●

**Testing**  
[Test the connection...](#)

Cancel < Back Next > Finish

Figure 13-12 New data source configuration: Entering connection details

5. Provide more details about the data source connection string. For our example, see Figure 13-13.

IBM Cognos Administration

Status Security Configuration Index Search

Data Source Connections

- Content Administration
- Distribution Lists and Contacts
- Printers
- Styles
- Portlets
- Dispatchers and Services
- Query Service Caching

Specify the IBM DB2 (JDBC) connection string - New Connection wizard

Edit the parameters to build a DB2 (driver: com.ibm.db2.jcc.DB2Driver) connection string.

Server name: localhost

Port number: 50000

Database name: SAMPLEDB

JDBC Connection Parameters:  
These optional parameters are appended to the URL and are specific to the driver.

Local Sort Options

Collation Sequence:

Level: Primary

Testing  
[Test the connection...](#)

Cancel < Back Next > Finish

Figure 13-13 New data source configuration: Entering more connection details

6. Click **Next** on the wizard to see the window as shown in Figure 13-14 on page 501.

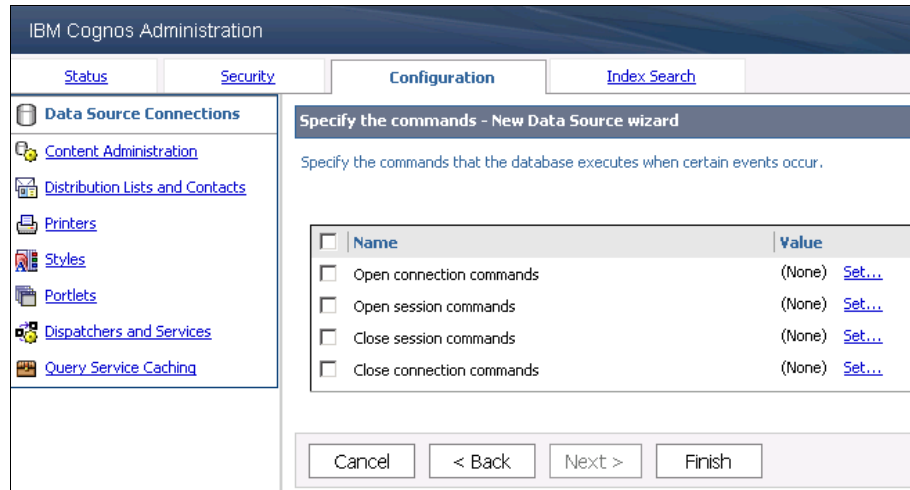


Figure 13-14 New data source configuration: Clicking Next

- Finalize the data source creation wizard by clicking **Finish** to see the new data source connection created. For our example, see Figure 13-15.

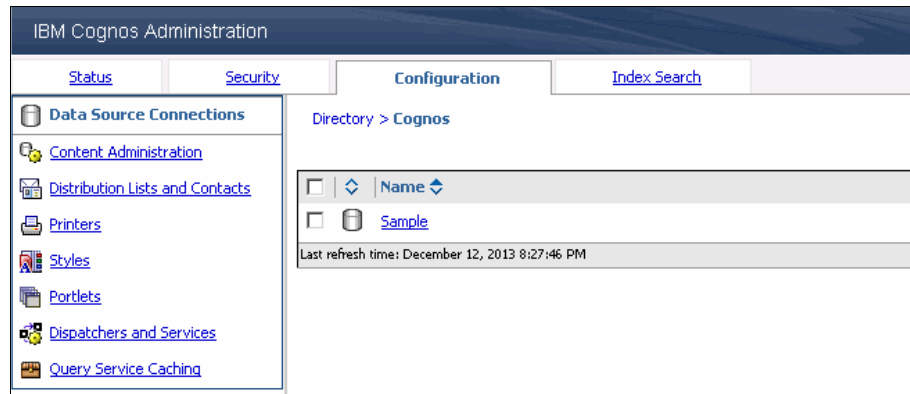


Figure 13-15 New data source: Created

### 13.3.3 Configuring default application user roles

The default application user role does not have privileges to create IBM Cognos BI reports. To assign privileges, follow these steps:

- Using Content Analytics administration console, go to the **Security** tab and choose **Specify default application user privileges** from the **Actions** menu button on the third section called **System-Level Security**, as shown in Figure 13-16 on page 502.

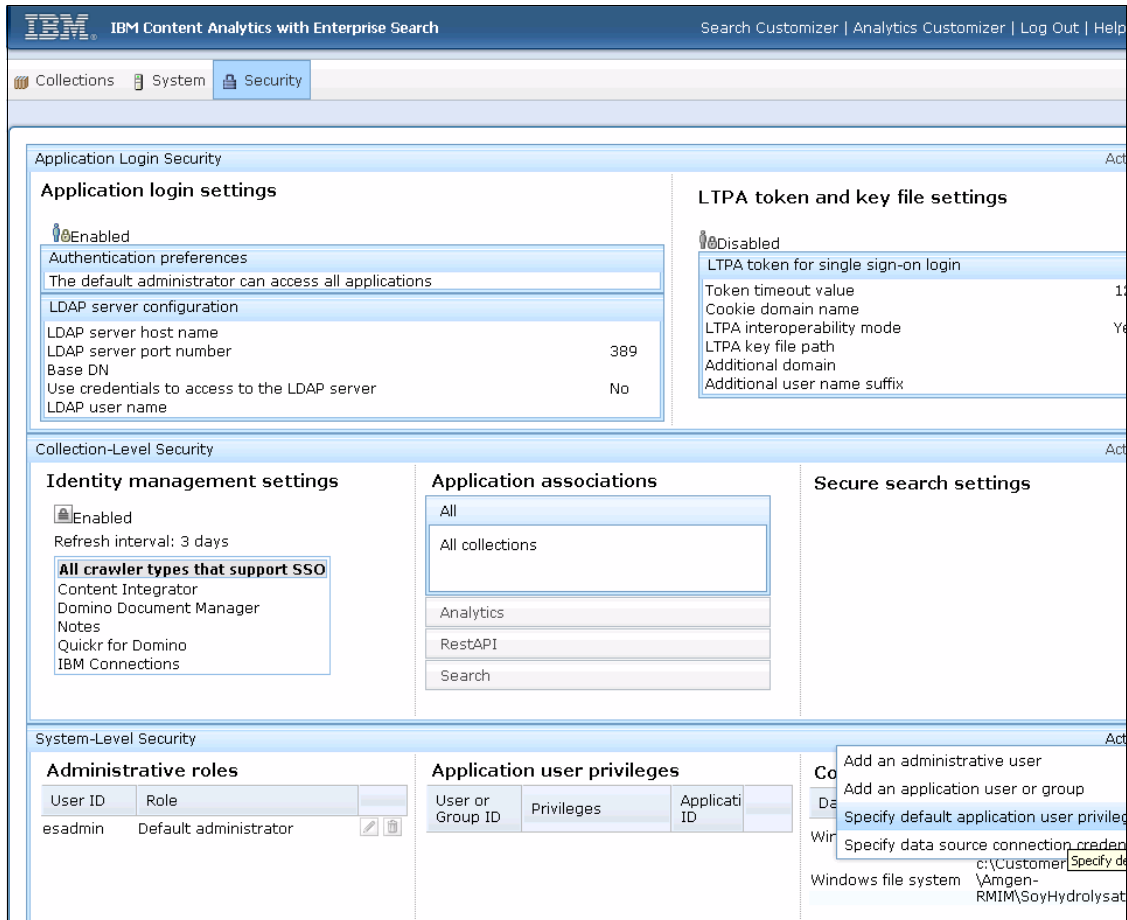


Figure 13-16 Content Analytics administration console: Setting up default user privileges

2. In the Configure application user roles panel under User privileges, select the following options and then click **OK**:
  - Save searches
  - Export documents
  - Create deep inspection reports
  - Add rules to categories
  - Manage document flags
  - Create IBM Cognos BI reports

Figure 13-17 on page 503 shows the selected default application user privileges.

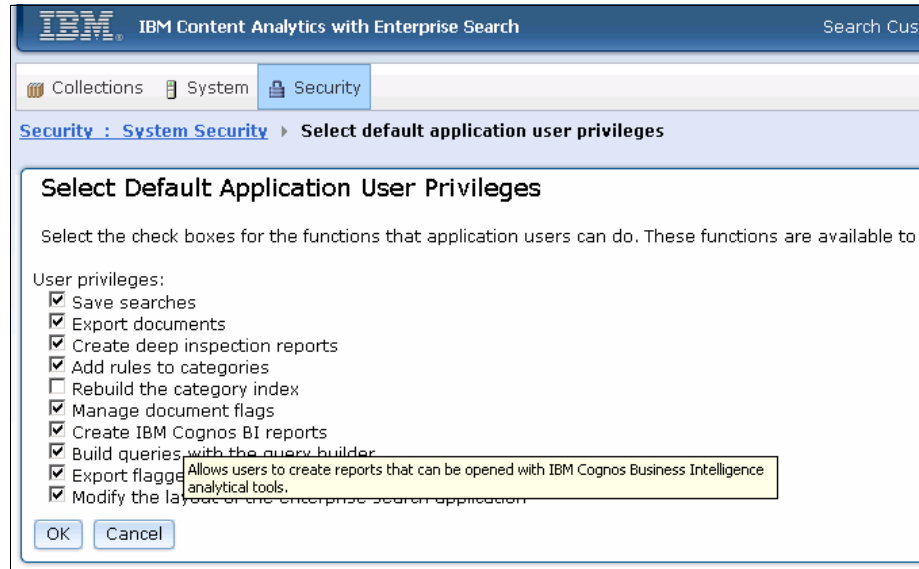


Figure 13-17 Default user privileges: Selecting user privileges

### 13.3.4 Configuring an export to a relational database using Content Analytics

A relational database is used as the common store of information to be shared between Content Analytics and IBM Cognos BI. This section guides you through the steps to configure Content Analytics to export searched documents to a relational database.

To configure export of searched documents to a relational database, follow these steps:

1. From Content Analytics administration console, click the collection to expand the details of the collection. Under the **Search and Content Analytics** area, also marked as 3, find and click the **Arrow** icon, as shown in Figure 13-18 on page 504.

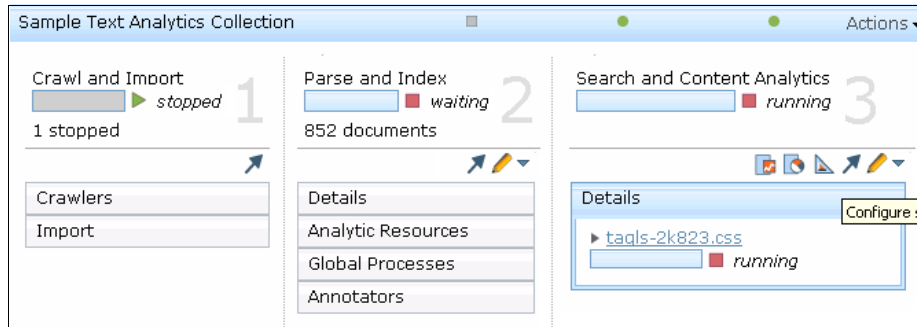


Figure 13-18 Content Analytics administration console: Configuring export to RDB

2. Choose the **Export documents into a relational database** option and click **Configure**, as shown in Figure 13-19.

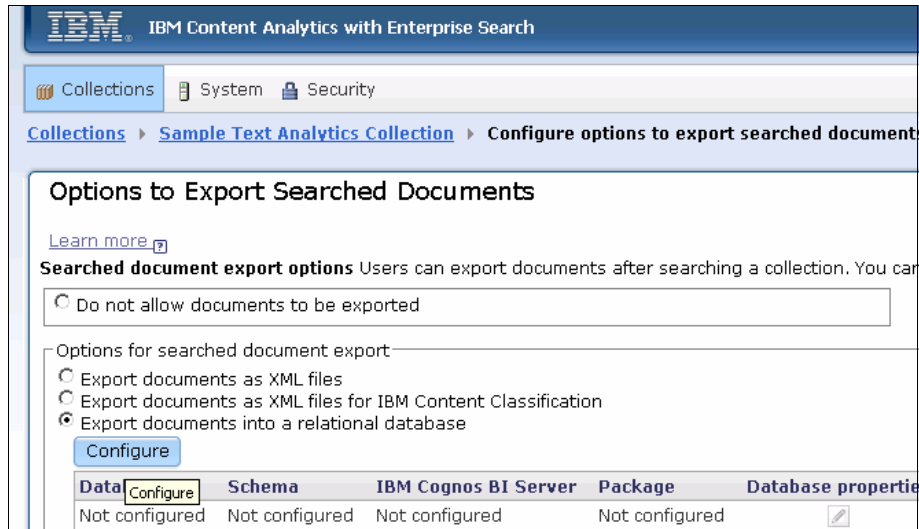


Figure 13-19 Export configuration: Choosing the export documents

3. In the Database Information panel, perform the following steps:
  - a. Select the JDBC database type. In our example, we use **DB2**, which is installed locally on the same machine as Content Analytics.
  - b. Specify the JDBC driver name and class path if the values are different from the default values.
  - c. Enter the database URL for your instance of the relational database. Examples are provided depending on the type of relational database



chosen. The variable *localhost* is used in the examples and most likely must be changed to the host name of the database server.

- d. Provide a valid user ID and password.
- e. Enter a schema name of tables to store data for report generation. The schema name differentiates the Content Analytics/Cognos tables from other tables in the database (if any). This name can be anything that you choose.
- f. Click **Next**, as shown in Figure 13-20.

**IBM** IBM Content Analytics with Enterprise Search

Collections System Security

Collections > Sample Text Analytics Collection > Database information

### Database Information for Exported Documents

[Learn more](#)

Documents will be exported into star-schema tables.  
Specify the following database connection credentials.  
The database must exist and the user must have privileges to create a table and drop a table.  
After you click Next, the system tries to create tables under the specified schema.

Tip: For Oracle, the schema name is usually the same as the user name.

\* JDBC database type:  
DB2

\* JDBC driver name:  
com.ibm.db2.jcc.DB2Driver

\* JDBC driver class path:  
C:\Program Files\IBM\tdsV6.3db2\java\db2jcc.jar

\* Database URL:  
Examples:  
- DB2 UDB: jdbc:db2://localhost:50000/sample  
- Oracle: jdbc:oracle:thin:@localhost:1521:sample  
- SQL Server 2008: jdbc:sqlserver://localhost:1433;DatabaseName=sample  
jdbc:db2://localhost:50000/SAMPLEDB

\* User ID:  
db2admin

\* Password:  
●●●●●●●●

\* Schema of tables to store data for report generation:  
COL\_SAMPLE

Back Next Finish Cancel

Figure 13-20 Export configuration: Setting relationship database connection details

4. Select the index field that needs to be exported. For each index field that might need exporting, there are two options in which it can be exported:
  - A column for a document fact table
  - A table for a dimension

The details for both these options and their meaning is specified at the top of the window. Figure 13-21 shows the portion of the window where we made the selections for our example.

Content to export	
<b>Content</b>	
Text content	<input checked="" type="radio"/> Do not export <input type="radio"/> A column for a document fact table
Binary content	<input checked="" type="radio"/> Do not export <input type="radio"/> A column for a document fact table
<b>Fields to export</b>	
<b>Index field name</b>	
doc_category	<input type="radio"/> Do not export <input checked="" type="radio"/> A column for a document fact table <div style="border: 1px solid gray; padding: 2px; margin-top: 5px;">           Column name: <input type="text" value="DOC_CATEGORY"/>            Data type of facet value column: <input type="text" value="CHAR(50)"/> </div> <input type="radio"/> A table for a dimension
doc_id	<input type="radio"/> Do not export <input checked="" type="radio"/> A column for a document fact table <div style="border: 1px solid gray; padding: 2px; margin-top: 5px;">           Column name: <input type="text" value="DOC_ID"/>            Data type of facet value column: <input type="text" value="CHAR(50)"/> </div> <input type="radio"/> A table for a dimension
doc_product	<input type="radio"/> Do not export <input checked="" type="radio"/> A column for a document fact table <div style="border: 1px solid gray; padding: 2px; margin-top: 5px;">           Column name: <input type="text" value="DOC_PRODUCT"/>            Data type of facet value column: <input type="text" value="CHAR(50)"/> </div> <input type="radio"/> A table for a dimension
doc_subcategory	<input type="radio"/> Do not export <input checked="" type="radio"/> A column for a document fact table <div style="border: 1px solid gray; padding: 2px; margin-top: 5px;">           Column name: <input type="text" value="DOC_SUBCATEGORY"/>            Data type of facet value column: <input type="text" value="CHAR(50)"/> </div> <input type="radio"/> A table for a dimension
<input type="button" value="Back"/> <input type="button" value="Next"/> <input type="button" value="Finish"/> <input type="button" value="Cancel"/>	

Figure 13-21 Export configuration: Selecting export index fields format

5. Now select the top-level facets that need to be exported. For each top-level facet in the facet tree that might need exporting, there is only one option in which it can be exported:

A table for a dimension

The details for this option and its meaning are specified at the top of the window. For our set up, see Figure 13-22.

Top-level facet in the facet tree	Subfacets	Export target
Part of Speech	Noun Noun, General Noun Noun, Others Verb Adjective Adverb	<input checked="" type="radio"/> Do not export <input type="radio"/> A table for a dimension
Phrase Constituent	Noun Phrase Noun Phrase, Noun Sequence Noun Phrase, Modified Noun Noun Phrase, Prep Noun Predicate Phrase Predicate Phrase, Predicate wi	<input checked="" type="radio"/> Do not export <input type="radio"/> A table for a dimension
Named entity	Person Location Organization	<input checked="" type="radio"/> Do not export <input type="radio"/> A table for a dimension
My Keywords		<input type="radio"/> Do not export <input checked="" type="radio"/> A table for a dimension Table name: <input type="text" value="MYKEYWORD"/> Data type of facet value column: <input type="text" value="CHAR(50)"/>
Category		<input type="radio"/> Do not export <input checked="" type="radio"/> A table for a dimension Table name: <input type="text" value="CATEGORY"/> Data type of facet value column: <input type="text" value="CHAR(50)"/>
Subcategory		<input type="radio"/> Do not export <input checked="" type="radio"/> A table for a dimension Table name: <input type="text" value="SUBCATEGORY"/> Data type of facet value column: <input type="text" value="CHAR(50)"/>
Product		<input type="radio"/> Do not export <input checked="" type="radio"/> A table for a dimension Table name: <input type="text" value="PRODUCT"/> Data type of facet value column: <input type="text" value="CHAR(50)"/>

Figure 13-22 Export configuration: Selecting format options for the facets to be exported

- Now in this step, you can choose to continue the Export wizard and configure the Cognos BI server, or choose to complete this wizard and save the export settings as shown in Figure 13-23 on page 508.

After finishing the wizard, you can always come back and configure the Cognos BI server separately or as a continuation of the export wizard as

described in 13.3.5, “Configuring the Cognos BI server for reporting by using Content Analytics” on page 510.

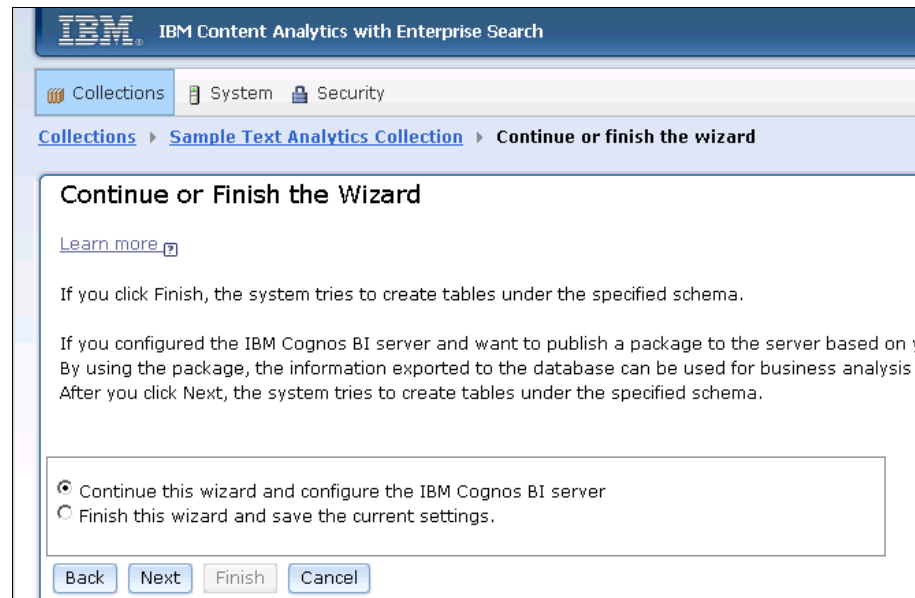


Figure 13-23 Export configuration: Finishing the Export wizard

7. Specify the Cognos BI Server URI as shown in Figure 13-24, and click **Next**.

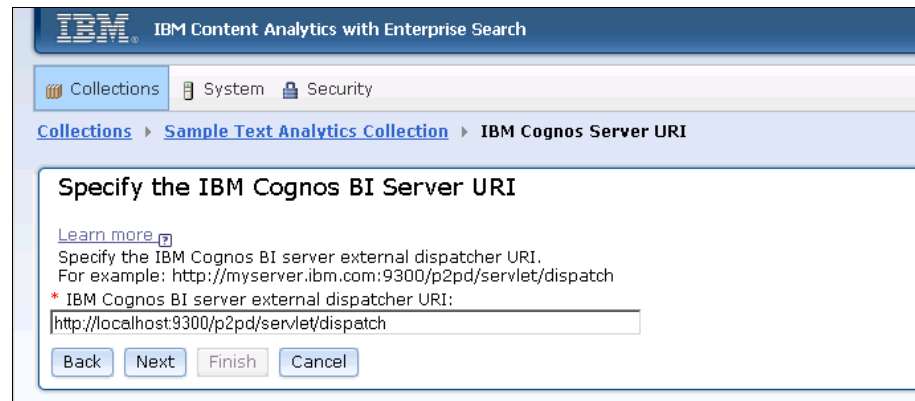


Figure 13-24 Specify Cognos BI Server URI

**Note:** Our example uses localhost. In a real environment, use FQDN.

8. In the “Specify information to publish a package” panel, complete the following steps, as shown in Figure 13-25:
  - a. Specify the data source connection to be used at the Cognos BI server. Make sure that the data source connection that you select is one that also connects to the same relational database that you specified as receiving the exported results.
  - b. Enter a package name, which can be anything that you choose and contains the model for your exported results in the Cognos BI server.
  - c. Enter a directory to store the IBM Cognos BI action log script. The action log script can be used by the Cognos BI server to generate the star schema model for your data.
  - d. Click **Finish**.

The screenshot shows the IBM Content Analytics with Enterprise Search web interface. The breadcrumb trail is: Collections > Sample Text Analytics Collection > Information about a package. The main content area is titled "Specify information to publish a package" and includes a "Learn more" link. The instructions state: "Select the Data Source Connection that is configured on the IBM Cognos BI server that points to the... Also specify an existing directory. The IBM Cognos BI Action Log file will be saved at the specified directory. This Action Log can be loaded by Cognos Framework Manager or Cognos BMT Script Player to create a... With this project, you can modify the model on your own. After you click Finish, IBM Content Analytics with Enterprise Search publishes the package." The form contains three fields: "Data source connection:" with a dropdown menu set to "Sample"; "Package Name:" with a text box containing "col\_sample"; and "Directory to save the IBM Cognos BI action log:" with a text box containing "C:\CustomerData\Cognos". At the bottom are four buttons: "Back", "Next", "Finish", and "Cancel".

Figure 13-25 Specify information to publish a package

You should be able to see a new export configuration panel show up under **Search and Content Analytics Details**, as shown in Figure 13-26.

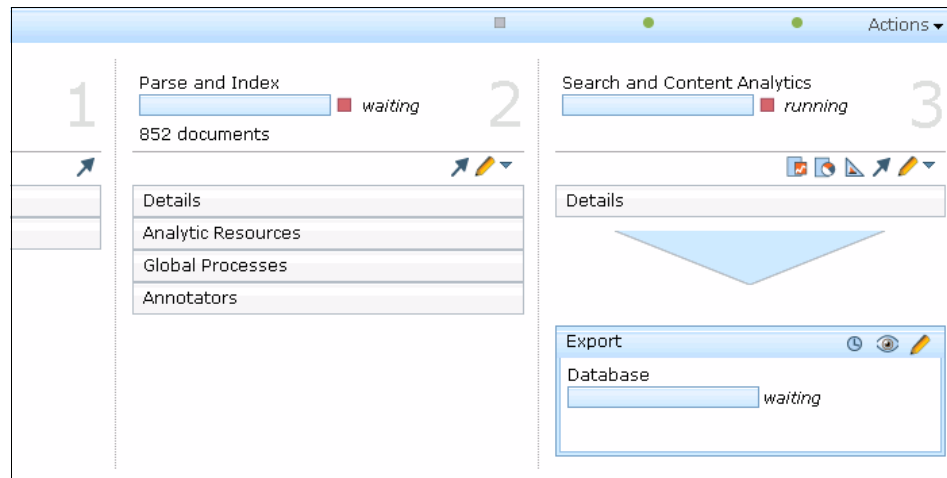


Figure 13-26 New export configuration created

An IBM Cognos BI action log XML file is generated in the file path specified in step 8c, as shown in Figure 13-27.

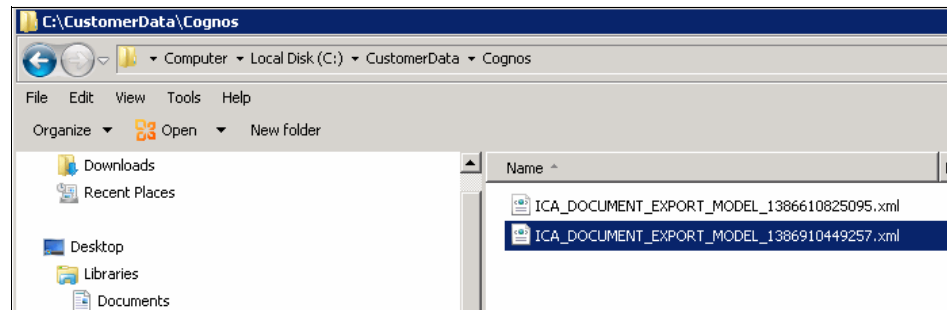


Figure 13-27 IBM Cognos BI action log XML file path

### 13.3.5 Configuring the Cognos BI server for reporting by using Content Analytics

In this section, we step through the process of configuring Cognos BI server for reporting by using Content Analytics administration console. Follow these steps to configure the Cognos BI server:

1. Click the collection to expand the details of the collection. Under the **Search and Content Analytics** area, also marked as 3, find and click the right angled triangle icon, as shown in Figure 13-28.

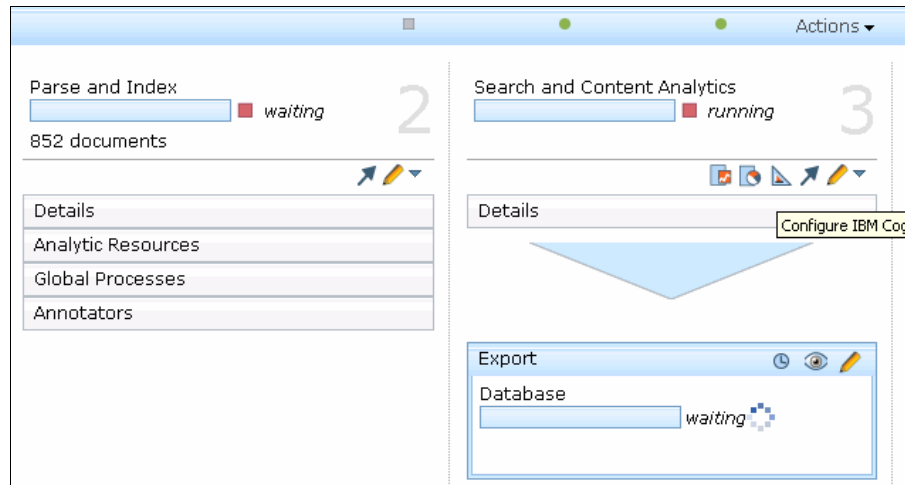


Figure 13-28 Content Analytics administration console: Launch Cognos BI configuration

2. In the database information panel, perform the following steps:
  - a. Select the JDBC database type. In this example, we use **DB2**, which is installed locally on the same machine as Content Analytics.
  - b. Specify the JDBC driver name and class path if the values are different from the default values.
  - c. Enter the database URL for your instance of the relational database. Examples are provided depending on the type of relational database chosen. The variable *localhost* is used in the examples and most likely must be changed to the host name of the database server.
  - d. Provide a valid user ID and password.
  - e. Enter a schema name of tables to store data for report generation. The schema name differentiates the Content Analytics/Cognos tables from other tables in the database (if any). This name can be anything that you choose.
  - f. Click **Next**, as shown in Figure 13-29 on page 512.

IBM Content Analytics with Enterprise Search Search C

Collections System Security

Collections > Sample Text Analytics Collection > Database information

### Database Information

[Learn more](#)

The results of content mining are stored in a database. IBM Cognos BI retrieves the stored data to... Specify the following information to determine which database is used. The database must exist. After you click Next, the system tries to create tables under the specified schema.

Tip: For Oracle, the schema name is usually the same as the user name.

\* JDBC database type:  
DB2

\* JDBC driver name:  
com.ibm.db2.jcc.DB2Driver

\* JDBC driver class path:  
C:\Program Files\IBM\tdsV6.3db2\java\db2jcc.jar

\* Database URL:  
Examples:  
- DB2 UDB: jdbc:db2://localhost:50000/sample  
- Oracle: jdbc:oracle:thin:@localhost:1521:sample  
- SQL Server 2008: jdbc:sqlserver://localhost:1433;DatabaseName=sample  
jdbc:db2://localhost:50000/SAMPLEDB

\* User ID:  
db2admin

\* Password:  
.....

\* Schema of tables to store data for report generation:  
COL\_SAMPLE

Back Next Finish Cancel

Figure 13-29 Content Analytics administration console: Configuring database information

3. Specify the Cognos BI Server URI. See Figure 13-30 on page 513.



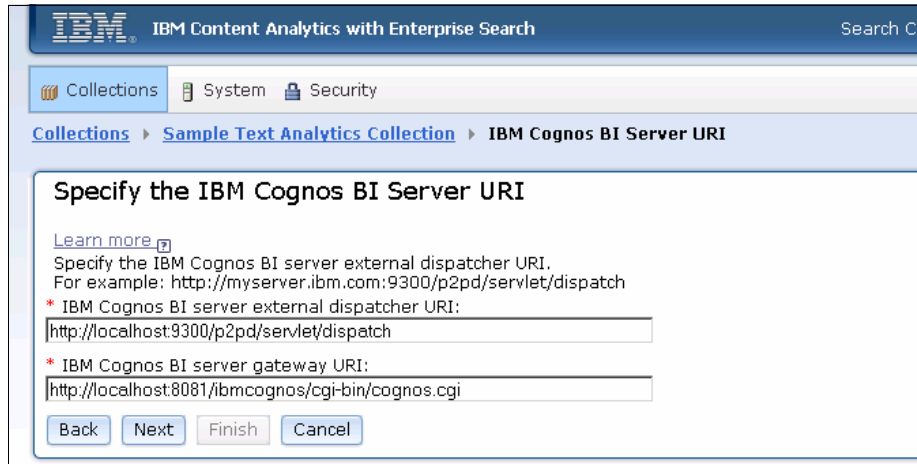


Figure 13-30 Specifying Cognos BI Server URI

4. Specify Cognos BI user ID and password if Cognos BI Server is configured to require authentication. See Figure 13-31.

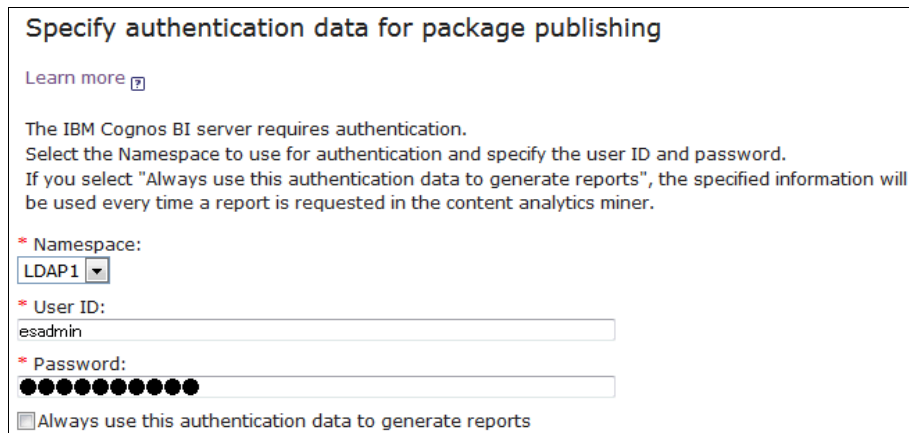


Figure 13-31 Specify Cognos BI server authentication for packaging publishing

5. In the "Specify information to publish a package" window, complete these steps:
  - a. Select the data source connection that points to the database that you just specified for storing the results of content mining. You can configure the data source connection in Cognos by using IBM Cognos Connections as described in 13.3.2, "Creating a data source connection by using Cognos BI Administration" on page 497.

- b. Enter the package name to use.
- c. Click **Finish**.

Content Analytics then publishes the package to the IBM Cognos BI server. This package includes a model for the IBM Cognos BI server to retrieve data from the database for report generation. See Figure 13-32.

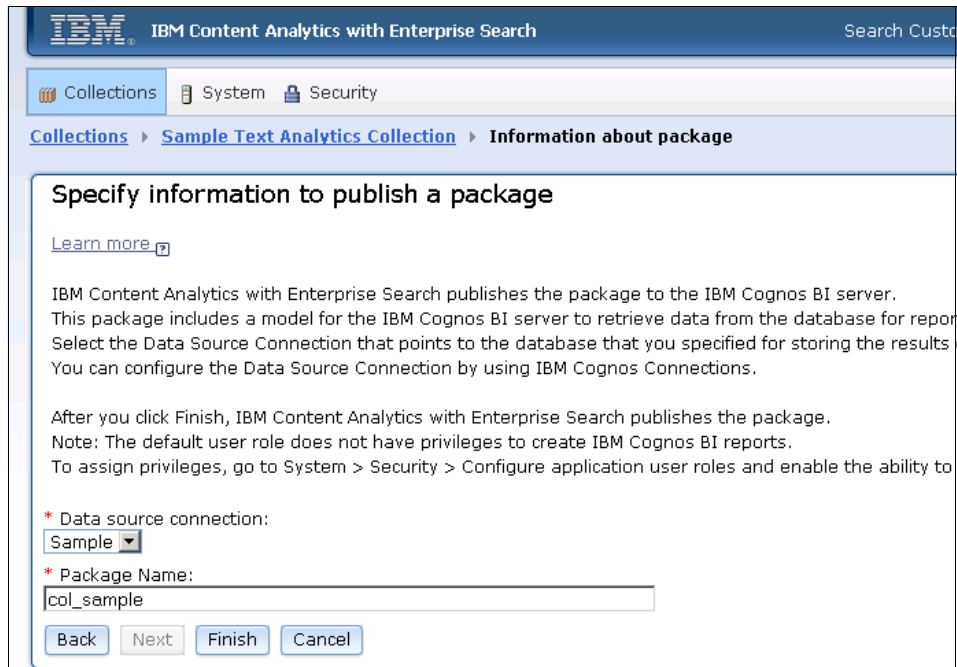


Figure 13-32 Specify information to publish a package

You have now established connectivity between Content Analytics and Cognos BI server. The Content Analytics administration console for the collection displays the configured Cognos, as shown in Figure 13-33 on page 515.

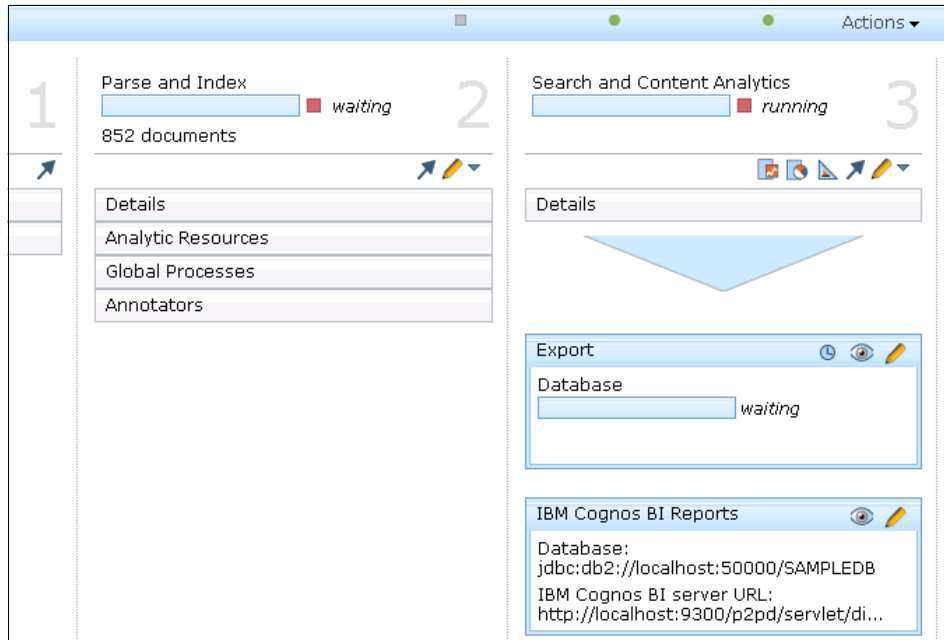


Figure 13-33 Content Analytics collection details after Cognos BI configuration

**Restarting the Content Analytics server:** After successfully creating a connection definition with the Cognos BI server, you can stop and restart the Content Analytics server, though this is not a necessary operation.

## 13.4 Generating Cognos BI reports

Now that you have successfully set up the initial configuration for connectivity with Cognos BI, you are ready to generate reports. This section explains how to generate the following predesigned Cognos BI reports for each corresponding Content Analytics view in the content analytics miner:

- ▶ Facets report
- ▶ Time series report
- ▶ Deviations report
- ▶ Trends report
- ▶ Facet pairs report

At any time during the content analytics miner discovery process, you can click a button to capture and generate a Cognos report that reflects the current state of your investigative work as seen through the current content analytics miner view. For the sake of simplicity and brevity, this section shows how to generate a Cognos BI report from the facets view only. You use the same steps to generate reports from the other views. The result from each view is a Cognos report that is designed for the corresponding content analytics miner view.

Figure 13-36 on page 517 shows a facet view of the Category facet in our sample collection. You might notice other categories indicating possible problems with the package containers. For example, many problems are related to the Package/Container Category.

To generate a Cognos report that captures these potential problems that are related to the Category facet, follow these steps:

1. Log in to content analytics miner after choosing the right collection for which Cognos BI server is configured.

If Cognos BI server is configured to require authentication and “Always use this authentication data to generate reports” is not checked during the configuration, click the **Preferences** link at the top of the window and go to the **Reports** tab. Enter the IBM Cognos BI user ID and IBM Cognos BI user password. See Figure 13-34.

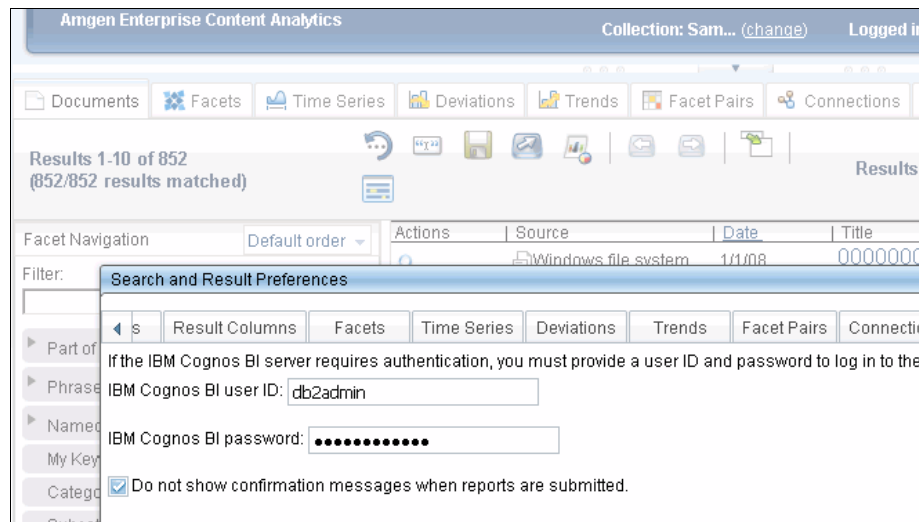


Figure 13-34 Log in to Cognos BI to work with reports

2. Ensure that a new **Reports** tab is available as a tab on the top of the content analytics miner pane, as shown in Figure 13-35.



Figure 13-35 Content analytics miner Reports tab

3. Click the **Cognos report** icon in the lower-right corner of the menu bar of the search results. As shown in Figure 13-36, the icon is displayed as a bar graph with a pie chart.

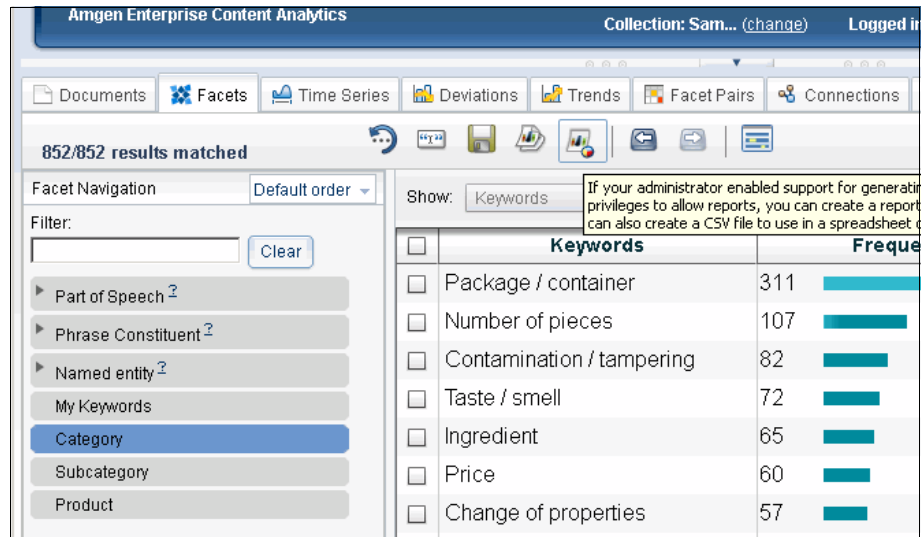


Figure 13-36 Submitting a Cognos facet report request

4. In the Create a Report window as shown in Figure 13-37, enter a meaningful name and optional description for your report. Verify that “Select an output format” is set to **IBM Cognos BI report** (the default setting). Then, click **Submit**.

The screenshot shows a dialog box titled "Create a Report". At the top, there is a close button (X). Below the title bar, the text reads: "To gain deeper insight into analyzed content, create a report that you can view with IBM Cognos BI. You can also create a report in CSV format to import into spreadsheets and other applications." Below this, there is a section "Select an output format:" with two radio buttons: "IBM Cognos BI report" (which is selected) and "CSV file". A text input field labeled "Name:" contains the text "Cognos\_report\_1386913070". Below the name field, there are two dropdown menus: "Maximum number of values (facet rows):" set to "100" and "Maximum number of results:" set to "500". To the right of these dropdowns is a section "Sort results by:" with two radio buttons: "Frequency" (selected) and "Index or correlation". Below the dropdowns is a text area labeled "Description:" which is currently empty. At the bottom right of the dialog box, there are two buttons: "Submit" and "Cancel".

Figure 13-37 Cognos report generation dialog box

A message box is displayed indicating that your report has been submitted for processing. The report is generated in the background. When it is ready, it is available for viewing from the **Reports** tab of the content analytics miner.

5. As an administrator, you can view the status of all Cognos report submissions from the Content Analytics administration console under the **IBM Cognos BI Reports** tab in the collection details, by clicking the **Eye** icon, as shown in Figure 13-38 on page 519.

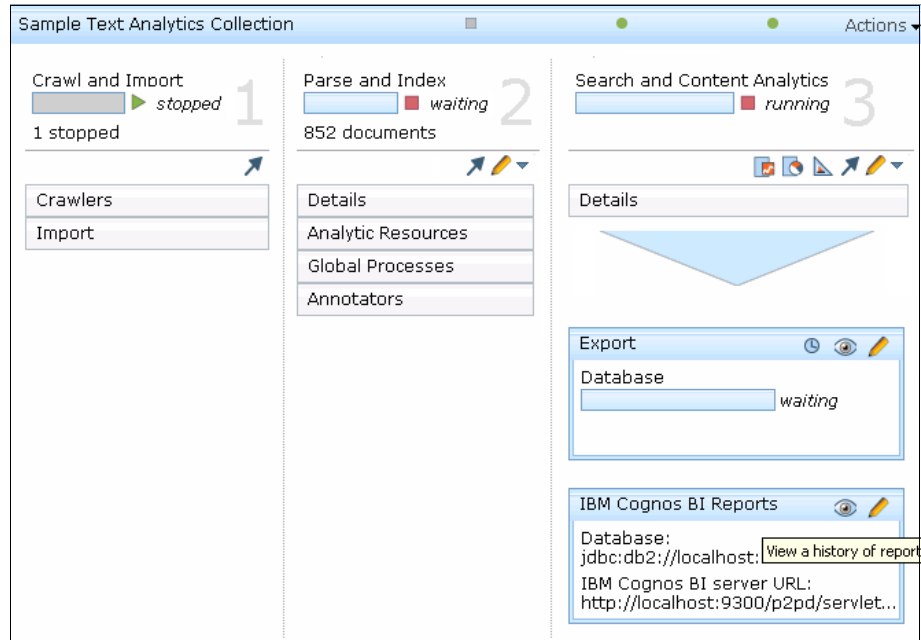


Figure 13-38 Click the Eye icon to view the details of Cognos BI generated reports

Figure 13-39 shows the status of the facet view report of Category facet for package containers. The green status icon on the far right side indicates a successful completion.

The screenshot shows the 'IBM Cognos BI Report History' page. The breadcrumb trail is 'Collections > Sample Text Analytics Collection > View the history of IBM Cognos BI report req'. The page title is 'IBM Cognos BI Report History' with a 'Learn more' link. Below the title, it says 'Last refreshed: Thursday, December 12, 2013 9:38:50 PM PST' and has a 'Refresh' button. The main content is a table titled 'IBM Cognos BI report history' with the following data:

Request ID	Report name	Stop time	User Name	Description	Status
13	Cognos_report_1386913070	12/12/13 9:38 PM	esadmin		🟢
12	Cognos_report_1386827257	12/11/13 9:48 PM	esadmin		🟢
11	Cognos_report_1386824566	12/11/13 9:03 PM	esadmin		🟢
10	Cognos_report_1386797479	12/11/13 1:31 PM	esadmin		🔴
9	Cognos_report_1386715838	12/10/13 2:50 PM	esadmin		🔴

Figure 13-39 Checking the status of submitted Cognos report generation requests

- For the business analyst working in the content analytics miner, the report normally takes about a minute or more to generate. When the report is completed, an entry for the report is displayed under the **Reports** tab in the content analytics miner as shown in Figure 13-40.

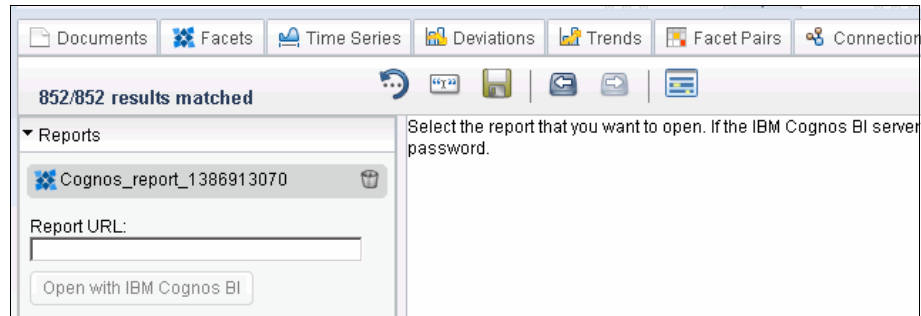


Figure 13-40 Cognos report ready to be viewed from content analytics miner

- You can click the **Open with IBM Cognos BI** button on the left side to view the selected report entry in the native Cognos Report Viewer within the content analytics miner user interface, as shown Figure 13-41.

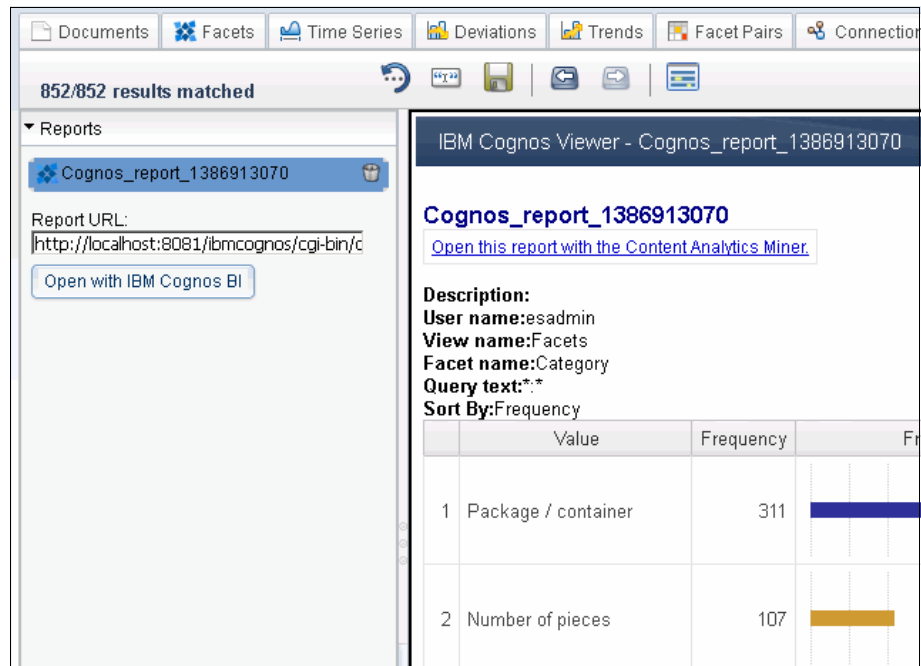


Figure 13-41 Facet report as viewed in the Cognos Report Viewer



8. The Cognos report that is shown in Figure 13-41 on page 520 includes a link underneath the report title. You can click this link to open the report in the content analytics miner. Alternatively, you can use IBM Cognos Connections URI to browse your newly generated facets reports as shown in Figure 13-42. Then, the content analytics miner is started with the same query that is used to generate the report, and the report is placed in the appropriate content analytics miner view. With this approach, you can continue where you left off in your investigative work. By using the powerful features of these navigation techniques, you can conveniently jump back and forth between the two products.

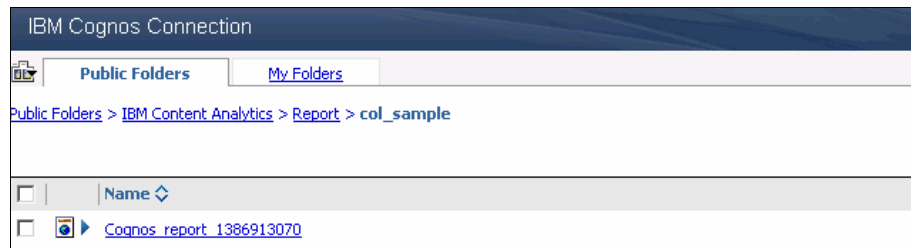


Figure 13-42 Finding a Cognos BI generated report on Cognos Connections

## 13.5 Creating custom Cognos BI reports

So far you have seen how Content Analytics works with Cognos BI to automatically generate predesigned Cognos reports. The data for the reports is stored in the relational database that you identified during the initial setup with Cognos. The reports definitions are stored in the Cognos package that you also specified during the initial setup. Consequently, it is possible to access and modify these reports from the advanced business reporting modules of Cognos BI.

A time might come when you want to build an entirely new Cognos report by using the text analytic information that was generated by Content Analytics. With Content Analytics support of this scenario, you can directly export content analytic data into a relational database and automatically generate an associated star schema for that data. With these two pieces in place, it is now business as usual for the Cognos report designer. The star schema describes the data and is used by the Cognos Framework Manager and report modules to design a broad array of reports. The data in the relational database is used to populate the newly designed reports.

This section does not explain how to build customized Cognos reports. Instead, it uses the configuration of export data to a relational database from Content Analytics as described in 13.3.4, “Configuring an export to a relational database using Content Analytics” on page 503, along with information corresponding to the star schema of the data in order for a Cognos report designer to be able to create customized reports.

The Cognos integration in this case has two major components. The first component is the relational database that is used as the repository of the exported content. The second component is the local directory where the corresponding activity log script is deposited. Now you must export some documents.

### 13.5.1 Exporting search results

This section explains how to export search result documents by using the Export button, not the Report generation button:

1. Perform a search query, for example, vanilla ice cream, and then click the **Export** button as shown in Figure 13-43.

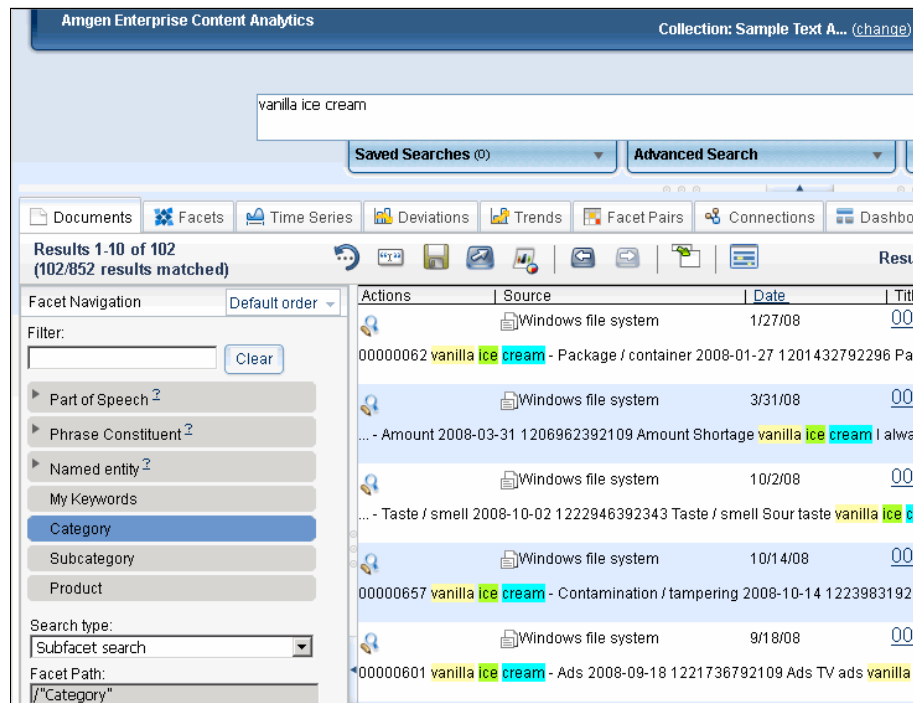


Figure 13-43 Sample search result documents to be exported

2. In the Export Search Results dialog box (Figure 13-44), complete these steps:
  - a. Enter a name for your export request.
  - b. Specify the content to be exported. In this case, we are exporting the crawled content and the parsed and analysis data. All of the fields and facets that are specified during the previous export configuration are also exported.
  - c. Optional: Enter a description.
  - d. Click **Submit**.

Export Search Results

Export search results to see metadata, binary content, and facet information about documents that match the query criteria. Your administrator determines how the results are exported, such as in XML format or according to a custom plug-in.

Name:  
results\_1387263226

Content to be exported:  
Parsed content with analysis results

Schedulable:  Yes  No

Description:

Export Cancel

Figure 13-44 Export search results options

You then see a message indicating that your export request has been submitted for background processing, as shown in Figure 13-45. Click **Close**.

Export Search Results

Export search results to see metadata, binary content, and facet information about documents that match the query criteria. Your administrator determines how the results are exported, such as in XML format or according to a custom plug-in.

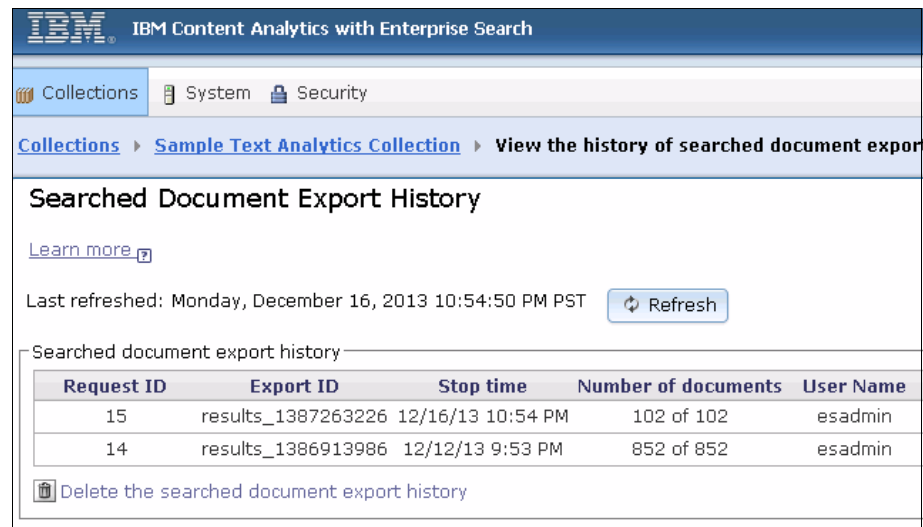
Your request to export search results was submitted. Your administrator can monitor the progress of the process.

The request ID is 15.

Export Close

Figure 13-45 Export search results job is submitted for background processing

- Using Content Analytics administration console, click the **Eye** icon for export configuration settings to determine when your export request has been processed, as shown in Figure 13-46.



IBM Content Analytics with Enterprise Search

Collections System Security

Collections > Sample Text Analytics Collection > View the history of searched document export

### Searched Document Export History

[Learn more](#)

Last refreshed: Monday, December 16, 2013 10:54:50 PM PST

Searched document export history

Request ID	Export ID	Stop time	Number of documents	User Name
15	results_1387263226	12/16/13 10:54 PM	102 of 102	esadmin
14	results_1386913986	12/12/13 9:53 PM	852 of 852	esadmin

Figure 13-46 Monitor status of export search documents job

## 13.5.2 Loading the exported data model into Cognos

Your export request has now been successfully processed by Content Analytics. Now you must examine whether the export tables have been created and populated with data by using the DB2 Control Center. Figure 13-47 on page 525 shows the product table in the sample database. As you can see on the right, the table is populated with a list of product names.

Figure 13-47 on page 525 also shows that many more tables were created by the export operation. Each table is assigned to the COL\_SAMPLE schema, which is the name provided for the schema during the configuration steps. Usage of this name is a convenient way to keep the export tables separate from other tables in the sample database.

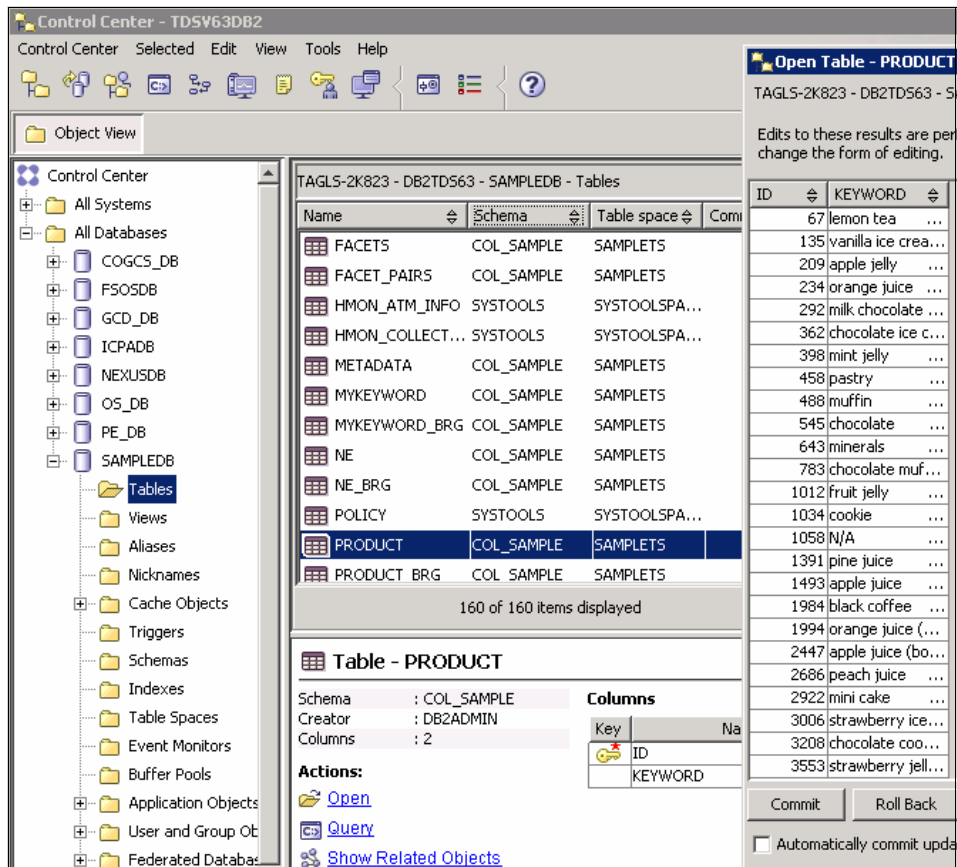


Figure 13-47 DB2 Control Center verification of export tables

You now have a relational database with several tables that are populated with data exported by Content Analytics. To generate custom Cognos reports on this data, you must first build a star schema by using the Cognos Framework Manager. The star schema describes all of the tables, their columns, and their relationships to Cognos. From this star schema, a Cognos report package can be published and made available to the Cognos Advanced and Business report modules.

### Building a star schema

To build a star schema, follow these steps:

1. Start **Cognos Framework Manager** from where Cognos Framework Manager is installed (Figure 13-48 on page 526).

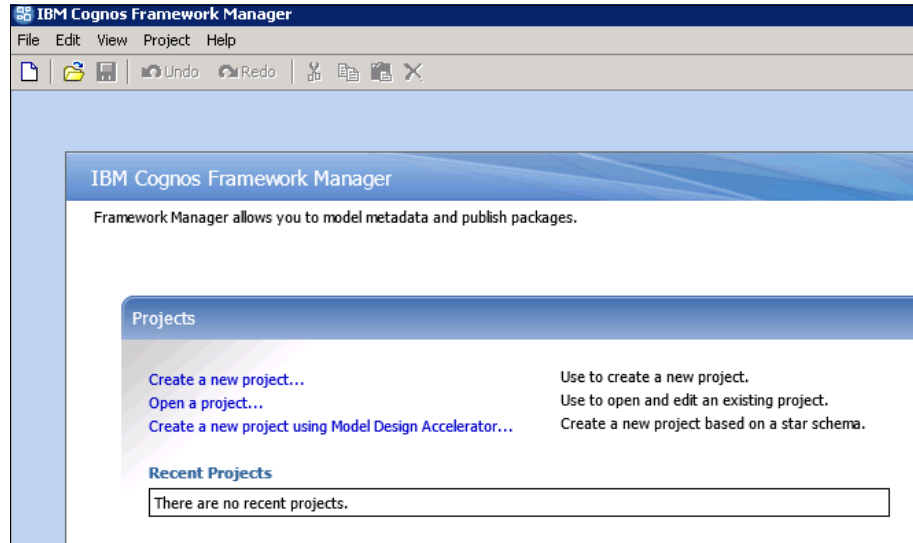


Figure 13-48 Cognos Framework Manager

2. Create a project. In this example, we create a project named CognosSample, as shown in Figure 13-49.

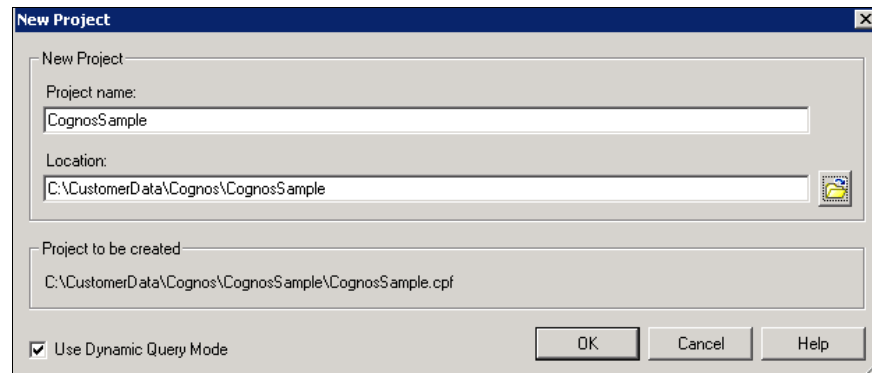


Figure 13-49 Create project named CognosSample

3. Select a language for the project, as shown in Figure 13-50.

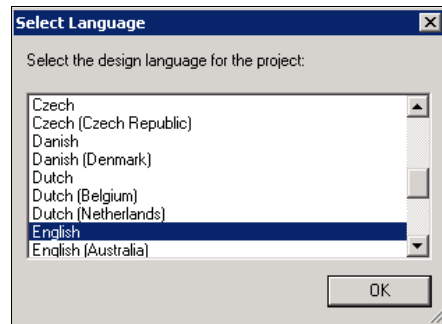


Figure 13-50 Language selection

4. Select the Metadata Source to be the **Data Source**, as shown in Figure 13-51.

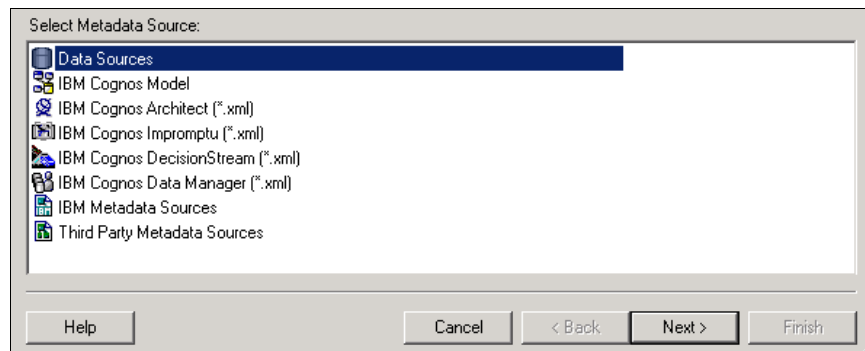


Figure 13-51 Select Metadata Source window

5. Run the activity log script that was stored in the export directory that was specified during the configuration. See the XML file generated in step #10 of 13.3.4, “Configuring an export to a relational database using Content Analytics” on page 503.
  - a. Select **Project** → **Run Script**.
  - b. In the Run Script window, select the XML file that was generated by Content Analytics. Click **Accept** to run the script.

Figure 13-52 on page 528 shows the Run Script window.

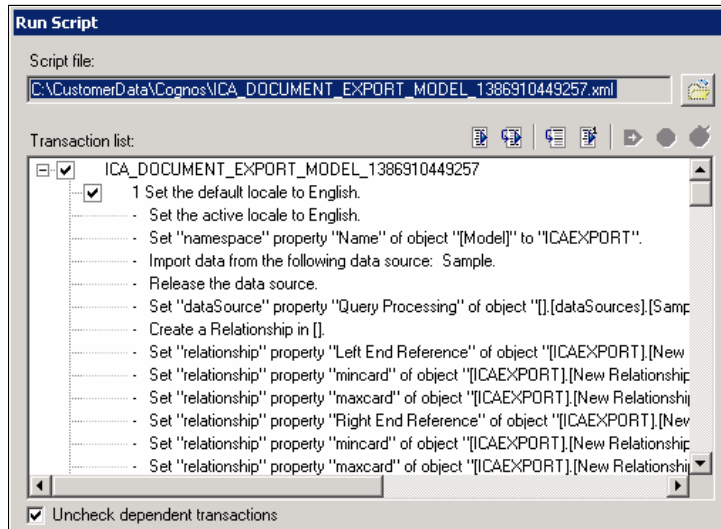


Figure 13-52 Browsing for the activity log script file

Figure 13-53 shows the results of running the activity log script.

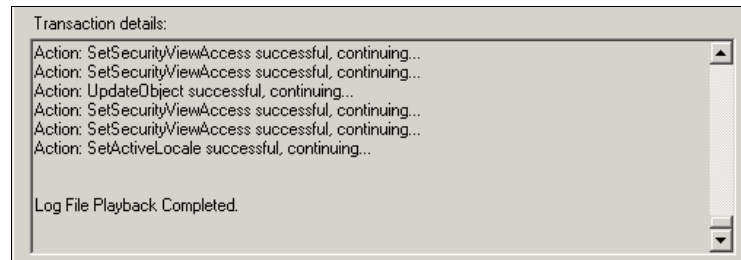


Figure 13-53 Running the activity log script



- Click the **Diagram** tab as shown Figure 13-54 to view the generated star schema.

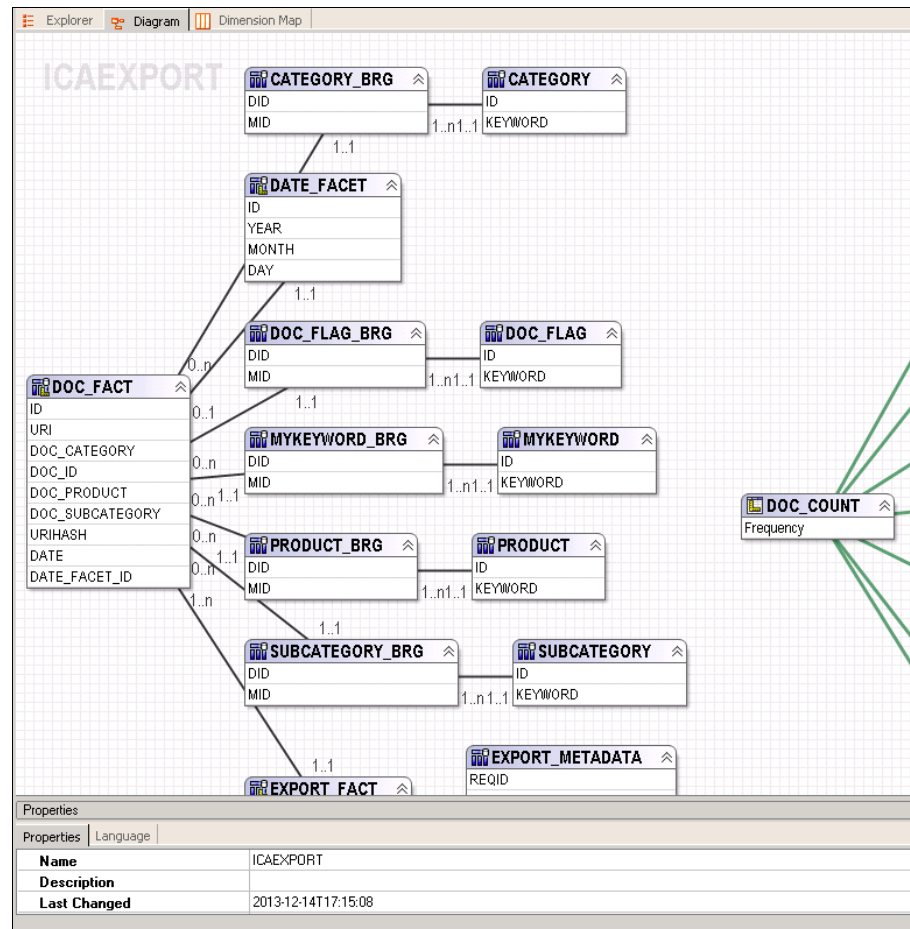


Figure 13-54 Star schema generated for Content Analytics exported data

## Publishing the package for the newly built star schema

To publish the package for the newly built star schema, follow these steps:

- Continue from the last step in “Building a star schema” on page 525; expand **Packages** on the Project viewer pane and click **col\_sample**. Publish the package for this star schema by selecting **Actions** → **Package** → **Publish packages** as shown in Figure 13-55 on page 530.

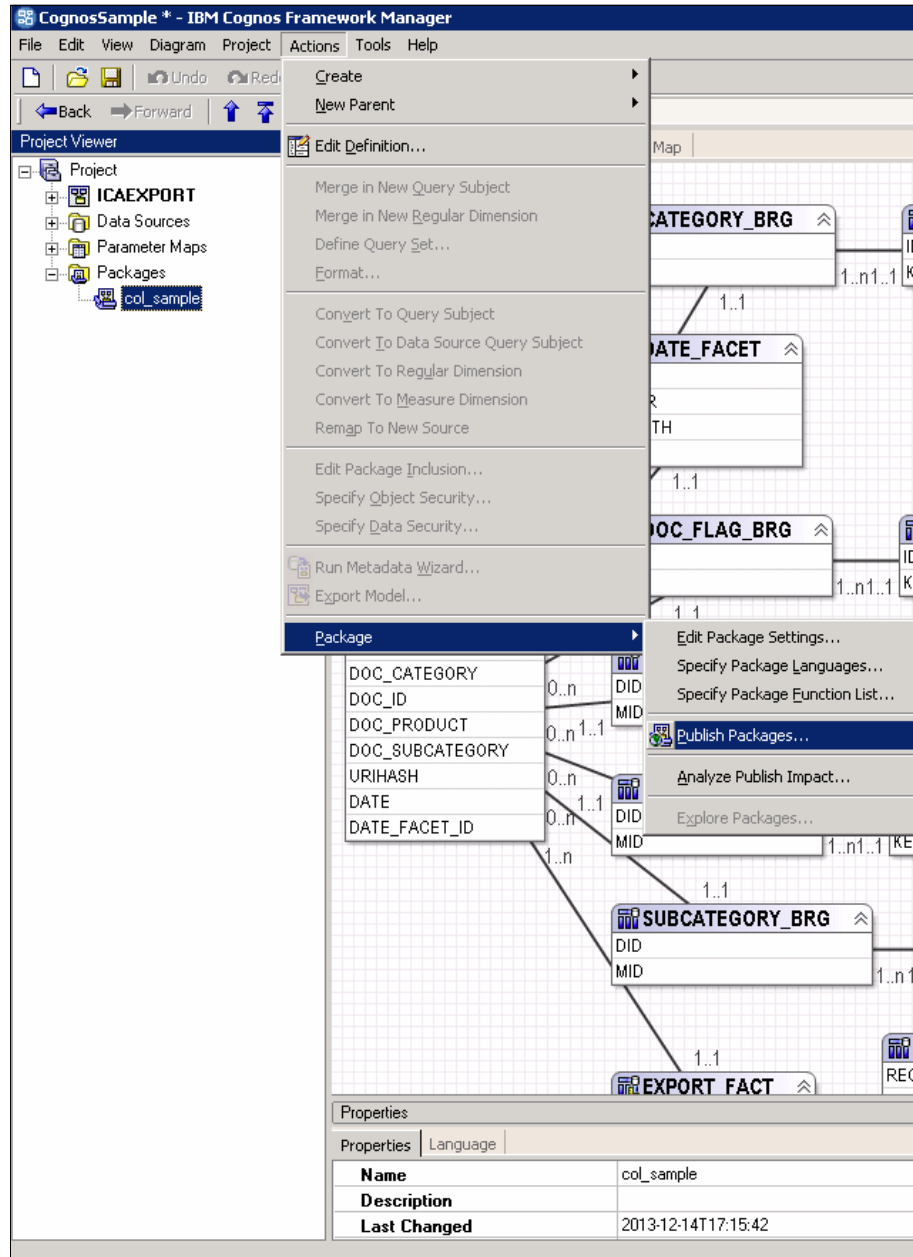


Figure 13-55 Launching publish package

2. In the Publish Wizard - Select Location Type window (Figure 13-56), select **IBM Cognos 10 Content Store**, enter a folder location, and then click **Next**.

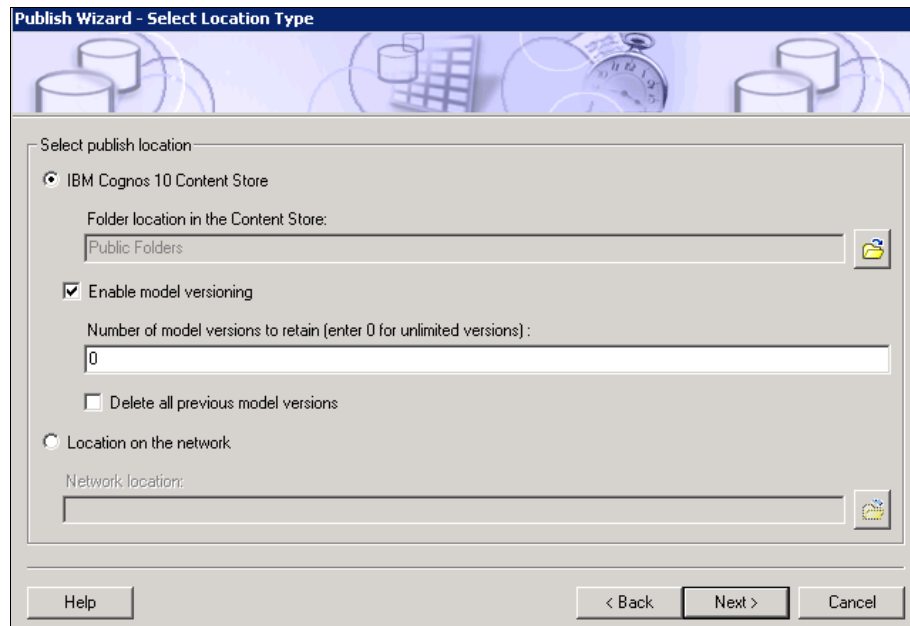


Figure 13-56 Publishing your package to a location

3. In the Publish Wizard - Add Security window (Figure 13-57), accept the defaults for security, and click **Next**.

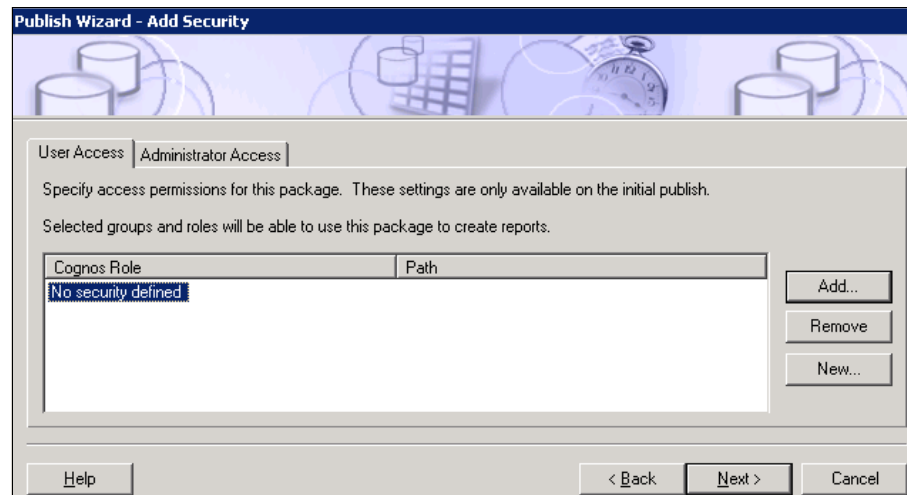


Figure 13-57 Setting package access rights

- In the Publish Wizard - Options window (Figure 13-58), accept the default to verify the package, and then click **Publish**.

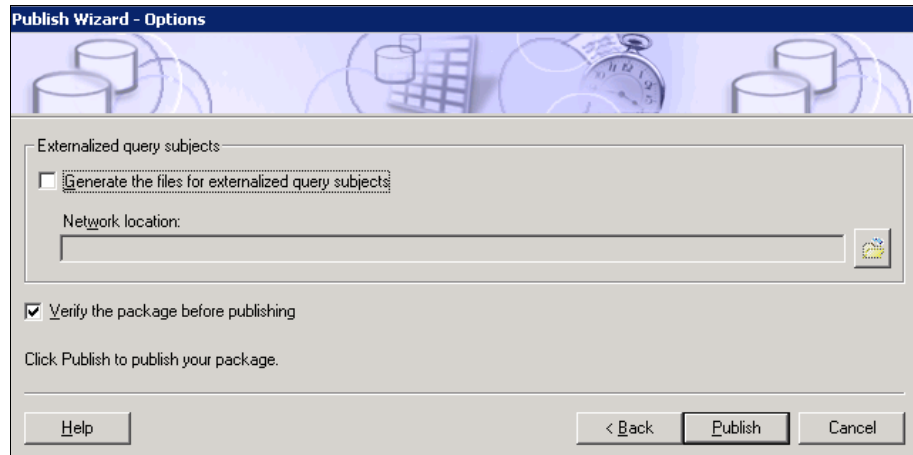


Figure 13-58 Verify the package before publishing

- Save the project and click Launch IBM Cognos. After you see the Cognos Connections browser, you can exit the wizard.
- Launch **Report Studio** from IBM Cognos Connections. See Figure 13-59.

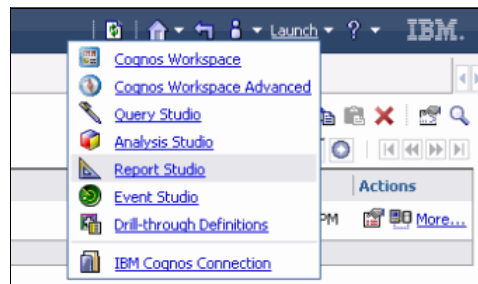


Figure 13-59 Launch Report Studio from Cognos Connections page

If the browser prompts for allowing pop-up windows for localhost, select to allow it. See Figure 13-60.

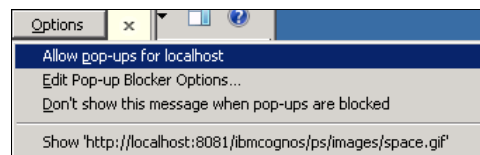


Figure 13-60 Allow pop-up windows prompt from browser

7. Select the package, col\_sample, that was recently created to be opened inside the Report Studio, as shown in Figure 13-61.

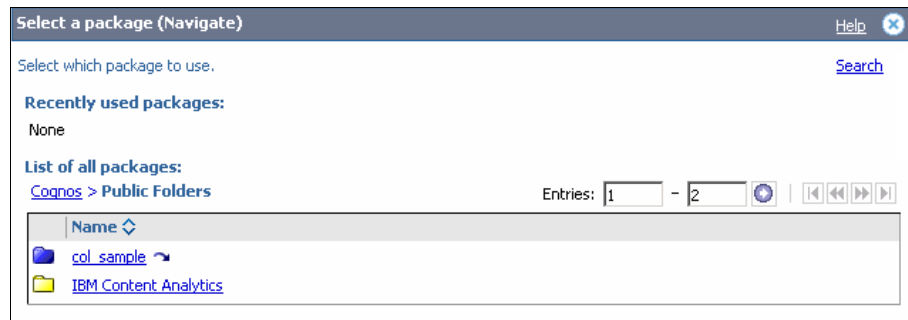


Figure 13-61 Select package name to be opened in Report Studio

## Building a custom report

Now that the COL\_SAMPLE package has been published, you can build a custom report in the Cognos Report Studio:

1. Load the report package, which is the COL\_SAMPLE report package in this example.
2. Select a simple list report to be created.
3. Drag the product table to the list report template.
4. Select **Run** → **Run report-HTML**. The list report of the product table is then displayed in the IBM Cognos Viewer as shown on the right side in Figure 13-62 on page 534.

The screenshot shows the IBM Cognos Viewer interface. On the left, the 'Insertable Objects' pane displays a tree structure of objects, with 'PRODUCT' selected. The 'Properties' pane is empty. The main workspace shows a report design area with a table structure defined by columns 'ID' and 'KEYWORD'. The 'Page layers' pane is empty. On the right, the 'IBM Cognos Viewer' window displays a list report of product data.

ID	KEYWORD
77	lemon tea
245	orange juice
376	chocolate ice cream
666	minerals
797	chocolate muffin
1013	fruit jelly
152	vanilla ice cream
408	mint jelly
1487	apple juice
1400	pine juice
1972	black coffee
2420	apple juice (bottle)
2661	peach juice
3192	chocolate cookie
3562	strawberry jelly
472	pastry
1985	orange juice (bottle)

Figure 13-62 List report of Content Analytics product data



## Customizing and extending the content analytics miner

The content analytics miner that comes with IBM Watson Content Analytics (Content Analytics) is a powerful text analysis tool that provides valuable insight to your unstructured content. If you want to customize and extend the existing functions, this chapter provides an overview of the application programming interfaces (APIs) that are available from Content Analytics and describes how you can customize and extend the application with a step-by-step tutorial.

This chapter includes the following sections:

- ▶ Reasons for custom development
- ▶ Analytics Customizer
- ▶ Creating the sample plug-in: Spatial Analysis

## 14.1 Reasons for custom development

The content analytics miner provides many different views from which to analyze your data, ranging from time series view, trend pattern analysis views to various facet correlation views. Although there are many readily available functions, you might want to extend the content analytics miner for several reasons.

First, you might use several tools (of which Content Analytics is just one of them) in the analysis of your data. If these tools are accessible by using a web browser or via programmable API, you can incorporate these tools into the content analytics miner and make them accessible from their own tabbed views. By doing this, the content analytics miner can consolidate your various analytic tools into a single portal. This approach greatly simplifies the switching back and forth between tasks.

In the opposite spectrum, you might want to integrate the content analytics miner or functionalities into your existing analytics tools and infrastructure.

Another reason could be, a more visualized view might better serve your data and analysis needs and thus there is a business requirement to implement such a view. In this case, you want to create your own view with the data provided by Content Analytics. Such visualization can, for example, be a word cloud plug-in as shown in Figure 14-1.

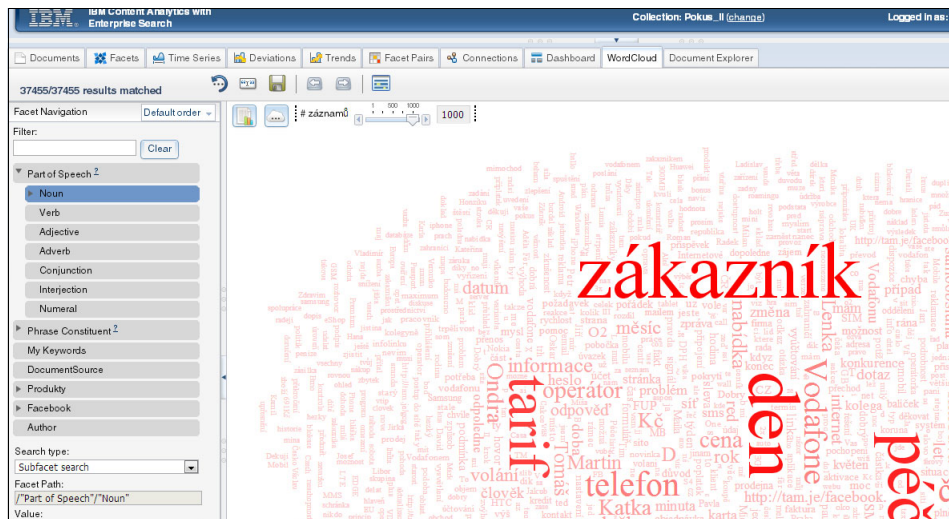


Figure 14-1 Custom visualization example: A word cloud plug-in





## 14.2 Analytics Customizer

Customizing the content analytics miner is useful when you examine the content analytics miner during the testing stage.

**Limiting user access to specific collections:** You can limit user access to specific collections for security purposes.

Content Analytics provides Analytics Customizer to help you customize parts of the content analytics miner. The advantage of using Analytics Customizer is that you can quickly update the properties that are used frequently by using a GUI. You do not have to edit the configuration file directly. You can examine and change the properties during the testing period.

Accessing the Analytics Customizer depends on your deployment. If you use the deployed Jetty web server, click the **Analytics Customizer** link in the upper right corner of the administration console (Figure 14-3).



Search Customizer | Analytics Customizer | Log Out | Help | About

Figure 14-3 Linking to the Analytics Customizer in the administration console

When you open the Analytics Customizer, you can update the following options on each tab:

- ▶ Server: Specify the host name, protocol (HTTP or HTTPS), logging level, and query timeout in seconds.
- ▶ Screen: Specify images and texts, links, and paths to view in the window, or the order of the tabs.
- ▶ Query options: Specify the query options, type ahead, or the file type filters.
- ▶ Results: Customize the look of results list.
- ▶ Result filters: Modify result filters for titles or URLs.
- ▶ Images: Specify the image files for the data sources.
- ▶ Facets: Specify the number of facets and their order.
- ▶ Time series: Specify the default time series scale.
- ▶ Deviations: Specify the parameters for the deviations tab.
- ▶ Trends: Specify the parameters for the trends tab.
- ▶ Facet pairs: Specify the default values for the facet pairs tab.

- ▶ Connections: Specify the default values for the connections tab.
- ▶ Dashboard: Create or modify custom dashboards for your collections.
- ▶ Reports: Specify the connections details to IBM Cognos BI Report.
- ▶ Sentiment: Specify the default values for the sentiment tab.

After you update the values, click **Close** in the Analytics Customizer window. To keep your settings, click **Save changes**. To discard them, click either on **Cancel changes** or **Exit**.

If the customization is finalized, and you do not want to allow further modifications by other users, you can set the `customizerDisabled` property to **true**.

## 14.3 Creating the sample plug-in: Spatial Analysis

Extending the content analytics miner follows a plug-in architecture where each plug-in creates a view tab on the results menu bar.

In this section, we walk you through the process of creating the Spatial Analysis plug-in step-by-step so that you see all important steps.

The goal is to create a plug-in that visualizes documents in Content Analytics on a map and allow us to incorporate their spatial information for our analysis. In the end, we will be able to limit documents in Content Analytics query by the location from which they originated. This helps us to do further analysis of the documents for a specific location.

**Disclaimer:** IBM does not warrant or represent that the code in this section is complete or up-to-date. IBM is under no obligation to update content nor provide further support.

All code is provided “as is,” with no warranties or guarantees whatsoever.

### 14.3.1 Preparation

Before we begin, we assume that you have at least the basic knowledge of JavaScript, Dojo, and CSS techniques for styling HTML DOM.

The content analytics miner is created using Dojo Toolkit. This is what we also use for our plug-in development. If you are not familiar with this technology, make sure that you gain the basic knowledge before you continue.

## Data source preparation

To create this plug-in, we need to know the location of each document. This example is based on the data that is acquired from publicly available Twitter feeds. The structure of the data is shown Figure 14-4. The `tw_coordinates` field provides the location of the document.

Source	Date	Title
<code>type: text/plain</code>		
<code>tw_retweeted: false</code>		
<code>tw_document_source: twitter</code>		
<code>tw_geo_longitude: 2.25425617</code>		
<code>tw_user_screen_name: [REDACTED]</code>		
<code>tw_document_source_specific_id: [REDACTED]</code>		
<code>tw_favorited: false</code>		
<code>tw_place_fullname: Paris, Paris</code>		
<code>tw_create_date: Mon Jul 15 12:19:43 CEST 2013</code>		
<code>tw_geo_latitude: 48.83914094</code>		
<code>tw_possibly_sensitive: false</code>		
<code>title: [REDACTED]</code>		
<code>tw_document_type: tweet</code>		
<code>tw_place_country: France</code>		
<code>tw_tweet_source: &lt;a href="http://twitter.com/download/iphone" rel="nofollow"&gt;Twitter for iPhone&lt;/a&gt;</code>		
<code>tw_favorite_count: 0</code>		
<code>tw_place_country_code: FR</code>		
<code>tw_orig_url: http://twitter.com/[REDACTED]/status/[REDACTED]</code>		
<code>tw_place_type: city</code>		
<code>date: 7/15/13</code>		
<code>tw_retweet_count: 0</code>		
<code>tw_contributors: null</code>		
<code>tw_place_street: null</code>		
<code>tw_document_source_specific_name: [REDACTED]</code>		
<code>tw_place_name: Paris</code>		
<code>tw_coordinates: 48.83914094, 2.25425617</code>		

Figure 14-4 Structure of documents with location information in metadata

To make the example simple for ease of understanding, we index the `tw_coordinates` field and create a facet for the location field. This makes the location coordinates to be easily accessible via the Search REST API. Figure 14-5 on page 541 shows what the data looks like.

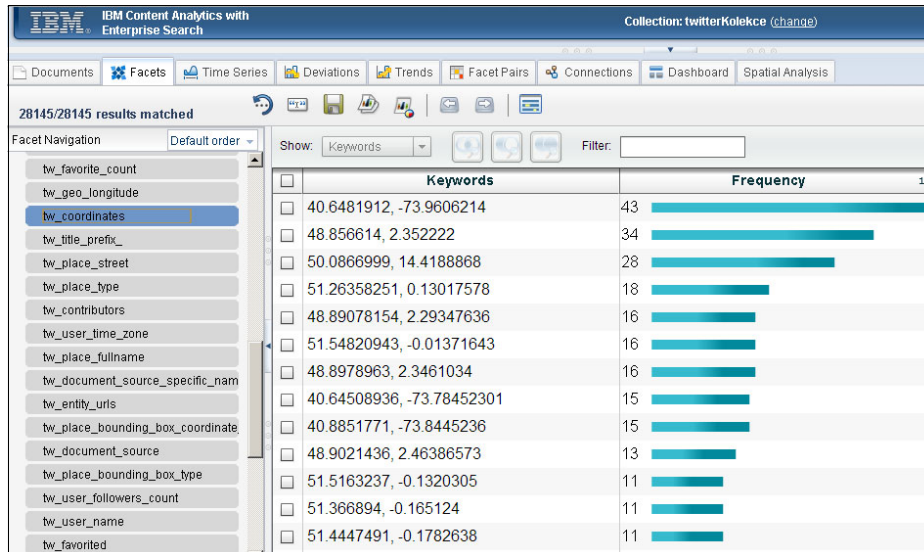


Figure 14-5 Input data for the sample Spatial Analysis plug-in: Facet with coordinates

As you can see in Figure 14-5, each tweet comes with information about its sent location combined in one attribute `tw_coordinates`. We use this value to show the document in the correct location on the map.

## Plug-in files preparation

To set up all necessary files, follow these steps:

1. Create a new folder “spatial” under the following directory:  
ES\_NODE\_ROOT/master\_config/searchapp/analytics/plugin/view
2. Inside the “spatial” folder, create three files:

```
plugin.js
plugin.html
style.css
```

3. Edit file `plugin.js` by adding the following lines:

```
dojo.provide("view.spatial.plugin");
dojo.require("widgets.analytics._AnalyticsPane");
dojo.declare("view.spatial.plugin",
widgets.analytics._AnalyticsPane, {
  templatePath: dojo.moduleUrl("view.spatial", "plugin.html"),
  onShow: function(){
    this.inherited("onShow", arguments);
  }
})
```

```
});
```

4. Edit file `plugin.html` by adding the following lines:

```
<div>
  <!-- Reference to cascade style sheet file -->
  <link rel="stylesheet" type="text/css"
href="wplugin/view/spatial/style.css" />

  <div id="spatialHeader"></div>
  <div id="mapHolder"></div>
</div>
```

5. Edit file `style.css` with the following line:

```
#spatialHeader{line-height:25px; height:25px; background:#00649D;}
```

6. Edit the `plugins.xml` file in the following directory:

`ES_NODE_ROOT/master_config/searchapp/analytics/plugin/view/plugins.xml`

Add the following lines to the file:

```
<plugin id="spatial" title="Spatial Analysis"
titleKey="resource.key.spatial" >
  <mainComponent file="plugin.js" dojoType="view.spatial.plugin"/>
</plugin>
```

Restart Content Analytics. Now you will see a new, empty tab in the content analytics miner, called “Spatial Analysis”.

## 14.3.2 Plug-in structure

Before we dive deeper into the code, we need to understand the structure of plug-in files and learn what functions can be called from within plug-ins to interact with Content Analytics.

The most important file for us is the JavaScript file, which in our case is the `plugin.js` file. The code in this file defines how the plug-in behaves, but it needs to be structured in the format of a Dojo class. The current content of the `plugin.js` file is the minimum version of a plug-in class. It does nothing as is. To add more functions, the default functions that Content Analytics provides can be used or overridden throughout our own code.

## Plug-in class default functions

The default functions can be segmented into four groups according to the way that they should be used in the code:

- ▶ Functions that you can and should override to get the wanted behavior:

```
/* Resize is triggered whenever content analytics miner windows is resized */
resize: function(){...},
/* onShow is triggered whenever the plugin is shown */
onShow: function(){...},
/* onHide is triggered whenever the plugin is hidden */
onHide: function(){...},
/* onSelectedFacetChange is triggered whenever user changes selected
facet */
onSelectedFacetChange: function(){...}
```

- ▶ Functions that you can call to get some value:

```
this.getSelectedFacetId(num /*0=row, 1=column*/);
this.getSelectedFacetType(num /*0=row, 1=column*/);
this.getSelectedFacetLabel(num /*0=row, 1=column*/);
this.getSelectedCollection();
this.getSelectedCollection().id;
this.getCurrentQuery();
this.getCurrentQueryLang();
/* is true when the plugin is currently visible */
if(this.isVisible);
```

- ▶ Events, to which you can listen to:

```
this.subscribe("refresh", function(){...});
this.subscribe("collectionChanged", function(){...});
this.subscribe("reset", function(){...});
this.subscribe("tabChanged", function(){...});
this.subscribe("verticalFacetChanged",function(){...});
this.subscribe("horizontalFacetChanged",function(){...});
```

- ▶ Functions that you can call to do something (advanced):

```
/* submits search query */
diigit.byId(EDR.prefix+"searchManager").submitQuickRefineSearch({
    "keywords": myQuery,
    "operator": operator,
});
```

For the Spatial Analysis plug-in, we need only a fraction of these methods:

- ▶ onShow method
- ▶ resize method

- ▶ `getCurrentQuery` method
- ▶ `submit search query` method

In addition, we also need to use the Search REST API for fetching the actual values (coordinates) for documents in the search query. After this step, the `plugin.js` file should be modified as follows:

```
dojo.provide("view.spatial.plugin");
dojo.require("widgets.analytics._AnalyticsPane");
dojo.declare("view.spatial.plugin", widgets.analytics._AnalyticsPane, {
    templatePath: dojo.moduleUrl("view.spatial", "plugin.html"),
    onShow: function(){
        this.inherited("onShow", arguments);

    },
    setListeners: function(){
        this.subscribe("refresh", reload);
        this.subscribe("collectionChanged", reload);
    },
    /* This method takes care of reloading all data when necessary */
    reload: function(){

    },
    resize: function(){

    }
});
```

### 14.3.3 Adding a map to the plug-in

The first step in creating the Spatial Analysis plug-in is to place a map to the plug-in. It should fill the entire space, except for a small line at the top, which will be used for a header later on.

Since we use Google Maps, we need to load all the associated libraries first and initialize the map. This can be a rather tricky part for most of the new developers because the plug-in files are loaded after the entire page (with content analytics miner code) has been loaded and more importantly after the HEAD of this page has been loaded. The default way of loading third-party JavaScript libraries is to put a reference to the HTML HEAD, which is not possible in this case. Also, Google Maps libraries load asynchronously, which adds to the challenge of loading them correctly from the scope of the plug-in. So we need to do a little more work to load them, as follows:

1. Use a Dojo Deferred object to catch the asynchronous load:



```

this.googleLoaderDef = new dojo.Deferred();
this.loadGoogleMaps();
this.googleLoaderDef.then(function() {
    /* place the map */
    _this.drawGoogleMap();
});

```

2. Place the SCRIPT tag to the header manually from the code:

```

loadGoogleMaps: function() {
    var _this = this;
    /* if Google has not yet been loaded, load it*/
    if (window["google"] == undefined) {
        window["SA_plugin"] = function() {
            _this.googleLoaderDef.resolve();
        };
        var script = document.createElement("script");
        script.type = "text/javascript";
        script.src =
"https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=false&libraries=visualization&callback=SA_plugin";
        document.body.appendChild(script);
    } else {
        /* Google is already loaded, resolve the deferred object*/
        this.googleLoaderDef.resolve();
    }
}

```

**Note:** If you copy the source code as shown above and paste it directly to an editor, additional line breaks might be added to the code. As a result, the final code might not work and requires your manual modification.

For example, in the above example, "https://maps.googleapis.com/...=false&libraries=visualization&callback=SA\_plugin" should all be one line. Make sure you remove the line break between them after you paste the code to your editor.

Alternatively, you can use the following code instead:

```

"https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=false&librar"+
"ies=visualization&callback=SA_plugin";

```

3. Create the resize function now, so the map gets resized correctly with every window size change:

```

resize: function() {
    /* HEIGHT - header size*/

```

```

25;         var newHeight = dojo.contentBox(dojo.byId("spatial")).h -
           dojo.style(
             dojo.byId("mapHolder"),
             "height",
             newHeight+ "px");

           if (this.isVisible)
             google.maps.event.trigger(this.map, "resize");
         }

```

4. Create the map and place it to the DOM:

```

drawGoogleMap: function() {
    var options = {
        zoom: 6,
        /* Prague */
        center: new google.maps.LatLng(50.08182, 14.440734),
        mapTypeId: google.maps.MapTypeId.ROADMAP
    };

    this.map = new google.maps.Map(dojo.byId("mapHolder"),
options);
}

```

### 14.3.4 Displaying documents on the map

Now, when we have the map in place, we want to display the documents on the map as markers in the respective location. To do that, we need to get the data first, parse them, and then place them on the map.

#### Getting the data

The right way to get the data of documents, currently matching the Content Analytics query, is to use the Search REST API. We need the following method:

`/api/v10/search/facet` with parameters:

```

collection=Current collection ID
output=application/json
facet={"namespace":"keyword","id":Facet ID,"depth":3,"count":number
of documents}
query=Current query

```

As you can see, we need to get some information from the Content Analytics environment before we can call this API. Namely, we need to know the ID of the currently selected collection, ID of a facet with location information (metadata),

wanted number of documents that the API call should return (the more documents, the more markers on the map) and last, but not least, the current query. All this can be done by using plug-in default functions as follows:

```
fetchData: function(){
    var _this = this
    return dojo.xhrGet({
        url: "/api/v10/search/facet?collection=" +
        _this.getSelectedCollection().id +
        "&output=application/json&facet={\"namespace\": \"keyword\", \"id\": \"$.T
        witter.tw_coordinates\", \"depth\": 3, \"count\": 1000}&query=" +
        _this.getCurrentQuery(),
        handleAs: "json",
        headers: {"Content-Encoding": "UTF-8"},
    });
}
```

### Parsing location information from document metadata

To parse the location information from the document metadata, use the following code:

```
if (data.es_apiResponse === null)
    return;

data.es_apiResponse.ibmesc_facet.ibmesc_facetValue.forEach(function(value
) {
    if (value.label == "null") return;
    /* coordinates should be in a format -41.415, 45.21212 */
    var position = value.label.indexOf(",");
    var latitude = parseFloat(value.label.substring(0, position));
    var longitude = parseFloat(value.label.substring(position + 2,
value.length));
});
```

### Creating markers and placing the documents on the map

To create markers and place the documents on the map, use the following code:

```
var markerpoint = new google.maps.LatLng(latitude, longitude);
var marker = new google.maps.Marker({position: point});
marker.setMap(_this.map);
```

## 14.3.5 The entire code for the Spatial Analysis plug-in so far

Example 14-1 on page 548 shows the code that we have so far for the Spatial Analysis plug-in.

```
dojo.provide("view.spatial.plugin");
dojo.require("widgets.analytics._AnalyticsPane");
dojo.declare("view.spatial.plugin", widgets.analytics._AnalyticsPane, {
  templatePath: dojo.moduleUrl("view.spatial", "plugin.html"),
  onShow: function() {
    var _this = this;
    this.inherited("onShow", arguments);

    this.googleLoaderDef = new dojo.Deferred();
    this.loadGoogleMaps();
    this.googleLoaderDef.then(function() {
      /* place the map */
      _this.drawGoogleMap();
      dojo.hitch(_this, "reload")();
    });
  },
  setListeners: function() {
    _this = this;
    this.subscribe("refresh", dojo.hitch(_this, "reload"));
    this.subscribe("collectionChanged", dojo.hitch(_this,
"reload"));
  },
  /* This method takes care of reloading all data when necessary */
  reload: function() {
    this.getCurrentData();
  },
  getCurrentData: function() {
    var _this = this;
    this.fetchData().then(function(data) {
      console.log(data);
      if (data.es_apiResponse === null)
        return;

data.es_apiResponse.ibm_sc_facet.ibm_sc_facetValue.forEach(function(value
) {
      if (value.label == "null")
        return;
      /* coordinates should be in a format -41.415,
45.21212 */
      var position = value.label.indexOf(",");
      var latitude = parseFloat(value.label.substring(0,
position));
```

```

        var longitude =
parseFloat(value.label.substring(position + 2, value.length));

        var point = new google.maps.LatLng(latitude,
longitude);

        var marker = new google.maps.Marker({position: point});
        marker.setMap(_this.map);
    });
});
},
fetchData: function() {
    var _this = this
    return dojo.xhrGet({
        url: "/api/v10/search/facet?collection=" +
_this.getSelectedCollection().id +
"&output=application/json&facet={\"namespace\": \"keyword\", \"id\": \"$.T
witter.tw_coordinates\", \"depth\": 3, \"count\": 1000}&query=" +
_this.getCurrentQuery(),
        handleAs: "json",
        headers: {"Content-Encoding": "UTF-8"},
    });
},
resize: function() {
    /* HEIGHT - header*/
    var newHeight = dojo.contentBox(dojo.byId("spatial")).h - 25;
    dojo.style(
        dojo.byId("mapHolder"),
        "height",
        newHeight + "px");

    if (this.isVisible && window['google'] != undefined)
        google.maps.event.trigger(this.map, "resize");
},
drawGoogleMap: function() {
    var options = {
        zoom: 6,
        /* Prague */
        center: new google.maps.LatLng(50.08182, 14.440734),
        mapTypeId: google.maps.MapTypeId.ROADMAP
    };

    this.map = new google.maps.Map(dojo.byId("mapHolder"),
options);
},
loadGoogleMaps: function() {

```

```

var _this = this;
/* IF google isn't already loaded */
if (window["google"] == undefined) {
    window["SA_plugin"] = function() {
        _this.googleLoaderDef.resolve();
    }
    var script = document.createElement("script");
    script.type = "text/javascript";
    script.src =
"https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=false&libraries
=visualization&callback=SA_plugin";
    document.body.appendChild(script);
} else {
    this.googleLoaderDef.resolve();
}
}
})

```

---

### 14.3.6 Adding selection mode

The plug-in that we created so far shows documents on a map. In this section, we add one more functionality to it: An interactive selection mode that allows a user to limit the Content Analytics search results only to documents within a certain geographic area. Enhancing the plug-in with the new function consists of four steps:

1. Preparation work.
2. Implement selection mode on the map.
3. Decide which documents belong to the selected geographic scope.
4. Submit refined query to Content Analytics.

#### Preparation work

The preparation work involves adding two buttons to the plug-in. The first button toggles the selection mode. The other button submits the AND query to Content Analytics. These buttons can be added by modifying the `plugin.html` file.

To add the buttons, we add the following code inside of the `<div id="spatialHeader"></div>` tag:

```

<a class="button blue" id="selectionButton">Selection mode</a>
<a class="button blue" id="and">AND</a>

```

To make the buttons look better, we add some CSS styles. For example, these (modifying the `style.css` file):

```
.button.blue{background-color: #00649D; display:inline-block;
width:100px; text-align:center; color:white;}
.button.blue:hover{background-color: #f63b00; text-decoration:none;
color:white;}
```

If we now restart Content Analytics, we should get a result that is similar to Figure 14-6.

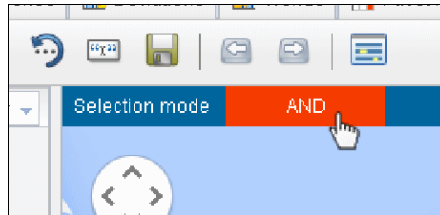


Figure 14-6 Plug-in with the new buttons

## Implementing selection mode

Before we start with this part of the task, we need to think first. Our goal is to select a square on the map (square because any other unusual shape would be later hard to put to simple Content Analytics query). To create a square, you need four points, but you do not necessarily need to click four times. You need to only click twice to select the opposite corner of the square because the other two points can be calculated from them (they have latitude of the first point and longitude of the other and the other way around for the second missing point).

Let us break down this step to several substeps:

1. Create a listener for the **Selection mode** button.

To create the listener, switch to the `plugin.js` file and update the `setListeners` so that it looks like this:

```
setListeners: function() {
    var _this = this;
    this.subscribe("refresh", dojo.hitch(_this, "reload"));
    this.subscribe("collectionChanged", dojo.hitch(_this,
"reload"));
    this.connect(dojo.byId("selectionButton"), "onclick",
dojo.hitch(_this, "selectionButtonClicked"));
}
```

Add the function `selectionButtonClicked`. See the next step.

2. Create a listener for the map to catch where a user clicks and places a marker.

Now we know when a user wants to start with the selection on the map. Because we use Google maps as our mapping engine, we need to reuse its API to listen to onClick events. We implement this in our newly created selectionButtonClicked() function. Do not forget that our map object is stored in the JavaScript variable "map" that is available within the scope of the class (therefore accessible via this.map):

```
selectionButtonClicked: function(){
    var _this = this;
    google.maps.event.addListener(this.map, "click", function() {
        /* TODO */
    });
};
```

3. After selecting two points, complete the square on the map.

This method is rather complex, but every step is commented directly in the code. We reuse Google API for the creation of the polygon (square) on the map. The complete method looks like Example 14-2.

*Example 14-2 Code for the selectionButtonClicked*

---

```
selectionButtonClicked: function(){
    var _this = this;
    /* Initialization a variable which will help us to create polygon
    (square) over the Google maps */
    this.polyLine = new google.maps.Polyline({ strokeColor: "#000000",
strokeOpacity: 1.0, strokeWeight: 3 });
    this.polyLine.setMap(this.map);
    /* Few variable we will need in the process */
    this.mapClickCounter = 0;
    this.selectionPoints = new Array();
    this.selectionPathMarkers = new Array();
    /* Add listener on the map */
    google.maps.event.addListener(this.map, "click", function(event) {
        /* Increment number of clicks */
        _this.mapClickCounter++;
        /* If user clicks more than twice, just do nothing */
        if (_this.mapClickCounter > 2)
            return;

        /* On the first click, just create the marker and place it to
        the map */
        if (_this.mapClickCounter == 1) {
            var marker = new google.maps.Marker({
                position: event.latLng,
                map: _this.map
            });
        }
    });
};
```



```

        /* Save both marker and Latitude/Longitude information to
an array. We will need it later */
        _this.selectionPathMarkers.push(marker);
        _this.selectionPoints.push(event.latLng);
    } else {
        /* Here we need to decide whether the second point has
higher/lower latitude or longitude than the first point*/
        var currentPoint = event.latLng;

        /* In order to create the query in later steps correctly,
we need to know the maximum & minimum latitude & longitude. Let's save
it to four variables */
        _this.latitudeMax = _this.selectionPoints[0].lat() >
currentPoint.lat() ? _this.selectionPoints[0].lat() :
currentPoint.lat();
        _this.latitudeMin = _this.selectionPoints[0].lat() <
currentPoint.lat() ? _this.selectionPoints[0].lat() :
currentPoint.lat();
        _this.longitudeMax = _this.selectionPoints[0].lng() >
currentPoint.lng() ? _this.selectionPoints[0].lng() :
currentPoint.lng();
        _this.longitudeMin = _this.selectionPoints[0].lng() <
currentPoint.lng() ? _this.selectionPoints[0].lng() :
currentPoint.lng();

        /* This is a second click on the map, we need to create the
selection polygon from all points.*/
        /* First corner will have the latitude of the first point
and longitude of the second point */
        var firstCorner = new
google.maps.LatLng(_this.selectionPoints[0].lat(), currentPoint.lng())
        /* Second corner of the polygon will have latitude of the
second point and longitude of the first point */
        var secondCorner = new
google.maps.LatLng(currentPoint.lat(), _this.selectionPoints[0].lng());
        /* We need to add all newly created points to the array so
we can create the polygon out of them */
        _this.selectionPoints.push(firstCorner, currentPoint,
secondCorner);

        /* Traverse all points and for each, create a marker &
place it to the polygon */
        for (var i = 0; i < _this.selectionPoints.length; i++) {
            _this.polyLine.getPath().push(_this.selectionPoints[i]);

```

```

        /* Marker for the first point has already been created,
skip it */
        if (i > 0) {
            var marker = new google.maps.Marker({position:
_this.selectionPoints[i], map: _this.map});
            _this.selectionPathMarkers.push(marker);
        }
    }
}

/* And now, close the loop (so it makes the square) */
_this.polygon = new google.maps.Polygon({
paths: _this.polyLine.getPath(),
map: _this.map,
strokeColor: "#FF0000",
strokeOpacity: 0.8,
strokeWeight: 2,
fillColor: "#00649D",
fillOpacity: 0.35
});
});
}

```

Restart Content Analytics and make sure that the plug-in is working as expected. To test, click **Selection mode**, and then select two points on the map, and the new plug-in code creates a square out of it. The result looks similar to the markup window as shown in Figure 14-7.

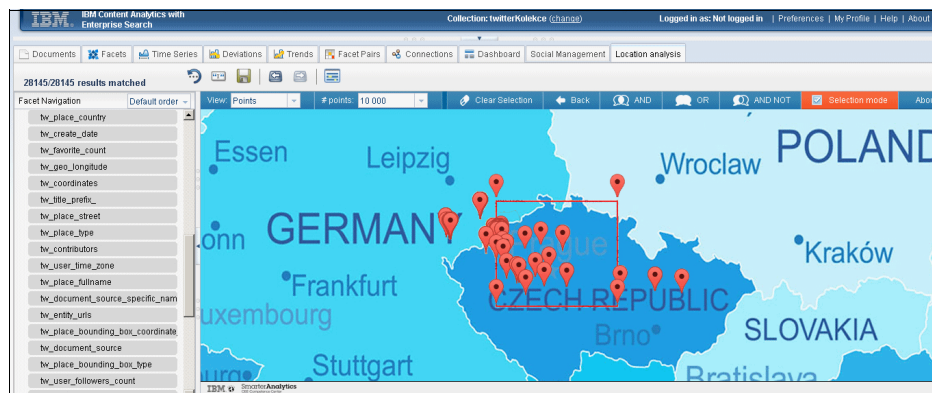


Figure 14-7 Markup of the resulting window if the user uses the selection mode

## Creating query

In this part, we assume that the selection mode is working properly and we have the square selected on the map. For the square, we know both latitude and longitude of all four points and we now need to create a query out of it to send it to Content Analytics. This step too can be divided into substeps for better clarity:

1. Create a listener for the **AND** button.

Similar to what we did in the previous step, update the `setListeners()` method for one more line:

```
this.connect(dojoo.byId("and"), "onclick", dojoo.hitch(_this, "andButtonClicked"));
```

2. Get the current query from Content Analytics.

We stored all information that we need (maximum and minimum latitude and longitude) to four variables: `latitudeMin`, `latitudeMax`, `longitudeMin`, `longitudeMax`. All that we need to do to get the current Content Analytics query is to call our well known function:

```
var currentQuery = this.getCurrentQuery();
```

3. Refine the query.

The last thing that must be done is creation of the query string for our geographic data. Now, we know that our data contains certain metadata holding latitude and longitude values and these are numbers. In the case of our collection (as can be seen in the beginning of this tutorial), these are two metadata: `tw_geo_latitude` and `tw_geo_longitude`. What we need now is to format the text query to something like:

```
existing query AND (tw_geo_latitude > latitudeMin AND  
tw_geo_latitude < latitudeMax AND tw_geo_longitude > longitudeMin  
AND tw_geo_longitude < longitudeMax)
```

In the code, it looks like this:

```
var geoQuery = "(tw_geo_latitude >\"" + this.latitudeMin + "\" AND  
tw_geo_latitude <\"" + this.latitudeMax + "\" AND  
tw_geo_longitude >\"" + this.longitudeMin + "\" AND  
tw_geo_longitude <\"" + this.longitudeMax + "\"";
```

## Submitting query and results

The last step of the process is submitting the query. The query is prepared in two variables: `currentQuery` and `geoQuery`. We want to do one more thing before the query execution: Clear the map. The reason is simple. Once we trigger the query, Content Analytics returns refined search results (only documents from within the selected area). We want our map to show only these new ones, so we need to remove all “old” markers. Luckily, it is not hard:

```

/* Remove all markers from the map */
this.markers.forEach(function(marker){ marker.setMap(null); })
/* Remove polygon (square) from the map */
this.polygon.setMap(null);
this.selectionPathMarkers.forEach(function(marker){
marker.setMap(null);

```

Now, the only thing left is to use API and trigger the search:

```

dijit.byId(EDR.prefix+"searchManager").submitQuickRefineSearch({
    "keywords": currentQuery+geoQuery,
    "operator": "AND"
});

```

After we do that, the result looks similar to the markup window as shown in Figure 14-8.

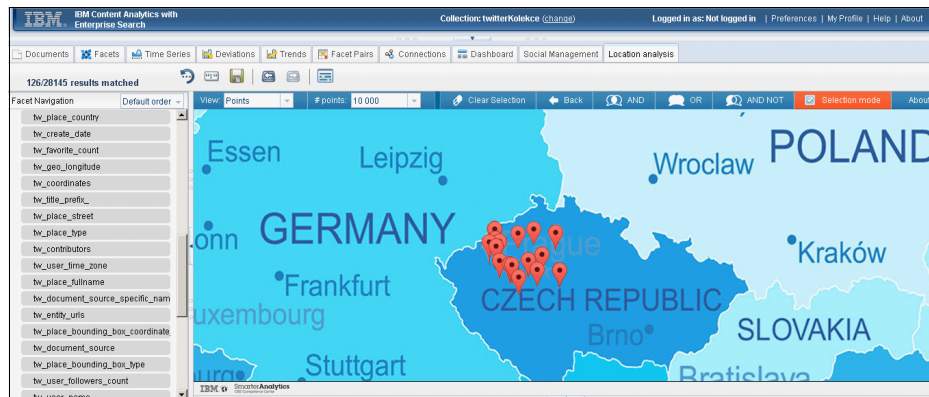


Figure 14-8 Markup of the resulting window after the old markers are removed

From the original 28,145 documents in the collection, we limited the scope of the search query to only 146 documents from within the selected geographic area.

The final version of the code can be found in Appendix A, “Spatial Analysis plug-in code” on page 557.

For more information, go to the IBM Content Analytics Information Center at the following address, and search on *customizing applications*:

<http://publib.boulder.ibm.com/infocenter/analytic/v3r0m0/index.jsp>



# A

## Spatial Analysis plug-in code

This appendix contains the source code for the Spatial Analysis plug-in for IBM Watson Content Analytics (Content Analytics).

The appendix includes the following sections:

- ▶ Spatial Analysis plug-in overview
- ▶ plugin.js
- ▶ plugin.html
- ▶ style.css

## **Disclaimer**

IBM does not warrant or represent that the code in this Appendix is complete or up-to-date. IBM does not warrant, represent, or imply reliability, serviceability, or function of the code presented in association with this book. IBM is under no obligation to update content nor provide further support.

All code is provided “as is,” with no warranties or guarantees whatsoever. IBM expressly disclaims to the fullest extent permitted by law all express, implied, statutory, and other warranties, guarantees, or representations, including, without limitation, the warranties of merchantability, fitness for a particular purpose, and non-infringement of proprietary and intellectual property rights. You understand and agree that you use these materials, information, products, software, programs, and services, at your own discretion and risk and that you will be solely responsible for any damages that may result, including loss of data or damage to your computer system.

In no event will IBM be liable to any party for any direct, indirect, incidental, special, exemplary, or consequential damages of any type whatsoever related to or arising from use of the code found herein, without limitation, any lost profits, business interruption, lost savings, loss of programs, or other data, even if IBM is expressly advised of the possibility of such damages. This exclusion and waiver of liability applies to all causes of action, whether based on contract, warranty, tort, or any other legal theories.

## Spatial Analysis plug-in overview

The Spatial Analysis plug-in visualizes documents from Content Analytics on a map and incorporates the documents' spatial information onto the map for analysis. Furthermore, this plug-in allows users to select a specific geographic location on the map and thus effecting the Content Analytics query result to be based on the selected location. This helps users to fine-tune the analysis of the documents for a specific geographic location.

For information about building this plug-in, see Chapter 14, "Customizing and extending the content analytics miner" on page 535.

### plugin.js

Example A-1 contains the content of the plugin.js file.

*Example A-1 plugin.js*

---

```
dojo.provide("view.spatial.plugin");
dojo.require("widgets.analytics._AnalyticsPane");
dojo.declare("view.spatial.plugin", widgets.analytics._AnalyticsPane, {
    templatePath: dojo.moduleUrl("view.spatial", "plugin.html"),
    onShow: function() {
        var _this = this;
        this.inherited("onShow", arguments);

        this.googleLoaderDef = new dojo.Deferred();
        this.loadGoogleMaps();
        this.googleLoaderDef.then(function() {
            /* place the map */
            _this.drawGoogleMap();
            _this.setListeners();
            dojo.hitch(_this, "reload")();
        });
    },
    setListeners: function() {
        var _this = this;
        this.subscribe("refresh", dojo.hitch(_this, "reload"));
        this.subscribe("collectionChanged", dojo.hitch(_this,
"reload"));
        this.connect(dojo.byId("selectionButton"), "onclick",
dojo.hitch(_this, "selectionButtonClicked"));
    }
});
```

```

        this.connect(dojو.byId("and"), "onclick", doجو.hitch(_this,
"andButtonClicked"));
    },
    andButtonClicked: function() {
        var currentQuery = this.getCurrentQuery();
        var geoQuery = "(tw_geo_latitude >\"" + this.latitudeMin + "\"
AND tw_geo_latitude<\"" + this.latitudeMax + "\" AND
tw_geo_longitude>\"" + this.longitudeMin + "\" AND tw_geo_longitude<\""
+ this.longitudeMax + "\")";

        /* Remove all markers from the map */
        this.markers.forEach(function(marker) {
            marker.setMap(null);
        })

        /* Remove polygon (square) from the map */
        this.polygon.setMap(null);
        this.selectionPathMarkers.forEach(function(marker) {
            marker.setMap(null);
        })

        dijit.byId(EDR.prefix +
"searchManager").submitQuickRefineSearch({
            "keywords": currentQuery + geoQuery,
            "operator": "AND"
        });
    },
    /* This method is triggered whenever the selection mode button is
clicked */
    selectionButtonClicked: function() {
        var _this = this;

        /* Initialization a variable which will help us to create
polygon (square) over the Google maps */
        this.polyLine = new google.maps.Polyline({strokeColor:
"#000000", strokeOpacity: 1.0, strokeWeight: 3});

        /* Few variable we will need in the process */
        this.mapClickCounter = 0;
        this.selectionPoints = new Array();
        this.selectionPathMarkers = new Array();

        /* Add listener on the map */
        google.maps.event.addListener(this.map, "click",
function(event) {

```



```

        /* Increment number of clicks */
        _this.mapClickCounter++;
        /* If user clicks more than twice, just do nothing */
        if (_this.mapClickCounter > 2)
            return;

        /* On the first click, just create the marker and place it
to the map */
        if (_this.mapClickCounter == 1) {
            var marker = new google.maps.Marker({
                position: event.latLng,
                map: _this.map
            });
            /* Save both marker and Latitude/Longitude information
to an array. We will need it later */
            _this.selectionPathMarkers.push(marker);
            _this.selectionPoints.push(event.latLng);
        } else {
            /* Here we need to decide whether the second point has
higher/lower latitude or longitude than the first point*/
            var currentPoint = event.latLng;

            /* In order to create the query in later steps
correctly, we need to know the maximum & minimum latitude & longitude.
Let's save it to four variables */
            _this.latitudeMax = _this.selectionPoints[0].lat() >
currentPoint.lat() ? _this.selectionPoints[0].lat() :
currentPoint.lat();
            _this.latitudeMin = _this.selectionPoints[0].lat() <
currentPoint.lat() ? _this.selectionPoints[0].lat() :
currentPoint.lat();
            _this.longitudeMax = _this.selectionPoints[0].lng() >
currentPoint.lng() ? _this.selectionPoints[0].lng() :
currentPoint.lng();
            _this.longitudeMin = _this.selectionPoints[0].lng() <
currentPoint.lng() ? _this.selectionPoints[0].lng() :
currentPoint.lng();

            /* This is a second click on the map, we need to create
the selection polygon from all points.*/
            /* First corner will have the latitude of the first
point and longitude of the second point */
            var firstCorner = new
google.maps.LatLng(_this.selectionPoints[0].lat(), currentPoint.lng())

```

```

        /* Second corner of the polygon will have latitude of
the second point and longitude of the first point */
        var secondCorner = new
google.maps.LatLng(currentPoint.lat(), _this.selectionPoints[0].lng());
        /* We need to add all newly created points to the array
so we can create the polygon out of them */
        _this.selectionPoints.push(firstCorner, currentPoint,
secondCorner);

        /* Traverse all points and for each, create a marker &
place it to the polygon */
        for (var i = 0; i < _this.selectionPoints.length; i++)
        {
            _this.polyLine.getPath().push(_this.selectionPoints[i]);

            /* Marker for the first point has already been
created, skip it */
            if (i > 0) {
                var marker = new google.maps.Marker({position:
_this.selectionPoints[i], map: _this.map});
                _this.selectionPathMarkers.push(marker);
            }
        }

        /* And now, close the loop (so it makes the square) */
        _this.polygon = new google.maps.Polygon({
            paths: _this.polyLine.getPath(),
            map: _this.map,
            strokeColor: "#FF0000",
            strokeOpacity: 0.8,
            strokeWeight: 2,
            fillColor: "#00649D",
            fillOpacity: 0.35
        });
    });
},
/* This method takes care of reloading all data when necessary */
reload: function() {
    this.getCurrentData();
},
getCurrentData: function() {
    var _this = this;
    this.fetchData().then(function(data) {

```

```

        console.log(data);
        if (data.es_apiResponse === null)
            return;

data.es_apiResponse.ibm_sc_facet.ibm_sc_facetValue.forEach(function(value
) {
    if (value.label == "null")
        return;
    /* coordinates should be in a format  -41.415,
45.21212 */
    var position = value.label.indexOf(",");
    var latitude = parseFloat(value.label.substring(0,
position));
    var longitude =
parseFloat(value.label.substring(position + 2, value.length));

    var point = new google.maps.LatLng(latitude,
longitude);
    var marker = new google.maps.Marker({position: point});
    marker.setMap(_this.map);
    _this.markers.push(marker);
});
});
},
fetchData: function() {
    var _this = this
    return dojo.xhrGet({
        url: "/api/v10/search/facet?collection=" +
_this.getSelectedCollection().id +
"&output=application/json&facet={\"namespace\": \"keyword\", \"id\": \"$.T
witter.tw_coordinates\", \"depth\": 3, \"count\": 1000}&query=" +
_this.getCurrentQuery(),
        handleAs: "json",
        headers: {"Content-Encoding": "UTF-8"},
    });
},
resize: function() {
    /* HEIGHT - header*/
    var newHeight = dojo.contentBox(dojo.byId("spatial")).h - 25;
    dojo.style(
        dojo.byId("mapHolder"),
        "height",
        newHeight + "px");
}
}

```

```

        if (this.isVisible && window["google"] != undefined)
            google.maps.event.trigger(this.map, "resize");
    },
    drawGoogleMap: function() {
        var options = {
            zoom: 6,
            /* Prague */
            center: new google.maps.LatLng(50.08182, 14.440734),
            mapTypeId: google.maps.MapTypeId.ROADMAP
        };

        /* Array to which we will store all markers */
        this.markers = new Array();

        this.map = new google.maps.Map(dojo.byId("mapHolder"),
options);
    },
    loadGoogleMaps: function() {
        var _this = this;
        /* IF google isn't already loaded */
        if (window["google"] == undefined) {
            window["SA_plugin"] = function() {
                _this.googleLoaderDef.resolve();
            }
            var script = document.createElement("script");
            script.type = "text/javascript";
            script.src =
"https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=false&libraries
=visualization&callback=SA_plugin";
            document.body.appendChild(script);
        } else {
            this.googleLoaderDef.resolve();
        }
    }
});

```

---

## plugin.html

Example A-2 contains the content of the plugin.html file.

*Example A-2 plugin.html*

---

```
<div>
  <!-- Reference to cascade style sheet file -->
  <link rel="stylesheet" type="text/css"
href="wplugin/view/spatial/style.css" />

  <div id="spatialHeader">
    <a class="button blue" id="selectionButton">Selection mode</a>
    <a class="button blue" id="and">AND</a>
  </div>
  <div id="mapHolder"></div>
</div>
```

---

## style.css

Example A-3 shows the content of the style.css file.

*Example A-3 style.css*

---

```
#spatialHeader{line-height:25px; height:25px; background:#00649D;}
.button.blue{background-color: #00649D; display:inline-block;
width:100px; text-align:center; color:white;}
.button.blue:hover{background-color: #f63b00; text-decoration:none;
color:white;}
```

---





# B

## Additional material

This book refers to additional material that can be downloaded from the Internet as described in the following sections.

### Locating the Web material

The Web material associated with this book is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser at:

<ftp://www.redbooks.ibm.com/redbooks/SG247877>

Alternatively, you can go to the IBM Redbooks website at:

[ibm.com/redbooks](http://ibm.com/redbooks)

Select **Additional materials** and open the directory that corresponds with the IBM Redbooks form number, SG247877.

## Using the Web material

The additional Web material that accompanies this book includes the following file:

<i>File name</i>	<i>Description</i>
<b>SG247877-01.PDF</b>	Previous version of this book

## System requirements for downloading the Web material

The Web material requires the following system configuration:

**Hard disk space:** 50 MB

## Downloading and extracting the Web material

Create a subdirectory (folder) on your workstation, and extract the contents of the Web material .zip file into this folder.



# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks

The IBM Redbooks publication *IBM Classification Module: Make It Work for You*, SG24-7707, provides additional information about the topic in this document.

You can search for, view, download, or order this document and other Redbooks, Redpapers, Web Docs, drafts, and additional materials, at the following website:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Other publications

The following publication is also relevant as a further information source:

- ▶ *IBM Content Analytics with Enterprise Search, Version 3.0, Administration Guide (SC19-3349-00)*

<http://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.ws?s?CTY=US&FNC=SRX&PBL=SC19-3349-00>

## Online resources

These websites are also relevant as further information sources:

- ▶ IBM Content Analytics with Enterprise Search, Version 3.0 Knowledge Center  
[http://www.ibm.com/support/knowledgecenter/SS5RWK\\_3.0.0/com.ibm.discovery.es.nav.doc/welcome.htm](http://www.ibm.com/support/knowledgecenter/SS5RWK_3.0.0/com.ibm.discovery.es.nav.doc/welcome.htm)
- ▶ IBM Content Analytics latest supported data sources  
<http://www.ibm.com/support/docview.wss?rs=4173&uid=swg27015094>
- ▶ IBM Content Analytics latest system requirements  
<http://www.ibm.com/support/docview.wss?rs=4173&uid=swg27015092>

- ▶ IBM Content Analytics support  
<http://www.ibm.com/support/docview.wss?rs=4173&uid=swg27015096>
- ▶ IBM Content Classification information  
<http://www.ibm.com/software/data/content-management/classification>
- ▶ IBM Content Classification information center  
<http://publib.boulder.ibm.com/infocenter/classify/v8r7/index.jsp>
- ▶ Collecting data for IBM Content Classification  
<http://www.ibm.com/support/docview.wss?uid=swg21417244>
- ▶ IBM Archive and eDiscovery Solution Information Center (for Content Collector, eDiscovery, and eDiscovery Manager)  
<http://publib.boulder.ibm.com/infocenter/email/v2r1m1/index.jsp>
- ▶ Technote “Building a knowledge base for IBM Classification Module V8.7” (IBM Content Classification was formerly known as IBM Classification Module)  
<http://www.ibm.com/support/docview.wss?uid=swg27015916>

## Help from IBM

IBM Support and downloads

[ibm.com/support](http://ibm.com/support)

IBM Global Services

[ibm.com/services](http://ibm.com/services)



**Redbooks**

# **IBM Watson Content Analytics: Discovering Actionable Insight from Your Content**

(1.0" spine)  
0.875" <-> 1.498"  
460 <-> 788 pages







# IBM Watson Content Analytics

## Discovering Actionable Insight from Your Content



**Learn how to perform effective content analytics and search**

**Learn how to gain insights from your data and detect problems early**

**Ultimately, improve your products, services, and offerings**

IBM Watson Content Analytics (Content Analytics) Version 3.0 (formerly known as IBM Content Analytics with Enterprise Search (ICAwES)) helps you to unlock the value of unstructured content to gain new actionable business insight and provides the enterprise search capability all in one product. Content Analytics comes with a set of tools and a robust user interface to empower you to better identify new revenue opportunities, improve customer satisfaction, detect problems early, and improve products, services, and offerings.

To help you gain the most benefits from your unstructured content, this IBM Redbooks publication provides in-depth information about the features and capabilities of Content Analytics, how the content analytics works, and how to perform effective and efficient content analytics on your content to discover actionable business insights.

This book covers key concepts in content analytics, such as facets, frequency, deviation, correlation, trend, and sentimental analysis. It describes the content analytics miner, and guides you on performing content analytics using views, dictionary lookup, and customization. The book also covers using IBM Content Analytics Studio for domain-specific content analytics, integrating with IBM Content Classification to get categories and new metadata, and interfacing with IBM Cognos Business Intelligence (BI) to add values in BI reporting and analysis, and customizing the content analytics miner with APIs.

The target audience of this book is decision makers, business users, and IT architects and specialists who want to understand and analyze their enterprise content to improve and enhance their business operations.

### **INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION**

### **BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:**  
[ibm.com/redbooks](http://ibm.com/redbooks)