SAS® Contextual Analysis 14.1: User’s Guide

Second Edition

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2015. *SAS® Contextual Analysis 14.1: User's Guide, Second Edition*. Cary, NC: SAS Institute Inc.

SAS® Contextual Analysis 14.1: User's Guide, Second Edition

Copyright © 2015, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

For a hard-copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

November 2015

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

Contents

Using This Book

Audience

This book is designed for users of SAS Contextual Analysis. It describes the terminology used in SAS Contextual Analysis and provides instructions for tasks.

What’s New

What’s New in SAS Contextual Analysis 14.1

Overview

SAS Contextual Analysis 14.1 is a major product release that offers an enhanced user interface and includes the following new and enhanced features:

- Project sharing is now supported.
- Exporting and importing SAS Contextual Analysis project models is now available.
- Additional project languages are supported.
- Application of sentiment can be enabled or disabled.
- Custom sentiment models can be applied to project data.
- Predefined concepts can be included or excluded during project creation.
- Interactive rule testing for concepts and categories is supported.
- Document matches for terms can be viewed.
- Concept matches in documents can be viewed.
- The Categories page has been redesigned.
- Category document frequency counts are supported.

- Category rules can be built in a tree view.
- Enhanced help for concept and category rule creation is available.
- Priority and case-sensitivity in concepts are supported.
- Topics and term density for all topics are supported.
- Rules are generated using the HPBOOLRULE procedure.
- Concepts can be referenced from within categories.

Project Sharing and Importing Project Models

You can now view and update projects that were created by another user if the projects are stored in a shared location. SAS Contextual Analysis project models can be exported and reused to create new projects. This enables the reuse of category and concept rules.

New Project Creation Options

The Create Project wizard now provides an option for choosing the project language. Thirteen languages, including English, can now be processed. A new sentiment feature enables you to enable or disable sentiment models, and to apply your own custom sentiment models during project creation. Predefined concepts can be included or excluded during project creation.

Redesigned Categories Page and Testing for Concepts and Categories

The Categories page has been redesigned and now includes document frequency counts and a tree view where you can build rules. Concept and category rules can be tested interactively with sample text. Enhanced documentation for writing rules can be accessed through the online Help.

New Properties for Concepts and Topics

New properties in the concept task enable you to apply priority and case-sensitivity to concept rules. New properties in the topics task enable you to specify term density and number of topics.

Enhanced Rule Generation and Processing

Rules are now generated using the HPBOOLRULE procedure. Multiple cores can now be accessed during project processing. The score code produced by SAS Contextual Analysis has been modified to include the DS2 procedure in Base SAS, which enables threaded processing. You can now reference LITI concepts from within category rules.

Accessibility

For information about the accessibility of this product, see Accessibility Features of SAS Contextual Analysis 14.1 at support.sas.com.

1

Introduction to SAS Contextual Analysis

What Is SAS Contextual Analysis?

SAS Contextual Analysis is a web-based text analytics application that uses contextual analysis to provide a comprehensive solution to the challenge of identifying and categorizing key textual data. Using this application, you can build models (based on training documents) that automatically analyze and categorize a set of documents. You can then customize your models in order to realize the value of your text-based data.

SAS Contextual Analysis combines the machine-learning capabilities of SAS Text Miner with the rules-based linguistic methods of categorization and extraction in SAS Enterprise Content Categorization. These capabilities, along with document-level sentiment scoring, are combined in a single user interface.

Using SAS Contextual Analysis, you can identify key textual data in your document collections, categorize those data, build concept models, and remove meaningless textual data.

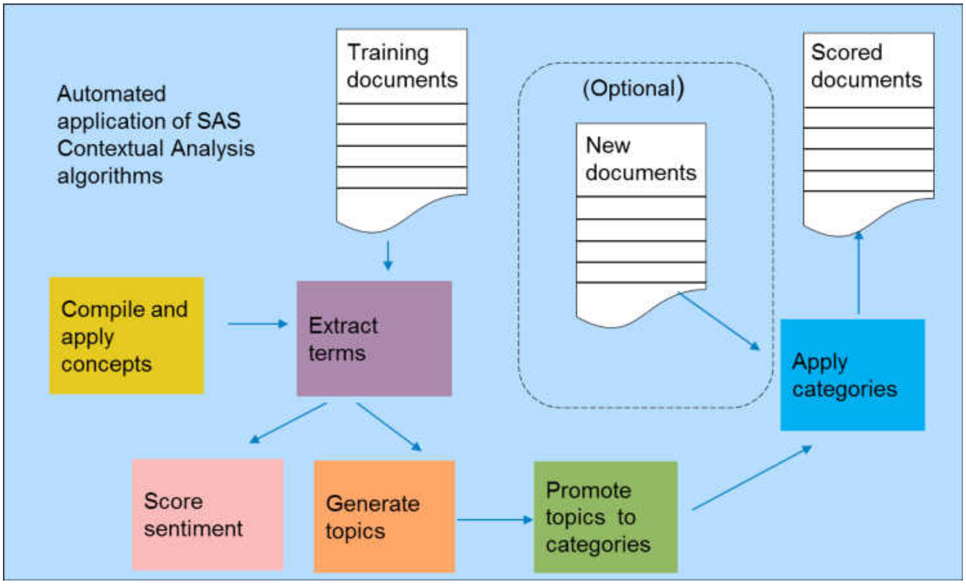
By default, words that provide little or no value are excluded from analysis. Examples of these words include the articles *a*, *an*, and *the* and conjunctions such as *and*, *or*, and *but*. Other terms that are specific to your document collection but provide little or no value are also identified and excluded.

SAS Contextual Analysis is designed for users who have no SAS programming or SAS macro language experience.

How Does SAS Contextual Analysis Work?

Figure 1.1 provides an overview of the SAS Contextual Analysis processes.

Figure 1.1 *Process Overview*



SAS Contextual Analysis enables you to extract pre-defined concepts or create additional custom concepts that you want to discover in a document or set of documents. For more information about concepts, see “Concepts” on page 28.

The SAS Contextual Analysis algorithms group similar documents in a collection into topics. The documents in each topic often contain similar subject matter, such as motorcycle accidents, computer graphics, or weather patterns. Automatic topic identification enables you to easily categorize each document in your collection.

You can create categories using these methods:

- import categories from SAS Enterprise Content Categorization
- specify category variables in your training documents
- create new categories
- promote topics to categories

Preliminary rules are generated when you promote a topic to a category or when you specify category variables in your training documents. These rules can be edited and refined.

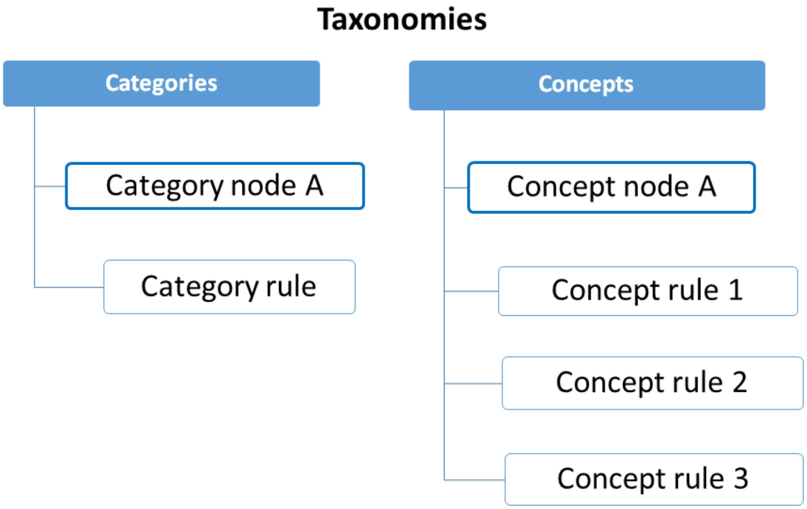
Whether you use the automated processes and rules that are available for extracting terms and subject matter or you customize the processes and rules, context sensitivity is an essential component of your model. To enhance context sensitivity, you can modify the preliminary rules. You can add or modify Boolean operators, characters, and other selections to make the rule matching more context-sensitive.

Finally, you deploy your model to automate the process of classifying a set of input documents.

Using Taxonomies

In SAS Contextual Analysis, you can create category and concept rules, which are displayed in a taxonomic structure. Each taxonomy consists of a tree of *nodes*. Each node is a container for rules. By contrast, under a concept node, there can exist multiple rules. Figure 1.2 on page 4 demonstrates how category and concept taxonomies differ.

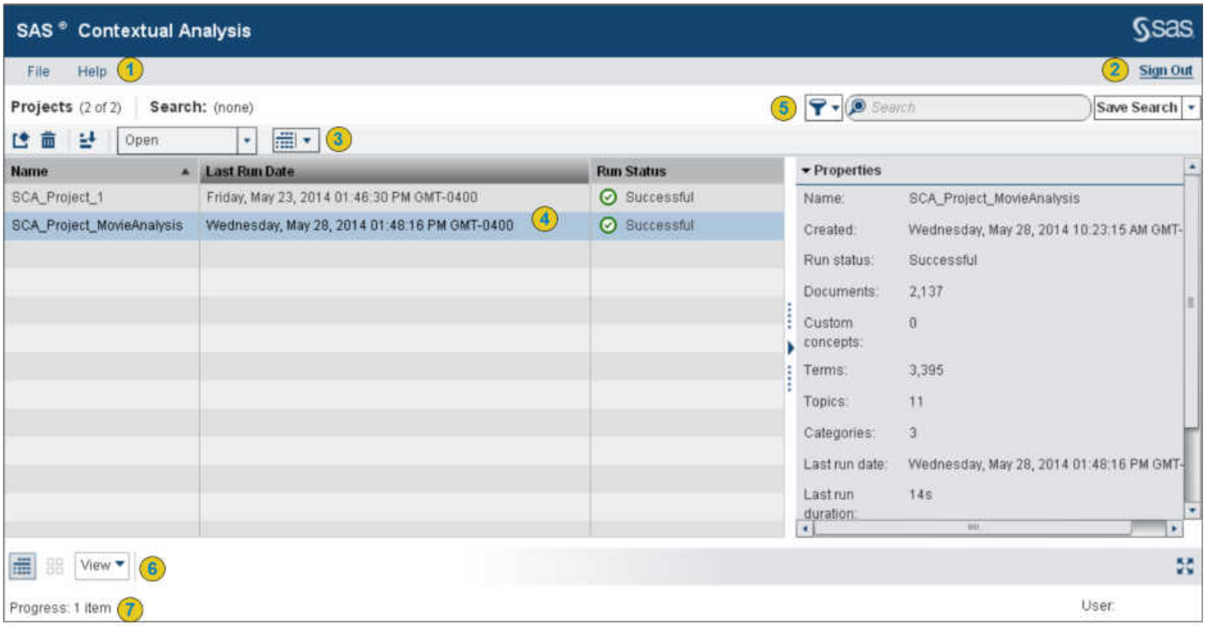
Figure 1.2 Taxonomies in SAS Contextual Analysis



Using the Interface

The main components of the user interface are shown in Figure 1.3.

Figure 1.3 *SAS Contextual Analysis Interface*



Projects (2 of 2) **Search:** (none)

Name	Last Run Date	Run Status
SCA_Project_1	Friday, May 23, 2014 01:46:30 PM GMT-0400	✓ Successful
SCA_Project_MovieAnalysis	Wednesday, May 28, 2014 01:48:16 PM GMT-0400	✓ Successful

▾ Properties

Name:	SCA_Project_MovieAnalysis
Created:	Wednesday, May 28, 2014 10:23:15 AM GMT-
Run status:	Successful
Documents:	2,137
Custom concepts:	0
Terms:	3,395
Topics:	11
Categories:	3
Last run date:	Wednesday, May 28, 2014 01:48:16 PM GMT-
Last run duration:	14s

Progress: 1 item

User:

1 Application menu

2 Sign out

3 Application toolbar

4 Project list

5 Search options

6 View options

7 Progress panel (click to open)

2

Projects in SAS Contextual Analysis

Overview of a Project

In SAS Contextual Analysis, you create projects, which are basically containers for your data and analysis. A project contains the input data, text mining options, and analysis tasks (working with concepts, terms, topics, and categories). SAS Contextual Analysis is designed so that you can create and run multiple projects simultaneously. Text analysis is performed in the background so that you can open one project while performing analysis on a different project. Projects can be shared among users and updated collaboratively.

When you build a model, you choose input data that contain the document collection that you want to use as a training data set. It is important to ensure that your training data are representative of the data to which this model will be applied. Topics and categories are built based on the terms in this document collection.

Next, you can choose to specify either a start list or a stop list. You can also specify whether to use a synonym list. Before you can run your project on a SAS data set, you must specify the text field that you want to analyze. You can also specify one or more category variables for the analysis.

After the project runs, you can view the terms and automatically discovered topics that were created during the initial text mining. Then, you use the topics to create categories. Categories are groups of documents that contain similar terms. SAS Contextual Analysis builds a set of rules for each category.

Preparing the Document Collection

Before you create a project in SAS Contextual Analysis, you need to prepare your document collection for analysis. SAS Contextual Analysis enables you to analyze a document collection that is stored as a SAS data set or in text-based file formats such as MS Office, OpenDocument (OpenOffice), PDF, XML, HTML, and others. You can select a SAS data set and then identify the text variables and category variables to be analyzed. Or you can specify a directory that contains the files that you want to use as training data.

When you prepare the input document collection, you should select a set of documents that is representative of the documents that you want to categorize later. The terms that exist in the input document collection are used to build the topics and categories.

There are no standard rules for creating an input document collection. However, the following guidelines can help you prepare your input document collection:

- Include at least 15 to 20 documents for each category that you want to discover.
- Be familiar with the contents of the documents in order to anticipate term discovery and rule creation.
- Do not store SAS data sets in the same directory where you store a collection of Microsoft Word or Adobe PDF documents.

Note: When you use a SAS data set, you must register that data set with the SAS Metadata Server before it is available in SAS Contextual Analysis. You can use SAS Management Console and SAS Enterprise Guide to register data sets. When you use a collection of documents in a folder (rather than a SAS data set), you must locate the folder on the server where the SAS Contextual Analysis workspace server is installed. For more information, see *SAS Contextual Analysis 14.1: Administrator's Guide*.

Importing Projects

Importing an Existing SAS Contextual Analysis Project Model

During project creation, you can import a SAS Contextual Analysis project so that you can reuse existing category and concept rules. Imported projects include only category and concept rules; other project components such as topics, terms, data, project settings, and so on, are not imported. The file that you import is in JSON (JavaScript Object Notation) format.

For information about exporting a SAS Contextual Analysis project, see "Exporting a Project Model" on page 22.

Importing an Existing SAS Enterprise Content Categorization Project

During project creation, you can import an existing SAS Enterprise Content Categorization project for analysis. If you plan to import a project, note the following:

- Concepts that were defined by using the LITI (language interpretation and text interpretation) syntax in an imported SAS Enterprise Content Categorization project can be used in your SAS Contextual Analysis project.
- Categories that were defined using Boolean rules (MCAT syntax) in an imported SAS Enterprise Content Categorization project can be used in your SAS Contextual Analysis project.
- Categories that were created using linguistic rules in SAS Enterprise Content Categorization are not supported.

Note: In order for the LITI concepts to be parsed correctly in SAS Contextual Analysis, the parsing priority for disabled concepts must be honored. To ensure this, open your existing project in SAS Enterprise Content Categorization. For any child concept that was disabled, modify its parent concept so that the parent has a higher priority than the child. Save the project before you import it into SAS Contextual Analysis.

Special Considerations When Importing Projects

When you import categories and concepts from a SAS Contextual Analysis or SAS Enterprise Content Categorization project into your new project, note that duplicate names might occur. Here are some items to consider when duplicate category or concept names occur during the importing process:

- Check messages to see how SAS Contextual Analysis handled the duplicate name. Click on the toolbar to see messages.
- Plan for possible duplicate names if you are using predefined concepts in your project. For example, suppose a LITI concept that you are importing has the same name as a predefined concept in the current project. In that case, the imported LITI concept’s rules are added to the predefined concept’s rules.
- When importing concepts from an existing project, the source of the concept determines which concept name takes precedence when the system resolves a naming conflict. Predefined concepts in the current project take first priority, followed by concepts imported from SAS Contextual Analysis, and finally, concepts imported from SAS Enterprise Content Categorization.
- When importing categories from an existing project, the source of the category determines which category name takes precedence when the system resolves a naming conflict. Categories imported from SAS Contextual Analysis take first priority, followed by external categories in the current project, and finally, concepts imported from SAS Enterprise Content Categorization.

Creating a New Project

Using the Create Project Wizard

The first time you log on to SAS Contextual Analysis, you must create a project before you can do anything else. To create a new project, click the icon near the upper left corner of the main window. The Create New Project wizard appears, where you can enter all the specifics for your project.

TIP Click the Help icon ? in the Create New Project wizard for information about a specific field or page.

Step 1: Identify Project Files, Servers, and Other Properties

1 Enter the project name, and specify where the project folder can be accessed in SAS metadata.

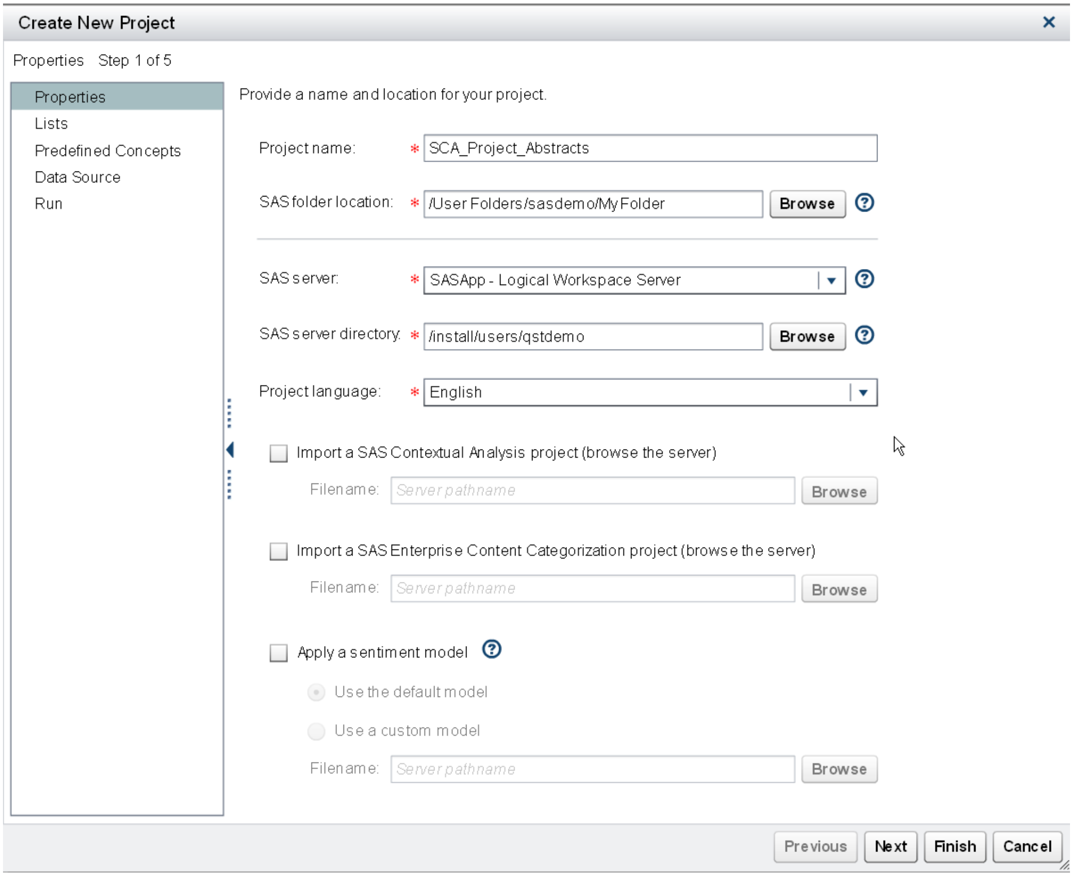
2 Indicate the SAS server and SAS server directory.

3 Select a language for the project data. This is the language that will be processed in your data files.

4 (Optional) Import a SAS Contextual Analysis project. For more information, see "Importing an Existing SAS Contextual Analysis Project Model" on page 10 .

5 (Optional) Import a SAS Enterprise Content Categorization project. For more information, see "Importing an Existing SAS Enterprise Content Categorization Project" on page 10.

6 (Optional) Apply a sentiment model. For more information about sentiment, see "Using SAS Sentiment Analysis Models in SAS Contextual Analysis" on page 26.



Create New Project

Properties Step 1 of 5

Provide a name and location for your project.

Project name: * SCA_Project_Abstracts

SAS folder location: * /User Folders/sasdemo/My Folder

SAS server: * SASApp - Logical Workspace Server

SAS server directory: * /install/users/qstdemo

Project language: * English

□ Import a SAS Contextual Analysis project (browse the server)

Filename: Server pathname

□ Import a SAS Enterprise Content Categorization project (browse the server)

Filename: Server pathname

□ Apply a sentiment model

● Use the default model

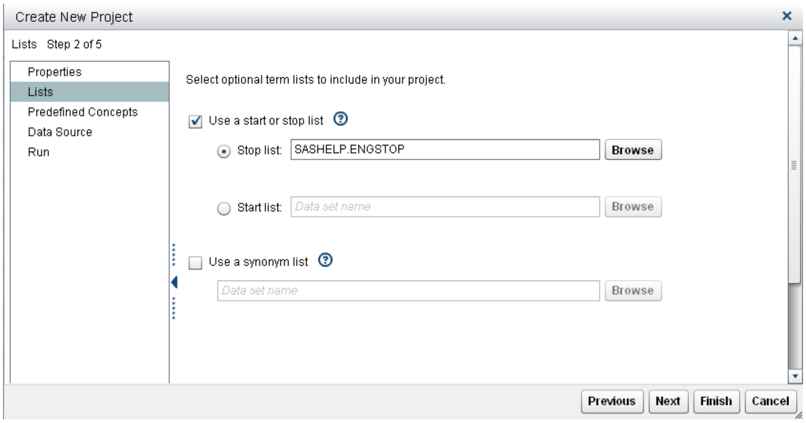
○ Use a custom model

Filename: Server pathname

Step 2: Specify Term and Synonym Lists

1 (Optional) Identify a start list or a stop list (but not both) to control which terms to include or exclude during text mining analysis. For more information, see “Start Lists and Stop Lists” on page 32. A default stop list is selected by default.

2 (Optional) Specify a synonym list to identify pairs of words that should be treated as single terms for analysis. For more information, see “Terms and Synonyms” on page 31.



Select optional term lists to include in your project.

☑ Use a start or stop list

● Stop list: SASHELP.ENGSTOP

○ Start list: Data set name

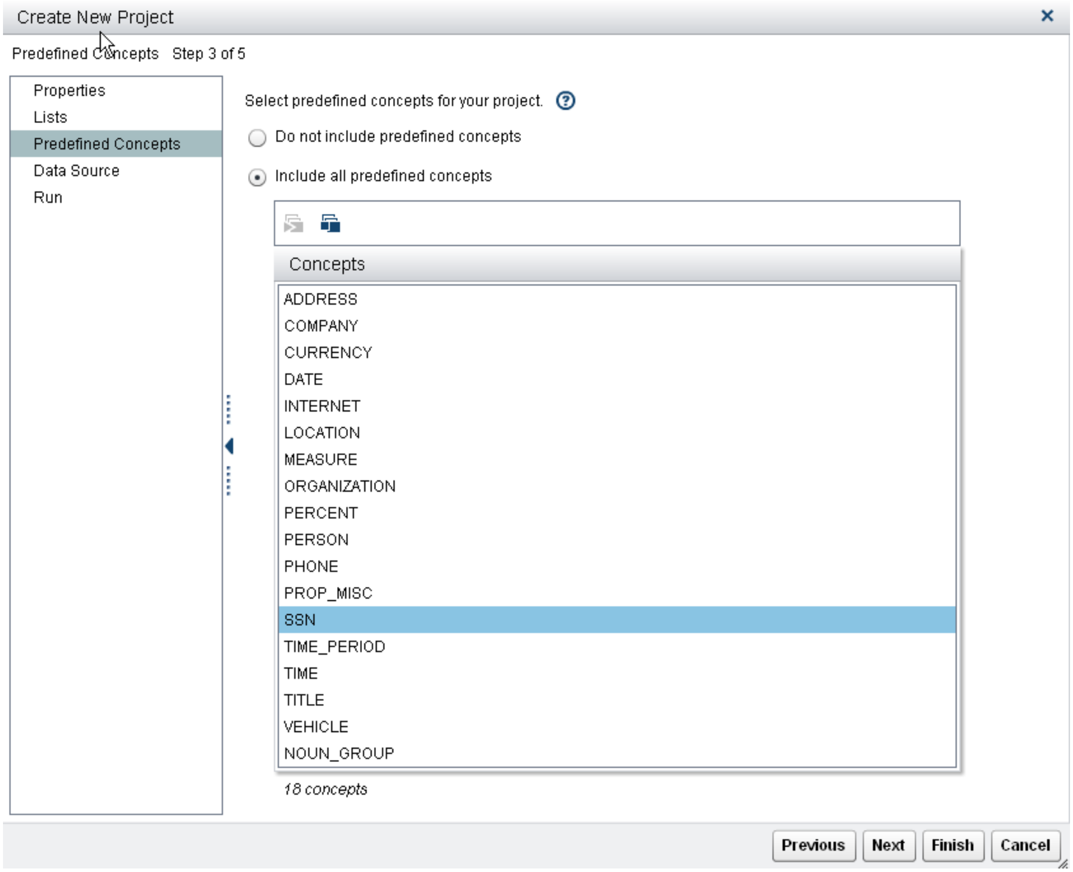
□ Use a synonym list

Data set name

Step 3: Choose Predefined Concepts

SAS Contextual Analysis provides *predefined concepts*, which are concepts whose rules are already written. Predefined concepts save time by providing you with commonly used concepts and their definitions, such as **COMPANY** or **TITLE**. You can choose to include them or not. If you do not include predefined concepts, they cannot be added later. If you include the predefined concepts, you can disable one or more predefined concepts by selecting one or more predefined concepts and then clicking . Disabled concepts are ignored during data processing. You can re-enable any predefined concept by selecting a concept and then clicking .

For information about predefined concepts, see “Concepts” on page 28.



Predefined Concepts Step 3 of 5

Select predefined concepts for your project.

○ Do not include predefined concepts

● Include all predefined concepts

Concepts

ADDRESS
COMPANY
CURRENCY
DATE
INTERNET
LOCATION
MEASURE
ORGANIZATION
PERCENT
PERSON
PHONE
PROP_MISC
SSN
TIME_PERIOD
TIME
TITLE
VEHICLE
NOUN_GROUP

18 concepts

Step 4: Identify a Data Source

Here are the options for selecting a data source:

- You can choose a data source now or later. If you choose a data source later, you can still enter more information for your project in the Edit Project wizard.
- You can select analysis variables from within a SAS data set. If you choose this option, you must specify the data set library and name and specify the text variable that you want to analyze. Rather than including the full text of a document in a SAS variable, you can enter a file reference, which identifies the location of a file. Using a file reference is the only way to analyze documents that are longer than 32,767 characters.

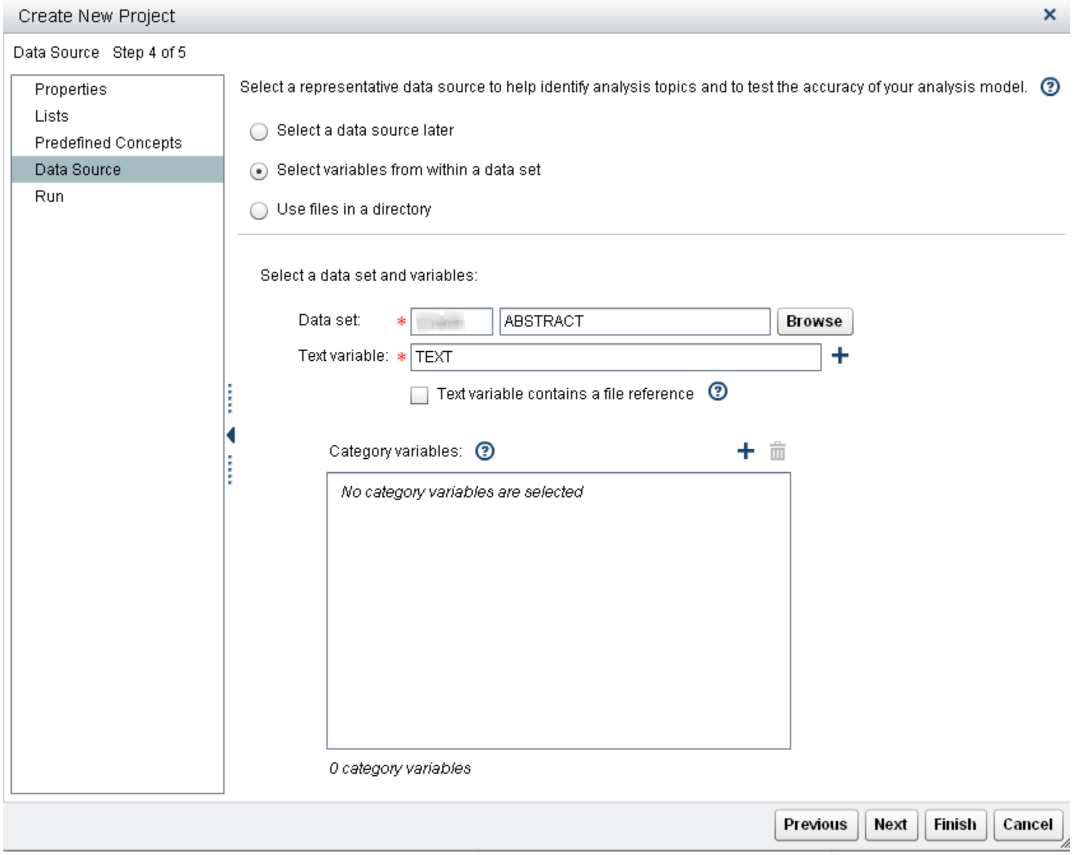
  Note: The referenced file must be in plain text format (TXT).

In addition, you can specify one or more category variables to indicate how you want the documents to be grouped. For example, suppose you are analyzing customer comments from hotel stays. The data column Hotels includes names of hotels where customers stayed. If you specify Hotels as a category variable, then category rules are automatically generated. Subcategory rules are also generated for each hotel that appears frequently in the data.

Note: SAS Contextual Analysis reserves variables names that begin with an underscore (_). Therefore, if you select a data set that includes variable names that begin with an underscore (_), you could encounter an error. If an error occurs, rename the variables in the data set and try again.

- You can specify a document collection that is stored in text-based file formats such as MS Office, OpenDocument (OpenOffice), PDF, XML, or HTML. The files must be located in a folder. You can define categories later.

In the following example, the text variable *TEXT* is selected from the data set *ABSTRACT*. There is no category variable.



Create New Project ✕

Data Source Step 4 of 5

Select a representative data source to help identify analysis topics and to test the accuracy of your analysis model.

○ Select a data source later

● Select variables from within a data set

○ Use files in a directory

Select a data set and variables:

Data set: * [redacted] ABSTRACT

Text variable: * TEXT

□ Text variable contains a file reference

Category variables:

No category variables are selected

0 category variables

Step 5: Run the Project

You can choose to run the entire project now or later. Select **None** on the Run page to run the project later. See the Help for more information about when to run the project.

If you choose to run the entire project, the following events occur for data sources that are provided:

- Parsing takes place.
- Topics are generated.
- Rules are generated and run for any category variables that you specified.
- Sentiment is applied (if you specified a sentiment model).

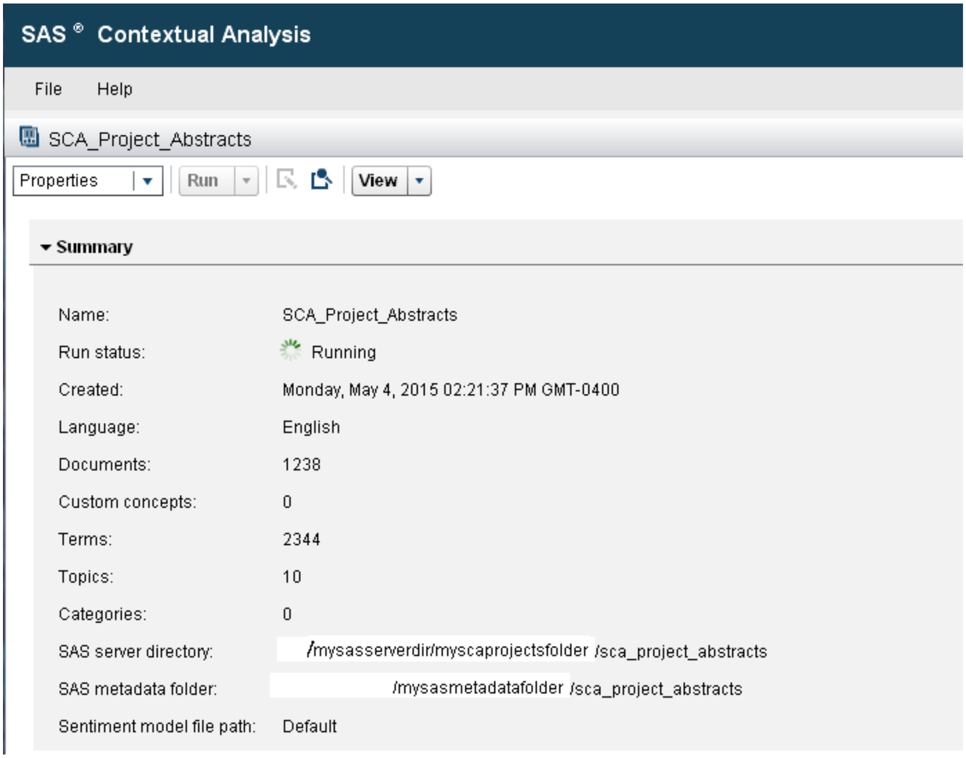
- Concepts are applied (if you included predefined concepts in the project).

If you import a SAS Enterprise Content Categorization project and then run the project, the following events occur:

- Imported concepts are compiled and applied to the data source that is provided.
- Imported categories are compiled and applied to the data source provided.

  Note: Subcategories are imported only if their parent categories were successfully imported.

The **Run status** field on the Properties page indicates whether a project is running.



Name: SCA_Project_Abstracts

Run status: Running

Created: Monday, May 4, 2015 02:21:37 PM GMT-0400

Language: English

Documents: 1238

Custom concepts: 0

Terms: 2344

Topics: 10

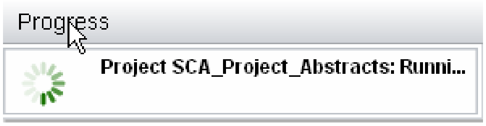
Categories: 0

SAS server directory: /mysasserverdir/myscaprojectsfolder /sca_project_abstracts

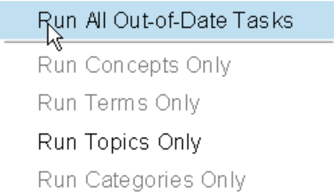
SAS metadata folder: /mysasmetadatafolder /sca_project_abstracts

Sentiment model file path: Default

Note: You can also check the **Progress** panel, which is located in the lower left corner of the main window. Click the word **Progress** to see which projects are running and which projects have finished running.



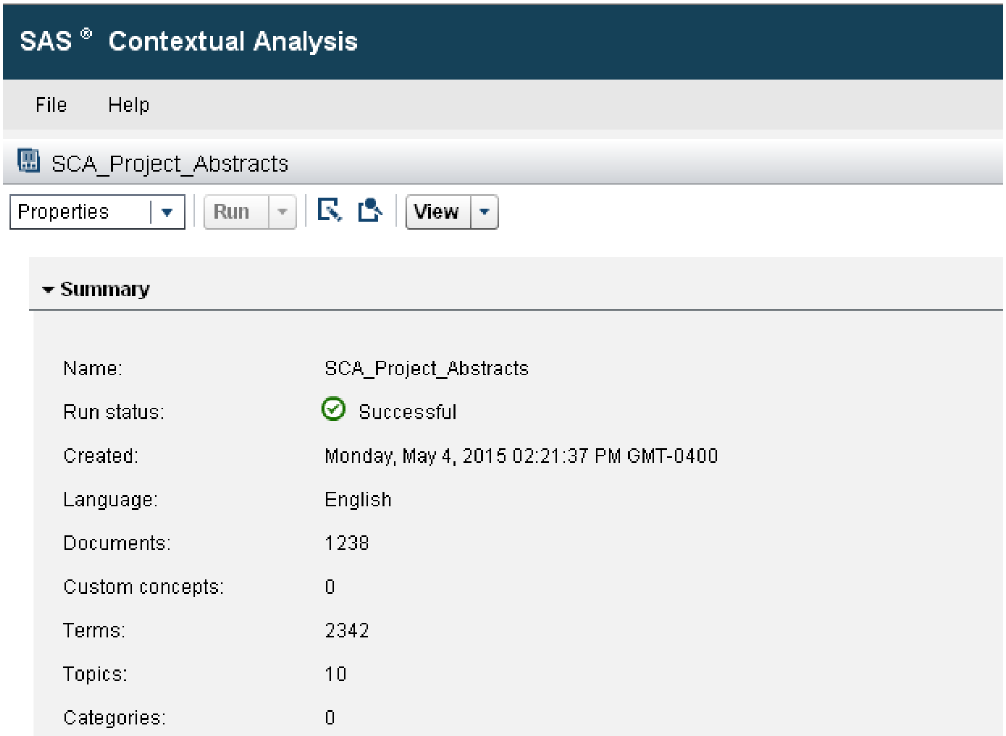
The **Run** menu enables you to run analysis tasks individually or run only the tasks that are out of date (and their dependent tasks, if any).



Using the Properties Page

Checking Project Status

The Properties page indicates whether the project ran successfully and provides basic information about the data that were analyzed.



Name: SCA_Project_Abstracts

Run status: ✓ Successful

Created: Monday, May 4, 2015 02:21:37 PM GMT-0400

Language: English

Documents: 1238

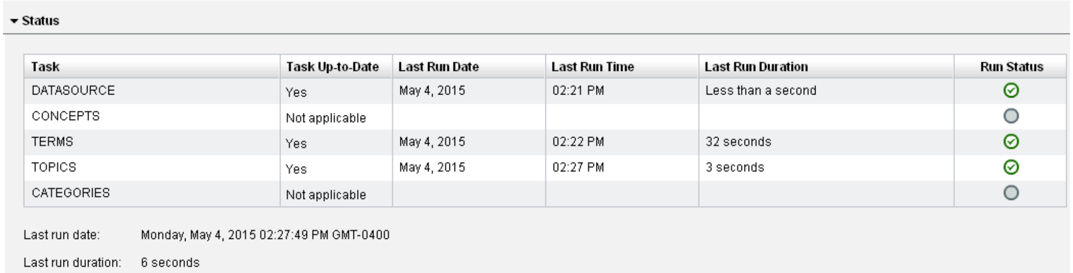
Custom concepts: 0

Terms: 2342

Topics: 10

Categories: 0

The resulting status of each analysis task that was run is displayed in the following fields in the **Status** section:



Task	Task Up-to-Date	Last Run Date	Last Run Time	Last Run Duration	Run Status
DATASOURCE	Yes	May 4, 2015	02:21 PM	Less than a second	✓
CONCEPTS	Not applicable				○
TERMS	Yes	May 4, 2015	02:22 PM	32 seconds	✓
TOPICS	Yes	May 4, 2015	02:27 PM	3 seconds	✓
CATEGORIES	Not applicable				○

Last run date: Monday, May 4, 2015 02:27:49 PM GMT-0400

Last run duration: 6 seconds

Task

The name of the analysis task

Task Up-to-Date

Indicates whether information in the task has changed since the last time the task was run. If no information has changed, the value is `Yes` and no further action is required. If information has changed in the task since the last run, then the value is `No` and the task should be rerun.

Last Run Date and Last Run Time

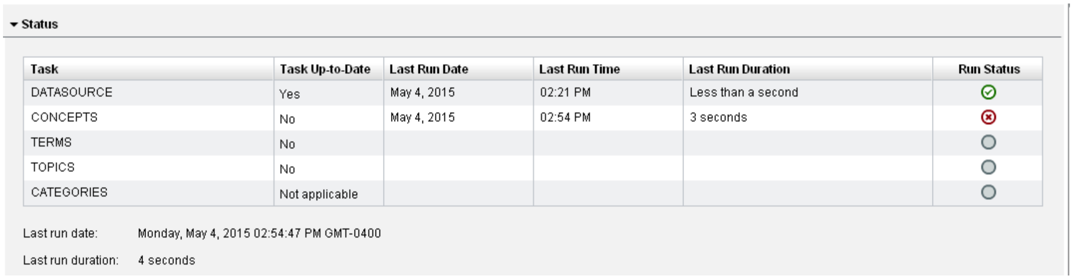
The last date and time the task was run

Last Run Duration

The duration of the task’s last run

Last Run Status

Indicates whether the task’s run was successful ✓, the run failed ⊗, the task did not run ○, or the task was unable to run (because of missing or insufficient information) or warnings occurred ⚠. Click **View messages** on the toolbar for specific information. A status of **Not run** ○ is displayed if the failure of a task prevents dependent tasks from running. In the following example, the failure of the `CONCEPTS` task prevented the `TERMS` and `TOPICS` tasks from running. You must correct the error and rerun the project until it runs successfully.



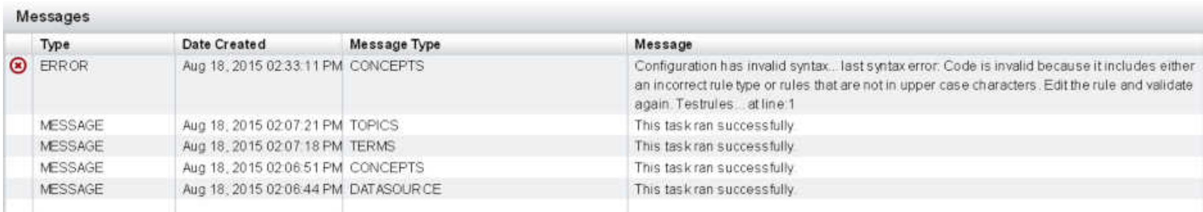
▾ Status

Task	Task Up-to-Date	Last Run Date	Last Run Time	Last Run Duration	Run Status
DATASOURCE	Yes	May 4, 2015	02:21 PM	Less than a second	✓
CONCEPTS	No	May 4, 2015	02:54 PM	3 seconds	⊗
TERMS	No				○
TOPICS	No				○
CATEGORIES	Not applicable				○

Last run date: Monday, May 4, 2015 02:54:47 PM GMT-0400

Last run duration: 4 seconds

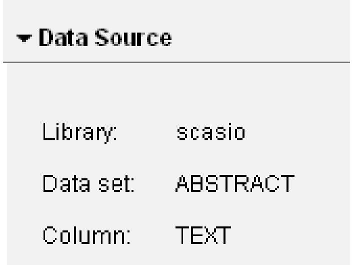
Click 🗨 on the toolbar to see messages about the status of each task. Here is an example of the Messages window for a task that has an error. The **Message Type** column indicates the corresponding analysis task for each message.



Messages

	Type	Date Created	Message Type	Message
⊗	ERROR	Aug 18, 2015 02:33:11 PM	CONCEPTS	Configuration has invalid syntax... last syntax error: Code is invalid because it includes either an incorrect rule type or rules that are not in upper case characters . Edit the rule and validate again. Testrules... at line:1
	MESSAGE	Aug 18, 2015 02:07:21 PM	TOPICS	This task ran successfully
	MESSAGE	Aug 18, 2015 02:07:18 PM	TERMS	This task ran successfully
	MESSAGE	Aug 18, 2015 02:06:51 PM	CONCEPTS	This task ran successfully
	MESSAGE	Aug 18, 2015 02:06:44 PM	DATASOURCE	This task ran successfully

The data source information is displayed at the bottom of the Properties page.



▾ Data Source

Library:	scasio
Data set:	ABSTRACT
Column:	TEXT

Editing Project Information

Click ✎ to edit basic information for your project (such as project name). The Edit Project wizard appears. Items that you cannot edit appear in gray.

Note: You must rerun the project to see the effects of your changes.

Viewing and Downloading Code

You can view and download SAS score code that is created. Score code enables you to apply the text analytic models in your project (concepts, categories, and sentiment) to other data.

On the Properties page, click **View**. Select **Concept code**, **Sentiment code**, or **Categories code**. Click ⧉ to copy the code for use in other programs. If you want to download the code to a file, click ✎ and follow the prompts to specify a location for the file.

Note: If you have run concepts while predefined concepts were enabled, the generated scored data displays matches that do not include a full path.

Exporting a Project Model

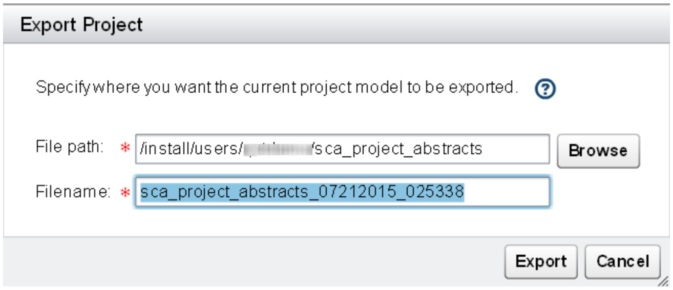
You can export a SAS Contextual Analysis project model so that you can reuse its rules in a new project. When you export a project, only the category and concept rules are exported. Any predefined concepts that you have selected are preserved. The other project components such as topics, terms, data, project settings, and so on, are not exported. The file that you export is in JSON (JavaScript Object Notation) format.

To export a project:

1. Open the project that you want to export. Click ↗ on the toolbar.

2. In the **File path** field, select a server location for the file that you are exporting. Note that the project name appears in the file path in all lowercase letters, and underscores (_) might be inserted.

3. In the **Filename** field, review the automatically generated name for the exported project and edit the name if desired. The generated filename consists of the project name plus the current date (in MMDDYY format) and time (in MMHHSS format), respectively.



Export Project

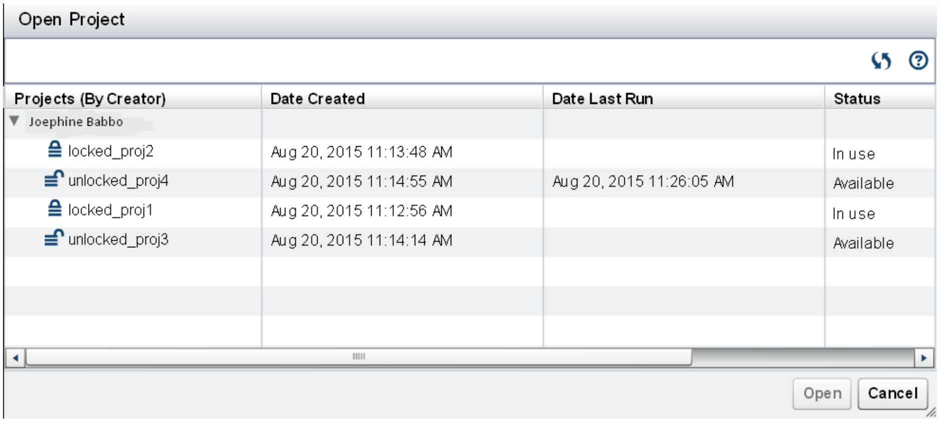
Specify where you want the current project model to be exported.

File path: * /install/users/[redacted]/sca_project_abstracts

Filename: * sca_project_abstracts_07212015_025338

Sharing Projects

You can open and edit a project that has been created by another user. From the main window, click 📂. The Open Project window displays all of the shared projects and their owners. Projects that are in use are locked and cannot be opened. The locked icon 🔒 appears next to projects that cannot be opened. The unlocked icon 🔓 appears next to projects that you can edit.



Projects (By Creator)	Date Created	Date Last Run	Status
▼ Joephine Babbo			
locked_proj2	Aug 20, 2015 11:13:48 AM		In use
unlocked_proj4	Aug 20, 2015 11:14:55 AM	Aug 20, 2015 11:26:05 AM	Available
locked_proj1	Aug 20, 2015 11:12:56 AM		In use
unlocked_proj3	Aug 20, 2015 11:14:14 AM		Available

Note: To keep your shared projects unlocked for other users, close open projects that are not in use and sign out of your SAS Contextual Analysis session (rather than closing your web browser).

After you open a project, you can edit the project information such as synonym lists. For information about editing project information, see "Editing Project Information" on page 22 . For information about editing concept or category rules, terms, or other analysis tasks, see Chapter 3, "Performing the Analysis Tasks," on page 27.

Scoring an External Data Set

You can use the model that you built in your SAS Contextual Analysis project to score an external data set. When you score an external data set, the category model is applied to the external data set (called the target data set). The categorization information for the document collection is then output into a scored data set.

Note: The data set must be stored in a file outside a folder. If your project uses a folder as a data source, you cannot score data sets within the same folder.

To score an external data set:

1 Select **File ->Score External Data Set** from the application's main menu.

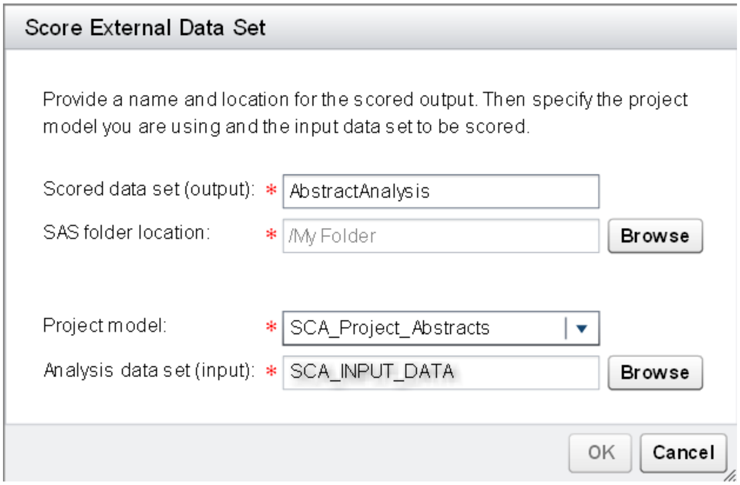
2 In the **Scored data set (output)** field, enter the name for the scored data set that is to be generated.

3 Enter the SAS folder location for saving the data.

4 In the **Project Model** field, select the name of the project that contains the analysis model that you are using.

5 In the **Analysis data set (input)** field, provide the data set to be scored. The analysis data set must have the same text variable as the selected project model's data source.

Note: To be eligible for scoring, a project must have compiled a category binary file. A category binary file is generated when you run a project that contains categories.



Score External Data Set

Provide a name and location for the scored output. Then specify the project model you are using and the input data set to be scored.

Scored data set (output): * AbstractAnalysis

SAS folder location: * /My Folder

Project model: * SCA_Project_Abstracts

Analysis data set (input): * SCA_INPUT_DATA

After the scoring begins, the project's run status changes to **Running**. When the scoring is complete, the scored data set is placed in the library folder where the project that you used as your project model is stored.

About Sentiment Analysis

Introduction to Document Scoring

Sentiment analysis is the process of identifying the author's tone or attitude (positive, negative, or neutral) expressed in a document. SAS Contextual analysis uses a set of proprietary rules that identify and analyze terms, phrases, and character strings that imply sentiment. A sentiment score is then assigned, based on that analysis. Using these rules, the software is able to provide repeatable, high quality results.

The assignment of sentiment to a document is based on the attitude that is associated with the document as a whole. For example, the following document would have a positive sentiment: `Had an awesome time yesterday. Glad I brought my tent from Store XYZ.`

Because documents can be associated with multiple words or terms that imply sentiment, SAS Contextual Analysis uses a scoring system to assign a final sentiment score. The following list provides basic information about how sentiment scoring works. (The information has been simplified to illustrate key concepts.)

- Each positive term or phrase is worth a single (positive) point.
- Each negative term or phrase is worth a negative point. If there are more positive terms or phrases than negative, the final sentiment score is positive.
- If there are more negative terms or phrases, the final sentiment score is negative.
- If there are an equal number of positive and negative terms or phrases, the sentiment score is neutral.

Using SAS Sentiment Analysis Models in SAS Contextual Analysis

Rules that are generated using SAS Sentiment Analysis are stored in a .sam binary file. When you create a project in SAS Contextual Analysis, you can use a .sam binary file that you have created to your specifications, or you can use the default file that is available for your project’s language.

Note: Not all languages have default sentiment models available for use.

For more information about the sentiment analysis and scoring, see *SAS Sentiment Analysis 12.2: User’s Guide*.

3

Performing the Analysis Tasks

Overview of the Analysis Tasks

Introduction

When you run a project, the following analysis tasks are performed (if data are present):

- concept extraction
- term identification (including synonyms)
- topic discovery
- category analysis

The following sections describe each task.

Concepts

A *concept* is a property such as a book title, last name, city, gender, and so on. Concepts are useful for analyzing information in context. You can write rules for recognizing concepts that are important to you, thereby creating custom concepts. For example, you can specify that the concept *kitchen* is identified when the terms *refrigerator*, *sink*, and *countertop* are encountered in text.

SAS Contextual Analysis provides *predefined concepts*, which are concepts whose rules are already written. Predefined concepts save time by providing you with commonly used concepts and their definitions, such as `COMPANY` or `TITLE`. You cannot rename predefined concepts, nor can you view or edit their base definitions. You can provide additional rules in the **Edit** tab for processing.

Note: If an imported concept has the same name as a predefined concept, the imported concept's rules are added to the predefined concept's rules.

Here is a list of the predefined concepts included with SAS Contextual Analysis. You can disable or enable any of the concepts during project creation (or in the Concepts task window).

Note: Some languages use a subset of the predefined concepts listed here.

ADDRESS
Postal address or number and street name

COMPANY
company name

Note: SAS Contextual Analysis uses a fixed dictionary of company and organization names in order to identify this concept. This concept is frequently associated with a parent. For example, if `IBM` appears in the text, it is returned with the predefined parent `International Business Machines`. Typically, the longest and most precise version of a name is used as the parent form.

CURRENCY
currency or currency expression. Examples: $300, 300 million dollars

DATE
date, day, month, or year

INTERNET
email address or URL

LOCATION
city, country, state, geographical place or region, or political place or region

MEASURE
measurement or measurement expression. Examples: 500kg, 2300 sq ft

NOUN_GROUP
: multiple words that act as a single term (for example, “window ” or “clinical trial”)

ORGANIZATION
: government, legal, or service agency. (See the note associated with COMPANY.)

PERCENT
: percentage or percentage expression. Examples: 97%, 12 percentage points

PERSON
: person’s name

PHONE
: phone number

PROP_MISC
: proper noun with an ambiguous (miscellaneous) classification. Example: Newton

SSN
: Social Security number

TIME
: time or time expression. Example: 0800, 1:15, 6pm

TIME_PERIOD
: measure of time expression. Example: 9a.m–6pm, 16 weeks

TITLE
: person’s title or position

VEHICLE
: motor vehicle including color, year, make, and model

A *custom concept* is a concept whose rules you must write.

Note: If you have imported a SAS Enterprise Content Categorization project, the concepts that were created using LITI rules appear in your project as custom concepts. You can edit them further by using the rules editor.

For more information about writing concept rules, see “Writing Concept Rules: Basic LITI Syntax” on page 53. For information about writing Boolean rules, see “Writing Category Rules: Boolean Rules” on page 79.

Terms and Synonyms

A *term* is defined as a label for a group of strings or patterns that represent a single concept (an idea) as defined by underlying rules or algorithms. In SAS Contextual Analysis, a term is the basic building block for topics, term maps, and category rules. Each term has an associated role that either is blank or identifies the term's part of speech. A term reflects one or more surface forms. A *surface form* is a variant of a term that is located in a matched subset of text. Surface forms can include inflected forms, synonyms, misspellings, and other ways of referring to a term. SAS Contextual Analysis can identify and classify misspellings of terms based on similarity and frequency. Because misspellings actually refer to another term, they are treated as synonyms during analysis.

A *synonym list* is a SAS data set that identifies pairs of words that should be treated as single terms for the purposes of analysis. Synonyms are applied at the parent level. You can specify a synonym list in the Create New Project wizard and in the Edit Project wizard. Synonym lists are stored in data sets and have a required format. You must include the following variables:

- TERM, which contains a term to treat as a synonym of the PARENT.
- PARENT, which contains the representative term to which the TERM should be assigned.

You can also include the following variables:

- TERMROLE, which enables you to specify that the synonym is assigned only when the TERM occurs in the role specified in this variable. A *term role* is a function performed by a term in a particular context; term roles include part-of-speech roles, entity roles, and user-defined roles.
- PARENTROLE, which enables you to specify the role of the PARENT.

> **TIP** If you want to use any of the terms in your synonym list to extract concepts, you must create custom concepts for the PARENTROLE that match the PARENT terms (or verify that the concepts exist). After the concepts are established, rerun

the terms. For example, suppose the parent term `Luke Skywalker` specifies the parent role `JEDI_MASTER`. You must create a custom concept called `JEDI_MASTER` that includes a rule that matches `Luke Skywalker` and then rerun the terms.

For more information about roles, see the section “Term Roles and Attributes” in *SAS Text Miner: Reference Help*.

Note: If a synonym list includes multiple entries that assign the same terms to different parents, then the parsing results reflect only the first entry.

Start Lists and Stop Lists

You use start lists and stop lists to control which terms are or are not used in a text mining analysis. A *start list* is a data set that contains a list of terms to include in the parsing results. If you use a start list, then only terms that are included in that list appear in parsing results. A *stop list* is a data set that contains a list of terms to exclude from the parsing results. You can use stop lists to exclude terms that contain little information or that are extraneous to your text mining tasks. A default stop list is provided for English (Sashelp.EngStop).

Start lists and stop lists have the same required format. You must include the variable TERM, which contains the terms to include (start) or exclude (stop). You can also include the variable ROLE, which contains an associated role. If you specify a ROLE variable, then terms are kept (for a start list) or dropped (for a stop list) only if their role is the one that is specified in the ROLE variable.

Topics

Topics are derived from natural groupings of important terms that occur in your documents. In SAS Contextual Analysis, topics are automatically generated and assigned to documents. A single document can contain more than one topic.

The Topics page displays all the topics that SAS Contextual Analysis identified. The default name of a topic is the top five terms that appear frequently in the topic. These terms are sorted in descending order based on their weight.

Categories

A *category* identifies a group of documents that share a common characteristic.

For example, you could use categories to identify the following:

- areas of complaints for hotel stays
- themes in abstracts of published articles
- recurring problems in a warranty call center

You create categories by promoting a topic to a category, specifying a category variable in the New Project wizard, or creating a new category. You can also import categories from SAS Enterprise Content Categorization. You can edit the rules that are automatically generated for category variables and for topics that are promoted to categories.

Note: The category rules are in the format that SAS Enterprise Content Categorization uses (MCAT), rather than in LITI format. You can refer to LITI concepts from within categories.

For more information about writing concept rules, see "Writing Concept Rules: Basic LITI Syntax" on page 53. For information about writing Boolean rules, see "Writing Category Rules: Boolean Rules" on page 79.

Using the Analysis Task Pages

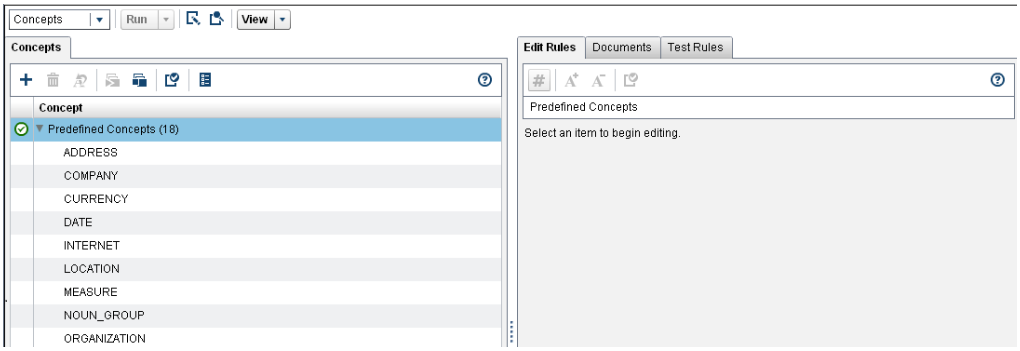
Concepts Page

The Concepts page enables you to view predefined and imported concepts, add custom concepts, test concept rules, edit concept properties, and view the documents that contain matches.

TIP Customize your view of the items that are associated with a concept node by dragging the **Edit Rules**, **Documents**, and **Test Rules** tabs from one pane to another.

Expand the list of predefined and custom concept nodes to see what is included in your analysis.

Note: If you chose to exclude predefined concepts during project creation, you cannot access predefined concepts on the Concepts page.



Concept

Predefined Concepts (18)

ADDRESS

COMPANY

CURRENCY

DATE

INTERNET

LOCATION

MEASURE

NOUN_GROUP

ORGANIZATION

Predefined Concepts

Select an item to begin editing.

Click the toolbar buttons to disable or enable a concept node.

Note: Any terms that are associated with a disabled concept are removed from the terms list and disregarded during parsing.

Here are other important actions that you can take on the Concepts page:

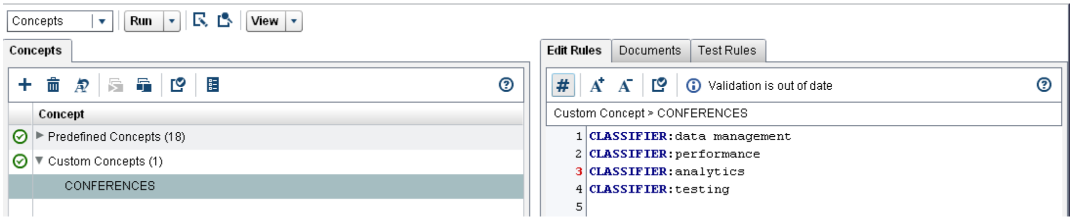
- **Add a custom concept**

  Click + to add a custom concept node for which you create your own rules.

  TIP When you create a custom concept node, use uppercase characters for its name. If you reference the concept later, it is easier to recognize the concept within the rule. Concept names are case-sensitive.

  TIP Do not use names for custom concept nodes that are also words (for example, PROBLEM or MECHANICS) that could be matched in your text. Instead, use names that cannot be interpreted as words, such as MYNEWCONCEPT.

On the **Edit Rules** tab, enter the LITI rules for the concept node. You must validate the rules before the concept node can be used in the analysis.



Validation is out of date

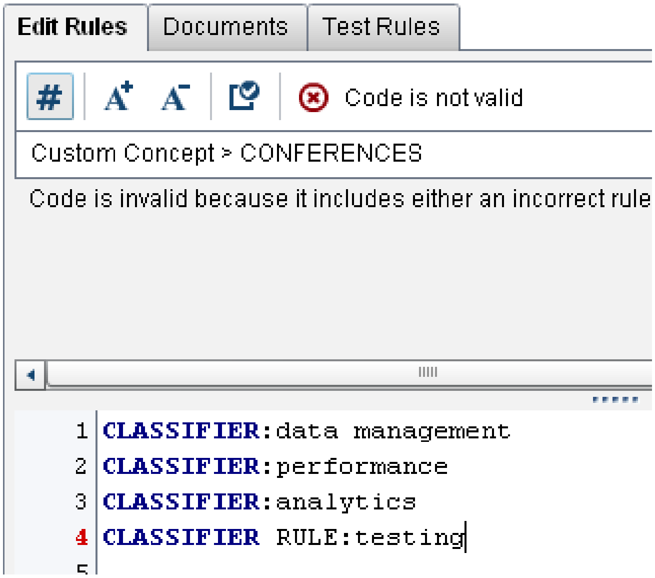
Custom Concept > CONFERENCES

Concept
▶ Predefined Concepts (18)
▼ Custom Concepts (1)
CONFERENCES

```
CLASSIFIER:data management
CLASSIFIER:performance
CLASSIFIER:analytics
CLASSIFIER:testing
```

TIP The Validation is out of date message reminds you to validate the rule after you make a change.

Click on the **Edit Rules** tab to validate each rule individually, or click in the **Concepts** to validate all the rules. Errors and other messages are displayed on the **Edit Rules** tab.



Code is not valid

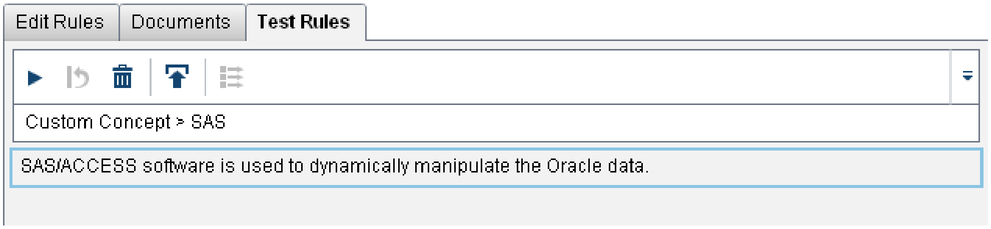
Custom Concept > CONFERENCES

Code is invalid because it includes either an incorrect rule

```
CLASSIFIER:data management
CLASSIFIER:performance
CLASSIFIER:analytics
CLASSIFIER RULE:testing
```

- **Test concept node rules**

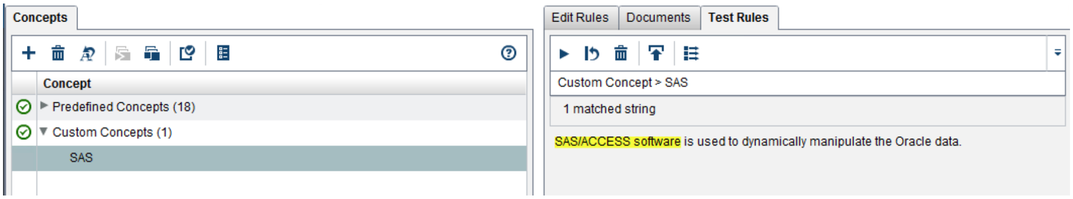
  Select a concept node in the **Concepts** tab and then click the **Test Rules** tab. Upload a file to test by clicking , or simply type test text for the rule that you have selected. Click ▶ to test the rule.



Custom Concept > SAS

SAS/ACCESS software is used to dynamically manipulate the Oracle data.

In the following sample screen, the matched strings are highlighted for the concept node **SAS**.



Predefined Concepts (18)
Custom Concepts (1)
SAS

Custom Concept > SAS

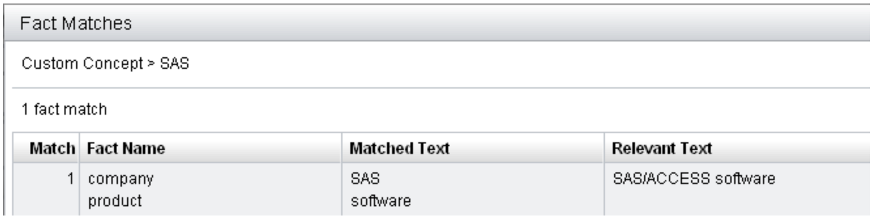
1 matched string

SAS/ACCESS software is used to dynamically manipulate the Oracle data.

Note: The matched strings (highlighted text) can contain multiple matches in concept rule testing.

You can view matches for *facts* (related pieces of information in text that are located and matched together) in a separate window so that you can examine the matched strings in your test text. Click ☰ to open the window and view the fact matches. The following sample screen shows the matches for testing the text **`SAS/ACCESS software is used to dynamically manipulate the Oracle data.`** with the rule

```
PREDICATE_RULE:(company, product):(AND, "_company{SAS}", "_product{software}")
```



Custom Concept > SAS

1 fact match

Match	Fact Name	Matched Text	Relevant Text
1	company product	SAS software	SAS/ACCESS software

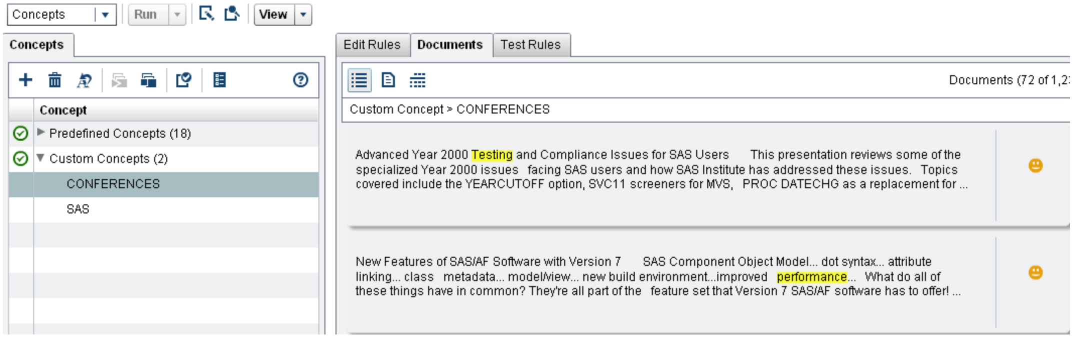
For more information about facts, see “Concepts versus Facts” on page 54.

- **View and explore matching documents**

  To view the training documents that contain matches, click the **Documents** tab. Click one of the icons ☰ 🗎 ▦ to switch between document views. Suppose you created a concept node **`CONFERENCES`**, which contains the rules

```
CLASSIFIER:testing
CLASSIFIER:performance
```

Matches within the documents are highlighted, as shown in the following sample screen:



Advanced Year 2000 Testing and Compliance Issues for SAS Users   This presentation reviews some of the specialized Year 2000 issues facing SAS users and how SAS Institute has addressed these issues. Topics covered include the YEARCUTOFF option, SVC11 screeners for MVS, PROC DATECHG as a replacement for ...

New Features of SAS/AF Software with Version 7   SAS Component Object Model... dot syntax... attribute linking... class metadata... model/view... new build environment...improved performance... What do all of these things have in common? They're all part of the feature set that Version 7 SAS/AF software has to offer! ...

Note: Sentiment values are displayed only if you applied a sentiment model when you created the project.

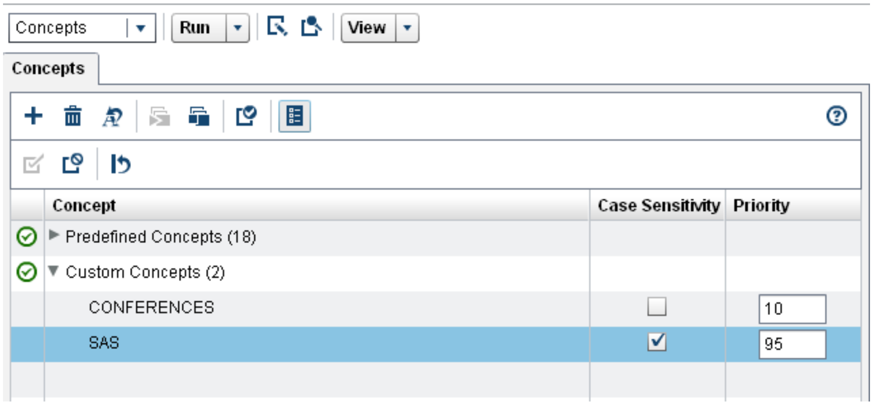
- **Edit custom concept node properties to refine concept matches**

  You can edit certain properties to help refine the matches from your custom concept rules. Click ☰ to view the properties.

  Select the **Case Sensitivity** check box to ensure that matches occur for the cases that are specified in the rule.

  You can prioritize which matches are returned when overlapping matches occur (for example, a concept node that matches **`New York`** and another concept node that matches **`New York City`**). In the case of overlapping matches, the concept node with the highest number entered in the **Priority** column is returned. The value must be a positive number (1 is the lowest priority). There is no limit to the highest priority value. The default value is 10.

  Reset all priority values to default values by clicking ↶.



	Concept	Case Sensitivity	Priority
✓	▸ Predefined Concepts (18)		
✓	▾ Custom Concepts (2)		
	CONFERENCES	□	10
	SAS	☑	95

Terms Page

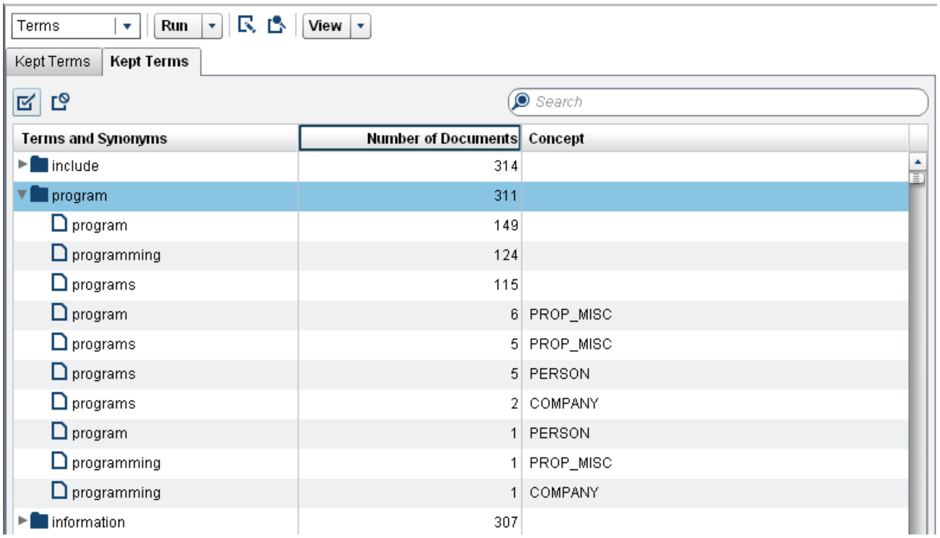
After a project is successfully run, open the **Terms** page to view the terms that were discovered in your document collection. The default view shows the **Kept Terms** on the left and the **Dropped Terms** on the right. Use the icons ☑ and ⊘ to switch views within a tab.

> **TIP** To customize your view, drag the **Document**, **Kept Terms**, or **Dropped Terms** tabs from one pane to another.

Here are other important tasks that you can complete in the Terms page:

- **View Terms**

  The **Kept Terms** displays all the terms in the document collection that were kept. The **Number of Documents** column displays the number of training documents that contain the selected term. The **Concept** column displays each term’s role, if one can be determined. To view the surface forms that were assigned to a term, click the triangle that appears next to that term.



Terms and Synonyms	Number of Documents	Concept
include	314	
program	311	
program	149	
programming	124	
programs	115	
program	6	PROP_MISC
programs	5	PROP_MISC
programs	5	PERSON
programs	2	COMPANY
program	1	PERSON
programming	1	PROP_MISC
programming	1	COMPANY
information	307	

- **Drag terms from one tab to another**

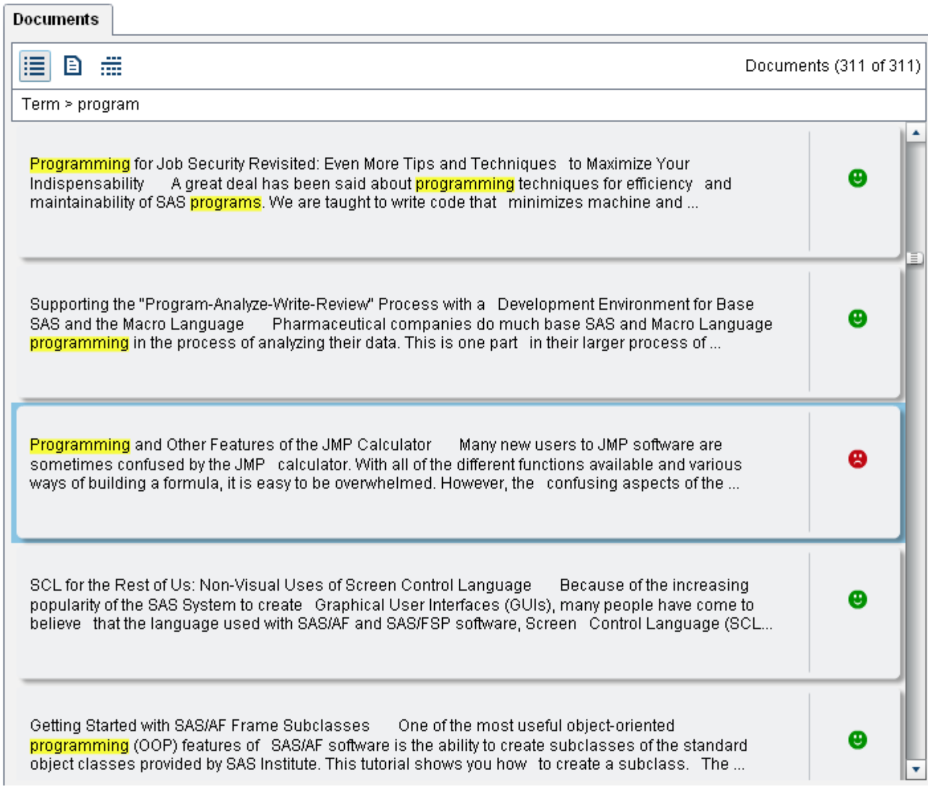
  By default, the lists of terms are sorted in descending order of the number of documents in which each term appears. You can drag parent terms from the **Kept** tab to the **Dropped** tab, and back again.

  Note: If you make changes to the terms and you want to see the effects of your changes, you must click **Run** to rerun the project.

  CAUTION! If concept rules are out-of-date when you rerun any tasks (all out-of-date tasks or topics only), any changes that you made to terms are overwritten with the original terms list.

- **View and explore matching documents**

  To view the training documents that contain matches, click the **Documents** tab. Click one of the icons ☰ 🗎 ▦ to switch between document views. Matches are highlighted, as shown in the following sample screen:



Documents (311 of 311)

Term > program

Programming for Job Security Revisited: Even More Tips and Techniques to Maximize Your Indispensability A great deal has been said about programming techniques for efficiency and maintainability of SAS programs. We are taught to write code that minimizes machine and ...

Supporting the "Program-Analyze-Write-Review" Process with a Development Environment for Base SAS and the Macro Language Pharmaceutical companies do much base SAS and Macro Language programming in the process of analyzing their data. This is one part in their larger process of ...

Programming and Other Features of the JMP Calculator Many new users to JMP software are sometimes confused by the JMP calculator. With all of the different functions available and various ways of building a formula, it is easy to be overwhelmed. However, the confusing aspects of the ...

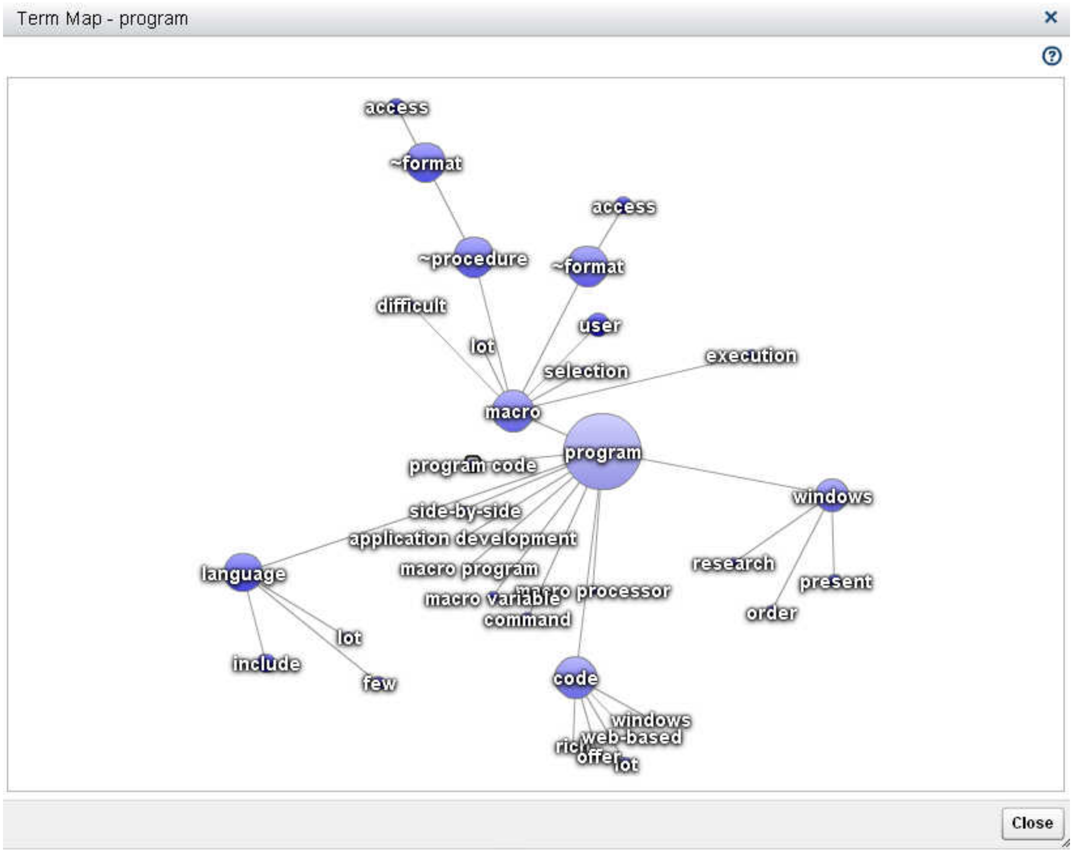
SCL for the Rest of Us: Non-Visual Uses of Screen Control Language Because of the increasing popularity of the SAS System to create Graphical User Interfaces (GUIs), many people have come to believe that the language used with SAS/AF and SAS/FSP software, Screen Control Language (SCL...

Getting Started with SAS/AF Frame Subclasses One of the most useful object-oriented programming (OOP) features of SAS/AF software is the ability to create subclasses of the standard object classes provided by SAS Institute. This tutorial shows you how to create a subclass. The ...

Note: Sentiment values are displayed only if you applied a sentiment model when you created the project.

- **View a term map**

  To view a **Term Map** for a term, select that term in the **Kept Terms** and click .



The Term Map window displays a term map for the selected term. In the preceding sample screen, the selected term is *program*, and it is represented by the largest circle in the map. For more information about reading the map, click ? above the term map.

Topics Page

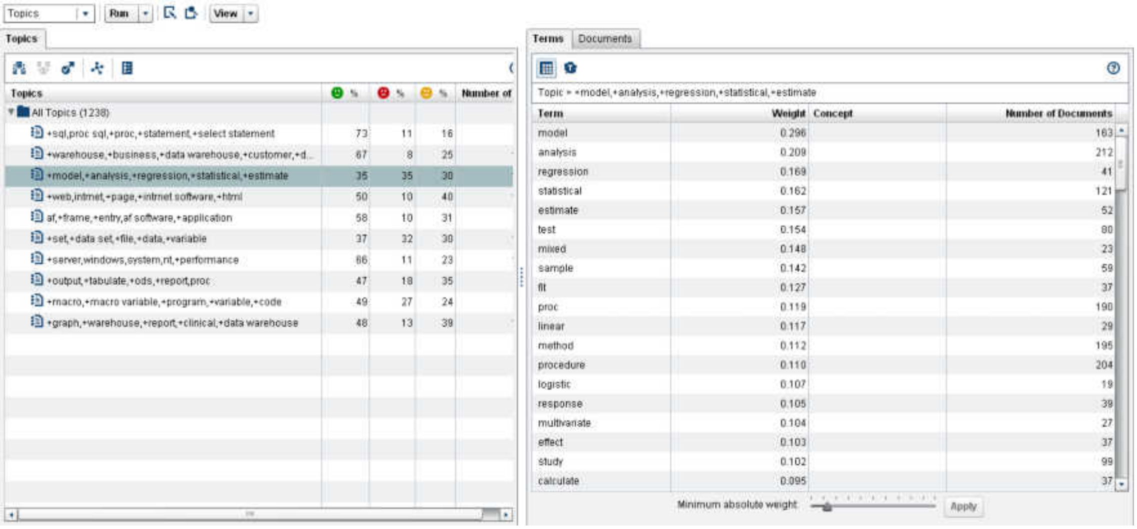
To analyze a topic, select that topic on the **Topics** tab. The selected topic is identified by its five most important terms. Here are the tasks that you can perform on the Topics page:

- **View terms that comprise the topic**

  In the following sample screen, the topic is identified by the terms **model**, **analysis**, **regression**, **statistical**, and **estimate**.

Note: The percentage of the documents in the topic that have a sentiment score of positive, negative, and neutral appears with each topic, provided that you included a sentiment model when you created the project.

When you select a topic, the view in the right is updated. Use the icons in the right 's toolbar to switch views. For information about each view, click ? in the right . Click ▦ in the right to view the terms list as a table. The terms table displays every term in the topic, its calculated weight, its assigned role (concept), and the number of documents that contain that term.

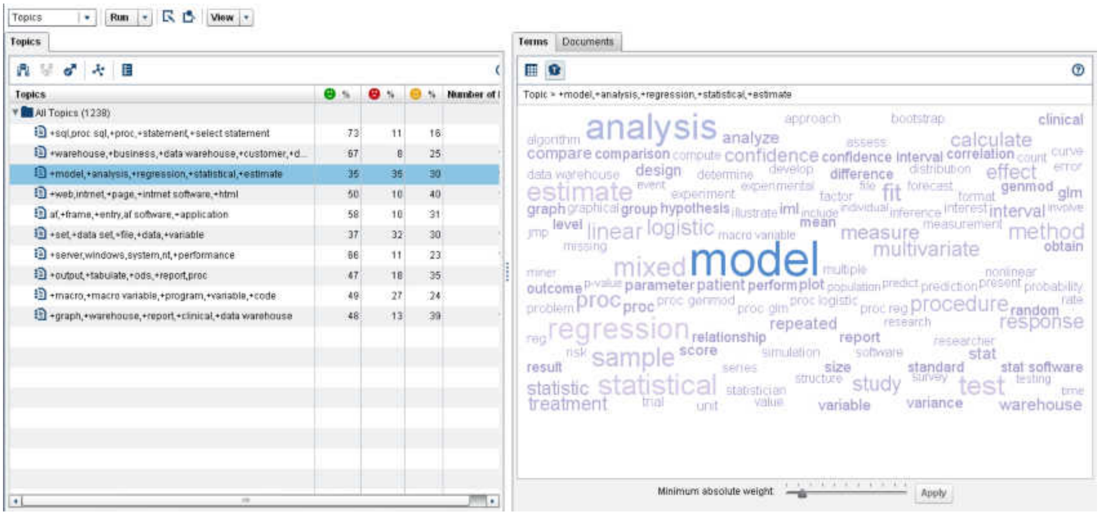


Topics	🟢 %	🔴 %	🟡 %	Number of
All Topics (1238)				
+sql,proc sql,+proc,+statement,+select statement	73	11	16	
+warehouse,+business,+data warehouse,+customer,+d...	67	8	25	
+model,+analysis,+regression,+statistical,+estimate	35	35	30	
+web,intrnet,+page,+intrnet software,+html	50	10	40	
af,+frame,+entry,af software,+application	58	10	31	
+set,+data set,+file,+data,+variable	37	32	30	
+server,windows,system,nt,+performance	66	11	23	
+output,+tabulate,+ods,+report,proc	47	18	35	
+macro,+macro variable,+program,+variable,+code	49	27	24	
+graph,+warehouse,+report,+clinical,+data warehouse	48	13	39	

Topic = +model,+analysis,+regression,+statistical,+estimate

Term	Weight	Concept	Number of Documents
model	0.296		163
analysis	0.209		212
regression	0.169		41
statistical	0.162		121
estimate	0.157		52
test	0.154		80
mixed	0.148		23
sample	0.142		59
fit	0.127		37
proc	0.119		190
linear	0.117		29
method	0.112		195
procedure	0.110		204
logistic	0.107		19
response	0.105		39
multivariate	0.104		27
effect	0.103		37
study	0.102		99
calculate	0.095		37

In the following sample screen, the word-cloud view icon T was selected. A slider at the bottom of the word-cloud pane enables you to adjust the minimum absolute weight necessary for a term to be included in this topic. The word cloud is updated as you move the slider to the right. Click **Apply** to finalize the changes that you make.



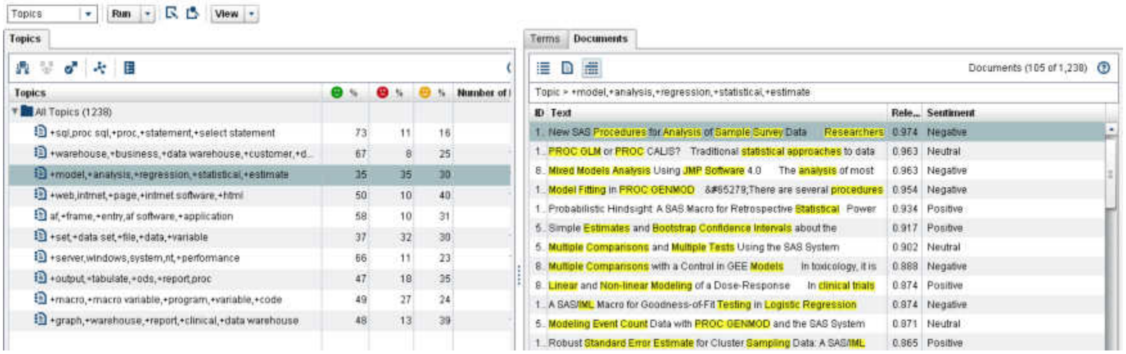
Topics	● %	● %	● %	Number of
All Topics (1238)				
+sql,proc sql,+proc,+statement,+select statement	73	11	16	
+warehouse,+business,+data warehouse,+customer,+d...	67	8	25	
+model,+analysis,+regression,+statistical,+estimate	35	35	30	
+web,intrnet,+page,+intrnet software,+html	50	10	40	
af,+frame,+entry,af software,+application	58	10	31	
+set,+data set,+file,+data,+variable	37	32	30	
+server,windows,system,nt,+performance	66	11	23	
+output,+tabulate,+ods,+report,proc	47	18	35	
+macro,+macro variable,+program,+variable,+code	49	27	24	
+graph,+warehouse,+report,+clinical,+data warehouse	48	13	39	

Topic > +model,+analysis,+regression,+statistical,+estimate

- **View documents associated with the topic**

  To view the training documents that are associated with a topic, click the **Documents** tab.

  Select one of the document view icons ☰ 🗎 ▦ to see the selected topic's terms. In the following sample screen, all of the terms that mark the documents as being a part of this topic are highlighted.



Topics	● %	● %	● %	Number of
All Topics (1238)				
+sql,proc sql,+proc,+statement,+select statement	73	11	16	
+warehouse,+business,+data warehouse,+customer,+d...	67	8	25	
+model,+analysis,+regression,+statistical,+estimate	35	35	30	
+web,intrnet,+page,+intrnet software,+html	50	10	40	
af,+frame,+entry,af software,+application	58	10	31	
+set,+data set,+file,+data,+variable	37	32	30	
+server,windows,system,nt,+performance	66	11	23	
+output,+tabulate,+ods,+report,proc	47	18	35	
+macro,+macro variable,+program,+variable,+code	49	27	24	
+graph,+warehouse,+report,+clinical,+data warehouse	48	13	39	

Documents (105 of 1,238)

Topic > +model,+analysis,+regression,+statistical,+estimate

ID	Text	Rele...	Sentiment
1...	New SAS Procedures for Analysis of Sample Survey Data Researchers	0.974	Negative
1...	PROC GLM or PROC CALIS? Traditional statistical approaches to data	0.963	Neutral
6...	Mixed Models Analysis Using JMP Software 4.0 The analysis of most	0.963	Negative
1...	Model Fitting in PROC GENMOD &#65279;There are several procedures	0.954	Negative
1...	Probabilistic Hindsight: A SAS Macro for Retrospective Statistical Power	0.934	Positive
5...	Simple Estimates and Bootstrap Confidence Intervals about the	0.917	Positive
5...	Multiple Comparisons and Multiple Tests Using the SAS System	0.902	Neutral
8...	Multiple Comparisons with a Control in GEE Models In toxicology, it is	0.888	Negative
8...	Linear and Non-linear Modeling of a Dose-Response In clinical trials	0.874	Positive
1...	A SAS/IML Macro for Goodness-of-Fit Testing in Logistic Regression	0.874	Negative
5...	Modeling Event Count Data with PROC GENMOD and the SAS System	0.871	Neutral
1...	Robust Standard Error Estimate for Cluster Sampling Data: A SAS/IML	0.865	Positive

- **Split and merge topics**

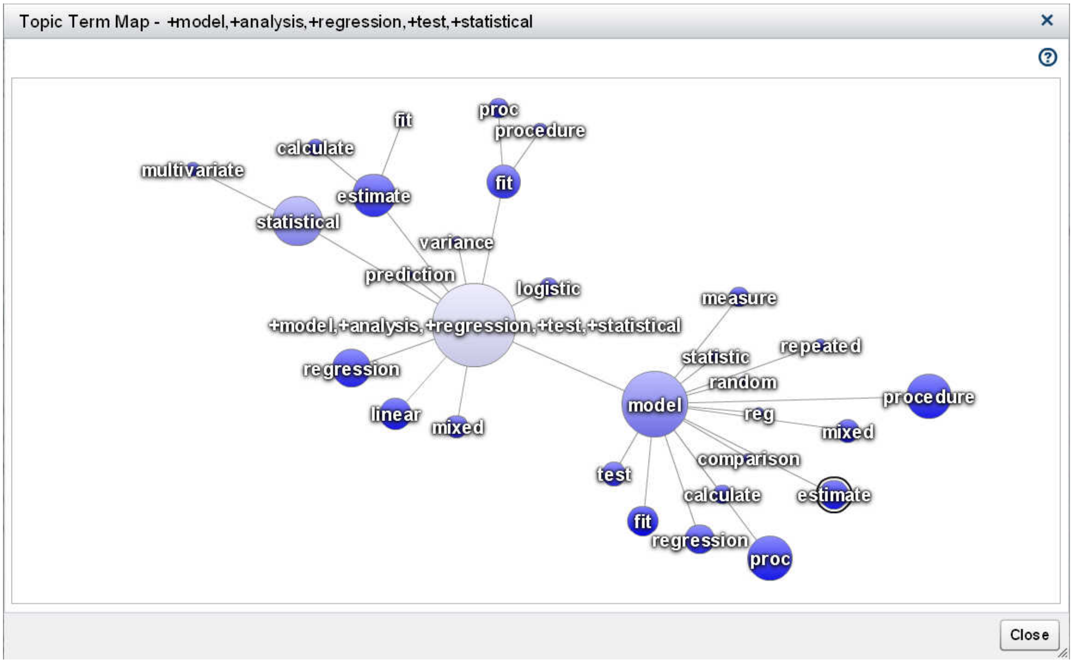
  If you consider a topic too broad for your purposes, you can split it by selecting it in the **Topics** and then clicking ⊞. This action splits the selected topic into two new topics.

If you see two topics that seem related, you can merge them by selecting them and clicking . This action combines all the selected topics into the same topic.

Note: If you want to revert to the unmerged topics after you merge them, you can do so by rerunning topics. Your changes to terms and topics up to that point will be lost.

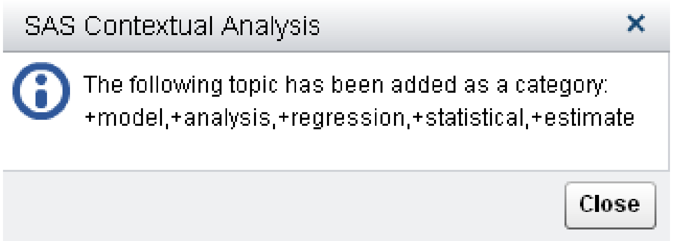
- **View a topic term map**

  You can view a topic term map from the **Topics** pane by selecting a topic and clicking . In a topic term map, the tilde at the beginning of a term is treated as a NOT operator. For more information about reading the map, click ? above the topic term map.



- **Promote topics to categories**

  A key step in the analysis is to identify which topics you want to promote to categories. To promote a topic to a category, select that topic in the **Topics** pane and click . When you click this icon, SAS Contextual Analysis adds the selected topic to the **Categories** page. You can promote multiple topics to categories at one time.



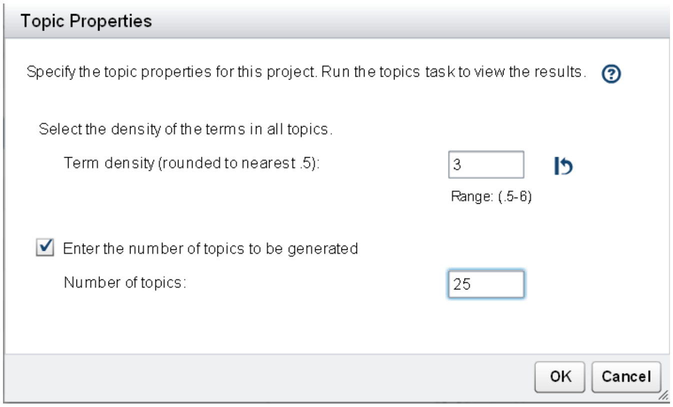
SAS Contextual Analysis

The following topic has been added as a category:
+model,+analysis,+regression,+statistical,+estimate

- **Edit topic properties**

  You can edit the properties that affect all topics. Term density refers to how topics are populated with terms; it is defined by a number between 0.5 and 6 (the default value is 2). When term density is closer to 0.5, topics are more densely populated by terms. When term density is closer to 6, topics are less densely populated by terms. This value affects the number of documents that belong to a topic (for example, having fewer terms in a topic captures fewer documents). Values that you enter are rounded to the nearest integer or half-integer.

  You can also designate a maximum number of topics that you want generated for the project. If you leave the setting blank, the software uses default methods to generate topics from important terms.



Topic Properties

Specify the topic properties for this project. Run the topics task to view the results.

Select the density of the terms in all topics.

Term density (rounded to nearest .5): 3

Range: (.5-6)

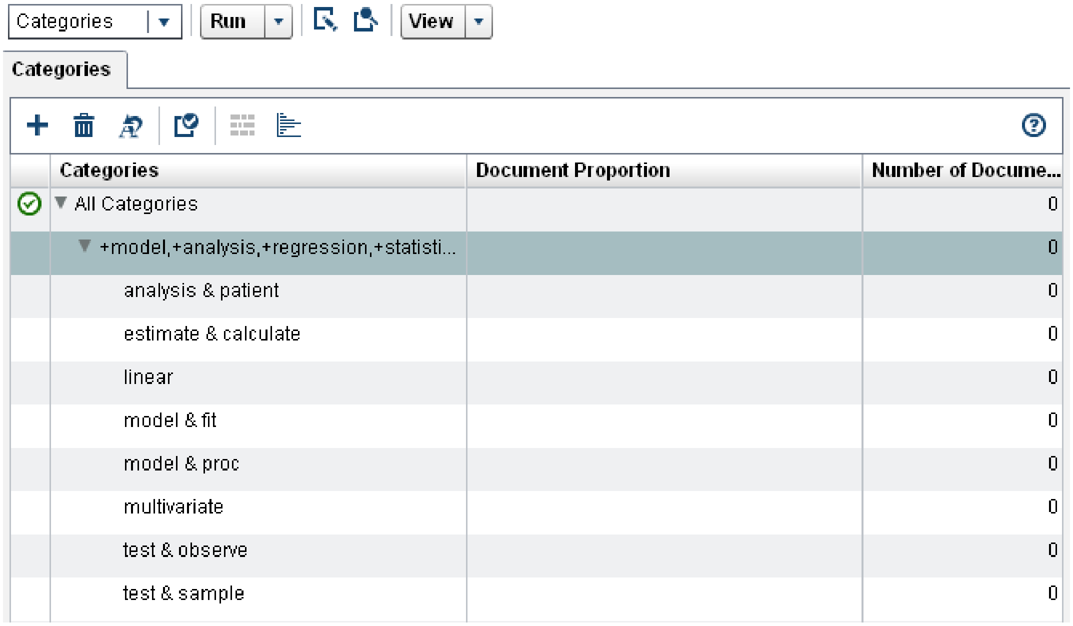
☑ Enter the number of topics to be generated

Number of topics: 25

Note: You must run the topics to see the results of your changes.

Categories Page

After you create a category from a topic in the **Topics** page, the category appears on the **Categories** page. In the **Edit Rules** tab, you see the rules that were generated for that category. The **Documents** tab is not populated until you run the category.



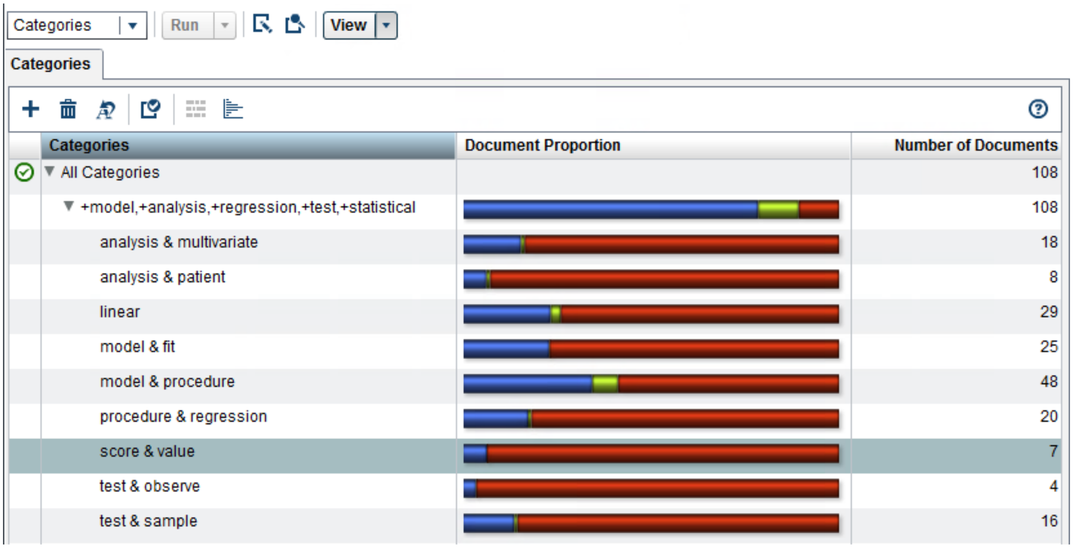
	Categories	Document Proportion	Number of Docume...
✓	▼ All Categories		0
	▼ +model,+analysis,+regression,+statisti...		0
	analysis & patient		0
	estimate & calculate		0
	linear		0
	model & fit		0
	model & proc		0
	multivariate		0
	test & observe		0
	test & sample		0

TIP Customize your view of the items associated with a category by dragging the **Edit Rules**, **Documents**, and **Test Rules** tabs from one to another.

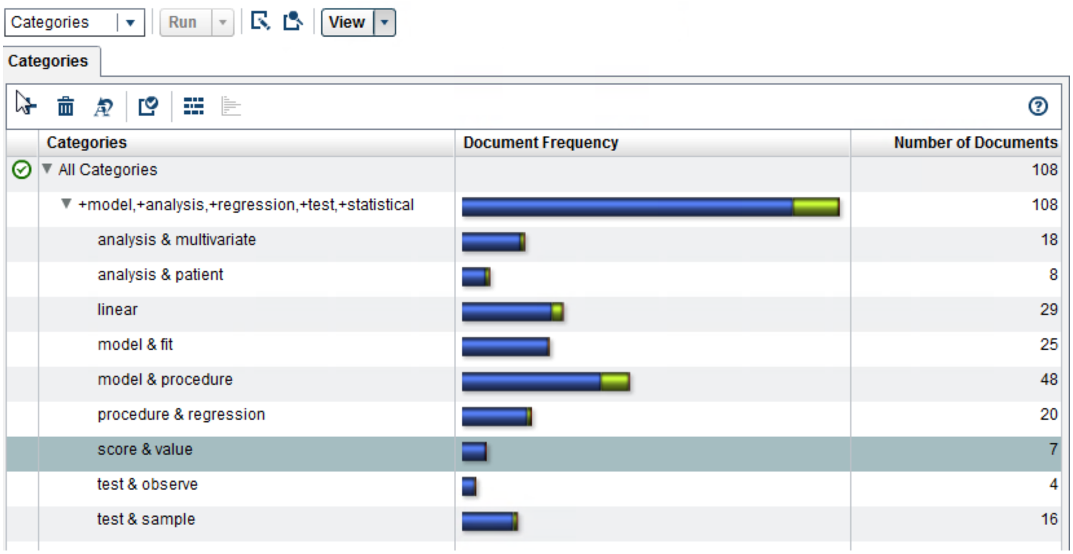
Here are the important tasks that you can perform on the **Categories** page:

- **Run the categories**

  Use the **Run** menu to compile your topics into categories. The rules for each category are executed against the input data set. This action updates the **Document Proportion** and the **Document Frequency** columns. Use ☰ and ▦ to switch between the column views.



Categories	Document Proportion	Number of Documents
All Categories		108
+model,+analysis,+regression,+test,+statistical		108
analysis & multivariate		18
analysis & patient		8
linear		29
model & fit		25
model & procedure		48
procedure & regression		20
score & value		7
test & observe		4
test & sample		16



Categories	Document Frequency	Number of Documents
All Categories		108
+model,+analysis,+regression,+test,+statistical		108
analysis & multivariate		18
analysis & patient		8
linear		29
model & fit		25
model & procedure		48
procedure & regression		20
score & value		7
test & observe		4
test & sample		16

The following colors are used In the **Document Proportion** and **Document Frequency** columns:

Blue - True Positive

The category rules captured documents that you intended to capture. You want to maximize this number.

Green - False Positive

The category rules captured documents that you did not intend to capture because the documents do not fit the category.

Red - False Negative

The category rules missed documents that were found in the topic.

Gray - Matches

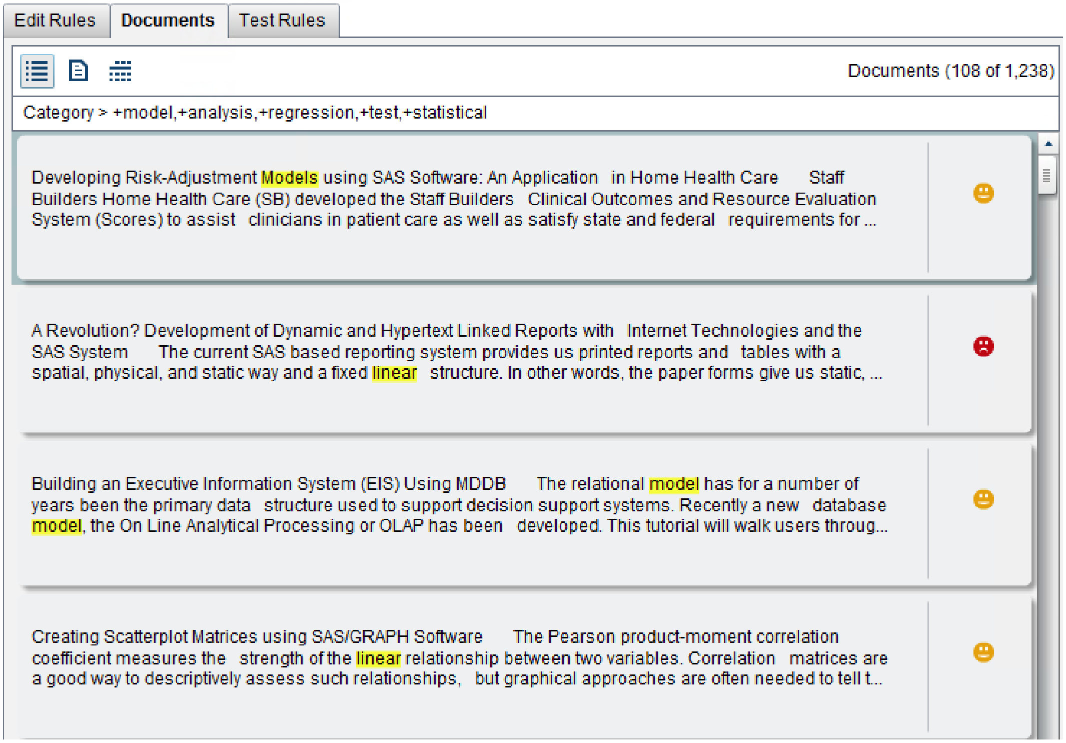
Gray bars show the statistics for matches on categories that are custom or imported.

Note: If you run categories that contain no rules or subcategory rules, the results show as false negatives (for categories generated from promoted topics or external category variables only).

To view only the documents that are true positives, click the blue portion of the bar and click the **Documents** tab. To view only the false positives, click the green portion of the bar and click the **Documents** tab, and so on.

- **View document matches for categories**

  The **Documents** tab is updated to display only the documents that meet your selection. Use the icons ☰ 🗎 ▦ to switch between views. Highlighted terms were used to determine the document’s membership in the category.



Documents (108 of 1,238)

Category > +model,+analysis,+regression,+test,+statistical

Developing Risk-Adjustment Models using SAS Software: An Application in Home Health Care Staff Builders Home Health Care (SB) developed the Staff Builders Clinical Outcomes and Resource Evaluation System (Scores) to assist clinicians in patient care as well as satisfy state and federal requirements for ...

A Revolution? Development of Dynamic and Hypertext Linked Reports with Internet Technologies and the SAS System The current SAS based reporting system provides us printed reports and tables with a spatial, physical, and static way and a fixed linear structure. In other words, the paper forms give us static, ...

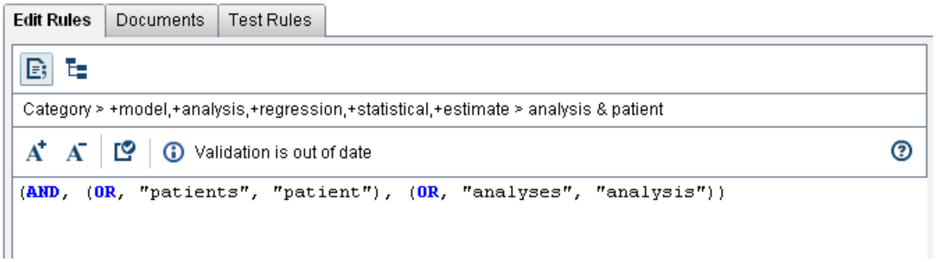
Building an Executive Information System (EIS) Using MDDB The relational model has for a number of years been the primary data structure used to support decision support systems. Recently a new database model, the On Line Analytical Processing or OLAP has been developed. This tutorial will walk users throug...

Creating Scatterplot Matrices using SAS/GRAPH Software The Pearson product-moment correlation coefficient measures the strength of the linear relationship between two variables. Correlation matrices are a good way to descriptively assess such relationships, but graphical approaches are often needed to tell t...

Note: The sentiment score for each document is displayed only if you specified a sentiment model when you created the project.

- **Edit category rules**

  To begin editing, select a rule and then click the **Edit Rules** tab. Use the rules code icon and rules tree icon to switch between editing modes. The following sample screen shows the rule code view.



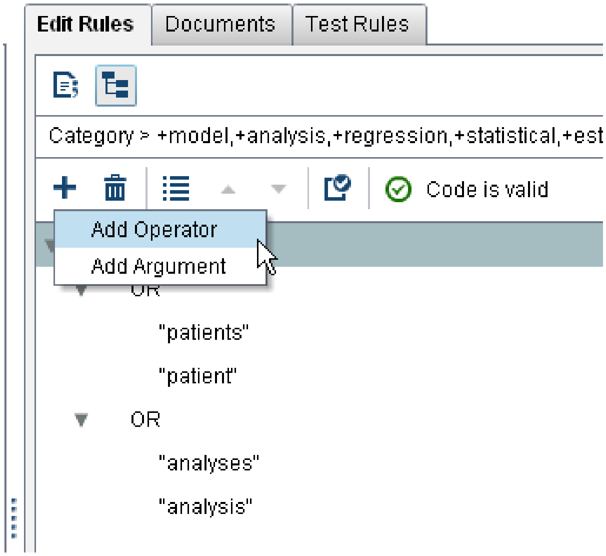
Category > +model,+analysis,+regression,+statistical,+estimate > analysis & patient

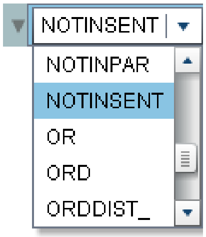
Validation is out of date

```
(AND, (OR, "patients", "patient"), (OR, "analyses", "analysis"))
```

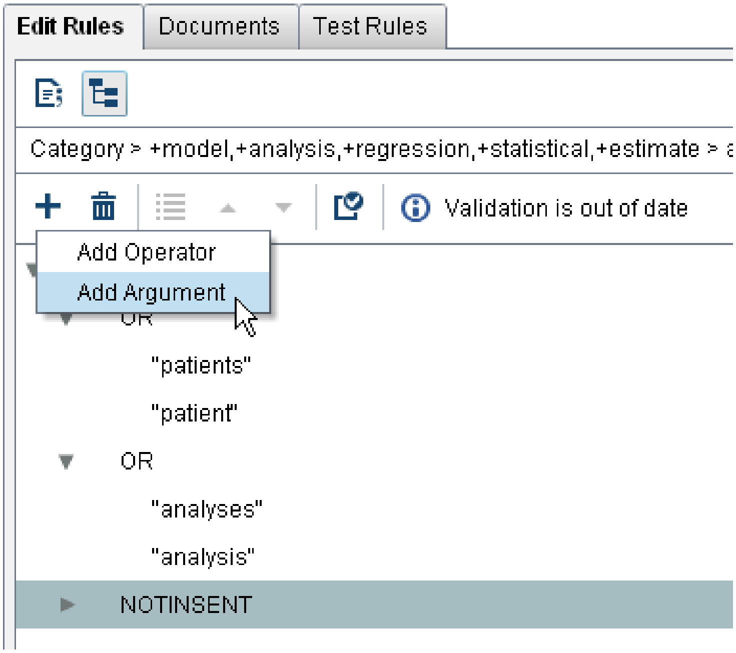
  Edit rule code by entering a rule and then validating the code.

The rule tree enables you to build rules by choosing operators and adding arguments in a visual display. Click + to add an operator or an argument to the rule. The following sample screen show adding an operator to a rule in the rule tree view.

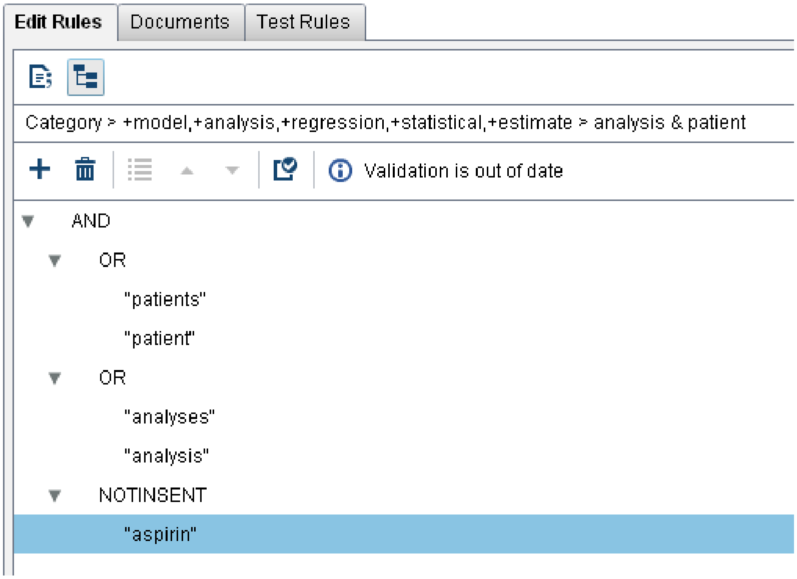




To add arguments, click + and select **Add Argument**.



Enter the argument in the space provided.


Edit Rules
Documents
Test Rules
Category > +model,+analysis,+regression,+statistical,+estimate > analysis & patient
Validation is out of date
AND
OR
"patients"
"patient"
OR
"analyses"
"analysis"
NOTINSENT
"aspirin"

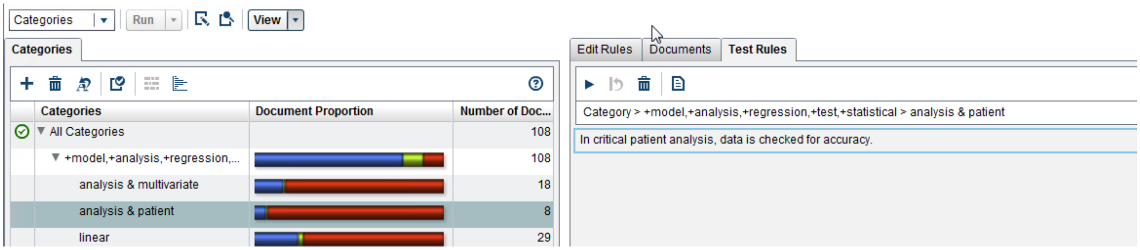
TIP The ⓘ Validation is out of date message in either view reminds you to validate the rule after you make a change.

Click ✓ on the **Edit Rules** tab to validate each rule individually, or click ✓ on the **Categories** tab to validate all the rules. Errors and other messages are displayed on the **Edit Rules** tab.

For information about writing category rules, see “Writing Category Rules: Boolean Rules” on page 79.

- **Test category rules**

  To test category rules, select a rule and then click the **Test Rules** tab. Upload a file to test by clicking ⇧, or simply type (or copy and paste) test text for the rule that you have selected. Click ▶ to test the rule.

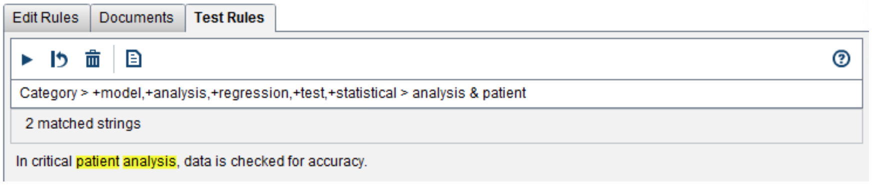


Categories	Document Proportion	Number of Doc...
✓ ▼ All Categories		108
▼ +model,+analysis,+regression,...		108
analysis & multivariate		18
analysis & patient		8
linear		29

Category > +model,+analysis,+regression,+test,+statistical > analysis & patient

In critical patient analysis, data is checked for accuracy.

In the following sample screen, the matched strings are highlighted for the rule **Analysis & patient**.



Category > +model,+analysis,+regression,+test,+statistical > analysis & patient

2 matched strings

In critical patient analysis, data is checked for accuracy.

Clear the highlighting by clicking ↺.

Writing Concept Rules: Basic LITI Syntax

Introduction to Concept Rules

Concept rules are written using LITI (language interpretation and text interpretation) syntax. Concept rules recognize items in context so that you can extract only the pieces of the document that match the rule. For example, you can create a custom concept node named `LaGuardia Airport Comments`, and then write a rule that extracts all documents in your document set that contain the word `LGA`. In other words, all of the documents displayed for the concept node `LaGuardia Airport Comments` would contain `LGA`.

Each document is evaluated separately for matches; matches do not span documents.

For information about editing rules by using the interface and by using properties settings, see "Concepts Page" on page 33. For a list of rule types, see "Which Rule Type Should I Use?" on page 55.

The following list provides basic guidelines for using LITI syntax to write concept rules. The syntax is flexible, and therefore the syntax elements can be combined in numerous ways.

- A rule consists of a rule type (which is written in uppercase letters), followed by a colon, then by arguments. For example, in the rule `CLASSIFIER:LGA`, `CLASSIFIER` is the rule type, `LGA` is the argument, and they are separated by a colon. Rule modifiers can be used to further refine the set of matches. The rule syntax varies greatly depending on the rule type; the basic syntax is included in the description of each rule in Table 3.1 on page 55 and Table 3.2 on page 57. For a list of rule modifiers, see "Adding Rule Modifiers" on page 59.
- Use descriptive concept rule names that cannot be used as single words (for example, BASEBALLSCORE). You can also include the type of rule as a prefix (for example, CONCEPT_BASEBALLSCORE).

- A single concept rule can reference one or more other concepts nodes. You can also write rules that recognize key words or elements within a specific context. For example, you can extract documents that contain the string `LGA` only if it appears before the word `Airport`.
- Use part-of-speech tags in rules to identify linguistic structures. For more information, see "Using Part-of-Speech and Other Tags" on page 68 .
- Use Boolean and proximity operators to enhance the precision of your rules. For more information, see "Using Boolean Operators for Extracting Concept Rules and Facts" on page 61.
- Use morphological expansion operators to return inflected forms of a word.
- Use coreference operators to resolve pronouns. For example, if the pronoun `he` were used to refer to `Walt Disney`, you can write a rule that specifies the canonical form (full form) and returns it in the concept. For more information, see "Using the Coreference Operator" on page 66.

Concepts versus Facts

Facts (also called predicates) are related pieces of information in text that are located and matched together.

Facts can be identified within a custom concept. For example, suppose you want to identify US universities that were named after presidents. You could write a rule that identifies `George Washington` as a US president (`US_President_Names`) and also identifies `George Washington University` as a university named for him (`UNIVERSITY`).

So, in the sentence `There are countless active student organizations at George Washington University`, the string `George Washington` would match the concept `US_President_Names` and `George Washington University` would match `UNIVERSITY`.

You can use the following special types of concept rules to locate facts:

- A predicate rule (PREDICATE_RULE) uses Boolean and proximity operators to help locate facts. For example, you can use Boolean and proximity operators to specify terms that you want to occur within a certain number of terms of each other. The following rule identifies occurrences of the term **`America`** (denoted as **`country`**) that occurs within three terms of **`flag`**, **`emblem`**, or **`crest`**:

```
PREDICATE_RULE:(country):(DIST_3,"_country{America}",
(OR, "flag", "emblem", "crest"))
```

- You can use a sequence rule (SEQUENCE) when the order of the items in the fact is important. A sequence rule can detect a structure so that each term in the fact matches in the order that you specify.

CAUTION! Although you can create and test fact rules (SEQUENCE and PREDICATE_RULE) in SAS Contextual Analysis, they are not applied to the project's documents when the project is run. As a result, you will not see fact matches within document views, topics, and auto-generated rules. To obtain fact rule matches, you can choose one of the following options: (1) Use the project's concept score code feature. For more information, see "Viewing and Downloading Code" on page 22. (2) Deploy the project's LITI binary file (which includes the fact rules) for use with SAS Enterprise Content Categorization Server. For more information, see SAS Notes for SAS Contextual Analysis, available at http://support.sas.com/notes/index.html.

Which Rule Type Should I Use?

There are several distinct types of rules for extracting concepts and facts. You can specify more than one rule in each custom concept or fact. It is important to understand the rule types so that you can select those that efficiently generate the most matches for your purposes.

Note: For the concept rule syntax listed in the following tables, < > denotes an optional syntax element. Items in *italics* denote values that you must supply, such as strings and concept node names.

Table 3.1 lists the types of rules that are used for extracting concepts. Included is a brief description of how each rule type is used, along with basic syntax. For examples of concept rule syntax, see "Concept Rule Types: Examples" on page 76.

Table 3.1 *Overview of Rules for Extracting Concepts*

Rule Type	Description and Basic Syntax

CLASSIFIER

Identifies single terms or strings that you want matched in context. For example, in a concept definition, you can create CLASSIFIER rules that contain specific airport codes. The portions of text that contain the airport codes are considered matches to the CLASSIFIER rules.

CLASSIFIER:*string* <, *information*>

CONCEPT

Identifies related information by referencing other concepts. For example, to capture documents that contain certain US airport names and locations, you can create a CONCEPT rule type in the definition. The CONCEPT rule type can reference a CLASSIFIER rule type by its name, thereby accessing a list of airport codes.

CONCEPT is a rule type. It is not to be confused with a “concept” in the general sense.

Note: The concept that you are referencing in the rule is also matched as a string. For example, in the rule `CONCEPT:SCORE`, the string `SCORE` is matched. Therefore, it is recommended that you use concept names that cannot be used as single words (for example, BASEBALLSCORE).

CONCEPT:<PRIORITY=*n*>:*argument-1*…<*argument-n*>
where *argument* can be a concept name, rule modifier, or string.

C_CONCEPT

Returns matches that occur in the specified context only. For example, to extract matches that include names of university professors, you could create a C_CONCEPT rule that identifies matches on a concept (previously defined) that identifies last names only when the matched names are preceded by the word `Professor`.

Note: This rule type requires the `_c` modifier.

C_CONCEPT:<*argument*> _c{*argument*}<*argument*>
where *argument* can be a concept name, rule modifier, or string.

CONCEPT_RULE

Uses Boolean and proximity operators to determine matches. For a list of Boolean operators, see “Boolean and Proximity Operators for Category Rules” on page 80.

Note: This rule type requires the `_c` modifier. Quotation marks (") must surround the strings that you want to match. The `c{}` can surround only one argument, which is highlighted when matches are returned. The other arguments that appear in quotation marks provide context for the match and must be present for a match to occur.

CONCEPT_RULE:(<*Boolean-rule-1*>…<*Boolean-rule-n*>
where *Boolean-rule* can be nested *n* times and is written as:
Boolean-operator “_c{*argument-1*}”,<“*argument-2*”>…<“*argument-n*”>)

NO_BREAK	Prevents partial matches by ensuring that a match occurs only if the entire string is located. For example, suppose you want to capture text that includes the item `National Gallery of Art`. You can create a rule that ensures that the entire string `National Gallery of Art` is matched and not `Gallery` and `Art` as separate items. **Note:** This rule type requires the `_c` modifier. **NO_BREAK**: _c{*argument*} where *argument* can be a concept name or string.
REGEX	Identifies recurring patterns of textual information that can be expressed in numbers and characters, such as telephone numbers, license plates number and character combinations, or word pairings such as `merry-go-round`. For example, you could write a REGEX rule that matches the expression `32,768`. For more information, see “Using Regular Expressions (Regex)” on page 71. **REGEX**:*regular-expression*
REMOVE_ITEM	Ensures that a correct match is made when one word is a unique identifier for more than one concept. For example, you can write a rule that distinguishes between the Arizona `Cardinals` football team and the St. Louis `Cardinals` baseball team. The context of each match is used to eliminate incorrect matches. **Note:** This rule type requires the `_c` modifier and the ALIGNED operator. Quotation marks (") must surround the strings that you want to match. **REMOVE_ITEM**:(ALIGNED, “_c{*concept name*}”, <“*argument*”> where *argument* can be a concept name or string.

Table 3.2 lists the rules used for extracting facts. Included is a brief description of how each rule type is used, along with basic syntax.

Table 3.2 *Overview of the Rules for Extracting Facts*

Rule Type	Description and Basic Syntax

PREDICATE_ RULE	Helps you define facts that you want identified in text. For information about facts, see “Concepts versus Facts” on page 54. **PREDICATE_RULE**:(*argument-name-1*… <*argument-name-n*>): (*Boolean-rule-1*…<*Boolean-rule-n*>) where *argument-name* refers to a name you specify for fact matching, and where *Boolean-rule* can be nested *n* times and is written as: (*Boolean-operator*, “_*argument-name* {*argument*}”, … “<_*argument-name*>{<*argument*>}”) The PREDICATE_RULE rule type requires arguments. It is more flexible than the SEQUENCE rule type because it does not always specify order.
SEQUENCE	Identifies facts in documents if the facts appear in the order specified. For information about facts, see “Concepts versus Facts” on page 54. **SEQUENCE**:(*argument-name-1*… <*argument-name-n*>):_*argument-name-1*{*argument*} <_*argument-name-n argument*> where *argument-name* refers to a name you specify for fact matching, and where *argument* refers to the name of one or more concept nodes. **Note:** This syntax is written in its simplest form. Additional modifiers and arguments for concept rule matching can be inserted. The SEQUENCE rule type requires arguments. The number of *argument-names* specified must match the number of _*argument-names*.

Using Punctuation

Use punctuation to qualify the matches for all rule types except CLASSIFIER and CONCEPT.

Colon :

Separates rule types and tags. When to use a colon:

- After a concept rule type (for example, `CLASSIFIER:`)
- Between the arguments list and the rules list in a SEQUENCE or PREDICATE_RULE definition.
- Before a part-of-speech tag (for example, `:Prep`).

Comma ,

Separates elements (such as arguments) in a concept rule definition. Add a space after the comma and before the next element. Also used to separate logical operators in a PREDICATE_RULE definition.

Single space

Separates strings, concept node names, part-of-speech tags, and rule modifiers in CONCEPT, CONCEPT_RULE, and C_CONCEPT rule types.

Quotation marks “ ”

Encloses concept node names and strings in CONCEPT_RULE, REMOVE_ITEM, and PREDICATE_RULE rule types.

Parentheses ( )

Groups the elements in CONCEPT_RULE, REMOVE_ITEM, SEQUENCE, and PREDICATE_RULE rule types.

Square braces [ ]

Groups elements in the REGEX rule type.

Curly braces { }

Delimits information that is returned as a match. Braces {} can be used in combination with parentheses `()` in some rule types.

Adding Rule Modifiers

Several types of concept rule modifiers can enhance the matching ability of a rule. Table 3.3 and Table 3.4 list the type of rule modifiers available and denote which rule types they can be used in.

Table 3.3 *Concept Rule Modifiers and Associated Rule Types*

Modifier	CLASSIFIER	CONCEPT	C_CONCEPT	CONCEPT_ RULE
Comments	X	X	X	X
Context (_c)			X (Required)	X (Required)
Word (_w)		X	X	X

Word with initial capital letter (_cap)		X	X	X
Multiple matches symbol (>)			X	X
Morphological expansion symbols (@, @A, @N, and @V)		X	X	X
Boolean andproximity operators				X
Part-of-speech tags		X	X	X
Export feature	X			
Coreference symbols (_ref, _P, and _F)		X	X	X
Regular expressions (Regex)				
Predefined concepts		X	X	X

Table 3.4 Concept Rule Modifiers and Associated Rule Types, Continued

Modifier	REMOVE_ITEM	NO_BREAK	SEQUENCE	PREDICATE _ RULE	REGEX
Comments	X	X	X	X	
Context (_c)	X (Required)	X (Required)			
Word (_w)	X	X	X	X	
Word with initial capital letter (_cap)	X	X	X	X	

> symbol					
Morphological expansion symbols (@, @A, @N, and @V)	X	X	X	X	
Boolean and proximity operators				X	
Part-of-speech tags	X	X	X	X	
Export feature					
Coreference symbols (_ref, _P, and _F)					
Regular expressions (Regex)					X (Required)
Predefined concepts	X	X	X	X	

Using Boolean Operators for Extracting Concept Rules and Facts

Table 3.5 lists Boolean operators that you can use when you write concept rules and identify facts.

Table 3.5 Boolean Operators for Extracting Concept Rules and Facts

Operator	Description

ALIGNED

Takes two arguments. Returns a match when both arguments are present (aligned) in a document. Used with the REMOVE_ITEM, CONCEPT_RULE, and PREDICATE_RULE rule types. For example, the following rule specifies that if a match on rules in the `LOC` concept node also matches rules in the `PERSON` concept node, then the match on `LOC` should be removed:

```
REMOVE_ITEM:ALIGNED, ("_c{LOC}", "PERSON")
```

Note: This operator type requires the `_c` modifier.

AND

Takes one or more arguments. Matches if all arguments occur in the document, in any order. For example, the following rule returns a match on `King Louis XIV` if it occurs in the document with `France`:

```
CONCEPT_RULE:(AND, "_c{King Louis XIV}", "France")
```

DIST_*n*

(Distance) Takes a value for *n* and two or more arguments. Matches if all arguments occur within *n* (or fewer) words of each other, regardless of their order. For example, the following rule returns a match in the phrase `the picture with the best lighting`:

```
CONCEPT_RULE:(DIST_5, "best", "_c{picture}")
```

Note: For calculation purposes, the distance between words is not inclusive. For example, the distance between the words `best` and `show` in the phrase `best in show` is two words. Words that include hyphens are counted as one word (for example, `merry-go-round` is one word).

NOT

Takes one argument. Matches if the argument does not occur in the document. Must be used with the AND operator. For example, the following rule returns a match if `cinema`, `theater`, or `theatre` occur in the document, but `Broadway` does not:

```
CONCEPT_RULE: (AND, (OR, "_c{cinema}", "_c{theater}", "_c{theatre}"), (NOT, "Broadway"))
```

Note: The NOT operator applies across the entire document. If you specify the OR operator in addition to the AND operator, you must enclose the OR arguments in parentheses.

OR

Takes one or more arguments. Matches if at least one argument occurs in the document. For example, the following rule returns a match if one or more of the items `U.S.`, `US`, or `United States` appear in the document:

```
CONCEPT_RULE:(OR, "_c{U.S.}", "_c{US} ", "_c{United States}")
```

Note: Rules that are generated by SAS Contextual Analysis nest the OR operator within the AND operator. However, the OR operator can stand alone.

ORD

(Order) Takes one or more arguments. Matches if all of the arguments occur in the order specified in the rule. For example, the following rule returns a match in the sentence **`The warranty claim for the washing machine was denied.`**:

```
(ORD,  “warranty”, “claim”, “denied”)
```

ORDDIST_*n*

(Order and distance) Takes a value for *n* and two or more arguments. Matches if all arguments occur in the same order that is specified in the rule and if all arguments are within *n* words of each other. For example, the following rule returns a match in the phrase **`the teacher introduced elementary statistics`** because the arguments appear in the correct order and within five words of each other:

```
CONCEPT_RULE:(ORDDIST_5, "elementary", "_c{statistics}")
```

Note: For calculation purposes, the distance between words is not inclusive. For example, the distance between the words **`best`** and **`show`** in the phrase **`best in show`** is two words. Words that include hyphens are counted as one word (for example, **`merry-go-round`** is one word).

PARA

(Paragraph) Matches if all the arguments occur in a single paragraph, in any order. For example, the following rule returns a match if the paragraph contains the term **`Manhattan`** and also includes the term **`apartment`**. (Only **`Manhattan`** is highlighted.)

```
CONCEPT_RULE:(PARA, "_c{Manhattan}", "apartment")
```

Note: PARA rules work properly only when they are applied to data sets that contain paragraph delimiters \n\n (newline), \t\t (tab), or <P> (paragraph). PARA cannot be applied on the **Test Rules** tab. PARA also cannot be applied to data that is contained in folders.

SENT

(Sentence) Takes two or more arguments. Matches if all the arguments occur in the same sentence, in any order. For example, the following rule returns a match when **Amazon** and **river** occur within the same sentence:

```
CONCEPT_RULE:(SENT, "_c{Amazon}", "river")
```

Delimiters are used for sentence tokenization, which is a process that breaks up sentences into words, phrases, symbols, or other meaningful elements (tokens). Note that a period (.) does not necessarily indicate an end of sentence (for example, **Mr. Quackenbush** or **Boston, Mass.** could occur in the middle of a sentence). Here is a list of sentence delimiters:

\r\n\r\n	Two consecutive carriage returns and new lines (for documents created in Windows)
\r\n \r\n	Two consecutive carriage returns and new lines, separated by a space
.<SPACE>	Period (.) followed by an ASCII space
.\n	Period (.) followed by a new line
.\r	Period (.) followed by a carriage return
!	Exclamation point
!\n	Exclamation point followed by a new line
!\r	Exclamation point followed by a carriage return
?	Question mark
?\n	Question mark followed by a newline
?\r	Question mark followed by a carriage return
.)	Period followed by a closing parenthesis
!)	Exclamation point followed by a closing parenthesis
?)	Question mark followed by a closing parenthesis
.”	Period followed by double quotation marks.

SENT_*n*

(Multiple sentences) Takes a value for *n* and two or more arguments. Returns matches within *n* sentences. For example, the following rule returns a match for the concept node **GENDER** and the term **he** within two sentences. Suppose the **GENDER** concept node contains the following rule:

```
CLASSIFIER: male
```

You can then write this rule:

```
CONCEPT_RULE:(SENT_2, "_c{GENDER}", "he")
```

For more information, see the SENT operator.

SENTEND_*n*

(End of sentence) Takes a value for *n* and one or more arguments. Returns matches within *n* words of the end of the sentence. For example, suppose the **GENDER** concept node contains the following rule:

```
CLASSIFIER: female
```

Then the following rule returns a match for the concept node **GENDER** and the term **she** within five words from the end of a sentence:

```
CONCEPT_RULE:(SENTEND_5, "_c{GENDER}", "she")
```

For more information, see the SENT operator.

Note: When you specify the value of *n*, consider that the end of the sentence is `0`. Words that include hyphens are counted as one word (for example, **merry-go-round** is one word).

SENTSTART_*n*

(Start of sentence) Takes a value for *n* and one or more arguments. Returns matches within *n* words of the beginning of the sentence. For example, the following rule locates matches for the sentence **The patient experienced breathing difficulty.** :

```
CONCEPT_RULE:(SENTSTART_5, "_c{patient}" "breathing", "difficulty")
```

For more information, see the SENT operator.

Note: When you specify the value of *n*, consider that the beginning of the sentence is `0`. Words that include hyphens are counted as one word (for example, **merry-go-round** is one word).

UNLESS	Takes two arguments. Restricts certain matches within the parameters that you specify when both arguments are matched in the same document. Used in rule types PREDICATE_RULE and CONCEPT_RULE only. Specify only these Boolean operators with the UNLESS operator: AND, SENT, DIST, ORD, and ORDDIST. The Boolean operators should appear with the second argument, as shown in the example. For example, the following rule does not include the word **`river`** in its matches and returns matches for **`Mississippi`** the state and not **`Mississippi`** the river: `CONCEPT_RULE:(UNLESS, "river", (SENT, "_c{Mississippi}", "United States"))` The rule ensures that **`river`** does not appear between **`Mississippi`** and **`United States`** in the matches. **Note:** When you specify a concept node in a rule that uses the UNLESS operator, specify a concept nodes that contains only CLASSIFIER or REGEX rules.

Using the Coreference Operator

Use the coreference operator (_ref) when you want to link pronouns and other words with the canonical form (full form) of the terms that they reference.

Suppose you have a concept node **`LEADERS`** that includes this rule:

```
CLASSIFIER:Congressional leaders
```

You can create a concept node **`THEY_SAID`** that enables **`they`** to reference its canonical form, **`Congressional leaders`**. Both forms are matched in the document.

```
C_CONCEPT:_c{LEADERS} said _ref{they}
```

You can use the following symbols with the coreference operator (_ref). Place the symbol after the _ref{*concept*} operator.

> (Multiple matches)

Locates multiple instances of a match that is specified by the coreference operator (_ref). For example, you might want to return the canonical form of the name **`Ms. Geraldine Jones`** each time the nickname **`Geri`** is encountered. The > symbol enables this match to occur after the first time the canonical form of the name is located.

```
C_CONCEPT:_c{Ms. Geraldine Jones} _ref{Geri}>
```

_F (Forward)
Returns only matches that occur after the coreference rule match. Sample syntax:

```
C_CONCEPT:_c{PERSON} as _ref{TITLE}_F
```

_P (Preceding)
Returns only matches that occur before the coreference rule match. Sample syntax:

```
C_CONCEPT:_c{MILITARY BRANCH} as _ref{HONOR}_P
```

Using the Export Feature

The Export feature enables you to find matching occurrences of terms or phrases found in CLASSIFIER rules and then export them to one or more concepts. This feature is useful for conditional matching of terms or phrases. You can export matches from multiple concepts to one concept, or to more than one concept.

Note: The Export feature can be used only with CLASSIFIER rules.

For example, suppose you want to find all the occurrences of the term **accounts receivable** that occur together with the name **Sokolov**, and export those matches to the concept **AR**. You could write the following rule in a concept node named **ACCOUNT_HOLDER**:

```
CLASSIFIER:[export=AR:accounts receivable]:Sokolov
```

The rule first matches the term **Sokolov**. If that match is found, the rule checks the documents for any occurrences of the term **accounts receivable** and assigns any matches to the concept **AR**. In the list of matches for **ACCOUNT_HOLDER**, the term **Sokolov** would be highlighted. In the list of matches for **AR**, the term **accounts receivable** would be highlighted. Note that in order for the rule to work, the primary term (in the example, **Sokolov**) needs to be present anywhere in the document before **accounts receivable** can be returned as a match for concept node **AR**.

Concepts that you are exporting to (such as **AR** in the example) must exist in the list of concepts and can contain additional rules (or be empty).

Using Part-of-Speech and Other Tags

Part-of-speech tags enable you to locate matches by the part of speech that the searched item belongs to, rather than locating a specific term. These tags are useful when you know the syntax but not the exact wording of an item that you are seeking. Also included are other tags that are not considered parts of speech (such as punctuation).

Because the parts of speech are sensitive to the context in which they appear, the same word might be tagged differently, depending on the surrounding text. For example, the word **will** could be tagged as modal (she will be a big star someday) or noun (a last will and testament).

Part-of-speech tags are preceded by a colon (:). The tags are case-sensitive. For example, suppose you want to match an attribution for a quotation in a news article. You know that the syntax for the match will appear as **Senator from** *state* or **Senator of** *state* but you do not know the name of the senator. You can use the following rule:

```
C_CONCEPT:SENATE_TITLE _c{_cap _cap} :Prep STATE
```

The rule assumes that there is a concept **SENATE_TITLE** that contains words such as **majority leader**, **senator**, and **senators**, and a concept **STATE** that includes names of states. The :Prep tag indicates a preposition (for example, **from** or **of**). A match on the C_CONCEPT rule would occur on the text **Senator Phineas Craymoor from North Carolina took the floor**. However, the following text would not produce a match because the word **and** is not a preposition: **Senators Phineas Craymoor and Garrett Garcia from North Carolina pushed the bill through**.

Table 3.6 lists the part-of-speech tags in English. For tags in other languages, see Appendix 1, "Part-of-Speech Tags (for Languages Other Than English)," on page 89.

Table 3.6 *Part-of-Speech Tags (for English)*

Part-of-Speech Tag	Definition	Examples
:ABBREV	Abbreviation	etc., Ms, cm

:Acomp	Comparative adjective	cooler, luckier, worse
:Adv	Adverb	lyrically, physically
:Asup	Superlative adjective	mellowest, merriest, best
:C	Conjunction	when, yet, after, except
:date	Date	2000-02-21, 04/03/2012
:digit	Sequence of numbers	2345, 234.22, 21/234
:Det	Determiner	the, an, every
:F	Foreign	facto, klieg, modus
:inc	Unknown word	slaster, lijer
:Int	Interjection	hah, hello, tallyho
:Md	Modal	can, should, will
:N	Noun	cake, love, shoe
:Npl	Plural noun	peas, sheep, shoes
:Num	Number	one, twenty, hundred
:PN	Proper noun	SAS, Cary, Goodnight
:PossDet	Possessive determiner	our, his, my
:PossPro	Possessive pronoun	mine, yours, hers
:PreDet	Pre-determiner	quite, such, all
:Prefix	Prefix	cross, ex, multi
:Prep	Preposition	on, under, across
:Pro	Pronoun Relative pronoun	he, one, somebody, me myself, oneself, themselves

:Ptl	Particle	away, forward, in
:sep	Separator and punctuation	; , /
:time	Time	7AM, 10:00 pm
:url	File names, pathnames, URL	A:/mydir/file.txt, www.sas.com
:V	Undeclined *be*, *do*, or *have* auxiliary Undeclined verb First person singular verb	be, do, have go, see, love am
:V3sg	Third person singular *be*, *do*, or *have* auxiliary Third person singular verb	is, does, has goes, sees, loves
:Ving	Present participle *be*, *do*, or *have* auxiliary Present participle	being, doing, having bucketing, climbing
:Vpp	Past participle *be*, *do*, or *have* auxiliary Past participle	been, done, had dashed, factored, gone
:Vpt	Past tense *be*, *do*, or *have* auxiliary Past tense verb	was, were, did, have dashed, factored, went
:WAdv	Adverbial *wh*	how, when, whereby
:Wdet	Demonstrative determiner *wh*	which, what, whatever
:WPossPro	Possessive determiner *wh*	whose
:WPro	Nominal *wh*	whose, what, whoever

Using Regular Expressions (Regex)

Use regular expressions (Regex syntax) to identify regularly occurring patterns in the text that include numbers and characters. You can use regular expressions to match patterns such as license plate numbers (example: ABX-0444), part numbers for manufacturing components (example: TMS1T3B1M5R-23), hyphenated words (example: fifty-nine), and so on.

The following guidelines apply to Regex syntax:

- Include one or more characters inside square brackets (**`[ ]`**) to match the specified characters. This provides flexibility in character matching. For example, the following rule matches **`c`**, **`r`**, **`a s`**, or **`h`**:

  ```
  REGEX:[crash]
  ```

  If you add a plus sign (**`+`**) as follows, the rule matches the characters specified in any combination, such as **`rash`** , **`cash`**, **`ash`**, and **`crass`** (but not **`crashpad`** or **`crashdummy`**):

  ```
  REGEX:[crash]+
  ```

- Characters are matched within a string in sequence when represented without square brackets (**`[ ]`**). For example, the following rule matches only the word **`any`** (**`anyone`** or **`anything`** would not be matched):

  ```
  REGEX:any
  ```

  To match words that contain **`any`**, you can modify the rule to use asterisks (*) to match other character occurrences (or none) surrounding **`any`**. For example, the following rule matches **`any`**, **`anyone`**, **`anything`**, and **`Many`**:

  ```
  REGEX:[A-Za-z]*any[A-Za-z]*
  ```

- You can specify a range of characters to be matched. For example, the following rule matches lowercase characters between **`a`** and **`f`**, inclusively:

  ```
  REGEX:[a-f]
  ```

  To add uppercase characters, use the following rule:

  ```
  REGEX:[A-Fa-f]
  ```

- You can specify characters that should not be matched (negated characters) by inserting a caret (^) before a set of characters. For example, the following rule matches all characters, numbers, and symbols in text except **a**, **e**, **i**, **o**, and **u**:

  ```
  REGEX:[^aeiou]
  ```

- Characters that are reserved for special meaning (metacharacters) must be escaped with a backward slash (\) to be literally matched in a regular expression. The metacharacters are: **[**, **]**, **(**, **)**, ?, *****, **+**, **.**, **-**, \, and |

  For example, **[\?]** matches a question mark **?** in text.

- Numbers are matched as-is within a string when represented without square brackets (**[ ]**). For example, the following rule matches part numbers that begin with **0125-** and end with a letter:

  ```
  REGEX:0125\-[A-Za-z]
  ```

- Numbers are matched by specifying ranges when enclosed in square brackets (**[ ]**) For example, the following rule returns a match on a number between **0** and **9**:

  ```
  REGEX:[0-9]
  ```

The special characters used for matching in Regex syntax can be used in combination and are shown in Table 3.7 on page 72.

Table 3.7 *Special Characters (Metacharacters) Used in Regular Expressions*

Character or Expression	Description
|	(Alternative) Indicates that matches occur when either regular expression *a* or *b* is matched. Example: *a* | *b*
( )	Grouping mechanism (non-remembering). Used in expressions for clarity. Example: (*ababab*) | *b*)
.	(Wildcard) Matches any character.
%	Matches % or *percent*
?	Matches 0 or 1 occurrences
*	Matches 0 or more occurrences

+	Matches 1 or more occurrences
{ }	Indicates repetition: {*n*} matches exactly *n* occurrences {*n,m*} matches at least *n* occurrences but no more than *m* occurrences {*n,*} matches at least *n* occurrences
\a	Alarm (beep)
\n	New line
\r	Carriage return
\t	Tab
\f	Form feed
\e	Escape
\d	Digit (same as `[0-9]`)
\D	Not a digit (same as `[^0-9]`)
\w	Word character (same as `[a-zA-Z_0-9]`)
\W	Non-word character (same as `[^a-zA-Z_0-9]`)
\s	White space character (same as `[ \t\n\r\f]]`)
\S	Non-white-space character (same as `[^ \t\n\r\f]]`)
\xh	Hexadecimal number, where *h* is a hexadecimal character
\xhh	Hexadecimal number, where *h* is a hexadecimal character
\0o	Octal number, where *o* is an octal digit
\0oo	Octal number, where *o* is an octal digit

The following restrictions apply to Regex syntax:

- Regex syntax works similarly to regular expressions in Perl; however, the two are not identical.
- Character matching for characters, numbers, or symbols that are specified inside square brackets ([ ]) does not occur at the word level. For example, the following rule matches the isolated letters **`x`**, **`y`**, and **`z`**, but no matching occurs for the words **`xylitol`**, **`yes`**, or **`recognize`**:

  ```
  REGEX:[xyz]
  ```

  If you add a plus sign (+) to match multiple occurrences (or one occurrence) as follows, the rule matches any combination of the characters that are specified, such as **`xzx`**, **`yz`**, and **`zyzy`**:

  ```
  REGEX:[xyz]+
  ```

  However, because word-level matching does not occur, there is no matching for words **`xxl`**, **`syzygy`**, or **`diy`**.
- You cannot refer to concepts in a Regex expression.
- Backward references to matches in the text are not supported.
- Parentheses `( )` as a grouping mechanism where matches are remembered are not supported. Parentheses are used merely for clarifying matching rules.

Using Morphological Expansion Symbols

You can use morphological expansion in all rule types except CLASSIFIER and REGEX. For example, to expand the word **`breathe`** to all verb forms, which include **`breathes`** and **`breathing`**, use the following syntax for the argument: **`"breathe@V"`**.

Table 3.8 Morphological Expansion Symbols in Concept Rules

Symbol	Description

@	Expands the concept rule to match all inflectional forms of the word in the argument. For example, the argument **`"wonder@"`** returns the matches **`wonder`**, **`wonders`**, **`wondered`**, **`wondering`**, and so on. **Note:** If you apply @ to a word that SAS Contextual Analysis does not recognize, no expansion occurs. Only the exact string specified before the @ is matched. For example, **`"grath"`** would not expand. Only the string **`grath`** would return a match in the rule.
@A	Expands the concept rule to match inflected comparative and superlative adjective forms of the word in the argument. For example, the argument **`"happy@A"`** returns the matches **`happier`** and **`happiest`**. **Note:** If you apply @A to a word that is not an adjective, no expansion occurs.
@N	Expands the concept rule to match all inflected noun forms of the word in the argument. For example, the argument **`"quality@N"`** returns the matches **`quality`** and **`qualities`**. **Note:** If you apply @N to a word that is not a noun, no expansion occurs.
@V	Expands the concept rule to match all inflected verb forms of the word in the argument. For example, the argument **`"transfer@V"`** returns the matches **`transfer`** , **`transfers`**, **`transferred`**, and **`transferring`**. **Note:** If you apply @V to a word that is not a verb, no expansion occurs.

Adding Comments

You can insert comments into rule definitions that have separate rules appearing on successive lines, such as CLASSIFIER rules. The comment continues until the end of the line. Comments are written as

```
# comment text
```

Note: The pound character (`#`) denotes a comment. If you want to match `#` in a rule definition, you must use a backward slash (\) as an escape character before the `#`. (Example: The expression `99\#` attempts to match the string `99#`.)

TIP You can comment out a rule by inserting a pound character (`#`) at the beginning of a line that contains a rule.

Concept Rule Types: Examples

Examine the syntax in the examples to understand how to write different types of concept rules.

CLASSIFIER

Example: To extract documents that contain US airport codes, you can create a concept node named **`US_AIRPORTS`** that includes these CLASSIFIER rules:

```
CLASSIFIER:BUF
CLASSIFIER:BUR
CLASSIFIER:BVK
```

So, documents that include a match on one or more of the airport codes **`BUF`**, **`BUR`**, or **`BVK`**, return a match for **`US_AIRPORTS`**.

CONCEPT

Example: To extract documents that contain flight arrival information, create a concept node **`ON_TIME_ARRIVALS`**. The rule definition for **`ON_TIME_ARRIVALS`** contains the CONCEPT rule type. The CONCEPT rule type can reference the concept node **`US_AIRPORTS`** , which enables airport codes to be detected. The rule definition for the concept node **`ON_TIME_ARRIVALS`** is as follows: **`CONCEPT:at US_AIRPORTS on time`** (where **`US_AIRPORTS`** includes CLASSIFIER rules that identify US airport codes).

C_CONCEPT

Example: To extract documents that include names of university professors, create a C_CONCEPT rule named **`PROFESSORS`** whose definition includes this rule: **`C_CONCEPT: Professor _c{FIRSTNAME LASTNAME}`**. The rule indicates that matches are returned when **`FIRSTNAME`** and **`LASTNAME`** (previously defined) are found, but only when they are preceded by the word **`Professor`**. Provide the context for the match by using the modifier **`_c`** and enclosing the argument that you want to match in braces (**`{}`**).

The rule modifier `_c` indicates that the match occurs within the context of the specified concept nodes.

NO_BREAK

Example: Suppose you want to extract **`National Gallery of Art`**. You defined a concept node **`US_ART_GALLERIES`** that includes the CLASSIFIER rule **`National Gallery of Art`**. There also exists a concept node called **`CLASS_TYPES`** that includes the CLASSIFIER rule **`Art`**. You can create the following rule that prevents a partial match on **`CLASS_TYPES`** and ensures that the entire string **`National Gallery of Art`** is matched: **`NO_BREAK:_c{US_ART_GALLERIES}`**

The rule modifier `_c` indicates that the match occurs within the context of another concept node.

REMOVE_ITEM

Example: Suppose you want to extract the baseball team St. Louis **`Cardinals`**, but not the football team Arizona **`Cardinals`**. You have a concept node named **`FOOTBALL`** that includes the rule **`CLASSIFIER:Cardinals`**. You have another concept node named **`BASEBALL`** that includes the rule **`CLASSIFIER:Cardinals`**. The following rule returns matches for the baseball team only:

```
REMOVE_ITEM(ALIGNED, "_c{FOOTBALL}", "BASEBALL")
```

Note: The REMOVE_ITEM rule type could influence matches outside of the concept node in which it is used. In this case, the rule could influence matches in the FOOTBALL rule because the rule specifies that items be removed.

REGEX

Example: To extract whole numbers in text (such as **`1`**, **`23`**, **`456`**, and so on), use the rule

```
REGEX:[0-9]+
```

This rule requires that one or more consecutive digits occur and are without decimals.

Example: To extract a number that uses decimal notation, such as **`392.55`**, **`45.25`**, and **`0,987654321`**, use the following rule:

```
REGEX:[0-9]+[,\.][0-9]+
```

This rule returns a match on any digit 0 to 9 followed by any number of digits, commas, or periods (in any combination), and then ending in a digit.

For more information about writing Regex rules, see "Using Regular Expressions (Regex)" on page 71.

CONCEPT_RULE

Example: Suppose you want to extract Amazon the company, not Amazon the river. You could use this rule, which would return a company name within three words of **`company`**, but not if there were nature-related words in the document.

```
CONCEPT_RULE:(AND, (DIST_3, "_c{COMPANY}", "company"), (NOT, "NATURE"))
```

SEQUENCE

Example: Suppose you want to extract first and last names only from a list of first, middle, and last names. You can use a SEQUENCE rule to define the arguments **`first`** and **`last`**. By using these arguments, matches are made on the concept nodes **`FIRST_NAME`**, **`MIDDLE_NAME`**, and **`LAST_NAME`**, but matches are returned on only **`FIRST_NAME`** and **`LAST_NAME`**.

```
SEQUENCE:(first, last): _first{FIRST_NAME} MIDDLE_NAME _last{LAST_NAME}
```

PREDICATE_RULE

Example: Suppose you want to match a company to its products. You could use the following PREDICATE_RULE, which assumes that the concept node **`COMPANY`** includes CLASSIFIER rules that list company names and the concept node **`PRODUCTS`** contains CLASSIFIER rules that list products. Items must appear in the same sentence.

```
PREDICATE_RULE:(company, product):(SENT, "_company{COMPANY}",
"produces", "_product{PRODUCTS}")
```

Writing Category Rules: Boolean Rules

Introduction to Category Rules

Category rules resolve to true or false. “True” results in a match. Boolean rules use Boolean and proximity operators, arguments, and modifiers to define the conditions that are necessary for category matches. Category rules are simpler to write than LITI rules and are recommended when there is no need to extract specific information from the data. For a list of operators, see Table 3.9 on page 80.

Use the following syntax for a Boolean rule:

(***OPERATOR***, <*argument1*>, <*argument2*>, ...)
where arguments can be terms, strings, or nested rules.

General rules for syntax:

- Boolean and proximity operators are enclosed in parentheses and separated with commas. Strings within arguments are included in quotation marks (“ ”). Example: (AND, “holiday”, “vacation”)
- Rules can be nested. Example: (AND, (OR, “courage”, “courageous”), (OR, “brave”, “bravery”))
- Reference a category from another category by using special syntax called *tmac syntax* (_tmac). For more information, see “Using _tmac for Referencing Categories” on page 87.
- Concept node names can be referenced in category rules. If you reference a concept node name, all concept matches will also match in the category. Concept node names must be enclosed in braces (`[]`) and quotation marks (`" "`) For example, to reference the concept node `GAME_SHOWS` in a category rule, you could write the rule (OR, “[GAME_SHOWS]”).

  Note: Concept nodes that are named in categories might return more matches than concepts that are run outside of categories. In categories, matches on concepts are based on an “all matches” method, which returns all matches found in the text. By contrast, in concepts, matches are based on a “best match” method. The best match method detects when text that matches one concept overlaps text that matches another concept (for example, a concept that matches **`New York`** and another concept that matches **`New York City`**). When concept matches overlap and the best match method is used, only the concept that is assigned the highest number for the priority is returned (1 is the lowest). When two or more concepts have the same priority assigned, SAS Contextual Analysis selects a match.

- The enabled or disabled status of concepts that are named in categories is ignored during category matching. As a result, the concepts are processed as if they were all enabled, regardless of whether they were previously disabled.
- Special symbols can be used to modify the rules to include, wildcards, case sensitivity, and so on. For a list of symbols, see Table 3.10 on page 86.

Note: XPath expressions are not supported.

Boolean and Proximity Operators for Category Rules

Table 3.9 shows a list of Boolean and proximity operators that you can use to write category rules.

Table 3.9 *Boolean and Proximity Operators for Category Rules*

Operator	Description
AND	Takes one or more arguments. Matches if all arguments occur in the document, in any order. For example, the rule (AND, “King”, “Louis”, “XIV”) returns a match if **`King`**, **`Louis`**, and **`XIV`** all occur in the document.

DIST_*n*

(Distance) Takes a value for *n* and two or more arguments. Matches if all arguments occur within *n* (or fewer) words of each other, regardless of their order. For example, the rule (DIST_5, “best”, “picture”) returns a match in the phrase `the picture with the best lighting`.

Note: For calculation purposes, the distance between words is not inclusive. For example, the distance between the words `best` and `show` in the phrase `best in show` is two words. Words that include hyphens are counted as one word (for example, `merry-go-round` is one word).

END_*n*

(From the end of the document) Takes a value for *n* and one or more arguments. Matches if the argument occurs within *n* words from the end of the document. For example, the rule (END_35, “conclusion”) returns a match if `conclusion` is found within 35 words from the last word in the document.

Note: Words that include hyphens are counted as one word (for example, `merry-go-round` is one word).

MIN_*n*

(Minimum) Takes a value for *n* and one or more arguments. Matches if the document contains at least *n* of the arguments specified (in any order). For example, the rule (MIN_2, “Hollywood”, “tinseltown”, “movies”) returns a match if `Hollywood` and `movies` occur in the document. However, there is no match if `Hollywood` occurs twice and no other arguments occur.

MINOC_*n*

(Minimum occurrence) Takes a value for *n* and one or more arguments. Matches if the document contains at least *n* occurrences of the arguments specified (in any order or combination). For example, the rule (MINOC_2, “Hollywood”, “tinseltown”, “movies”) returns a match if `Hollywood` and `movies` occur in the document. There is also a match if `Hollywood` occurs twice and no other arguments occur.

MAXOC_*n*

(Maximum occurrence) Takes a value for *n* and one or more arguments. Matches if the document contains *n* or fewer occurrences of the arguments (in any order or combination). Useful for filtering spam documents. For example, the rule (MAXOC_8, “savings”, “offer”, “best”) returns a match if `savings` occurs in the document six times. There is also a match if `offer` occurs in the document six times and `best` occurs twice.

MAXPAR_*n*

(Maximum paragraph) Takes a value for *n* and one or more arguments. Matches if all arguments occur within the first *n* (or fewer) paragraphs of the document, in any order. For example, the rule (MAXPAR_4, “seasonal", “herbs”, “native”) returns a match if **`seasonal`** occurs in paragraph 4, **`herbs`** occurs in paragraph 2, and **`native`** occurs in paragraph 2.

Note: MAXPAR rules work properly only when applied to data sets that contain paragraph delimiters (\n\n). MAXPAR cannot be applied on the **Test Rules** tab. MAXPAR also cannot be applied in the **Categories** tab to data that is contained in folders.

MAXSENT_*n*

(Maximum sentence) Takes a value for *n* and one or more arguments. Matches if all arguments occur within the first *n* sentences of the document, in any order. For example, the rule (MAXSENT_4, “weight loss”, “plan”) returns a match if **`weight loss`** and **`plan`** occur in sentence 3 of the document. For a list of sentence delimiters, see the SENT operator.

NOT

Takes one argument. Matches if the argument does not occur in the document. Must be used with the AND operator. For example, the rule (AND, (OR, “cinema”, “theater”, “theatre”), (NOT, “Broadway”)) returns a match if **`cinema`**, **`theater`**, or **`theatre`** occur in the document and **`Broadway`** does not.

Note: The NOT operator applies across the entire document. If you specify the OR operator in addition to the AND operator, you must enclose the OR arguments in parentheses.

NOTIN

(Not in) Takes two arguments and matches if the first argument does not appear within the second argument. For example, the rule (NOTIN, “butter”, “peanut butter”) identifies **`butter`** when it does not appear within the noun phrase **`peanut butter`**. This sentence returns a match: **`Early American colonists churned their own butter.`**

NOTINDIST_*n*

(Not in distance) Takes a value for *n* and two arguments. Matches if the arguments do not occur within *n* words of each other, or if the first argument listed in the rule occurs in the document and the second argument does not. For example, the rule (NOTINDIST_3 “orange”, “green”) returns a match if `orange` and `green` do not occur within three words of each other, or if only `orange` appears in the document. The following sentence returns a match because the words that are specified in the rule are more than three words apart: `How green is my valley, how orange is the sunset?`

Note: For calculation purposes, the distance between words is not inclusive. For example, the distance between the words `best` and `show` in the phrase `best in show` is two words. Words that include hyphens are counted as one word (for example, `merry-go-round` is one word).

NOTINPAR

(Not in paragraph) Takes two or more arguments and matches if all arguments occur within the document but appear in separate paragraphs. For example, the rule (NOTINPAR, “China”, “export”) returns a match if `China` and `export` occur in separate paragraphs (without the other argument present).

Note: NOTINPAR rules work properly only when applied to data sets that contain paragraph delimiters (\n\n). NOTINPAR cannot be applied on the **Test Rules** tab. NOTINPAR also cannot be applied in the **Categories** tab to data that is contained in folders.

NOTINSENT

(Not in sentence) Takes two or more arguments and matches only if all arguments occur within the document but appear in separate sentences. For example, the rule (NOTINSENT, “China”, “trade”) returns a match if `China` and `trade` occur in separate sentences (without the other argument present), as in `China is our biggest partner. The trade it generates is huge.` For a list of sentence delimiters, see the SENT operator.

OR

Takes one or more arguments. Matches if at least one argument occurs in the document. For example, the rule (OR, "U.S.", "US ", "United States") returns a match if one or more of the items `U.S.`, `US`, or `United States` appear in the document.

Note: Rules that are generated by SAS Contextual Analysis nest the OR operator within the AND operator. However, the OR operator can stand alone.

ORD

(Order) Takes one or more arguments. Matches if all of the arguments occur in the order that is specified in the rule. It cannot be used with SENT (or any other operator that limits the scope of matches). For example, the rule (ORD, “warranty”, “claim”, “denied”) returns a match in the sentence **`The warranty claim for the washing machine was denied.`**

ORDDIST_*n*

(Order and distance) Takes a value for *n* and two or more arguments. Matches if both arguments occur in the same order that is specified in the rule and if both arguments are within *n* words of each other. For example, the rule (ORDDIST_5, “elementary”, “statistics”) returns a match in the phrase **`the teacher introduced elementary statistics`**.

Note: For calculation purposes, the distance between words is not inclusive. For example, the distance between the words **`best`** and **`show`** in the phrase **`best in show`** is two words. Words that include hyphens are counted as one word (for example, **`merry-go-round`** is one word).

PAR

(Paragraph) Takes one or more arguments. Matches if all the arguments occur in a single paragraph, in any order. For example, the rule (PAR, “director”, “budget”) returns a match if the paragraph includes both **`director`** and **`budget`**.

Note: PAR rules work properly only when applied to data sets that contain paragraph delimiters (\n\n). PAR cannot be applied on the **Test Rules** tab. PAR also cannot be applied in the **Categories** tab to data that is contained in folders.

PARPOS_*n*

(Paragraph position) Takes a value for *n* and one or more arguments. Matches if all arguments occur within the n^{th} paragraph, in any order. For example, the rule (PARAPOS_2, “journalists”, “detained”, “overseas”) returns a match if **`journalists`**, **`detained`**, and **`overseas`** occur within paragraph 2 of the document.

Note: PARPOS rules work properly only when applied to data sets that contain paragraph delimiters (\n\n). PARPOS cannot be applied on the **Test Rules** tab. PARPOS also cannot be applied in the **Categories** tab to data that is contained in folders.

<table>
<tr><td>SENT</td><td>(Sentence) Takes two or more arguments. Matches if all the arguments occur in the same sentence, in any order. For example, the rule (SENT, “growth”, “hormone”) returns a match in the sentence Patients who take a growth hormone might experience side effects. Sentence delimiters are as follows:
<table>
<tr><td>\r\n\r\n</td><td>Two consecutive carriage returns and new lines (for documents created in Windows)</td></tr>
<tr><td>\r\n \r\n</td><td>Two consecutive carriage returns and new lines, separated by a space</td></tr>
<tr><td>.<SPACE></td><td>Period (.) followed by an ASCII space</td></tr>
<tr><td>.\n</td><td>Period (.) followed by a new line</td></tr>
<tr><td>.\r</td><td>Period (.) followed by a carriage return</td></tr>
<tr><td>!</td><td>Exclamation point</td></tr>
<tr><td>!\n</td><td>Exclamation point followed by a new line</td></tr>
<tr><td>!\r</td><td>Exclamation point followed by a carriage return</td></tr>
<tr><td>?</td><td>Question mark</td></tr>
<tr><td>?\n</td><td>Question mark followed by a newline</td></tr>
<tr><td>?\r</td><td>Question mark followed by a carriage return</td></tr>
<tr><td>.)</td><td>Period followed by a closing parenthesis</td></tr>
<tr><td>!)</td><td>Exclamation point followed by a closing parenthesis</td></tr>
<tr><td>?)</td><td>Question mark followed by a closing parenthesis</td></tr>
<tr><td>.”</td><td>Period followed by double quotation marks</td></tr>
</table>
</td></tr>
<tr><td>START_<i>n</i></td><td>(From the start of the document) Takes a value for <i>n</i> and one or more arguments. Matches if the argument occurs within <i>n</i> words from the start of the document. For example, the rule (START_22, “infection”) returns a match if infection occurs within 22 words of the first word in the document.
<b>Note:</b> Words that include hyphens are counted as one word (for example, merry-go-round is one word).</td></tr>
</table>

Using Symbols in Boolean Rules

You can use the symbols listed in Table 3.10 to modify your Boolean rules for category matching. Symbols are written as suffixes to strings in arguments. For example, to expand the word `breathe` to all inflected verb forms, which include `breathes` and `breathing`, use the following syntax for the argument: `“breathe@V”`.

Table 3.10 Special Symbols Used in Boolean Rules

Symbol	Description
*	(Wildcard matching) Matches any characters that occur at the beginning or end of the word. For example, the argument `“travel*”` returns the matches `travels` , `traveled`, `traveler`, `traveling`, and so on. The argument `“*room”` matches `bedroom`, `cloakroom`, `ballroom`, `room`, and so on.
^	(Beginning of sentence) Starts searching at the beginning of the sentence to find a match. For example, the argument `“^Independent”` returns a match in this sentence: `Independent research was conducted.` **Note:** Tokens (words, phrases, symbols, or other meaningful elements) need to be entered specifically to be considered for matching. For example, if you are searching for `**In this case`, use the argument `“^\*\*In this case”`. Also note that backward slashes (\) are used as escape characters for the asterisks (*) so that the asterisks are not treated as wildcards.
$	(End of sentence) Starts searching at the end of the sentence to find a match. For example, the argument `“deleted.$”` returns a match on the following sentence: `All the files were hastily deleted.` **Note:** Tokens (words, phrases, symbols, or other meaningful elements) need to be entered specifically to be considered for matching. For example, the argument `“deleted$”` would not produce a match on the following sentence: `All the files were hastily deleted.` because the ending period (.) was not specified.
@	(Morphological expansion) Expands the category rule to match all inflectional forms of the word in the argument. For example, the argument `“wonder@”` returns the matches `wonder`, `wonders`, `wondered`, `wondering`, and so on (but does not return a match on `wonderful`). **Note:** If you apply @ to a word that SAS Contextual Analysis does not recognize, no expansion occurs. Only the exact string specified before the @ is returned. For example, `“grath”` would not expand. Only the string `grath` would return a match in the rule.

@A	(Morphological expansion for adjectives) Expands the category rule to match inflected comparative and superlative adjective forms of the word in the argument. For example, the argument **"happy@A"** returns the matches **happier** and **happiest**. **Note:** If you apply @A to a word that is not an adjective, no expansion occurs.
@N	(Morphological expansion for nouns) Expands the category rule to match all noun forms of the word in the argument. For example, the argument **"quality@N"** returns the matches **quality** and **qualities**. **Note:** If you apply @N to a word that is not a noun, no expansion occurs.
@V	(Morphological expansion for verbs) Expands the category rule to match all verb forms of the word in the argument. For example, the argument **"transfer@V"** returns the matches **transfer** , **transfers**, **transferred**, and **transferring**. **Note:** If you apply @V to a word that is not a verb, no expansion occurs.
_L	(Literal matching) Matches a literal string. Useful when you want to match a string that includes symbols. For example, the argument **"$USD_L"** returns the match **$USD**. **Note:** Tokens (words, phrases, symbols, or other meaningful elements) need to be entered specifically to be considered for matching.
_C	(Case matching) Specifies case-sensitive matching. For example, the argument **"Iris_C"** returns the match **Iris**, but not **iris**.

Using _tmac for Referencing Categories

Referencing a category enables you to use the rules in an existing category without having to duplicate the rules. Use tmac syntax (_tmac) to reference an existing category in a category rule. The definition of the existing rule is processed in the category that references it.

To reference a category, you must identify its path. All category paths begin with **@Top/**. From there, you can specify the path by following the category hierarchy.

For example, suppose you have the following category structure under **All Categories**:

NAME
    FIRST

    LAST

You would reference the category **FIRST** as **@Top/NAME/FIRST**.

You can use the tmac syntax with Boolean operators. For example, suppose you want to reference the category **FIRST** from a category called **FIRST_NAME**. You could add this rule in the **FIRST_NAME** definition:

```
(OR,_tmac:"@Top/NAME/FIRST")
```

To enforce a first name followed by last name (FIRST LAST), you could add this rule in a category called **COMPLETE_NAME**::

```
(ORD,_tmac:"@Top/NAME/FIRST",_tmac:"@Top/NAME/LAST")
```

The definitions written in **FIRST** and **LAST** are automatically processed.

Appendix 1

Part-of-Speech Tags (for Languages Other Than English)

Introduction to Part-of-Speech and Other Tags

The part-of-speech tags for languages other than English are listed in the following tables. Also included are other tags that are not considered parts of speech (such as punctuation). All tags are case-sensitive and are preceded by a colon (:) in concept rules. For more information, including English tags, see "Using Part-of-Speech and Other Tags" on page 68.

Part-of-Speech Tags

Chinese

Table A1.1 *Part-of-Speech Tags for Chinese*

Part-of-Speech Tag	Description	Examples
:A	Adjective	俊俏, 开心, 兇險, 凌亂
:ASCII	ASCII characters	sas, do, happy, day2136456
:C	Conjunction	或, 与, 雖然
:D	Adverb	非常, 偏偏, 稍微, 永遠
:digit	Number	1051, 1.9
:E	Interjection	咦, 呸, 哦喲
:F	Location / direction	中間, 下边, 南侧
:G	Other morpheme	馨, 慚
:H	Other prefix	亚, 非
:K	Other suffix	们, 者, 們
:L	Idiom (chengyu)	囫囵吞枣, 博古通今, 一廂情願
:M	Quantifier	十, 卅, 成千上万, 上萬, １０５１
:N	Noun	人, 桌子, 香蕉, 枷鎖

:NR	Proper noun, name	习近平, 梁振英, 奥巴马
:NS	Proper noun, geographic	中国, 美國, 山東
:NT	Proper noun, organization	北京大学, 上汽集團
:NZ	Proper noun, miscellaneous	潘婷, 劍南春
:O	Onomatopoeia	吱呀, 叽叽喳喳, 劈裏啪啦
:P	Preposition	依照, 对于
:Q	Classifier	个, 斤, 艘, 加侖
:R	Pronoun	我, 他們, 这
:S	Subcountry location (general; specifics only within sinosphere)	地上, 上空, 高处, 內廳
:T	Temporal phrase	今天, 夜间, 十月, 去歲
:U	Particle	的, 了, 着
:UNKNOWN	Unknown word	婳, 繟
:V	Verb	看, 认为, 彈奏, 徵納
:W	Punctuation or symbols	！, 。, $, ￥
:Y	Interjectional particle	吧, 吗, 麼

Dutch

Table A1.2 *Part-of-Speech Tags for Dutch*

Part-of-Speech Tag	Definition	Examples
:A	Adjective	betrouwbaar, gelukkig, mooi

:ABB	Abbreviation	enz, kg, zgn
:ADV	Adverb	eenmaal, hier, nu
:CONJ	Conjunction	als, doch, hoe
:DET	Determiner	de, der, een
:digit	Number	21
:DNUM	Determiner, number	acht, elf, miljard, duizend
:inc	Unknown word	xrxx
:N	Noun	geluk , schoonheid
:PFX	Prefix	anti
:PN	Proper noun	Amerika, Nederland
:PREP	Preposition	met, per, te, van
:PREPDET	Preposition and determiner contraction	ten, ter
:PRO	Pronoun	alles, beide, hetgeen
:sep	Separator or punctuation	,
:url	URL	www.sas.com
:V	Verb	helpt, vernieuwt
:VB	Infinitive	helpen, vernieuwen
:VE	Present progressive	helpende, vernieuwende
:VH	Past participle	geholpen, vernieuwd
:XI	Archaic form	hoofde, tijde, voordele

Finnish

Table A1.3 Part-of-Speech Tags for Finnish

Part-of-Speech Tag	Definition	Examples
:A	Adjective	loistava, korkea
:ADV	Adverb	ohitse, juuri
:CLX	Clitic	kinko, pas
:CONJC	Coordinating conjunction	ja, vaan
:CONJS	Subordinating conjunction	ellei, jotta
:date	Date	12-14, 2001-12-02
:digit	Number	1234, 7
:inc	Unknown word	auttonkkan, eggs
:N	Noun	siltoineen, postiksi
:PN	Proper noun	Pertti, Fazer
:PREP	Preposition	pitkin, kanssaan
:PRO	Pronoun	noihin, muussa, ketkä
:sep	Separator or punctuation	; / +
:time	Time	12:00:00, 7PM
:PROP	Personal pronoun	sinun, heissä, me
:url	URL	http://www.sas.com
:VB	Infinitive verb	heilahtamassa, heilauttaen, olla

:VC	Potential present verb	lähennemme, luvannette
:VE	Present participle verb	kumarrettava, ilmaisevaa
:VH	Past participle verb	jaettu, ilmaistu
:VJ	Indicative preterit verb	meditoitpa, matkattu
:VP	Indicative present verb	ihastele, hörähdä
:VS	Conditional present verb	omistautuisi, hehkuisikaan
:VY	Imperative verb	parannuttako, pakkaa

French

Table A1.4 *Part-of-Speech Tags for French*

Part-of-Speech Tag	Definition	Examples
:A	Adjective	comparable, compassionnelle, intraduisibles
:ADV	Adverb	plutôt, individuellement
:CONJC	Coordinating conjunction	et, ou
:CONJS	Subordinating conjunction	lorsque, puisque
:DET	Determiner	sa, tes
:digit	Number	123, 12.3, 12.3.2003, 12/3/2003
:inc	Unknown word	analytics
:INTJ	Interjection	tralala, zzz

:N	Noun	zèbre, encyclopédie
:PN	Proper noun	Eurotunnel, Égypte
:PFX	Prefix	anglo, éco
:PREP	Preposition	après, jusque
:PREPDET	Preposition and determiner contraction	aux, du
:PRO	Pronoun	lui, ce
:sep	Separator or punctuation	, . !
:url	URL	http://www.sas.com
:V	Verb	vais, obligez
:Vpp	Past participle	mangé, relaxée, travaillées
:VB	Infinitive	traduire, rompre
:VE	Present participle	ceignant, tramant
:XI	Foreign words	vitae, ab

German

Table A1.5 *Part-of-Speech Tags for German*

Part-of-Speech Tag	Definition	Examples
:A	Adjective	schön, zuverlässig
:ADV	Adverb	gern, sehr
:CONJ	Conjunction	und, oder

:CPO	Compounding (prefix only)	Lustigkeits
:DET	Determiner	der, eine
:digit	Number	21
:DNUM	Determiner, number	fünf, zwölf
:EMP	Emphatic/intensifier	ganz
:inc	Unknown word	xrxx
:N	Noun	Schönheit, Zuverlässigkeit
:PFX	Prefix	Irr, lob
:PN	Proper noun	Mozart
:PN.gen	Proper noun, genitive	Nirvanas
:PNG.dat	Proper noun, geographic, dative	Niederlanden
:PREDET	Predeterminer	manch
:PREP	Preposition	kontra, ober
:PRO	Pronoun	er, sie
:PXPRO	Pronominal adverb	heraus
:sep	Separator or punctuation	,
:url	URL	www.sas.com
:V	Verb	ging, half
:VI	Infinite (infinitives and participles)	gehen, helfen

Italian

Table A1.6 Part-of-Speech Tags for Italian

Part-of-Speech Tag	Definition	Examples
:A	Adjective	affidabile, bellissimo, felice
:AVV	Adverb	felicemente, rapidamente
:CONG	Conjunction	ma, oppure, sebbene
:DET	Determiner	il, la, uno
:digit	Number	21
:ESC	Interjection	ah, ahimè
:inc	Unknown word	Xrxx
:N	Noun	affidabilità, bellezza, felicità
:PN	Proper noun	Roma, Italia
:PRON	Pronoun	io, ne, tu
:PREFIX	Prefix	anti, ri
:PREP	Preposition	con, in, per
:sep	Separator or punctuation	,
:SUFFIX	Suffix	anza, issimo
:url	URL	www.sas.com
:V	Verb	andare, vedono
:VGerund	Gerund	andando, vedendo

:VH	Past historic	andasse, vedessero
:Vpastpart	Past participle	andato, visto

Japanese

Table A1.7 *Part-of-Speech Tags for Japanese*

Part-of-Speech Tag	Description	Examples
:AJ	Adjective	長い, いい, 忙しい,便利だ
:AV	Adverb	別に, 相変わらず, 年年歳歳
:AVC	Adverbs of form or condition (adverbs that show forms or condition of the predicates)	正々堂々, 淡々と, きらり
:AVD	Adverbs of degree	結構, とっても
:AVE	Adverbs of evaluation	たまたま, 幸い, 無論
:AVF	Adverbs of frequency	次々と, 次次に
:AVO	Adverbs of opinion	実は, 即ち, すなわち
:AVQ	Adverbs of quantity	大方, いくら, 半分
:AVS	Adverbs of statement or declaration	何でも, 多分, 絶対
:AVT	Adverbs of tense or aspect	徐々に, 急遽, 直ぐ
:AX	Auxiliary verbs	らしい,みたいだ, 様だ, わけだ
:CN	Conjunctions	並びに, でも, 但し, だけど
:CP	Copula	だ, なんだ

:DA	Demonstratives, adverbials	こう, そう, あのように, この様に
:DM	Demonstratives, prenominal modifiers (prenominal forms of DN or other pronouns)	この, あの, そのような, そんな
:DN	Demonstratives, nominals (similar to *this*, *that*, and other pronouns in English)	あれ, こちら, あそこ
:MD	Prenominal modifiers	明くる, 小さな, 主たる, 色んな
:IT	Interjection words	あれれ, あれー, あ～, ええ, ええと
:NA	Adverbial noun	所所, 前, 間, 後, 挙句
:NC	Common noun	別室, 紅花, ベネズエラ社会, 便所
:NT	Noun of time	永年, 長年, 夏, 先月
:NK	Content nouns—act as a relative pronoun of nonhead relative clauses (similar to *that* clause or *what* clause in English)	ぐらい, の, もの, こと
:NN	Numeral	千, 十, ○, 零, レイ, 六, 6
:NP	Proper noun	ＷＴＯ繊維協定, 米州
:NPO	Proper noun of organization	米軍, 米国, 米国際貿易委員会
:NL	Noun of place	米国, 越南, 奈井江町, 奈央島
:NH	Noun of person	中川秀直, 中川浩明, 中川勝
:NHM	Subclass of NH	城三郎, 譲三郎, 菜衣子
:NHS	Subclass of NH	名内, 苗川, 苗田, 苗場

:NV	Verbal noun	米国開催, 米国進出, 勉学
:VSN	Verbs provided with a part-of-speech **Note:** When kango-verbs appear alone, they are categorized as a type of noun. When they appear with the verbal conjugating morpheme *suru*, kango-verbs are categorized as VSN	くるくる, くよくよ, 伸びのび
:PC	Particles of case marker	を, で, の, へ, から
:PE	Particles that appear at the end of the sentence and describe the mood of the speaker	っけ, な, ナ, なぁ
:PN	Particles that combine nominals (similar to *and* in English)	ないし, ないしは, 並びに
:PP	Particles that combine clauses	ながら, なら, なり, のに, きり
:PQ	Particles of quotation	って, っと, て, と
:PS	Particles that mean *only* or *too* in English	ったら, って, 等, など, なら
:PRJ1	Prefixes to adjectives (that end with い)	か, こ, 真
:PRJ2	Prefixes to adjectives (that end with な)	無, 不, 非
:PRN	Prefixes to nominals	高, 前, 全
:PRV	Prefixes to predicates	相, 猛, 最
:SC	Special-category comma	、 ， ，

:SCP	Special-category closed parentheses	', ", > ,), 》
:SCP	Special-category period	。 , .
:SS	Special-category space	
:SJN	Suffixes to nouns and configure adjectival nouns	っぽい, くさい
:SJV	Suffixes to verbs and configure adjectives	ない, たい, づらい
:SNA	Suffixes to adjectives	さ
:SNC	Suffixes to classifiers or counters	頁, ぺーじ, 杯, 版
:SNN	Suffixes to nouns	っ子, 下手, 内, 等, など, 制, 性, 生, 製, 席, 説, 線, 船
:SNV	Suffixes to nominal predicates	っきり, っぱなし, っ放し
:SV	Suffixes to verbs	る,ある,得る
:V1	Ichidan dan verbs (ending with *-eru* or *-iru*)	飛び始める, べんじる, 便じる, 直せる, 流れ落ちる
:V5	Godan verb	並ぶ,せめぐ, 泣付く,流し込む,往ぬ
:VK	Kuru verb	やってくる, くる
:VS1	Suru verb	する
:VS2	Suru verb	辞する, じする, 無くする
:VZ	Zuru verb	べんずる, 便ずる, べんずる

Korean

Table A1.8 *Part-of-Speech Tags for Korean*

Part-of-Speech Tag	Description	Examples
:AD	Adverb	매우, 정말, 빨리
:AJ	Adjective	예쁘다, 귀엽다, 차분하다
:ASCII	Foreign	Korean, iPhone, SK
:DATE	Date	2015-04-28, 20150428
:DEFAULT	Unknown word	하페즈, 샤리프, 쿠레쉬
:GAC	Case grammatical affix	가, 를, 로
:GAD	Determinative grammatical affix	은, 을, 는
:GAH	Change grammatical affix	이다, 기, 음
:GAJ	Conjunctive grammatical affix	는데, 는지, 느라고
:GAP	Predicate grammatical affix	다, 습니다, 더구만
:GAR	Respect grammatical affix	시, 으시, 옵
:GAT	Time grammatical affix	겠, 었, 였었
:GAX	Auxiliary grammatical affix	도, 만, 까지
:IJ	Interjection	아, 네, 그래
:NN	Noun	하늘, 산, 바다
:NNB	Bound noun	것, 수, 개
:NNP	Proper noun	서울, 이순신, 국립국어원

:NUMBER	Numeral	하나, 둘, 셋
:PF	Prefix	제-, 햇-, 명-
:PN	Prenoun	각, 첫,기초적
:PR	Pronoun	이것, 언제, 이분
:PUNC	Punctuation	. ? ! ( )
:SF	Suffix	-꾼, 꾸러기, -감
:TIME	Time	23:59:59
:URL	URL	www.sas.com
:VB	Verb	웃다, 뛰다, 날다

Portuguese

Table A1.9 *Part-of-Speech Tags for Portuguese*

Part-of-Speech Tag	Definition	Examples
:A	Adjective	confiável, belo, feliz
:ADV	Adverb	belamente, felizmente
:CONJ	Conjunction	e, que
:DET	Determiner	alguns, cada, os
:digit	Number	21
:DNUM	Numeric determiner	bilionésimo, cinco
:inc	Unknown word	xrxx
:INTJ	Interjection	caramba, eh

:N	Noun	beleza, felicidade
:PFX	Prefix	anti, circum
:PN	Proper noun	Brasil, Portugal
:PREP	Preposition	com, de, em
:PREPDET	Preposition and determiner contraction	dessas, dum
:PRO	Pronoun	me, nós, quem
:sep	Separator or punctuation	,
:url	URL	www.sas.com
:V	Verb	agradecem, garanto
:VB	Infinitive	agradecer, garantir
:VG	Gerund	agradecendo, garantindo
:VH	Past historic	agradecido, garantido
:XL	Foreign word	cf, ibid, sic

Russian

Table A1.10 *Part-of-Speech Tags for Russian*

Part-of-Speech Tag	Definition	Examples
:A	Adjective	духовитый, красивая, лучших
:ABBREV	Abbreviation	др, км
:adv	Comparative adverb	дальше

:adverbial	Adverb	хорошо, сколько-нибудь
:conj	Conjunction	если, и
:digit	Number	123, 12.3, 12.3.2003, 12/3/2013
:inc	Unknown word	геминг, analytics
:idet	Interrogative determiner	который
:INT	Interjection	ах
:intadv	Interrogative adverb	где
:intquant	Interrogative quantifier	сколькие, почём
:N	Noun	велосипед, история, малолетство
:NONDECL	Nondeclinable word	мартини, маэстро
:NONDECL-ADJ	Nondeclinable adjective	баскервиллей
:NONDECL-PN	Nondeclinable proper noun	Шевроле, Айдахо
:NONDECL-PRO	Nondeclinable pronoun	всяко
:num	Number	один, десятью
:particle	Particle	бы, же
:PN	Proper noun	Миа, Тузла
:PNA	Proper adjective	Роханский, Сашина
:PNN	Proper noun, name	Свердловск, Мария, Давыдович
:prep	Preposition	до, вроде
:pron	Pronoun	я, её

:sep	Separator or punctuation	, . !
:url	URL	http://www.sas.com
:VB	Infinitive	автоматизировать, менять, кончить
:V	Verb	нажимает, кладите, плавала
:VG	Gerund	адаптировав, вальсируя

Spanish

Table A1.11 Part-of-Speech Tags for Spanish

Part-of-Speech Tag	Definition	Examples
:A	Adjective	confiable, feliz, hermoso
:ABBREV	Abbreviation	km, pág, Sra
:Adv	Adverb	ahora, felizmente
:CONJ	Conjunction	ni, pero, y
:DET	Determiner	el, las, mi, nuestro
:digit	Number	21
:inc	Unknown word	xrxx
:INTJ	Interjection	hola
:N	Noun	belleza, felicidad
:PN	Proper noun	Chile, España
:PREP	Preposition	con, de, en, por

:PREPDET	Preposition and determiner contraction	al, del
:PRON	Pronoun	alguien, ellos, me
:sep	Separator or punctuation	,
:url	URL	www.sas.com
:V	Verb	ayudan, pide
:VB	Infinitive	ayudar, pedir
:VE	Present progressive	ayudando, pidiendo
:VH	Past participle	ayudado, pedido

Turkish

Table A1.12 *Part-of-Speech Tags for Turkish*

Part-of-Speech Tag	Description	Examples
:A	Adjective	iyi, zor
:ADV	Adverb	yine, zaten
:CONJ	Conjunction	veya, hem
:date	Date	12/30/2000, 12/30/00, 2000-30-12
:digit	Number	12.302.000, 5
:inc	Unknown word	wug
:N	Noun	kitap, insan

:NUMERAL	Numeral	dokuz, onbir
:PARTICLE	Particle	beri, diye
:PN	Proper noun	Ayşe, Türkçe
:PRONOUN	Impersonal pronoun	bunlar, kendi
:PROP	Personal pronoun	onlar, sen
:QUANT	Quantifier	çok, her
:sep	Separator or punctuation	! . ,
:time	Time	12:30:00
:url	URL	sas.com, www.sas.com, http://sas.com
:V	Verb (can be followed by any number and combination of any of the following):	
	A (habitual/nomic)	bilir
	B (infinitive)	bilmek
	C (conditional)	bilse
	F (future)	bilecek
	I (indefinite/inferential)	bilmiş
	J (past)	bildi
	N (necessitative)	bilmeli
	P (continuous, including archaic forms)	biliyor, bilmekte
	O (subjunctive)	bile
	Y (imperative)	bil
:V_GER_STEM	Untagged verb	Not applicable

Recommended Reading

Here is the recommended reading list for this title:

- *SAS Contextual Analysis: Administrator's Guide*
- *SAS Content Categorization Single User Servers: Administrator's Guide*
- *SAS Text Miner: Reference Help*
- *SAS Encoding: Understanding the Details*
- *SAS Enterprise Content Categorization Studio: User's Guide*
- *Text Mining and Analysis: Practical Methods, Examples, and Case Studies Using SAS*

For a complete list of SAS publications, go to sas.com/store/books. If you have questions about which titles you need, please contact a SAS Representative:

SAS Books
SAS Campus Drive
Cary, NC 27513-2414
Phone: 1-800-727-0025
Fax: 1-919-677-4444
Email: sasbook@sas.com
Web address: sas.com/store/books

Glossary

category

a classification for documents that is based on a common characteristic. Category membership is indicated as a binary property. In order to determine when a document is likely to be a member of a category, one or more Boolean rules comprising the category text definition must be satisfied.

concept

an abstract class of meanings. In order to determine when a concept is likely to be referenced in a subset of text, the rules comprising the concept text definition must be satisfied.

model scoring

the process of applying a model to new data in order to compute outputs.

parse

to analyze text, such as a SAS statement, for the purpose of separating it into its constituent words, phrases, punctuation marks, values, or other types of information. The information can then be analyzed according to a definition or set of rules.

relevancy score

a score that indicates how well a document satisfies a rule or model. The best match has a score of 1 and reflects a perfect (100%) match.

scoring

See model scoring.

sentiment

an attitude that is expressed about an item that is being analyzed, which can be a segment of text, a grouping of text segments, or a specific subject of interest.

sentiment analysis

the use of natural language processing, computational linguistics, and text analytics to determine the attitude of a speaker or writer with respect to a topic, document, or other item of analysis. Sentiment analysis results in a positive, negative, or neutral score on the target of analysis.

stemming

the process of finding and returning the root form of a word. For example, the root form of grind, grinds, grinding, and ground is grind.

stop list

a SAS data set that contains a simple collection of low-information or extraneous words that you want to remove from text mining analysis.

string

See text string.

subset of text

the matched text for a concept text definition; this consists of one or more strings that are contained in a document.

surface form

a variant of a term that is contained in a matched subset of text in one or more documents. These forms include stems, synonyms, misspellings, and alternate ways of referring to the same entity.

taxonomy

a hierarchical relationship of parent and child category nodes. In a true taxonomy, whenever a category is detected, it is implied that all parents are also represented. For example, if something is identified as human, it must also be a primate, mammal, animal, and so on.

term

a representation of a single concept in one or more textual forms, as defined by rules or algorithms.

term map

a node-arc graph that centers around an "object of interest," which could be a category, concept, topic, or term. Corresponding nodes in the graph indicate rules that are predictive of the object of interest. Better rules are shown as larger nodes. The arcs represent the addition or exclusion of terms that are used to build up the rules.

term role

a function that is performed by a term in a particular context. A term can function as a part of speech, entity type, or other purpose that is user-defined.

term table

a list of every term in a collection of documents including the representative text form for each term, its role, and all of its surface forms that appear within that collection.

text string

a subset of text that consists of adjacent characters of any type. Depending on the specified options, strings can be either case-sensitive or case-insensitive.

token

in the SAS programming language, a collection of characters that communicates a meaning to SAS and that cannot be divided into smaller functional units. A token such as a variable name might look like an English word, but can also be a mathematical operator, or even an individual character such as a semicolon. A token can contain a maximum of 32,767 characters.

topic

a machine-generated category, the purpose of which is to indicate what documents are about. A topic identifies groupings of important terms in a document collection. A single document can contain one or more topics, or no topics.

topic document weight

See topic-specific document weight

topic term weight

See topic-specific term weight

topic-specific document weight

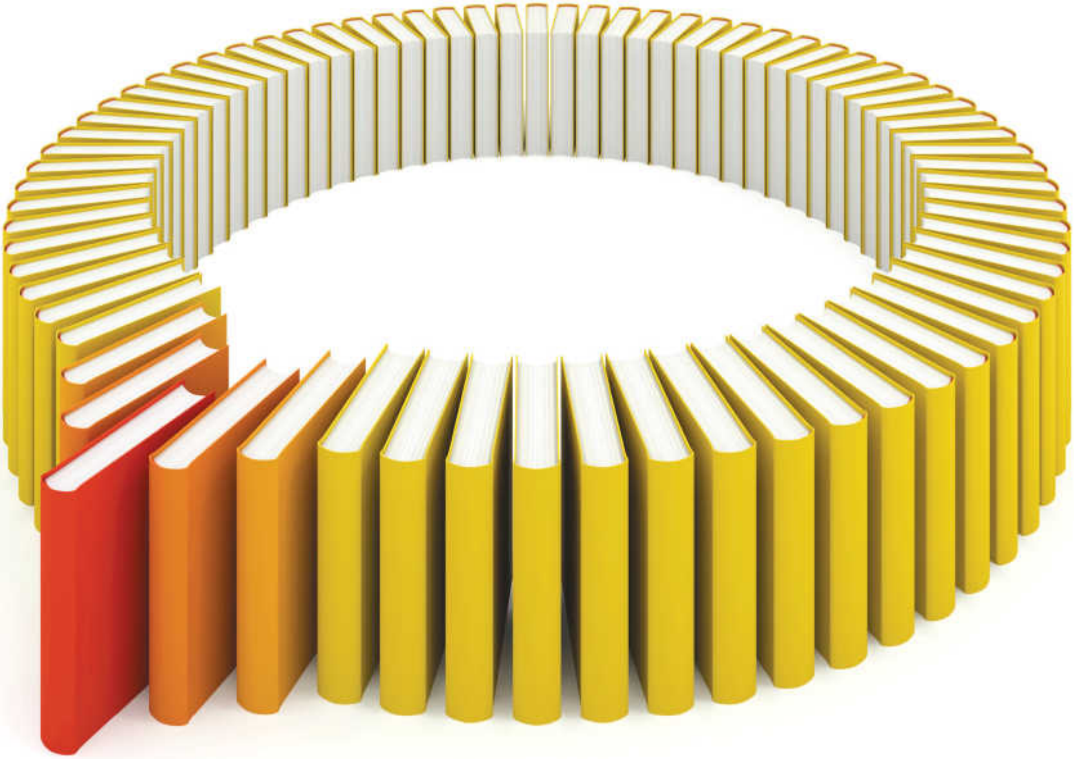
an indicator of the importance of a topic to a document. A value that is above a specified cutoff value indicates that a document contains that topic.

topic-specific term weight

an indicator of the relative importance of a term in a topic as compared to other terms. A term with a value above a specified cutoff value contributes to the assignment of a document to the topic.

weight

a numeric indicator that is assigned to an item and that indicates the relative importance of the item in a frequency distribution or population.



Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

support.sas.com/bookstore
for additional books and resources.

sas.
THE POWER TO KNOW.