



SAS[®] Visual Text Analytics 8.2: User's Guide

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2017. *SAS® Visual Text Analytics 8.2: User's Guide*. Cary, NC: SAS Institute Inc.

SAS® Visual Text Analytics 8.2: User's Guide

Copyright © 2017, 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, NC 27513-2414

June 2018

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.

8.2-P1:ctxtug

Contents

| | |
|---|------------|
| <i>About this Book</i> | v |
| <i>Accessibility</i> | vii |
| Chapter 1 • Introduction to SAS Visual Text Analytics on Viya | 1 |
| What Is SAS Visual Text Analytics on Viya? | 1 |
| How Does SAS Visual Text Analytics Work? | 2 |
| Supported Languages | 4 |
| Visual Text Analytics Basics | 4 |
| Chapter 2 • Working in Projects | 11 |
| Getting Started | 11 |
| Using SAS Sentiment Analysis Models in SAS Visual Text Analytics | 14 |
| Project Sharing | 15 |
| Chapter 3 • Working In Pipelines | 17 |
| About Working with Pipelines | 17 |
| Using the Default Text Analytics Pipeline | 17 |
| Adding Text Analytics Nodes to the Pipeline | 18 |
| Setting Options for the Analysis Nodes | 19 |
| Scoring an External Data Set | 24 |
| Chapter 4 • Using the Interactive Windows for the Nodes | 27 |
| The Interactive Window for the Concepts Node | 27 |
| The Interactive Window for the Text Parsing Node | 30 |
| The Interactive Window for the Topics Node | 34 |
| The Interactive Window for the Categories Node | 36 |
| Chapter 5 • Writing Rules | 43 |
| Writing Concept Rules: Basic LITI Syntax | 43 |
| Writing Category Rules | 62 |
| Appendix 1 • Part-of-Speech Tags (for Languages Other Than English) | 71 |
| Introduction to Part-of-Speech and Other Tags | 71 |
| Part-of-Speech Tags for Rule Writing | 72 |
| Appendix 2 • Pre-Defined Concept Priorities (for Languages Other Than English) | 103 |
| Using Priority Values in Predefined Concepts | 103 |
| Priority Values for Predefined Concepts | 104 |
| Recommended Reading | 119 |
| Glossary | 121 |

About this Book

Audience

This book is designed for users of SAS Visual Text Analytics on Viya. It describes the terminology used in SAS Visual Text Analytics on Viya and provides instructions for tasks. Where appropriate, it guides users to information about Viya.

Accessibility

For information about the accessibility of this product, see [Model Studio: Accessibility Features](#).

Chapter 1

Introduction to SAS Visual Text Analytics on Viya

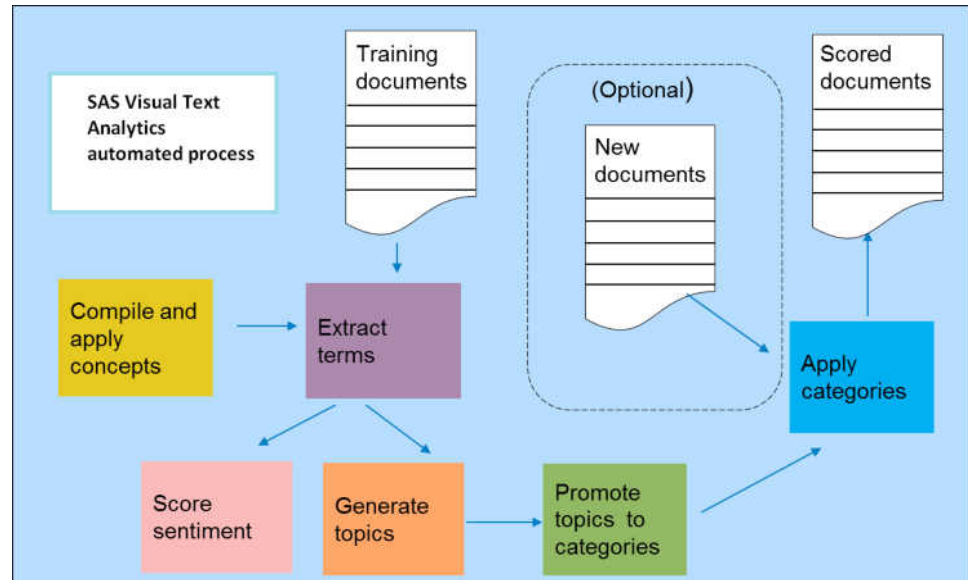
| | |
|---|----------|
| What Is SAS Visual Text Analytics on Viya? | 1 |
| How Does SAS Visual Text Analytics Work? | 2 |
| Supported Languages | 4 |
| Visual Text Analytics Basics | 4 |
| Introduction | 4 |
| Concepts | 5 |
| Text Parsing—Terms and Synonyms | 6 |
| Start Lists and Stop Lists | 7 |
| Topics | 7 |
| Sentiment Scoring | 7 |
| Categories | 8 |
| Using Taxonomies | 8 |

What Is SAS Visual Text Analytics on Viya?

SAS Visual Text Analytics on Viya is a web-based text analytics application that uses context 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.

Figure 1.1 provides an overview of the SAS Visual Text Analytics processes.

Figure 1.1 Process Overview



SAS Visual Text Analytics on Viya combines the visual programming flow of SAS Text Miner with the rules-based linguistic methods of categorization and extraction in SAS Contextual Analysis. These capabilities, along with document-level scoring for each component, are combined in a single user interface.

Using SAS Visual Text Analytics on Viya, 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 informational value (stop words) are excluded from topic 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 Visual Text Analytics on Viya uses a graphical user interface that is useful for all users, regardless of whether they have programming experience.

How Does SAS Visual Text Analytics Work?

SAS Visual Text Analytics provides a number of text analysis pipeline nodes, arranged in a sequence that you control. The pipeline empowers you to analyze your document collection with considerable flexibility.

The *Concepts* analysis node in SAS Visual Text Analytics enables you to extract predefined concepts or create additional custom concepts that you can discover in a document or set of documents. For more information about concepts, see [“Concepts” on page 5](#).

The *Text Parsing* analysis node finds all the terms that are in your document collection. This is also true for concepts, if defined in a preceding Concepts node and if the Concepts node precedes the Text Parsing node. In addition, the Text Parsing node displays useful groups of words such as nouns with their modifiers that can be used for topic discovery.

The *Topics* analysis node groups similar documents in a collection into related themes, or *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.

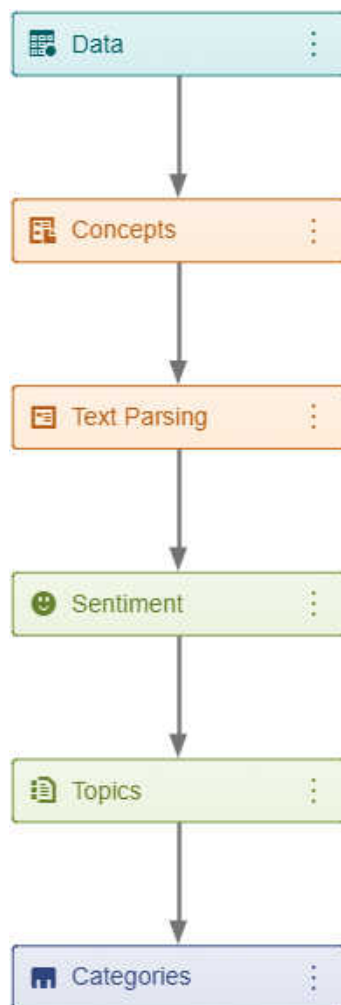
The *Category* analysis node labels documents based on their content. You can create *categories* using these methods:

- specify category (target) variables in your training documents
- create new categories that correspond to your organization's interests
- promote discovered 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 using simple Boolean and proximity operators.

The *Sentiment* analysis node determines whether documents express positive, neutral, or negative attitudes. Analysis performed after the Sentiment Analysis node will display a sentiment indicator for each document.

Finally, each of the analysis nodes (except parsing) provide score code that enables you to deploy your models. Use deployed models to automate the process of labeling a set of input documents into their respective concepts, categories, topics, and sentiment.



Supported Languages

Table 1.1 shows a full list of project languages that are supported. See your SAS sales representative for information about licensing additional languages.

Table 1.1 SAS Visual Text Analytics 8.2 Supported Languages

| | |
|-----------------------|-----------------------|
| Arabic | Chinese (Simp./Trad.) |
| Croatian | Czech |
| Danish | Dutch |
| English | Farsi |
| Finnish | French |
| German | Greek |
| Hebrew | Hindi |
| Indonesian | Italian |
| Japanese | Korean |
| Norwegian (Bok./Nyn.) | Polish |
| Portuguese | Russian |
| Slovak | Slovene |
| Spanish | Swedish |
| Tagalog | Thai |
| Turkish | Vietnamese |

Visual Text Analytics Basics

Introduction

When you run a pipeline, the following analyses are performed in their respective nodes (if data are present):

- Concepts node — concept extraction

- Text Parsing node — term identification (including synonyms)
- Topics node — topic discovery
- Sentiment node — sentiment analysis
- Categories node — category analysis

The following sections describe the primary function of each pipeline node.

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 and for extracting useful information. 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 Visual Text Analytics 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 an organization name or a date. You cannot rename predefined concepts, nor can you view or edit their base definitions. You can provide additional rules in the **Edit** to modify or extend their behavior.

For custom concepts, 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). You do this by setting a priority value. When setting priority values, it is helpful to know the preset values of predefined concepts so that you can set a custom concept's priority at a higher value. For more information about setting priorities, see [“Which Rule Type Should I Use?” on page 45](#).

[Table 1.2 on page 5](#) shows a list of the predefined concepts for English that are included with SAS Visual Text Analytics, along with their preset priority values. For predefined concepts and priority values for other languages, see [Appendix 2, “Pre-Defined Concept Priorities \(for Languages Other Than English\),” on page 103](#).

Table 1.2 *Predefined Concepts and Priorities for English*

| Predefined Concept | Description | Priority Value |
|--------------------|---|----------------|
| nlpDate | Any date expression (month, day, year, date) | 18 |
| nlpMeasure | Measurement or measurement expression (for example, 500kg or 2300 sq ft) | 20 |
| nlpMoney | Currency or currency expression | 18 |
| nlpNounGroup | Nouns and close modifiers that identify a single object or item (for example, <i>clinical trial</i>). Noun groups are typically 2- to 3- word combinations (but can be longer) | 15 |

| | | |
|-----------------|--|-----|
| nlpOrganization | Name of a company or government, legal, or service agency (for example, FBI) | 25* |
| nlpPercent | Percentage or percentage expression (for example, 96% or 12 percentage points) | 18 |
| nlpPerson | Person's name, including any associated title | 20 |
| nlpPlace | Name of a city, country, state, geographical place or region, or political place or region | 20 |
| nlpTime | Time or time expression (for example, 6pm or Friday morning) | 18 |

* Highest value for this language

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

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

For more information about writing concept rules, see [“Writing Concept Rules: Basic LITI Syntax” on page 43](#). For information about writing category rules, see [Writing Category Rules on page 62](#).

Text Parsing—Terms and Synonyms

A *parent term* is defined as a label for one or more tokens that represent a grouping of variants (one or more surface forms) that are related, as defined by underlying rules or algorithms. In SAS Visual Text Analytics, 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 that term's part of speech. A *surface form* is a variant of a parent 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 parent term. SAS Visual Text Analytics 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 way for users to create custom parent terms or to add terms grouped under a parent term. It is a SAS data set that identifies pairs of words that should be combined as single terms for the purposes of analysis. Synonyms are applied at the parent level; all variants of each parent term are combined together into one group. You can specify a synonym list in the Text Parsing node. 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 (label) 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. Users can define these roles in the Concepts node. In order for the user-defined roles to be available in the Text Parsing node, the Concepts node needs to precede it in the pipeline.

- PARENTROLE, which enables you to specify the role of the PARENT.

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 topic discovery. 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 and many other languages in the Reference Data library.

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 Visual Text Analytics, topics are automatically generated and assigned to documents. A single document can contain more than one topic.

The interactive window for the Topics node displays all the topics that SAS Visual Text Analytics 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.

Sentiment Scoring

Sentiment analysis is the process of identifying the author's tone or attitude (positive, negative, or neutral) expressed in a document. SAS Visual Text Analytics 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 Visual Text Analytics 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.

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 while creating a new project, or creating a new category in the **Categories** node. 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 Contextual Analysis 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 43](#). For information about writing category rules, see [Writing Category Rules on page 62](#).

Using Taxonomies

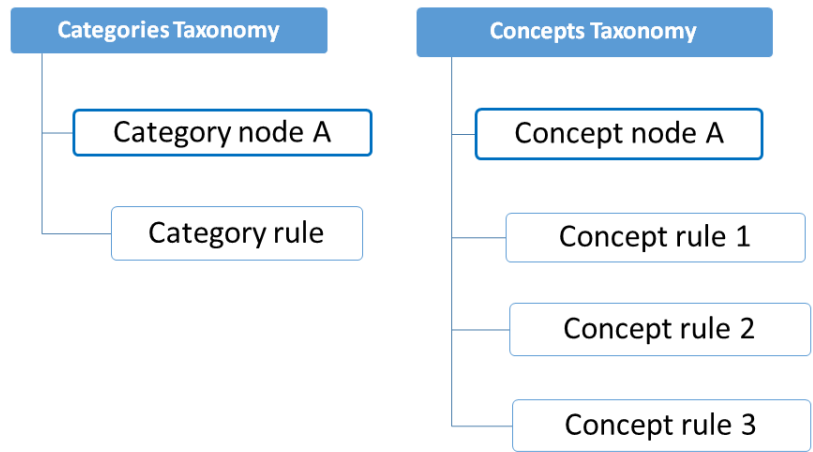
In SAS Visual Text Analytics, you can create category and concept rule sets, which are organized into a taxonomic structure. Each taxonomy consists of *tree nodes* (not to be confused with analysis nodes). Each tree node is a container for one or more rules. The taxonomy is used to organize rules and reflect the overall model design and to make testing, refinement, and maintenance of rules easier. Rules explicitly may reference other tree nodes, but there are no implied dependencies within the tree that impact results (like dependencies of inheritance).

Concept and category taxonomy trees can be organized in any way that is useful for your objectives. However, using a careful and principled design process is recommended for larger projects. For example, commonly referenced rules should be placed in a location where they are easy to find and their shared status is apparent. Naming concept or category tree nodes should enable easy navigation among nodes. See guidelines for naming nodes for more information.

Each category node in the tree is a container for a rule. By contrast, under a concept node, there can exist multiple rules. [Figure 1.2 on page 9](#) demonstrates how category and concept taxonomies differ.

Figure 1.2 Taxonomies in SAS Visual Text Analytics

Working with Taxonomies



Chapter 2

Working in Projects

| | |
|---|-----------|
| Getting Started | 11 |
| Preparing the Document Collection | 11 |
| Creating a Project | 11 |
| Assigning Variables in the Data Tab | 12 |
| Customizing Views in the Data Tab | 13 |
| Using SAS Sentiment Analysis Models in SAS Visual Text Analytics | 14 |
| Project Sharing | 15 |

Getting Started

Preparing the Document Collection

Before you create a project in SAS Visual Text Analytics, you need to prepare your document collection for analysis. SAS Visual Text Analytics enables you to analyze document collections that are stored in various formats. For a list of supported formats, see [Making Data Available to CAS](#). You can select a data source and then identify the text variables and category variables to be analyzed.

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.
- In order to take advantage of interactive visual displays, reduce the size of very large document collections. Very large collections will take a longer time to render in term maps, for example.

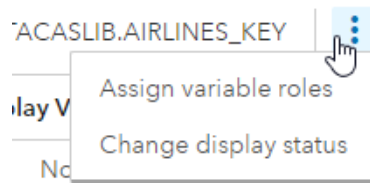
Creating a Project

To create a project in **Model Studio**, click **New Project** in the upper right corner of the **Projects** page. A **New Project** window appears. Within the window, the user can: assign

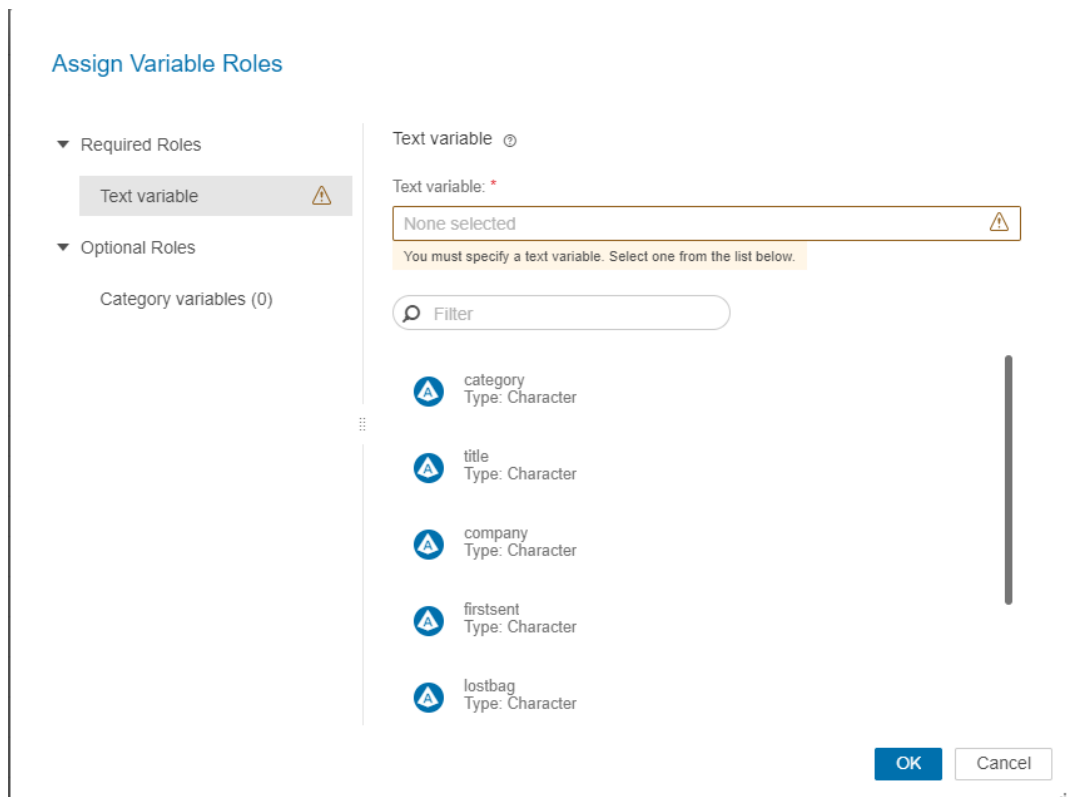
a name to the project; select the type of project they want to create (“Text Analytics” is the default); choose a data source; and select the language to be used for analyzing the text in the document collection. For a list of supported languages, see [Table 1.1 on page 4](#). Once all fields are populated, click **Save** in the lower right corner of the **New Project** window.

Assigning Variables in the Data Tab


Once a project has been created, click on the project to open it. This will bring you to the **Variables table** in the **Data** tab, which displays the variables in the data set, the variable type (Numeric or Character) of each variable, each variable's role (Category, Text, or Key), and display status (Yes or No). To assign variable roles, select the drop-down menu in the top right corner of the **Data** tab.



Select “Assign variable roles” to access the **Assign Variable Roles** window. To assign a text variable, select **Text variable** from the **Required Roles** list on the left side of the window. The text variable identifies the text data to be analyzed. Select the variable to be used from the list provided.



If you are not going to assign a category variable, click **OK** to submit your changes. To assign a category variable or variables, select **Category variables** under **Optional Roles** on the left side of the window. Select the variable or variables of choice from the list


using the  icon; if selecting multiple variables, you can only add one at a time. Once you are done assigning roles, click **OK** in the bottom right corner of the window to submit your changes.

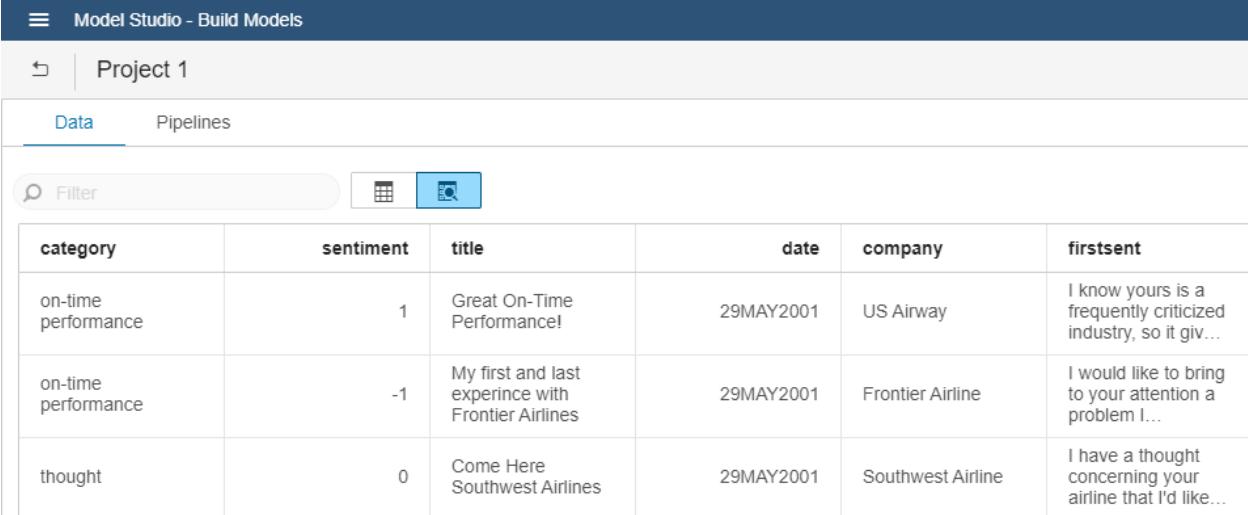
After assigning variable roles, you can select variables to be used as *display variables*. Display variables become columns in the **Documents** tab of all pipeline nodes with the exception of the **Data** node and **Sentiment** node. To change the display status of variables, click the check box to the left of each variable that you want to modify. Once variables have been selected, use the drop-down menu in the top right corner of the **Data** tab and select **Change display status**. The display status of the selected variable or variables changes instantly.

Note: Your **Text** variable will always have a display status of “Yes”; however, you can choose whether to display **Key** and **Category** variables.

Customizing Views in the Data Tab

In the Data Tab, there are two different ways of viewing the information present. The default view in the Data Tab shows the **Variables table**, which has columns for “Variable Name”, “Type”, “Role”, and “Display Variable”. The second option for viewing information about the data set being used is the **View table** option. To switch

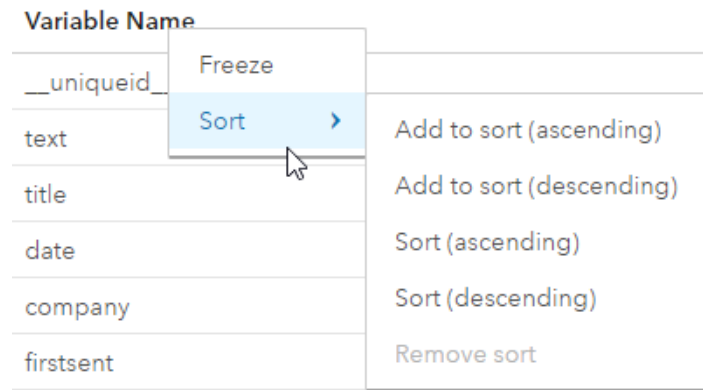
from the **Variables table** to the **View table**, click the  icon in the top left corner of the Data Tab, next to the search bar. The **View table** shows greater detail, and has a column for each of the variables in the data set.




The screenshot shows the Model Studio interface. At the top, there is a header "Model Studio - Build Models" and a breadcrumb "Project 1". Below that, there are tabs for "Data" and "Pipelines". A search bar labeled "Filter" is present, along with icons for a calendar and a view toggle. The main content area displays a table with the following data:

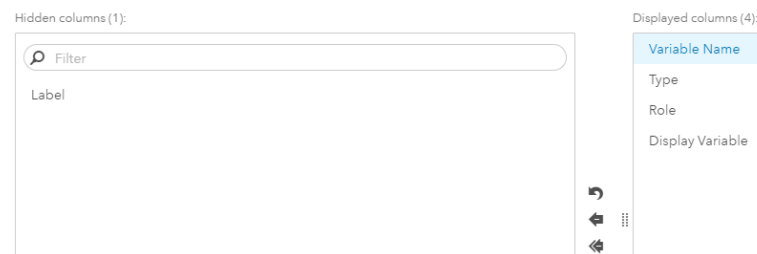
| category | sentiment | title | date | company | firstsent |
|---------------------|-----------|---|-----------|-------------------|--|
| on-time performance | 1 | Great On-Time Performance! | 29MAY2001 | US Airway | I know yours is a frequently criticized industry, so it giv... |
| on-time performance | -1 | My first and last experience with Frontier Airlines | 29MAY2001 | Frontier Airline | I would like to bring to your attention a problem I... |
| thought | 0 | Come Here Southwest Airlines | 29MAY2001 | Southwest Airline | I have a thought concerning your airline that I'd like... |

To customize your view in either the **Variables table** or the **View Table**, you can right click on column headers to sort or freeze a column.



sentiment You can also choose to add or discard columns in the **Manage columns** window, which is made available by clicking the  icon in the top right corner of the Data Tab. In the window, a list of **Hidden columns** and a list of **Displayed columns** are shown.

[Manage Columns](#)



Using the icons between the two lists, you can move variables from the **Displayed columns** list to the **Hidden columns** list, and from the **Hidden columns** list to the **Displayed columns** list.

Note: By default, the **View table** displays all variables as columns and therefore does not display any variables in the **Hidden columns** list.

Using SAS Sentiment Analysis Models in SAS Visual Text Analytics

Rules that are generated using SAS Sentiment Analysis are stored in a .sam binary file. When you create a project in SAS Visual Text Analytics, 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.

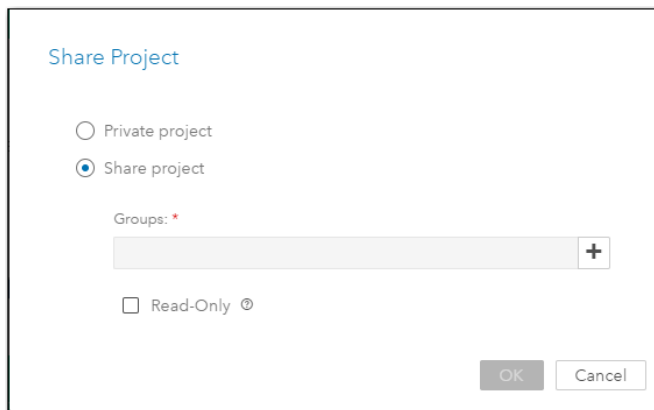
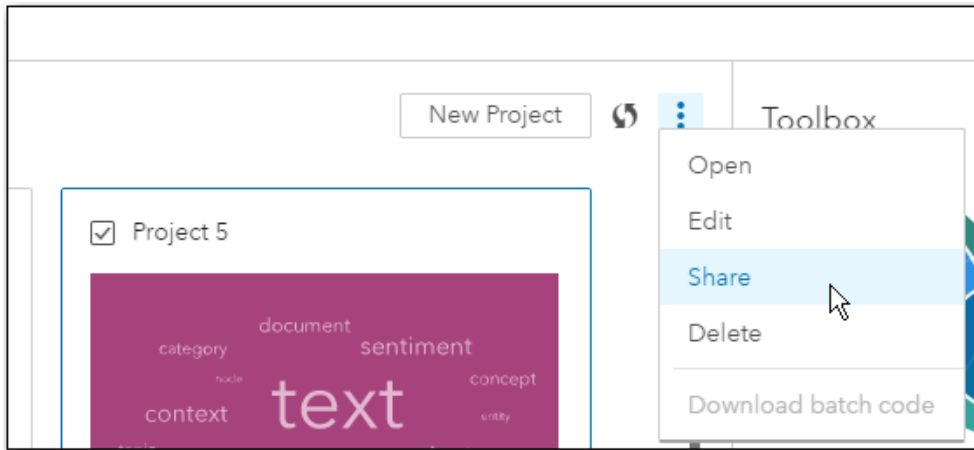
Note: Not all languages have default sentiment models available for use.

Note: Before you use .sam binary files from SAS Sentiment Analysis within SAS Visual Text Analytics, this file must be uploaded to CAS using the loadTableFromDisk CAS action. For more information, see [SAS Cloud Analytic Services: Analytics Programming Guide](#).

For more information about sentiment analysis and scoring, see [SAS Sentiment Analysis 12.2: User's Guide](#).

Project Sharing

It is possible to share projects with other users. In Model Studio, check the selection box in the project that you want to share. Then select the menu and select **Share**.



Note that in shared Read-only mode, all operations that involve changing of existing data, such as running a pipeline, adding or editing concepts or categories, and so on, are disabled. You can still perform actions such as viewing document matches, viewing term maps, and test rules against text. It's important to note that when the project is in Read-only mode, even the project owner cannot make changes to the data.

Chapter 3

Working In Pipelines

| | |
|--|-----------|
| About Working with Pipelines | 17 |
| Using the Default Text Analytics Pipeline | 17 |
| Adding Text Analytics Nodes to the Pipeline | 18 |
| Setting Options for the Analysis Nodes | 19 |
| Locating the Node Options | 19 |
| Concepts | 20 |
| Text Parsing | 20 |
| Sentiment | 21 |
| Topics | 22 |
| Categories | 23 |
| Scoring an External Data Set | 24 |

About Working with Pipelines

A pipeline is a process flow diagram that can be used to represent a sequence of analytical tasks. These analytical tasks are represented as individual nodes in a pipeline.

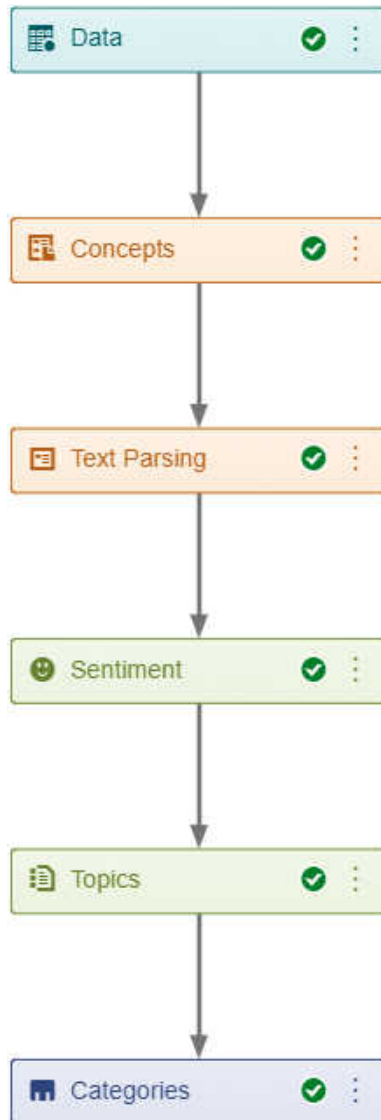
A project can be composed of one or more pipelines. For general information about working with pipelines, see the Pipelines section in *SAS Visual Data Mining and Machine Learning 8.2: User's Guide* .

Using the Default Text Analytics Pipeline

In SAS Visual Text Analytics, six nodes are provided:

- Data
- Concepts
- Text Parsing
- Sentiment
- Topics
- Categories

Each of these nodes is designed to solve a specific problem related to text analytics. These nodes and their associated properties are explained in detail in the following sections. When a new project is created, a default pipeline associated with the project is pre-populated. This default pipeline represents a typical workflow of a text analytics project. It looks like this:



For detailed information about each analytic task performed by the nodes, see [“Visual Text Analytics Basics”](#) on page 4.

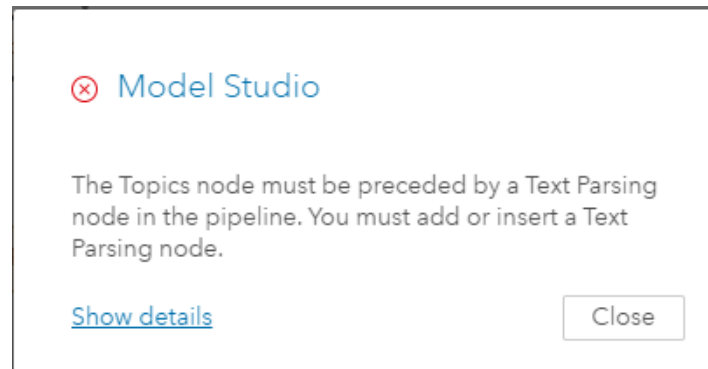
Adding Text Analytics Nodes to the Pipeline

Pipelines are flexible. You can create additional pipelines or modify the default pipeline by adding different nodes. The different nodes within a pipeline are organized into groupings of nodes that share similar characteristics, and are visually grouped by color. The pipeline groupings in SAS Visual Text Analytics are:

1. Natural Language Processing, which includes the **Concepts** and **Text Parsing** nodes.

2. Feature Extraction, which includes the **Topics** node.
3. Text Modeling, which includes the **Categories** node.
4. Miscellaneous, which includes the **Sentiment** node.

When you build a pipeline, a set of governing rules are applied to ensure the proper ordering of the nodes. For example, a Topics node requires a Parse node as one of the predecessors. If such a predecessor does not exist, then the governing rules will prevent the inclusion of a Topics node.



Where applicable, the output of a given node is used within (flows into) its successors. Here are some examples:

- When a Text Parsing node runs, it uses the concepts from all its predecessor nodes during text parsing and extracts relevant terms
- When a Text Parsing node precedes a Concepts or Categories node, all the kept terms from the Text Parsing node are included in the concepts and categories interactive view as textual elements. These textual elements can be used to develop rules for concept extraction or categorization.
- From the Topics interactive window, you can select one or more topics and promote them as categories. These categories and the associated category rules are automatically created when any of the succeeding Category nodes run.
- Within the rules in a Categories interactive window, you can refer to concepts defined in the immediately preceding Concepts node. For more information about referring to concepts in categorization rules, see “[Introduction to Category Rules](#)” on page 62.
- Within the interactive views that follow a Sentiment node, the document level sentiment information is shown alongside the document text.

Note: The Data and Sentiment nodes do not have interactive windows.

Setting Options for the Analysis Nodes

Locating the Node Options

When you select a node in the pipeline, its options are displayed to the right of the node. Each node has different options, detailed in the remainder of this chapter. The defaults are displayed. When you modify one of the node properties, the changes are immediately

saved. This action can also change the node status to “Out of Date” if it was previously marked as “Completed”.

Concepts

Concepts 🗑️ ?

Description:

Extracts specific information from text.

Include predefined concepts

The only option you can specify for the **Concepts** node is whether or not to include predefined concepts in your analysis. You can also adjust the minimum number of documents to view by using the slider. The default, which is set automatically, is 4. Predefined concepts identify items in context such as a person, name, or an address. They save time by providing you with commonly used concepts and their definitions. (Predefined concept availability depends on the project data language.) For more information about concepts and predefined concepts, see “[Concepts](#)” on page 5.

Text Parsing

The options for the **Text Parsing** node include adjusting the minimum number of documents that a term must appear in to be kept in the analysis; specifying a custom start or stop list; and specifying a custom synonym list. If the number of matching documents for a term is less than the minimum number, the term is dropped when the **Text Parsing** node is run.

Start lists and stop lists enable you to control which terms are or are not used, respectively, in the text parsing and terms analysis. You can use a start list or a stop list, but not both. A start list is a data set that contains a list of terms to include in the analysis results. If you use a start list, then only terms that are included in that list appear in the results. No start list is applied by default. To select a start list, check the check box and select the table that represents the start list data set. A stop list is a data set that lists terms to exclude from the analysis results, such as terms that contain little information or that are outside the realm of your analysis. A stop list is provided and automatically applied by default for the following languages: Croatian, Czech, Danish, Dutch, English, Finnish, French, German, Greek, Hebrew, Italian, Norwegian, Polish, Portuguese, Russian, Slovak, Slovene, Spanish, Swedish, and Turkish. To override the default stop list with a custom stop list, check the check box in the options pane and select the table that represents the stop list data set.

A synonym list is a SAS data set that identifies pairs of words that should be treated as a single term for generating topics and textual elements. The data set can include both a term and different forms of that term, including misspellings or abbreviations. For example, you can specify that the words *advert* and *advertising* are to be treated as the term *advertisement*. For more information, see “[Text Parsing—Terms and Synonyms](#)” on page 6.

Text Parsing 🔍 ?

Description:

Prepares text for terms analysis.

Minimum Number of Documents:

4

1 ————— 51 ————— 100

▼ Lists

Specify a custom start or stop list

List type:

Stop list ▼

Start list:

Select a table

Browse

Stop list:

Select a table

Browse

Select a table

Sentiment

You can specify and apply a sentiment model if you want document-level sentiment to appear in your analysis within the application. (Score code can be generated for feature-level sentiment.) If you do not specify a sentiment model, a default model is used (not available for all languages).

Add analysis nodes after the sentiment node in order to see document-level sentiment. There is no interactive window for the Sentiment analysis node.

Sentiment



Description:

Analyzes attitudes expressed in documents.

Specify a sentiment model

Sentiment model:

Select a table

Browse

Topics

- You can choose for the software to generate topics, or you can designate a maximum (or exact) number of topics that you want generated for the analysis. This setting determines the number of documents that are displayed in the Topics node interactive window.

- Term density determines the term cutoff value for each topic. Terms that have an absolute value of weight that is above this value are considered to be included in the topic. Terms that have values below the cutoff are not included in the topic.

Term density is defined by an integer between 0 and 10 (the default value is 1).

When term density is closer to 0, term topic cutoff will be lower and therefore topics will be more densely populated by terms. When term density is closer to 10, topics are less densely populated by terms. Use this setting in conjunction with document density.

- Document density affects the cutoff for each topic, which in turn determines the number of documents that belong to a topic. Only documents with a value higher than the cutoff are assigned to the topic. Use this setting in conjunction with term density.

Topics 🔍 ?

Description:

Assigns documents to topics.

▼ Topic Discovery

Automatically determine number of topics

Maximum topics:

25

Term density:

0

1

10

5

Document density:

0

1

10

5

You must rerun the Topics analysis node to see the results of your changes to these settings.

Categories

You can choose to have the application generate category rules and also rules for category variables. (You specify category variables in the **Data** tab.) Category rules are also generated when topics are promoted to categories.

Categories 🔍 ?

Description:

Classifies documents by subject.

Automatically generate categories and rules

TIP You must run the Category analysis node to see any of the generated categories or rules.

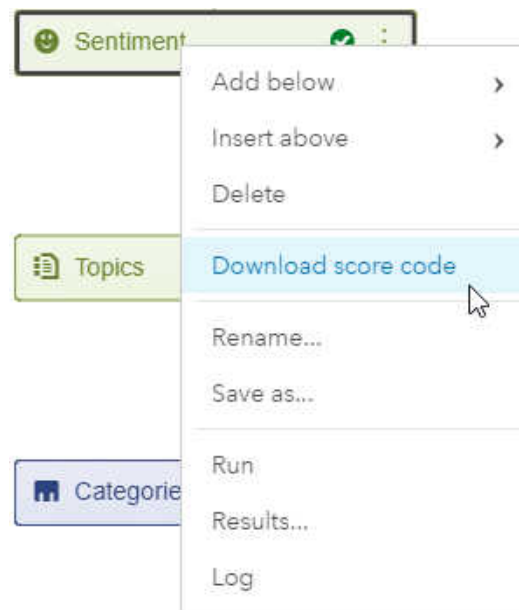
Scoring an External Data Set

You can use the model that you built in your SAS Visual Text Analytics project to score an external data set. When you score an external data set, the category and sentiment models are applied to the external data set (the target data set). The categorization information for the document collection is then output into a scored data set.

Score code can be viewed and downloaded from the following nodes

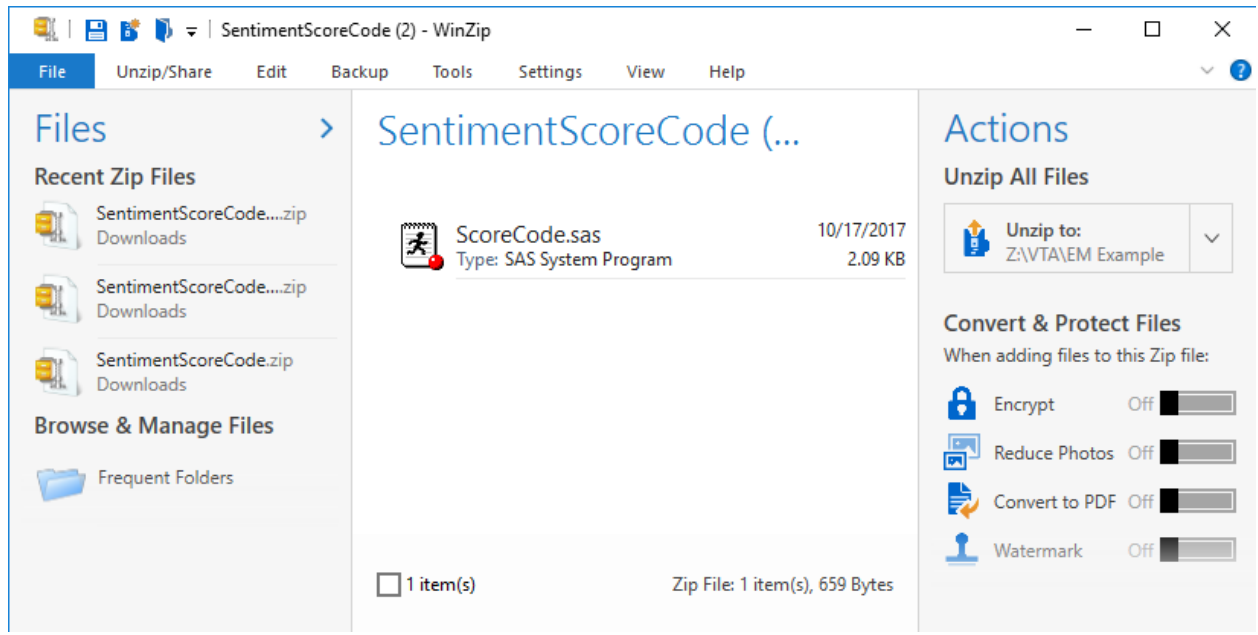
- Concepts
- Sentiment
- Topics
- Categories

This score code can be used to score an external data set using the models created in the corresponding nodes.



When you download score code from a node, the resulting ZIP file contains two entries:

- SAS score code for the node - This code can be used to score an external CAS table within a SAS Viya environment (for example, in SAS Studio).
- A copy of the model created within the node - The model can be used to score external SAS data sets within a SAS 9.4 environment. The models created in SAS Visual Text Analytics 8.2 are compatible with SAS 9.4M5 or higher. The score code in this case can be obtained from SAS Contextual Analysis (for concepts, categories, and sentiment) or SAS Text Miner (for Topics).



Chapter 4




Using the Interactive Windows for the Nodes

| | |
|--|----|
| The Interactive Window for the Concepts Node | 27 |
| The Interactive Window for the Text Parsing Node | 30 |
| The Interactive Window for the Topics Node | 34 |
| The Interactive Window for the Categories Node | 36 |

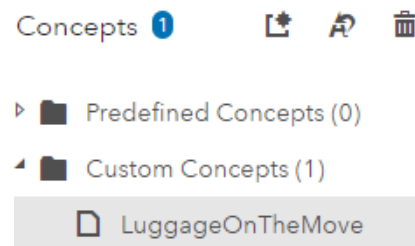
The Interactive Window for the Concepts Node

The interactive window for the **Concepts** node enables you to do the following:

- View predefined and imported concepts
- Add and delete custom concepts
- Test concept rules
- Edit concept properties
- View the documents that contain matches

TIP Use the  and  icons in the **Documents** tab to switch between Document View (shows one document at a time) and Tabular View (Shows multiple documents at once). You can also select the  icon to view only the documents that match a predefined concept or a custom concept.


Expand Predefined Concepts and Custom Concepts to see what is included in your analysis. To expand the list, click the arrow to the left of **Predefined Concepts** or **Custom Concepts**.



Note: If you choose to exclude predefined concepts during project creation, you cannot access predefined concepts in the interactive window for the **Concepts** node.

Here are other important actions that you can execute in the interactive window for the **Concepts** node:

- **Add a custom concept**


Select the  icon to add a custom concept for which you create your own rules.


Note: No more than 400 concepts (including child concepts) can be present.

In the **Edit a Concept** pane, enter the LITI rules for a selected concept. (For more information about writing LITI rules, see [“Writing Concept Rules: Basic LITI Syntax” on page 43](#). Validating the rules before running the **Concepts** node enables you to see and correct errors more easily.

Edit a Concept

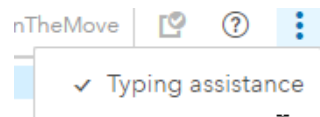
```
1 C_CONCEPT:luggage be@ _c{:V}
```

To validate concept rules, select the  icon in the toolbar in the **Edit a Concept** pane. Otherwise, a warning message will appear at the bottom of the **Edit a Concept** pane.

 Validation is out of date. Once the rules have been validated, rerun the **Concepts** node so that only documents matching the most recent criteria will show in the matched documents tab.


Note: Matching documents are only shown for concepts with the behavior of **Primary**. Concepts with concept behavior set to **Supporting** will not yield any matching documents.

TIP When writing LITI rules for custom concepts, activate **Typing Assistance** from the drop-down list in the **Edit a Concept** toolbar to quickly find operators that can be used.






Edit a Concept

```
1 C_CONCEPT|
```

 C_CONCEPT

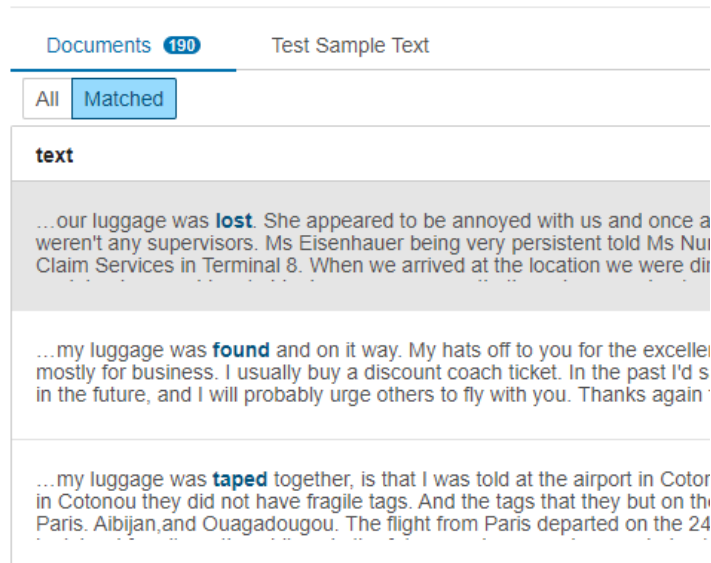
TIP Press Shift+F6 to exit from the code editor.




- **View and explore matching documents**

To view the training documents that contain matches, click the **Documents** tab. Click the  icon to display or hide columns such as **Fact Matches** and **Relevancy**. Click either of the icons  or  to switch between document views. Suppose you created a concept **LuggageOnTheMove**, which contains the rule

```
C_CONCEPT:luggage be@ _c{:V}
```

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



You can also test for matches on a single document or string of text. To test a document, right-click the document in the **Documents** tab and select **Paste to Test Sample Text**. You must then click the  icon in the **Test Sample Text** tab to test the document against the selected concept. To test a string of text, simply enter the desired text into the text box in the **Test Sample Text** tab and click the  icon. If a **Matched item**, **Matched fact**, or **Overlapping match** is discovered, the match is indicated by certain visual cues. Select the  icon to see the legend for each type of cue.



Note: Sample text can be tested on newly created custom concepts without running the **Concepts** node. However, you must run the **Concepts** node to see updated document matches in the **Documents** tab.

Note: The Matched Documents tab and the Test Sample Text tab offer different scopes for concept rule matching. The Matched Documents pane displays relevant matches for all concepts in the project being applied, including those concept types with global impact (for example, REMOVE_ITEM). The Test Sample Text tab shows matches for rules in the selected concept only, plus the concepts that are explicitly referenced in rules of the highlighted concept.

Note: When using **Test Sample Text** feature, global rule types not defined in the specific concept being tested will not affect results. Global rule types include NO_BREAK and REMOVE_ITEM.

- **Guidelines for Naming Concepts**

When you create a custom concept node, follow these naming guidelines:

- Use valid characters – numbers, letters, and underscores (_). (See the Note below regarding the use of underscores).
- Concept names are case-sensitive.
- Create names that are not regular words; using mixed case is recommended to help with readability. For example, MyConcept or myConcept are good names. Do not use names for custom concepts 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.

If underscores (_) are used in concept names, follow these guidelines to ensure that your concept rules will work as expected:

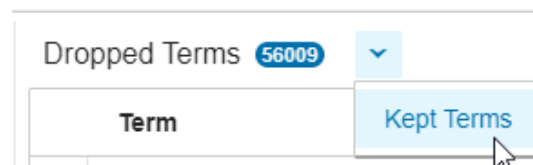
- If you use underscores at either end of the concept name, be sure there is a matched pair at both ends. For example, **_Domestic_** is permitted, but **Domestic_** is not permitted.
- Do not include **_Q**, a character combination reserved by the application, anywhere in a concept name.
- If a concept name begins with an underscore, the next character must be a letter. For example, the concept name **_25anniv_** is not permitted.

TIP Use mixed case to enhance the readability of concept names. For example, **truckMechanicalIssues** is easier to read than **truckmechanicalissues**.

The Interactive Window for the Text Parsing Node

After a pipeline is successfully run, right-click the **Text Parsing** node and select **Open** 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.

TIP To customize your view, use the arrow in the **Dropped Terms** pane to change it to a **Kept Terms** pane, as shown below. You can also resize each pane by using the splitter bars between the two panes.



Here are other important tasks that you can complete in **Terms Management**:

- **View Terms**

In the interactive window for the **Text Parsing** node, view terms in the following contexts:

- The **Kept Terms** pane displays all of the terms in the document collection that were kept
- The **Role** column displays the part of speech from which each term is derived

Note: In some languages, the roles displayed might not be the same as the ones used for rule writing in the Concepts node.

- The **Documents** column displays the number of training documents that contain the selected term
- The **Frequency** column displays the number of times that each term is used
- To view the surface forms that were assigned to a term, click the triangle that appears next to that term

Note: If you chose to exclude predefined concepts in the **Concepts** node, you can still see terms with the role **nlpNounGroup** in the interactive window for the **Text Parsing** node.

| Term | Role | Documents | Frequency |
|------------------------------------|------|-----------|-----------|
| <input type="checkbox"/> not | ADV | 5004 | 21685 |
| <input type="checkbox"/> » flight | N | 4705 | 21630 |
| <input type="checkbox"/> » airline | N | 5861 | 20788 |
| <input type="checkbox"/> » fly | V | 5630 | 14412 |
| <input type="checkbox"/> » time | N | 5510 | 11431 |
| <input type="checkbox"/> » ticket | N | 5000 | 10448 |
| <input type="checkbox"/> quot | N | 1769 | 8383 |
| <input type="checkbox"/> » tell | V | 3040 | 7925 |
| <input type="checkbox"/> 's | PRO | 5405 | 7780 |
| <input type="checkbox"/> here | ADV | 4378 | 7645 |
| <input type="checkbox"/> » know | V | 3471 | 7563 |

| Term | Role | Documents |
|---------------------------------|------|-----------|
| <input type="checkbox"/> i | PRO | 61 |
| <input type="checkbox"/> » be | V | 61 |
| <input type="checkbox"/> to | PPOS | 61 |
| <input type="checkbox"/> the | DET | 61 |
| <input type="checkbox"/> and | CONJ | 60 |
| <input type="checkbox"/> a | DET | 61 |
| <input type="checkbox"/> » have | V | 61 |
| <input type="checkbox"/> in | PPOS | 59 |
| <input type="checkbox"/> » will | V | 61 |
| <input type="checkbox"/> my | DET | 58 |
| <input type="checkbox"/> for | PPOS | 61 |

TIP To customize a view within the **Kept Terms**, **Dropped Terms**, or **Documents** pane, use the **E** icon to access the **Options** menu. From there, you can reorder, add, or drop columns such as **Relevancy**, **Similarity**, or **Role**. Select the column type or types and use the operators between the **Hidden Columns** and **Displayed Columns** panes to customize the content displayed. You can also right-click column headings to change the sorting parameters for each column.

Manage Columns

Note: The above options appear for the **Kept Terms** pane, and might differ from those in the **Dropped Terms** and **Documents** pane.

- **Select and drop 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 select parent terms from the **Kept** tab and move them to the **Dropped** tab by using the icon, and back again using the icon.

Note: If you make changes to the terms and you want to see the effects of your changes in the matched documents table, you must click the icon on the **Pipelines** tab to rerun the pipeline.

CAUTION:

If concept rules are out-of-date when you rerun any nodes (all out-of-date nodes 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  icon. Select the  icon or the  to switch between document views in either the **All documents** tab or the **Matched documents for kept terms** tab. If viewing matching documents, the matching terms are highlighted.

Documents 4705

All Matched

text


...upgrade on next **flight**). Thank you for taking the time to read this. Sincerely, John E.

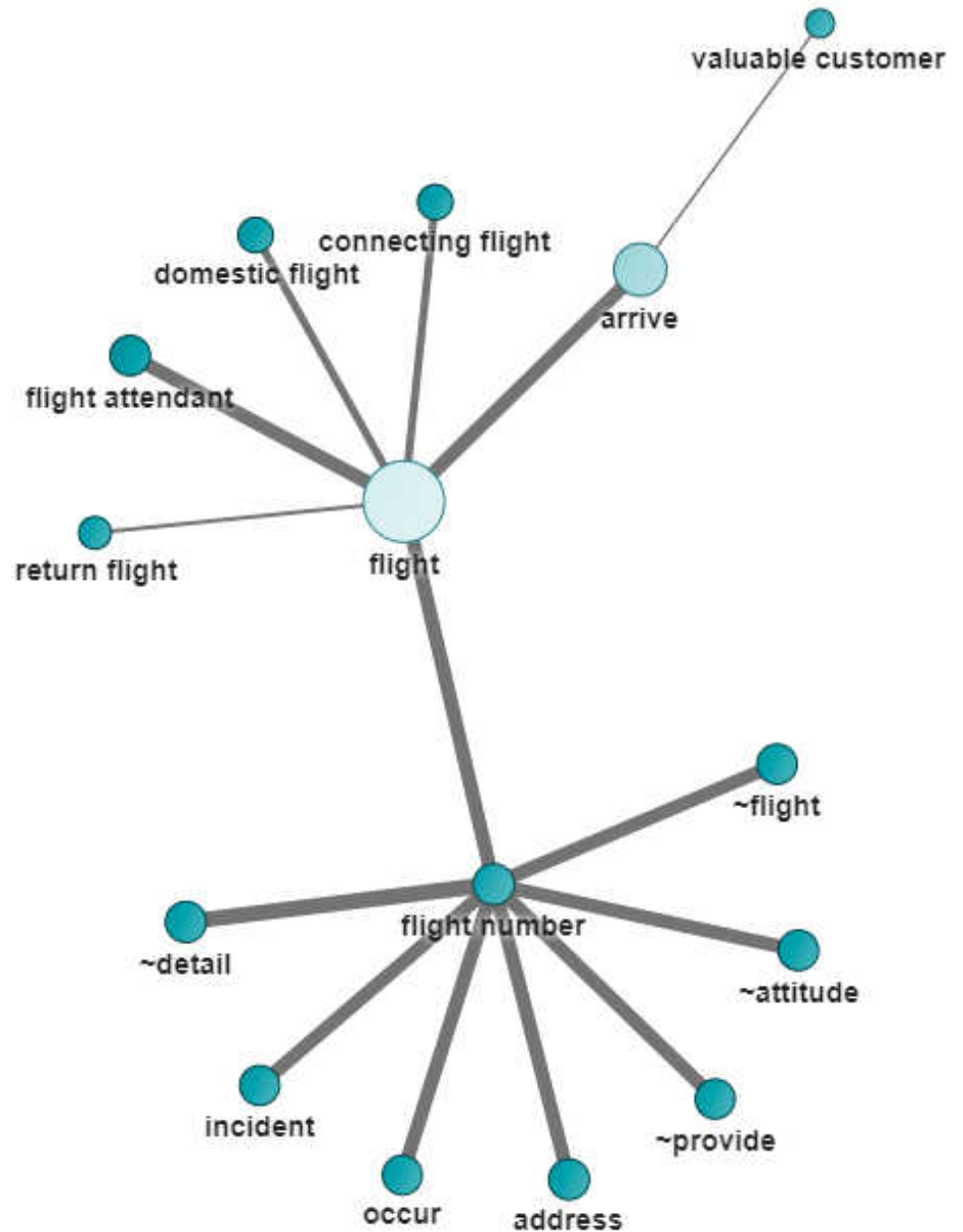
...reference, my last **flight** on your airline was UA1814. It was a domestic **flight**. I traveled on programs in order to maximize their rewards and avoid their miles expiring. Perhaps it would h only one I have access to. When I buy a ticket from you I usually spend \$800.00, and I think I'

...was a domestic **flight**. I traveled on Thanksgiving Day, 2000. I had the pleasure of flying you their way to make us comfortable. The pilot invited my son up to the cockpit, which he loved. T help you to know a little bit about me. I am a true fan of your company, and this reinforces my

Note: Sentiment values are displayed only if a **Sentiment** node precedes the **Text Parsing** node.

- **View a term map**

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



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

Note: Term maps for more frequently occurring terms will take longer to produce. The time it takes to produce a term map can vary dramatically and is dependent upon these factors: the number of documents in your collection, the number of terms being searched, and the number of documents that include the center term.

The Interactive Window for the Topics Node

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 in the interactive window for the **Topics** node:

- **View terms that comprise the topic**

In the following sample screen, the topic is identified by the terms **work**, **good work**, **great**, **compliment**, and **great experience**.

To view **Matched topic terms**, select a topic from the **Topics** pane, and select **Matched** in the **Terms** pane. For more information about **Topics**, select the ⓘ icon in the upper right corner of the **Topics** pane.

The screenshot shows the Model Studio interface with the 'Topics' pane on the left and the 'Terms' pane on the right. The 'Topics' pane contains a table with 12 topics and their associated documents. The 'Terms' pane shows a list of terms with checkboxes and a 'Matched' button.

| Topic | Documents |
|---|-----------|
| <input type="checkbox"/> +work, good work, +great, +compliment, +great experience | 1251 |
| <input type="checkbox"/> +gate, +hotel, +arrive, +hour, +delay | 1176 |
| <input type="checkbox"/> +refund, +credit, +call, +reservation, +fee | 1125 |
| <input type="checkbox"/> quot, +line, +counter, +gate, +agent | 985 |
| <input type="checkbox"/> +seat, +attendant, +flight attendant, +child, +seat | 950 |
| <input type="checkbox"/> gt, lt, northwest, customer, planetfeedback | 802 |
| <input type="checkbox"/> +bag, +baggage, +claim, +luggage, +baggage claim | 791 |
| <input type="checkbox"/> delta, +delta, lines, air, atlanta | 790 |
| <input type="checkbox"/> +mile, frequent, +flyer, +program, +flyer program | 761 |
| <input type="checkbox"/> valuable, +valuable customer, all-time high, all-time, +air travel | 631 |
| <input type="checkbox"/> +suggestion, +benefit, +thought, +consider, +interaction | 477 |

The 'Terms' pane shows a list of terms with checkboxes and a 'Matched' button. The terms are: not, flight, airline, fly, time, ticket, quot, tell, 's, here.

The **Terms** pane displays the following:

- Every kept term from the **Text Parsing** node
- The relevancy of a term to a topic (if a topic has been selected)
- The assigned role (concept) of each term
- The number of documents containing each term
- **View documents associated with the topic**

To view the training documents that are associated with a topic, refer to the **Documents** pane at the bottom of the interactive window for the **Topics** node.

Note: Only one topic can be selected when testing for matching documents.


Select **Matched** to view the documents associated with the selected topic. Within each document, the terms that match the selected topic are highlighted.

Note: In the case that emoji characters are present in the data source, they are rendered as a diamond character with a “?” in it within Model Studio.

- **Merge 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.

- **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 . Once you click this icon, you must rerun the **Categories** node in order for your new category to appear. Upon promoting a topic to a category, the following text will appear in the **Documents** pane.

Added 1 topic as category.

You can promote multiple topics to categories at one time.

Note: When a topic is promoted to a category, any category node succeeding the topic node will have an automatically generated category corresponding to the promoted topic. If you do not want the automatically generated category, ensure that **Automatically generate categories and rules** in the options panel for the Categories node is not selected.

- **Edit topic properties**

You can edit the properties affecting all topics from the main pipeline view. Term density refers to how topics are populated with terms; it is defined by a number between 1 and 10 (the default value is 1). When term density is closer to 1, more documents are captured. When term density is closer to 10, fewer documents are captured.

You can also designate a maximum number of topics that you want generated for the project (between 1 and 500). The default value for **Maximum topics** is 25.

Document density can also be adjusted. Similar to term density, a lower number implies more documents will be captured, and a higher number implies that fewer documents will be captured.

Topics

Description:

Assigns documents to topics.

▼ Topic Discovery

Automatically determine number of topics

Maximum topics:

25

Term density:


0 1 5 10

Document density:

0 1 5 10

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

- **Customize your view**

Use the  icon to select which column types will appear in each pane. In the **Topics** pane, there are two options: Topics added as category and Documents. The **Terms** pane offers more options, such as Relevancy, Similarity, Role, Documents, and Frequency. The **Documents** tab offers two columns, Sentiment and Relevancy. You can also resize columns by using the splitter bars between panes, and change the sort order of each column by right-clicking on the column headings to access sorting options.

The Interactive Window for the Categories Node

After you create a category from a topic in the interactive window for the **Topics** node, ensure that **Automatically generate categories and rules** is checked in the **Categories** node options. With this option checked, the category that was created from the **Topics** node will appear in the interactive window for the **Categories** node after you rerun the **Categories** node. In the **Edit Category** pane, you see the rules that were generated for that category. The **Documents** tab is not populated until you select the category of interest.

Here are the important tasks that you can perform in the interactive window for the **Categories** node:

- **View document matches for categories**

The **Documents** pane offers two tabs, **All** and **Matched**. To see which documents match a particular category, select a category from the **Categories** pane, and then select **Matched** in the **Documents** tab. The highlighted terms are the terms that determined the document’s membership in the category.

Note: In the case that emoji characters are present in the data source, they are rendered as a diamond character with a “?” in it within Model Studio.

| text | Relevancy | Sentiment |
|--|-----------|-----------|
| ... a frequently criticized industry , so it gives me great pleasure to send this compliment about your airline's experience. I hope you'll let everyone involved in this great experience know that their work is greatly appreciated . During the past two months, I have made 3 round trip flights on Southwest Airlines. On every flight, the flight attendants have been friendly, outgoing and enjoyable to be on a flight with. At the airport, Southwest Airline employees are always willing to help, and go above their call of... | 13,000 | 😊 |
| ... a frequently criticized industry , so it gives me great pleasure to send this compliment about your airline's price and value. I hope you'll let everyone involved in this great experience know that their work is greatly appreciated . This is a bit delayed, but I am writing to compliment you on both your prices and service. Due to a family emergency, I had to fly from Burlington, Vermont to Oakland on three days notice. The rest of the industry was charging \$500.00 on up, I paid a measly \$335.00 for a round trip fare. Not... | 13,000 | 😊 |
| ... a frequently criticized industry , so it gives me great pleasure to send this compliment about your airline's airport check-in. I hope you'll let everyone involved in this great experience know that their work is greatly appreciated . My 9-year-old daughter flew from Kansas City to Manchester, NH at 7:20am July 23, flight 970. She was flying by herself and the check-in crew was so very nice to her and whisked her right on the plane as soon as she got to the gate. She said they were very nice to her on the flight. As... | 12,000 | 😊 |

Note: The sentiment for each document is displayed only if you preceded the **Categories** node with a **Sentiment** node.

Note: If a rule is edited or if a category is renamed, the **Categories** node must be rerun in order to display the document matches. A dotted red line underneath a term indicates an out-of-date match.



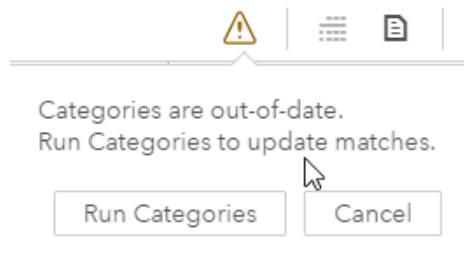
text

... a frequently criticized **industry**, so it gives me great pleasure to **send** this compliment about your airline's experience. I hope you'll let everyone involved in this **great experience** know that their work is greatly **appreciated**. During the past two months, I have made 3 round trip flights on Southwest Airlines. On every flight, the flight attendants have been friendly, outgoing and enjoyable to be on a flight with. At the airport, Southwest Airline employees are always willing to help, and go above their call of...




... a frequently criticized **industry**, so it gives me great pleasure to **send** this compliment about your airline's price and value. I hope you'll let everyone involved in this **great experience** know that their work is greatly **appreciated**. This is a bit delayed, but I am writing to compliment you on both your prices and service. Due to a family emergency, I had to fly from Burlington, Vermont to Oakland on three days notice. The rest of the **industry** was charging \$500.00 on up, I paid a measly \$335.00 for a round trip fare. Not...

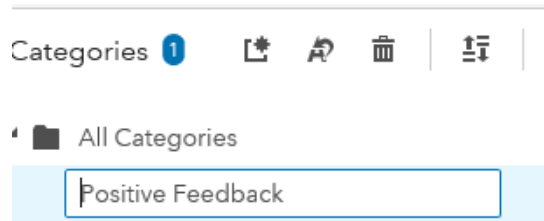
... a frequently criticized **industry**, so it gives me great pleasure to **send** this compliment about your airline's airport check-in. I hope you'll let everyone involved in this **great experience** know that their work is greatly **appreciated**. My 9-year-old daughter flew from Kansas City to Manchester, NH at 7:20am July 23, flight 970. She was flying by herself and the check-in crew was so very nice to her and whisked her right on the plane as soon as she got to the gate. She said they were very nice to her on the flight. As...

Note: Once a category has been modified, a warning icon appears in the top right corner of the **Documents** pane. Upon clicking the icon, you are given the option to rerun Categories in order to update matches.



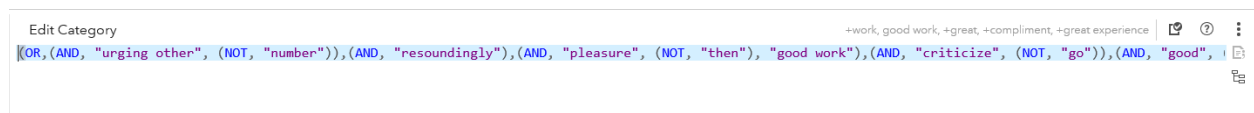
- **Edit category rules**

To begin editing, select a category. In the **Edit Rules** pane, the rule for that category will appear. Use the tree view icon  or the rule view icon  to switch between views. The option to rename a category is also available. Select the category to be renamed and select the  icon in the **Categories** pane. The category name can then be modified.



To edit a rule, you can use either the rule view or the tree view in the **Edit Category** pane.

To edit a rule in the default rule view, select the rule to be edited. The rule will then appear in the **Edit Category** pane.



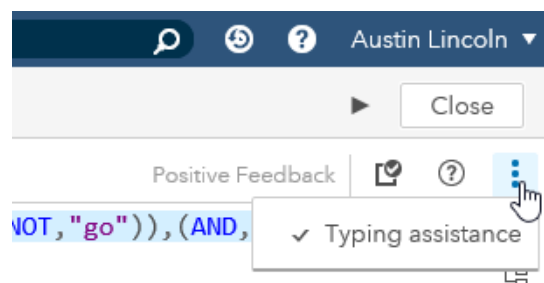
A rule has two main components: arguments and operators. In the rule view, arguments are displayed in purple text, and the operators are displayed in blue text. To edit the selected rule, simply click inside the rule and modify the arguments and the operators as desired.

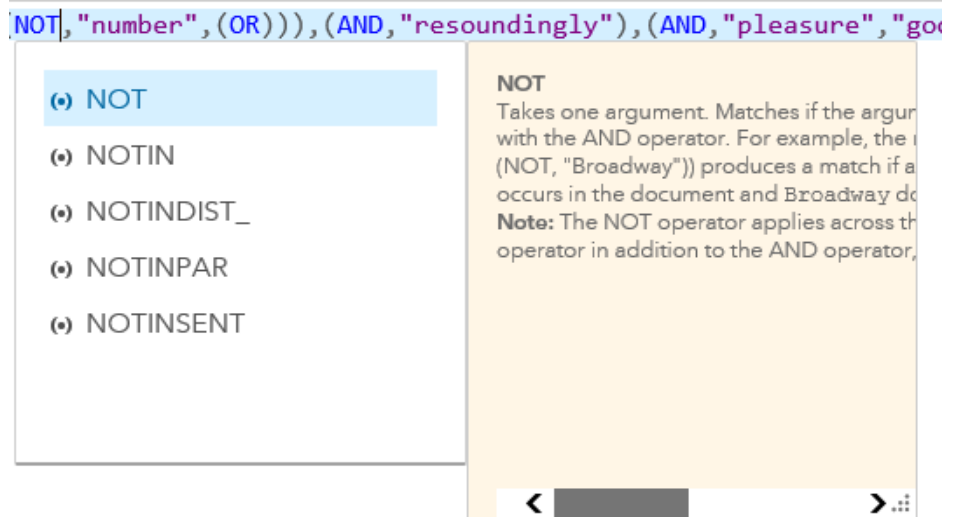
TIP By default, the typing assistance tool is active unless manually disabled.

When modifying an operator, this feature will provide a list of available operators as well as an explanation of what each operator does.



TIP Press Shift+F6 to exit from the code editor.

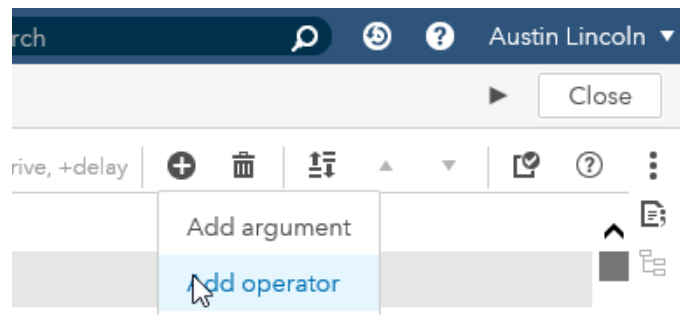
Note: If typing assistance is not desired, open the drop-down menu in the **Edit a Category** toolbar and click **Typing assistance** to turn it off.




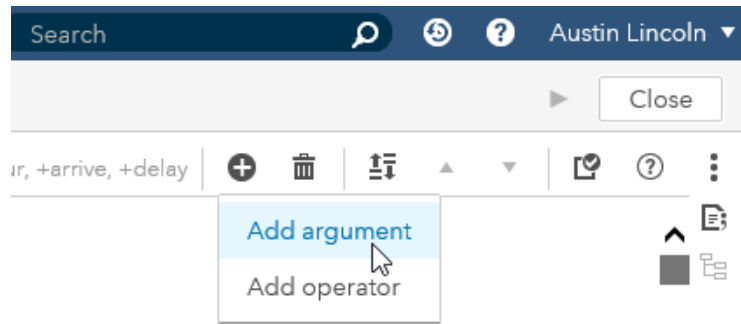


To edit a rule in the **Tree view**, complete the following steps:

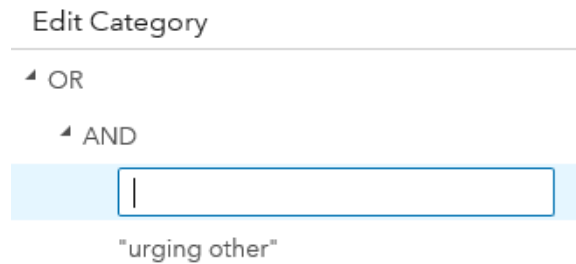
1. Select the category rule to be edited.
2. Select the  icon in the upper right corner of the **Edit a Category** pane.
3. Select the  icon from the **Edit a Category** toolbar and select **Add operator**. A drop-down list of valid operators appears in the **Edit a Category** pane from which you can choose.




4. Once an operator is in place, you must create an argument. Select the operator that you added and select the  icon.
5. Select **Add argument**. A text box appears underneath the operator in the **Edit a Concept** pane.



6. Enter the desired argument in the text box and press **Enter** on the keyboard when you are finished.



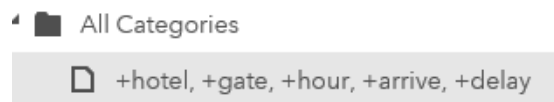
TIP The  Validation is out of date. message in either view reminds you to validate the rule after you make a change.



Click  above the **Edit Category** tab to validate all the rules.

For information about writing category rules, see [“Writing Category Rules” on page 62](#).

- **Test category rules**

To test category rules, select a rule, and then click the **Test Sample Text** tab.



Simply type (or copy and paste) the test text into the **Test Category** tab for the rule that you have selected, or select a document from the **Documents** tab and click the  icon to paste that document into the **Test Category** tab. Click the  to test the text.

Once the testing is complete, any matched items and overlapping matches are highlighted.

Documents **6163**

Test Sample Text



My coworkers and I had a **great experience** with the crew.

Note: When using **Test Sample Text** feature, global rule types not defined in the specific concept being tested will not affect results. Global rule types include NO_BREAK and REMOVE_ITEM.

Clear the highlighting by clicking the **A** icon, or clear the sample text entirely by selecting the **X** icon.

- **Using Textual Elements**

In the **Textual Elements** pane, the terms that were kept from the **Text Parsing** node appear. Use the **Textual Elements** pane to create a rule for an existing category, or to create a rule for a new category. To create a rule in the **Textual Elements** pane, a category must first be selected. Once a category is selected, select the terms from the **Textual Elements** pane that will be used to create the new rule. Select the **+** icon to view and edit the new rule before it is applied to the category selected.

Create Rules from Textual Elements

Select an operator (and any corresponding properties) to generate a rule. (?)

Operator:

Or (OR)

Rule:

(OR,"airlines","airline","problem","problems")



This action will overwrite the existing rule code.




OK



Cancel

Note: The new rule created will replace any previous rule associated with the selected category .

Note: There can be no more than 400 categories (including sub-categories) present.


It might also be useful to know which terms are “similar” to -- that is, likely to appear in the same context as -- a selected term in your documents. Select a kept term and click the  icon to generate similarity scores. The higher the score, the more the term is likely to appear in the same context as the selected term. A score of 1.0 is an exact match (in other words, the term itself). To turn off similarity scores, click the  icon on the right side of the pane, and the terms table will return to its original format.

Textual Elements 10923   

| | String  | Role  |
|--------------------------|--|--|
| <input type="checkbox"/> | not | ADV |
| <input type="checkbox"/> | ▸ flight | N |
| <input type="checkbox"/> | ▸ airline | N |
| <input type="checkbox"/> | ▸ fly | V |
| <input type="checkbox"/> | ▸ time | N |
| <input type="checkbox"/> | ▸ ticket | N |
| <input type="checkbox"/> | quot | N |
| <input type="checkbox"/> | ▸ tell | V |
| <input type="checkbox"/> | 's | PRO |
| <input type="checkbox"/> | here | ADV |
| <input type="checkbox"/> | ▸ know | V |

TIP Use similarity scores to create rules from the most similar terms to capture more documents relating to the selected term.

- **Customize your view**

In the **Textual Elements** and **Documents** panes, use the  icon to select which columns to display or to hide. In **Textual Elements**, there are five options: Documents, Frequency, Similarity, String, and Role. The **Documents** pane offers three options: Relevancy, Sentiment, and text.

Chapter 5

Writing Rules

| | |
|--|-----------|
| Writing Concept Rules: Basic LITI Syntax | 43 |
| Introduction to Concept Rules | 43 |
| Concepts versus Facts | 44 |
| Which Rule Type Should I Use? | 45 |
| Using Punctuation | 47 |
| Adding Rule Modifiers | 48 |
| Using Boolean Operators for Extracting Concept Rules and Facts | 50 |
| Using the Coreference Modifier | 53 |
| Using the Export Feature | 54 |
| Using Part-of-Speech and Other Tags | 55 |
| Using Regular Expressions (Regex) | 57 |
| Using Morphological Expansion Symbols | 59 |
| Adding Comments | 60 |
| Concept Rule Types: Examples | 60 |
| Writing Category Rules | 62 |
| Introduction to Category Rules | 62 |
| Boolean and Proximity Operators for Category Rules | 63 |
| Using Symbols in Boolean Rules | 67 |
| Using <code>_tmac</code> for Referencing Categories | 69 |

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 `LaGuardiaAirportComments`, 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 `LaGuardiaAirportComments` 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 [“The Interactive Window for the Concepts Node” on page 27](#). For a list of rule types, see [“Which Rule Type Should I Use?” on page 45](#).

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 5.1 on page 45](#) and [Table 5.2 on page 47](#). For a list of rule modifiers, see [“Adding Rule Modifiers” on page 48](#).
- Use descriptive concept rule names that cannot be used as single words (for example, **BASEBALLSCORE**). You can also include information about how you will use the concept in other rules by using a prefix (for example **Helper_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 55](#).
- 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 50](#).
- 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 Modifier” on page 53](#).

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 with no intervening items.

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 5.1 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 60.

Table 5.1 Overview of Rules for Extracting Concepts

| Rule Type | Description and Basic Syntax |
|------------|---|
| CLASSIFIER | <p>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.</p> <p>CLASSIFIER:<i>string</i> <, <i>information</i>></p> |
| CONCEPT | <p>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.</p> <p>CONCEPT is a rule type. It is not to be confused with a “concept” in the general sense.</p> <p><i>Note:</i> 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).</p> <p>CONCEPT:<i>argument-1</i> <<i>argument-n</i>> where <i>argument</i> can be a concept name, rule modifier, or string.</p> |

| | |
|--------------|--|
| C_CONCEPT | <p>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.</p> <p><i>Note:</i> This rule type requires the <code>_c{ }</code> modifier.</p> <p>C_CONCEPT:<code><argument> _c{argument}<argument></code> where <i>argument</i> can be a concept name, rule modifier, or string.</p> |
| <hr/> | |
| CONCEPT_RULE | <p>Uses Boolean and proximity operators to determine matches. For a list of operators, see “Boolean and Proximity Operators for Category Rules” on page 63.</p> <p><i>Note:</i> This rule type requires the <code>_c{ }</code> modifier. Quotation marks (") must surround the strings that you want to match. The <code>_c{ }</code> 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.</p> <p>CONCEPT_RULE:<code>(<Boolean-rule-1>...<Boolean-rule-n></code> where <i>Boolean-rule</i> can be nested <i>n</i> times and is written as: <i>Boolean-operator</i> “<code>_c{argument-1}</code>”,<code><“argument-2”></code>...<code><“argument-n”></code>)</p> |
| <hr/> | |
| NO_BREAK | <p>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.</p> <p><i>Note:</i> This rule type requires the <code>_c{ }</code> modifier.</p> <p><i>Note:</i> NO_BREAK applies across the entire taxonomy regardless of where the rule appears or whether the rule is enabled or disabled.</p> <p><i>Note:</i> Do not insert NO_BREAK rules just anywhere. It is helpful to insert them all in one concept. That is, create a concept that contains globally implemented rules only (NO_BREAK or REMOVE_ITEM). Having such rules all in one place aids in troubleshooting the matching behavior across your taxonomy.</p> <p>NO_BREAK: <code>_c{argument}</code> where <i>argument</i> can be a concept name (not recommended) or string.</p> |
| <hr/> | |
| REGEX | <p>Identifies patterns of information that can be represented as a series of character types, as in telephone numbers, ZIP code, product numbers, or hyphenated words. No other elements can be placed in a REGEX rule with the exception of the regular expression itself. Also, the boundaries of the match must coincide with token boundaries; you cannot match a partial token with a REGEX rule.</p> <p>For example,</p> <p>REGEX: <code>[0-9]{5}</code></p> <p>matches any five digit number to help find ZIP codes in the USA.</p> <p>REGEX:regular-expression</p> |

| | |
|-------------|--|
| REMOVE_ITEM | <p>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.</p> <p><i>Note:</i> This rule type requires the <code>_c{ }</code> modifier and the ALIGNED operator. Quotation marks (") must surround each of the two arguments of ALIGNED.</p> <p><i>Note:</i> The REMOVE_ITEM rule type is a global rule type that can influence matches outside of the concept node in which it is used.</p> <p>REMOVE_ITEM:(ALIGNED, "<code>_c{concept name}</code>", "<code>argument</code>") where <i>argument</i> can be a concept name, rule modifier, or string.</p> |
|-------------|--|

Table 5.2 lists the rules used for extracting facts. Included is a brief description of how each rule type is used, along with basic syntax.

Table 5.2 Overview of the Rules for Extracting Facts

| Rule Type | Description and Basic Syntax |
|----------------|---|
| PREDICATE_RULE | <p>Helps you define facts that you want identified in text. For information about facts, see “Concepts versus Facts” on page 44.</p> <p>PREDICATE_RULE:(<i>argument-name-1</i>... <i><argument-name-n></i>): (<i>Boolean-rule-1</i>...<i><Boolean-rule-n></i>) where <i>argument-name</i> refers to a name you specify for fact matching, and where <i>Boolean-rule</i> can be nested <i>n</i> times and is written as: (<i>Boolean-operator</i>, "<code>_argument-name {argument}</code>", ... "<code><_argument-name> {<argument></code>")</p> <p>The PREDICATE_RULE rule type is more flexible than the SEQUENCE rule type because it does not always specify order.</p> |
| SEQUENCE | <p>Identifies facts in documents if the facts appear in the order specified with no intervening elements. For information about facts, see “Concepts versus Facts” on page 44.</p> <p>SEQUENCE:(<i>argument-name-1</i>... <i><argument-name-n></i>): <code>_argument-name-1 {argument}</code> <code><_argument-name-n {argument}></code> where <i>argument-name</i> refers to a name you specify for fact matching, and where <i>argument</i> can be a concept name, rule modifier, or string.</p> <p><i>Note:</i> This syntax is written in its simplest form. Additional modifiers and arguments for concept rule matching can be inserted.</p> <p>The SEQUENCE rule type requires the number of <i>argument-names</i> specified must match the number of <i>_argument-names</i> applied.</p> |

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 SEQUENCE or PREDICATE_RULE definition.
- Before a part-of-speech tag (for example, **:Prep**).

Comma ,

Separates operators and arguments in a CONCEPT_RULE or PREDICATE_RULE definition. Add a space after the comma and before the next argument.

Single space

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

Quotation marks “ ”

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

Parentheses ()

Groups the arguments with each operator in CONCEPT_RULE, REMOVE_ITEM, SEQUENCE, and PREDICATE_RULE rule types.

Square braces []

Character class in the REGEX rule type.

Curly braces { }

Delimits information that is returned as a match.

Adding Rule Modifiers

Several types of concept rule modifiers can enhance the matching ability of a rule. [Table 5.3](#) and [Table 5.4](#) list the type of rule modifiers available and denote which rule types they can be used in.

Table 5.3 Concept Rule Modifiers and Associated Rule Types

| Modifier | CLASSIFIER | CONCEPT | C_CONCEPT | CONCEPT_RULE |
|--|------------|---------|--------------|--------------|
| Comments | X | X | X | X |
| Context (<code>_c{}</code>) | | | X (Required) | X (Required) |
| Word (<code>_w</code>) | | X | X | X |
| Word with initial capital letter (<code>_cap</code>) | | X | X | X |
| Multiple matches symbol (<code>></code>) | | | X | X |
| Morphological expansion symbols (<code>@</code> , <code>@A</code> , <code>@N</code> , and <code>@V</code>) | | X | X | X |

| | | | | |
|---|---|---|---|---|
| Boolean and proximity operators | | | | X |
| Part-of-speech tags | | X | X | X |
| Export feature | X | | | |
| Coreference symbols (<code>_ref{}</code> , <code>_P</code> , and <code>_F</code>) | | X | X | X |
| Regular expressions (Regex) | | | | |
| Predefined concepts | | X | X | X |

Table 5.4 Concept Rule Modifiers and Associated Rule Types, Continued

| Modifier | REMOVE_ITEM | NO_BREAK | SEQUENCE | PREDICATE_RULE | REGEX |
|--|-----------------|-----------------|----------|----------------|-------|
| Comments | X | X | X | X | |
| Context (<code>_c{}</code>) | X (Required) | X (Required) | | | |
| Word (<code>_w</code>) | X | X | X | X | |
| Word with initial capital letter (<code>_cap</code>) | X | X | X | X | |
| > symbol | | | | | |
| Morphological expansion symbols (<code>@</code> , <code>@A</code> , <code>@N</code> , and <code>@V</code>) | X | X | X | X | |
| Boolean and proximity operators | | | | X | |
| Part-of-speech tags | X | X | X | X | |
| Export feature | | | | | |
| Coreference symbols (<code>_ref{}</code> , <code>_P</code> , and <code>_F</code>) | | | | | |

| | | | | | |
|-----------------------------|---|---|---|---|--------------|
| Regular expressions (Regex) | | | | | X (Required) |
| Predefined concepts | X | X | X | X | |

Using Boolean Operators for Extracting Concept Rules and Facts

Table 5.5 lists Boolean operators that you can use when you write concept rules and identify facts.

Table 5.5 Boolean Operators for Extracting Concept Rules and Facts

| Operator | Description |
|--------------------------|---|
| ALIGNED | <p>Takes two arguments. Returns a match when both arguments are matched in the same span of text in a document. Used with the REMOVE_ITEM rule type only. 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:</p> <pre>REMOVE_ITEM: (ALIGNED, "_c{LOC}", "PERSON")</pre> |
| AND | <p>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:</p> <pre>CONCEPT_RULE: (AND, "_c{King Louis XIV}", "France")</pre> |
| DIST _{<i>n</i>} | <p>(Distance) Takes a value for <i>n</i> and two or more arguments. Matches if all arguments occur within <i>n</i> (or fewer) tokens of each other, regardless of their order. For example, the following rule returns a match in the phrase the picture with the best lighting:</p> <pre>CONCEPT_RULE: (DIST_5, "best", "_c{picture}")</pre> <p><i>Note:</i> For calculation purposes, the distance between tokens is not inclusive. For example, the distance between best and show in the phrase best in show is two tokens. Tokens that include hyphens are counted as one (for example, merry-go-round is one token).</p> |
| NOT | <p>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:</p> <pre>CONCEPT_RULE: (AND, (OR, "_c{cinema}", "_c{theater}", "_c{theatre}"), (NOT, "Broadway"))</pre> <p><i>Note:</i> The NOT operator applies across the entire document. All operators must have their own parentheses around themselves and their associated arguments.</p> |

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 Visual Text Analytics 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.**:

```
CONCEPT_RULE: (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* tokens 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 tokens is not inclusive. For example, the distance between **best** and **show** in the phrase **best in show** is two tokens. Tokens that include hyphens are counted as one (for example, **merry-go-round** is one token).

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 token **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 Sample Text** 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:

| | |
|-----------------------------|---|
| <code>\r\n\r\n</code> | Two consecutive carriage returns and new lines (for documents created in Windows) |
| <code>\r\n \r\n</code> | Two consecutive carriage returns and new lines, separated by a space |
| <code>.<SPACE></code> | Period (.) followed by an ASCII space |
| <code>.\n</code> | Period (.) followed by a new line |
| <code>.\r</code> | Period (.) followed by a carriage return |
| <code>!</code> | Exclamation point |
| <code>!\n</code> | Exclamation point followed by a new line |
| <code>!\r</code> | Exclamation point followed by a carriage return |
| <code>?</code> | Question mark |
| <code>?n</code> | Question mark followed by a newline |
| <code>?r</code> | Question mark followed by a carriage return |
| <code>.)</code> | Period followed by a closing parenthesis |
| <code>!)</code> | Exclamation point followed by a closing parenthesis |
| <code>?)</code> | Question mark followed by a closing parenthesis |
| <code>.”</code> | 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* tokens 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 tokens 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. Tokens that include hyphens are counted as one (for example, **merry-go-round** is one token).

SENTSTART_*n* (Start of sentence) Takes a value for *n* and one or more arguments. Returns matches within *n* tokens 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. Tokens that include hyphens are counted as one (for example, **merry-go-round** is one token).

UNLESS Takes two arguments, the second of which is one of the following operators (with its arguments): AND, SENT, DIST, ORD, or ORDDIST. Restricts certain matches by specifying a relationship between two arguments and allowing a match only if a third argument does not intervene. Used in rule types PREDICATE_RULE and CONCEPT_RULE only.

For example, the following rule does not include the token **river** in its matches; in addition, the rule 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 governed directly by the UNLESS operator, specify concepts that contain only CLASSIFIER or REGEX rules.

Using the Coreference Modifier

Use the coreference modifier (`_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 modifier (`_ref{}`). Place the symbol after the `_ref{concept}` modifier.

> (Multiple matches)

Locates multiple instances of a match that is specified by the coreference modifier (`_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 from the coreference rule match onward. Sample syntax:

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

_P (Preceding)

Returns only matches that occur up to and including 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).

The following example illustrates how to export two sets of terms to the same concept.

```
CLASSIFIER: [export=text2]:text1
```

If **text1** and **text2** appear in a document, return **text1** and **text2** as separate matches for the concept where this line is located. For example, suppose you have written the following rule:

```
CLASSIFIER: [export=SAS]:institute
```

The string **SAS institute** returns **SAS** and **institute** as matches to the concept where this line is located. The string **institute** (occurring alone) is a match, but not **SAS** occurring alone.

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 a modal verb (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 5.6 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 71. Note that in some languages, the tags documented in these sections might be different from the tags displayed in the Role column of the Text Parsing node.

Table 5.6 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 | Undeclared <i>be, do, or have</i> auxiliary Undeclared verb First person singular verb | be, do, have go, see, love am |
| :V3sg | Third person singular <i>be, do, or have</i> auxiliary Third person singular verb | is, does, has goes, sees, loves |
| :Ving | Present participle <i>be, do, or have</i> auxiliary Present participle | being, doing, having bucketing, climbing |
| :Vpp | Past participle <i>be, do, or have</i> auxiliary Past participle | been, done, had dashed, factored, gone |

| | | |
|-----------|--|--|
| :Vpt | Past tense <i>be, do, or have</i> auxiliary Past tense verb | was, were, did, have dashed, factored, went |
| :WAdv | Adverbial <i>wh</i> | how, when, whereby |
| :Wdet | Demonstrative determiner <i>wh</i> | which, what, whatever |
| :WPossPro | Possessive determiner <i>wh</i> | whose |
| :WPro | Nominal <i>wh</i> | whose, what, whoever |

Using Regular Expressions (Regex)

Use regular expressions (Regex syntax) to identify regularly occurring patterns in the text that might include numbers, punctuation, 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 one or more of 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]
```

Note: Matches returned by `^` are limited to ASCII characters.

- 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 5.7 on page 58](#).

Table 5.7 Special Characters (Metacharacters) Used in Regular Expressions

| Character or Expression | Description |
|-------------------------|--|
| | (Alternative) Indicates that matches occur when either regular expression <i>a</i> or <i>b</i> is matched. Example: <i>a b</i> |
| () | Grouping mechanism (non-remembering). Used in expressions for clarity. Example: <i>(?:?ababab) b</i> |
| . | (Wildcard) Matches any single ASCII character. |
| % | Matches % |
| ? | Matches 0 or 1 occurrences |
| * | Matches 0 or more occurrences |
| + | Matches 1 or more occurrences |
| { } | Indicates repetition: <div style="display: flex; justify-content: space-between; margin-top: 5px;"> <div style="text-align: left;"> <p><i>{n}</i> matches exactly <i>n</i> occurrences</p> <p><i>{n,}</i> matches at least <i>n</i> occurrences</p> </div> <div style="text-align: right;"> <p><i>{n,m}</i> matches at least <i>n</i> occurrences but no more than <i>m</i> occurrences</p> </div> </div> |
| \a | Alarm (beep) |
| \n | New line |
| \r | Carriage return |
| \t | Tab |

| | |
|-------------------|---|
| <code>\f</code> | Form feed |
| <code>\e</code> | Escape |
| <code>\d</code> | Digit (same as <code>[0-9]</code>) |
| <code>\D</code> | Not a digit (same as <code>[^0-9]</code>) |
| <code>\w</code> | Word character (same as <code>[a-zA-Z_0-9]</code>) |
| <code>\W</code> | Non-word character (same as <code>[^a-zA-Z_0-9]</code>) |
| <code>\s</code> | White space character (same as <code>[\t\n\r\f]</code>) |
| <code>\S</code> | Non-white-space character (same as <code>[^\t\n\r\f]</code>) |
| <code>\xh</code> | Hexadecimal number, where <i>h</i> is a hexadecimal character |
| <code>\xhh</code> | Hexadecimal number, where <i>h</i> is a hexadecimal character |
| <code>\0o</code> | Octal number, where <i>o</i> is an octal digit |
| <code>\0oo</code> | Octal number, where <i>o</i> 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 **xy***litol*, **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 **xxx**, **yz**, and **zyzy**:

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

However, because of the presence of characters other than **x**, **y**, and **z**, 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 5.8 Morphological Expansion Symbols in Concept Rules

| Symbol | Description |
|--------|---|
| @ | <p>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.</p> <p><i>Note:</i> If you apply @ to a word that SAS Visual Text Analytics 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.</p> |
| @A | <p>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.</p> <p><i>Note:</i> If you apply @A to a word that is not an adjective, no expansion occurs.</p> |
| @N | <p>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.</p> <p><i>Note:</i> If you apply @N to a word that is not a noun, no expansion occurs.</p> |
| @V | <p>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.</p> <p><i>Note:</i> If you apply @V to a word that is not a verb, no expansion occurs.</p> |

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 the 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 one or more digits, a comma, or a period, and then ending in one or more digits.

For more information about writing Regex rules, see [“Using Regular Expressions \(Regex\)” on page 57](#).

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

Introduction to Category Rules

Category rules resolve to true or false. “True” results in a match. Category 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 5.9 on page 63](#).

Use the following syntax for a category rule:

```
(OPERATOR, argument1, <argument2>, ...)
```

where arguments can be terms, strings, or nested rules.

General rules for syntax:

- Boolean and proximity operators and their arguments are enclosed in parentheses and separated with commas. The arguments are included in quotation marks (“”). Example: (AND, “my_w holiday”, “_cap”)
- 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 69.
- Concept node names can be referenced in category rules. If you reference a concept node name, the concept matches are used to contribute to the true/false match of the category rule. Concept node names must be enclosed in braces ([]). 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. 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 Visual Text Analytics 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 5.10 on page 67](#).

Note: XPath expressions are not supported.

Boolean and Proximity Operators for Category Rules

[Table 5.9](#) shows a list of Boolean and proximity operators that you can use to write category rules.

Table 5.9 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_ <i>n</i> | <p>(Distance) Takes a value for <i>n</i> and two or more arguments. Matches if all arguments occur within <i>n</i> (or fewer) tokens 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.</p> <p><i>Note:</i> For calculation purposes, the distance between tokens is not inclusive. For example, the distance between the tokens best and show in the phrase best in show is two tokens. Words that include hyphens are counted as one token (for example, merry-go-round is one token).</p> |
| <hr/> | |
| END_ <i>n</i> | <p>(From the end of the document) Takes a value for <i>n</i> and one or more arguments. Matches if the argument occurs within <i>n</i> tokens from the end of the document. For example, the rule (END_35, “conclusion”) returns a match if conclusion is found within 35 tokens from the last token in the document.</p> <p><i>Note:</i> Words that include hyphens are counted as one word (for example, merry-go-round is one word).</p> |
| <hr/> | |
| MIN_ <i>n</i> | <p>(Minimum) Takes a value for <i>n</i> and one or more arguments. Matches if the document contains at least <i>n</i> 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.</p> |
| <hr/> | |
| MINOC_ <i>n</i> | <p>(Minimum occurrence) Takes a value for <i>n</i> and one or more arguments. Matches if the document contains at least <i>n</i> 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.</p> |
| <hr/> | |
| MAXOC_ <i>n</i> | <p>(Maximum occurrence) Takes a value for <i>n</i> and one or more arguments. Matches if the document contains <i>n</i> or fewer occurrences of the arguments (in any order or combination). 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.</p> |
| <hr/> | |
| MAXPAR_ <i>n</i> | <p>(Maximum paragraph) Takes a value for <i>n</i> and one or more arguments. Matches if all arguments occur within the first <i>n</i> (or fewer) paragraphs of the document, in any order. For example, the rule (MAXPAR_4, “seasonal”, “herbs”, “plants”) returns a match if seasonal occurs in paragraph 4, herbs occurs in paragraph 2, and plants occurs in paragraph 2.</p> <p><i>Note:</i> MAXPAR rules work properly only when applied to data sets that contain paragraph delimiters (\n\n). MAXPAR cannot be applied on the Test Sample Text tab. MAXPAR also cannot be applied in the Categories node to data that is contained in folders.</p> |
| <hr/> | |
| MAXSENT_ <i>n</i> | <p>(Maximum sentence) Takes a value for <i>n</i> and one or more arguments. Matches if all arguments occur within the first <i>n</i> 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.</p> |

| | |
|---------------------|--|
| NOT | <p>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.</p> <p><i>Note:</i> The NOT operator applies across the entire document.</p> |
| NOTIN | <p>(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.</p> |
| NOTINDIST_ <i>n</i> | <p>(Not in distance) Takes a value for <i>n</i> and two arguments. Matches if the arguments do not occur within <i>n</i> tokens 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 tokens of each other, or if only orange appears in the document. The following sentence returns a match because the tokens that are specified in the rule are more than three words apart: How green is my valley, how orange is the sunset?</p> <p><i>Note:</i> For calculation purposes, the distance between tokens is not inclusive. For example, the distance between the tokens best and show in the phrase best in show is two tokens. Tokens that include hyphens are counted as one token (for example, merry-go-round is one token).</p> |
| NOTINPAR | <p>(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).</p> <p><i>Note:</i> NOTINPAR rules work properly only when applied to data sets that contain paragraph delimiters (\n\n). NOTINPAR cannot be applied on the Test Sample Text tab. NOTINPAR also cannot be applied in the Categories node to data that is contained in folders.</p> |
| NOTINSENT | <p>(Not in sentence) Takes two or more arguments and matches when the first of the two arguments is present and the second of the two arguments does NOT occur. For example, the rule (NOTINSENT, “trade”, “China”) indicates that “trade” will match if the word “China” does not occur in the same sentence. For a list of sentence delimiters, see the SENT operator.</p> |
| OR | <p>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.</p> <p><i>Note:</i> Rules that are generated by SAS Visual Text Analytics nest the OR operator within the AND operator. However, the OR operator can stand alone.</p> |

| | |
|-------------------|--|
| ORD | <p>(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.</p> |
| ORDDIST_ <i>n</i> | <p>(Order and distance) Takes a value for <i>n</i> 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 <i>n</i> tokens of each other. For example, the rule (ORDDIST_5, “elementary”, “statistics”) returns a match in the phrase the teacher introduced elementary statistics.</p> <p><i>Note:</i> For calculation purposes, the distance between tokens is not inclusive. For example, the distance between the tokens best and show in the phrase best in show is two tokens. Words that include hyphens are counted as one token (for example, merry-go-round is one word).</p> |
| PAR | <p>(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.</p> <p><i>Note:</i> PAR rules work properly only when applied to data sets that contain paragraph delimiters (\n\n). PAR cannot be applied on the Test Sample Text tab. PAR also cannot be applied in the Categories node to data that is contained in folders.</p> |
| PARPOS_ <i>n</i> | <p>(Paragraph position) Takes a value for <i>n</i> and one or more arguments. Matches if all arguments occur within the <i>n</i>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.</p> <p><i>Note:</i> PARPOS rules work properly only when applied to data sets that contain paragraph delimiters (\n\n). PARPOS cannot be applied on the Test Sample Text tab. PARPOS also cannot be applied in the Categories node to data that is contained in folders.</p> |

| | |
|-----------------|---|
| SENT | <p>(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:</p> <p>\r\n\r\n Two consecutive carriage returns and new lines (for documents created in Windows)</p> <p>\r\n \r\n Two consecutive carriage returns and new lines, separated by a space</p> <p>.<SPACE> Period (.) followed by an ASCII space</p> <p>.\n Period (.) followed by a new line</p> <p>.\r Period (.) followed by a carriage return</p> <p>! Exclamation point</p> <p>!\n Exclamation point followed by a new line</p> <p>!\r Exclamation point followed by a carriage return</p> <p>? Question mark</p> <p>?n Question mark followed by a newline</p> <p>?r Question mark followed by a carriage return</p> <p>.) Period followed by a closing parenthesis</p> <p>!) Exclamation point followed by a closing parenthesis</p> <p>?) Question mark followed by a closing parenthesis</p> <p>.” Period followed by double quotation marks</p> |
| START_ <i>n</i> | <p>(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.</p> <p><i>Note:</i> Words that include hyphens are counted as one word (for example, merry-go-round is one word).</p> |

Using Symbols in Boolean Rules

You can use the symbols listed in [Table 5.10](#) 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 5.10 Special Symbols Used in Boolean Rules

| Symbol | Description |
|--------|--|
| * | <p>(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.</p> |

| | |
|----|--|
| ^ | <p>(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.</p> <p><i>Note:</i> 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.</p> |
| \$ | <p>(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.</p> <p><i>Note:</i> 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.</p> |
| @ | <p>(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).</p> <p><i>Note:</i> If you apply @ to a word that SAS Visual Text Analytics 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.</p> |
| @A | <p>(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.</p> <p><i>Note:</i> If you apply @A to a word that is never an adjective, no expansion occurs.</p> |
| @N | <p>(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.</p> <p><i>Note:</i> If you apply @N to a word that is never a noun, no expansion occurs.</p> |
| @V | <p>(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.</p> <p><i>Note:</i> If you apply @V to a word that is never a verb, no expansion occurs.</p> |

| | |
|-----------------|---|
| <code>_L</code> | (Literal matching) Matches a literal string. Useful when you want to match a string that includes symbols. For example, the argument " <code>\$USD_L</code> " returns the match <code>\$USD</code> . <i>Note:</i> Tokens (words, phrases, symbols, or other meaningful elements) need to be specified by the user to be considered for matching. |
|-----------------|---|

| | |
|-----------------|--|
| <code>_C</code> | (Case matching) Specifies case-sensitive matching. For example, the argument " <code>Iris_C</code> " returns the match <code>Iris</code> , but not <code>iris</code> . |
|-----------------|--|

Using `_tmac` for Referencing Categories

Referencing a category enables you to leverage the rule in an existing category without having to duplicate it. 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 and proximity 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 | 71 |
| Part-of-Speech Tags for Rule Writing | 72 |
| Arabic | 72 |
| Chinese | 73 |
| Croatian | 74 |
| Czech | 75 |
| Danish | 76 |
| Dutch | 77 |
| English | 77 |
| Farsi | 78 |
| Finnish | 79 |
| French | 80 |
| German | 81 |
| Greek | 82 |
| Hebrew | 83 |
| Hindi | 83 |
| Indonesian | 84 |
| Italian | 85 |
| Japanese | 86 |
| Korean | 91 |
| Norwegian | 92 |
| Polish | 93 |
| Portuguese | 94 |
| Russian | 94 |
| Slovak | 95 |
| Slovene | 96 |
| Spanish | 97 |
| Swedish | 98 |
| Tagalog | 98 |
| Thai | 99 |
| Turkish | 100 |
| Vietnamese | 101 |

Introduction to Part-of-Speech and Other Tags

The part-of-speech tags for rule writing 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 55.

Part-of-Speech Tags for Rule Writing

Arabic

Table A1.1 Part-of-Speech Tags for Arabic

| Part-of-Speech Tag | Description | Examples |
|--------------------|-------------------|----------------|
| :ADJ | Adjective | أبدي، أثري |
| :ADV | Adverb | أيضاً، ربما |
| :CONJ | Conjunction | بل، حتى |
| :DET | Determiner | ال |
| :DIALECT | Dialect | أسم، أثول |
| :FUT | Future particle | س، سوف |
| :INTERJ | Interjection | أجل، لا |
| :INTERROG | Interrogative | أين، عمّا |
| :NEGPART | Negative particle | لم |
| :NOUN | Noun | تفاحة، شجرة |
| :NUM | Number | آلاف، أربعة |
| :PART | Particle | قد، لقد |
| :PREP | Preposition | إلا، على |
| :PRON | Pronoun | أنا، أنت |
| :PROP | Proper noun | أمريكا |
| :PUNC | Punctuation | ، ، ؟ |
| :CV | Imperative verb | انتبها، العبان |
| :IV | Present verb | تأتون، تلعبا |
| :PV | Past verb | أنتا، لعبت |

| | | |
|----------|--------------|---|
| :ASCII | English word | memory, tablets |
| :DEFAULT | Unknown word | اعتبَادِيًا، وشيئًا |
| :NUMBER | Number | 1.8, 200 |
| :URL | URL | http://www.sas.com |

Chinese

Table A1.2 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 | 噢, 呸, 哦喲 |
| :F | Location / direction | 中間, 下边, 南側 |
| :G | Other morpheme | 馨, 慚 |
| :H | Other prefix | 亚, 非 |
| :K | Other suffix | 们, 者, 們 |
| :L | Idiom (chengyu) | 囫圇吞枣, 博古通今, 一廂情願 |
| :M | Quantifier | 十, 卅, 成千上万, 上萬, 1 0 5 1 |
| :N | Noun | 人, 桌子, 香蕉, 枷鎖 |
| :NR | Proper noun, name | 习近平, 梁振英, 奥巴马 |
| :NR_xing | Proper noun, last name for Chinese (most are single characters) | 赵, 邹, 诸葛, 趙 |
| :NS | Proper noun, geographic | 中国, 美國, 山東 |

| | | |
|----------|---|----------------|
| :NS_abbr | Proper noun, abbreviation for country names (all are single characters) | 俄, 匈, 葡, 緬 |
| :NT | Proper noun, organization | 北京大學, 上汽集團 |
| :NZ | Proper noun, miscellaneous | 潘婷, 劍南春 |
| :O | Onomatopoeia | 吱呀, 叭叭喳喳, 劈裏啪啦 |
| :P | Preposition | 依照, 對於 |
| :Punct | Punctuation(English comma) | , |
| :Q | Classifier | 個, 斤, 艘, 加侖 |
| :R | Pronoun | 我, 他們, 這 |
| :S | Subcountry location (general; specifics only within sinosphere) | 地上, 上空, 高處, 內廳 |
| :T | Temporal phrase | 今天, 夜間, 十月, 去歲 |
| :U | Particle | 的, 了, 着 |
| :UNKNOWN | Unknown word | 嫻, 繹 |
| :V | Verb | 看, 認為, 彈奏, 徵納 |
| :W | Punctuation or symbols | !, ., \$, ¥ |
| :Y | Interjectional particle | 吧, 嗎, 麼 |

Croatian

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

| Part-of-Speech Tag | Description | Examples |
|--------------------|--------------|--|
| :ADJ | Adjective | svaki, hrvatskim, koje |
| :ADV | Adverb | uistinu, tamo |
| :CONJ | Conjunction | a, ali, kad |
| :INTJ | Interjection | hej, hajde, oh |
| :N | Noun | dan, april, http://www.sas.com , dr, itd. |

| | | |
|-------|--------------------------|--|
| :PTCL | Particle | ne, bilo (as in “bilo koje”) |
| :PPOS | Preposition | sa, bez, o |
| :PRO | Pronoun | ja, me, ih, nas, vam, njihovoj, svašta |
| :V | Verb | voli, došao, pozvala, dodíte, bih |
| :NUM | Number | 2, dva, sedmi, 1.23.2015 |
| :time | Time | 23:30:01 |
| :PUNC | Separator or punctuation | , . |
| :PN | Proper noun | Aleksandar, Jelenu, Gorenje, Zagreb |

Czech

Table A1.4 Part-of-Speech Tags for Czech

| Part-of-Speech Tag | Description | Examples |
|--------------------|--------------------------|---|
| :A | Adjective | duchovní, celý, všechny, čertvíjaký, která, jakém, žádněj |
| :ADV | Adverb | například, dál, zároveň, někam, ne |
| :CONJ | Conjunction | a, nebo |
| :INTJ | Interjection | ahoj, fuj |
| :N | Noun | autorů, lidem |
| :NUM | Spelled out number | tři, dvoje, šestatřicáté |
| :PPOS | Preposition | v, z |
| :PRO | Pronoun | kdo, sobě, nás, tomto, tím, nikoho, nic, její, mou |
| :V | Verb | nebyl, jdou |
| :sep | Separator or punctuation | . , : |
| :PN | Proper noun | Pavel, Valenta, Chotěbořským |

| | | |
|-------|-------------------------|--|
| :inc | Unknown or foreign word | mp3, larger |
| :time | Time | 23:30:01 |
| :url | URL | www.sas.com , http://www.sas.com |

Danish

Table A1.5 Part-of-Speech Tags for Danish

| Part-of-Speech Tag | Description | Example |
|--------------------|--------------------------|---|
| :A | Adjective | socialest, udartendere |
| :ADV | Adverb | sydsydøst |
| :CONJ | Conjunction | Såsom |
| :INTJ | Interjection | joh, pøj |
| :N | Noun | thyboernes, centerer, DVS, FL, ibm, netscape, tirsdag |
| :NUM | Spelled out number | tyvefem, tredive |
| :PN | Proper noun | Egholm, Franck, Carlos, Mallorca, Groth, Leth, Renault, Corel |
| :PPOS | Preposition | fra, trods |
| :PRO | Pronoun | dens, hans, jerselv, sigselv |
| :V | Verb | opofre, læsende, anvender, bliver, tredjebehandlet, læste, læse, tilvirk, bemyndiges, fuldkommengøredes |
| :date | Date | 23-12-2012, 12/12/2012 |
| :time | Time | :23:50, 09:23 |
| :digit | Digit | 2012, 12.23 |
| :url | Internet address | http://www.sas.com |
| :sep | Separator or punctuation | . , ; |
| :inc | Unknown word | bl, erne |

Dutch**Table A1.6** Part-of-Speech Tags for Dutch

| Part-of-Speech Tag | Definition | Examples |
|--------------------|--------------------------|---|
| :A | Adjective | betrouwbaar, gelukkig, mooi |
| :ADV | Adverb | eenmaal, hier, nu |
| :CONJ | Conjunction | als, doch, hoe |
| :DET | Determiner | de, der, een, ten, ter |
| :digit | Number | 21 |
| :NUM | Numeral | acht, elf, miljard, duizend |
| :inc | Unknown word | xrxx |
| :N | Noun | geluk, schoonheid, kg, zgn, anti, hoofde, tijde, voordele |
| :PN | Proper noun | Amerika, Nederland |
| :PPOS | Preposition | met, per, te, van |
| :PRO | Pronoun | alles, beide, hetgeen |
| :sep | Separator or punctuation | , |
| :url | URL | http://www.sas.com |
| :V | Verb | helpt, vernieuwt, helpen, vernieuwen, helpende, vernieuwende, geholpen, vernieuwd |

English

| Part-of-Speech Tag | Description | Examples |
|--------------------|-------------|---------------------------------------|
| :A | Adjective | luckier, worse, mellowest, merriest |
| :ADV | Adverb | lyrically, physically, luckier, worse |

| | | |
|--------|---------------------------|---|
| :CONJ | Conjunction | when, yet, how, when, whereby |
| :date | Date | 04/03/2012 |
| :digit | Sequence of Numbers | 2345, 234.22, 21/234 |
| :DET | Determiner | the, an, every, our, his, my, such, all |
| :inc | Unknown word | slaster, lijer |
| :INTJ | Interjection | hah, hello |
| :N | Noun | love, sheep, shoes, etc., Ms, cm, facto, klieg, modus |
| :NUM | Number | twenty, hundred |
| :PN | Proper noun | SAS, Cary, Goodnight |
| :PPOS | Preposition | on, under, across, after, except, away, forward, in, ex, multi |
| :PRO | Pronoun | he, one, somebody, me, myself, oneself, yours, hers, which, whatever, whose, whoever |
| :sep | Separator and punctuation | ;, / |
| :time | Time | 7AM, 10:00 |
| :url | Filenames, pathnames, URL | A:/mydir/file.txt, www.sas.com |
| :V | Verb | be, do, have, am, can, should, will, goes, sees, is, does, doing, having, climbing, been, had, was, were, did, have, dashed, factored, went |

Farsi

Table A1.7 Part-of-Speech Tags in Farsi

| Part-of-Speech Tag | Description | Examples |
|--------------------|-------------|---------------|
| :A | Adjective | خوشگل, خوشحال |

| | | |
|----------|--|-------------------------|
| :Acomp | Comparative adjective | خوشگل‌تر, خوشحال‌تر |
| :Asup | Superlative adjective | خوشگل‌ترین, خوشحال‌ترین |
| :Appl | Participle used as adjective | آسایانیده, آباتانده |
| :ADV | Adverb | هنوز, آنکه, ابتدائاً: |
| :CLASS | Classifier | باب, تخته, رأس |
| :CONJ | Conjunction | اگر, تااینکه |
| :DET | Determiner | اون, این |
| :INTJ | Interjection | آه, آفرین, ای |
| :N | Noun | آذوقه, آرنج, چشم |
| :Npl | Plural noun | آرنج‌ها, چشم‌ها |
| :NUM | Numeral | دو, صد, میلیون |
| :NUMord | Ordinal numeral | دومین, سوم, صدمین |
| :PN | Proper noun | اسرائیل, اتوسا |
| :PPOS | Preposition | از, الا, چون |
| :PRO | Pronoun | ن, او, شما |
| :PUNC | Punctuation or symbol | “ % ؟ ” |
| :Vinf | Infinitive (usage similar to English gerund) | خواندن, خوردن |
| :V | Verb | بخوان, بخوانم, خواندم |
| :ASCII | ASCII characters and digits | happy, 2017, love123 |
| :DEFAULT | Unknown word | بخوانبخوان |

Finnish

Table A1.8 Part-of-Speech Tags for Finnish

| Part-of-Speech Tag | Definition | Examples |
|--------------------|------------|------------------|
| :A | Adjective | loistava, korkea |

| | | |
|--------|--------------------------|--|
| :ADV | Adverb | ohitse, juuri |
| :CONJ | Conjunction | ja, vaan, ellej, jotta |
| :date | Date | 2001-12-02 |
| :digit | Number | 1234, 7 |
| :inc | Unknown word | auttonkkan, eggs |
| :N | Noun | siltoineen, postiksi |
| :PN | Proper noun | Pertti, Fazer |
| :PPOS | Preposition | pitkin, kanssa |
| :PRO | Pronoun | noihin, muussa, ketkä |
| :sep | Separator or punctuation | ; / + |
| :time | Time | 12:00:00, 7PM |
| :url | URL | http://www.sas.com |
| :V | Verb | heilahtamassa, heilauttaen, olla, kinko, pas, lähennemme, kumarrettava, jaettu, meditoitpa, ihastele, omistautuisi, pakkaa |

French

Table A1.9 Part-of-Speech Tags for French

| Part-of-Speech Tag | Definition | Examples |
|--------------------|--------------|---|
| :A | Adjective | comparable, compassionnelle, intraduisibles |
| :ADV | Adverb | plutôt, individuellement |
| :CONJ | Conjunction | et, ou, lorsque, puisque |
| :DET | Determiner | sa, tes, ce |
| :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 |
| :AFX | Affix | anglo, éco |
| :PPOS | Preposition | jusque, aux, du |
| :PRO | Pronoun | lui |
| :sep | Separator or punctuation | , . ! |
| :url | URL | http://www.sas.com |
| :V | Verb | vais, obligez, travaillées, traduire, tramant |
| :PTCL | Particle | vitae, ab |

German

Table A1.10 Part-of-Speech Tags for German

| Part-of-Speech Tag | Definition | Examples |
|--------------------|----------------------|--------------------------------|
| :A | Adjective | zuverlässig |
| :ADV | Adverb | gern, sehr |
| :CONJ | Conjunction | und, oder |
| :DET | Determiner | eine, manch |
| :digit | Number | 21 |
| :NUM | Numeral | fünf, zwölf |
| :EMP | Emphatic/intensifier | ganz |
| :inc | Unknown word | xrxx |
| :N | Noun | Schönheit, Zuverlässigkeit |
| :PN | Proper noun | Mozart, Nirvanas, Niederlanden |
| :PPOS | Preposition | kontra, ober, lob |

| | | |
|------|--------------------------|---|
| :PRO | Pronoun | er, sie, der, heraus |
| :sep | Separator or punctuation | , |
| :url | URL | http://www.sas.com |
| :V | Verb | ging, half, gehen, helfen |

Greek

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

| Part-of-Speech Tag | Description | Examples |
|--------------------|--------------------------|--|
| :A | Adjective | ενορμητικός, άβαθος |
| :ADV | Adverb | πολύ, επίσης |
| :CONJ | Conjunction | και, αλλά |
| :DET | Determiner | ένας, ο |
| :INTJ | Interjection | χαίρε, όπα |
| :N | Noun | μήλο, δέντρο |
| :PTCL | Particle | πάρα |
| :PPOS | Preposition | άχρι, διά |
| :PRO | Pronoun | εσύ, αυτός |
| :V | Verb | παίσαμε, παίνεψε, παίξει, παίζαμε, παίζουμε, παίζοντας, παίρνοντάς, κατασκευαστώ, έλα |
| :url | URL | http://www.sas.com |
| :date | Date | 2015-12 |
| :digit | Number | 1, 20 |
| :sep | Separator or punctuation | . , » |
| :inc | Unknown word | χλμ |
| :time | Time | 23:59 |

| | | |
|-----|-------------|-----------|
| :PN | Proper noun | Μάντισσες |
|-----|-------------|-----------|

Hebrew

Table A1.12 Part-of-Speech Tags for Hebrew

| Part-of-Speech | Descriptions | Examples |
|----------------|--------------------------|---|
| :A | Adjective | יפה, אדיר |
| :ADV | Adverb | באמת, בבטחה |
| :CONJ | Conjunction | או, בגלל |
| :INTJ | Interjection | אוף, אהה |
| :N | Noun | רחוב, ברחוב, אבזור, אבטחה |
| :PN | Proper noun | ישראל, אבוג'ה, אדוארד |
| :PPOS | Preposition | אודות, אצל |
| :PRO | Pronoun | אנחנו, באתה, ה"הן, מהיכן |
| :NUM | Quantifier | אחד, ביליון, שתיהן |
| :V | Verb | שמח, אבטח, אהבו |
| :date | Date | 12/31/2016, 2016-12-31 |
| :digit | Number | 100, 6,666, 6.000 |
| :inc | Unknown word | happy, happy123, בוויטנאם |
| :sep | Separator or punctuation | ., ! - |
| :time | Time | 14:30:30 |
| :url | :URL | http://www.sas.com |

Hindi

Table A1.13 Part-of-Speech Tags for Hindi

| Part-of-Speech Tag | Definition | Examples |
|--------------------|------------|----------|
|--------------------|------------|----------|

| | | |
|--------|--------------------------|------------------|
| :A | Adjective | ज्ञात, ज्ञानी |
| :PRO | Pronoun | तेरा, मेरा |
| :N | Noun | मेयर, मैग्नोलिया |
| :ADV | Adverb | यथायोग्य, यथोचित |
| :CONJ | Conjunction | यदि, यद्यपि |
| :DET | Determiner | ऐसा, इसी |
| :INTJ | Interjection | आह, अहा |
| :NUM | Number | अस्सी, अड़तालीस |
| :PN | Proper noun | अग्नीवो |
| :PPOS | Particles | का, का |
| :V | Verb | खरीदना, गुजर |
| :PUNC | Separator or punctuation | !, |
| :sep | Separator or punctuation | ,.) |
| :inc | Unknown words | आिद, २२५ |
| :digit | Number | 0, 3 |

Indonesian

Table A1.14 Part-of-Speech Tags for Indonesian

| Part-of-Speech Tag | Definition | Examples |
|--------------------|--------------|-----------------------------|
| :A | Adjective | lonjong, menjengkelkan |
| :N | Noun | kosmologiku, lotengnya, dpa |
| :ADV | Adverb | mingguan, perlahan |
| :CONJ | Conjunction | sambil, biarpun |
| :V | Verb | biarkanlah, membuntutiku |
| :DET | Determiners | sebuah |
| :NUM | number words | empat, delapan |

| | | |
|--------|--|--------------------|
| :INTJ | Interjections | hai, hoi |
| :PRO | Pronoun | dikau, engkau |
| :PN | Proper noun | irlandia, filipina |
| :PPOS | Phrasal; the word can be combined with another word to form a phrase | sebiru, secantik |
| :sep | Separator or punctuation | "(, |
| :inc | Unknown words | jpg, png |
| :digit | Number | 22, 490 |
| :url | URL | www.jakarta.go.id |
| :date | Date | 12/31/2016 |

Italian

Table A1.15 Part-of-Speech Tags for Italian

| Part-of-Speech Tag | Definition | Examples |
|--------------------|--------------------------|--|
| :A | Adjective | affidabile, bellissimo, felice, felicemente, rapidamente |
| :CONJ | Conjunction | ma, oppure, sebbene |
| :DET | Determiner | il, la, uno |
| :digit | Number | 21 |
| :INTJ | Interjection | ah, ahimè |
| :inc | Unknown word | Xrxx |
| :N | Noun | affidabilità, bellezza, felicità |
| :PN | Proper noun | Roma, Italia |
| :PRO | Pronoun | io, ne |
| :PPOS | Preposition | con, in, per, anti, ri, anza, issimo |
| :sep | Separator or punctuation | , |

| | | |
|------|------|---|
| :url | URL | http://www.sas.com |
| :V | Verb | andare, andando, andasse, andato |

Japanese

Table A1.16 Part-of-Speech Tags for Japanese

| Part-of-Speech Tag | Description | Examples |
|--------------------|------------------------------|---------------|
| :AJ | Adjective | 長い, 忙しい, 便利だ |
| :AV | Adverb | いかが, やはり |
| :AVC | Adverbs of form or condition | 直に, ぐっすり |
| :AVD | Adverb of degree | とっても, 大して |
| :AVE | Adverb of evaluation | たまたま, 無論 |
| :AVF | Adverb of frequency | あくまで, しばしば |
| :AVO | Adverb of opinion | いわば, 概して |
| :AVQ | Adverb of quantity | 大方, いくら |
| :AVS | Adverb of statement | いかに, あたかも |
| :AVT | Adverb of tense or aspect | 急遽, 直ぐ |
| :AX | Auxiliary verbs | べきだ, らしい, ようだ |
| :CN | Conjunction | 並びに, 但し, だけど |
| :CP | Copula | だ, なんだ |
| :DA | Adverbial demonstrative | こう, そう, あのよう |
| :DM | Prenominal demonstrative | この, あの, そんな |
| :DN | Pronoun | あれ, こちら, あそこ |
| :MD | Prenominal modifier | 小さな, 主たる, 色んな |
| :IT | Interjection | あれれ, あ~, ええと |
| :NA | Adverbial noun | おおむね, なにぶん |

| | | |
|-------|--|------------------|
| :NC | Common noun | 風, 学校, 雑誌 |
| :NK | Content noun | の, もの, こと |
| :NT | Noun of time | 長年, 夏, 先月 |
| :NV | Verbal noun | 請求, 弁解, 勉強 |
| :NP | Proper noun | W T O 繊維協定, 米州 |
| :NH | Proper noun of Person | 中川秀直, 中川浩明, 中川勝 |
| :NHM | Proper noun of Given name | 奈江子, 太郎, 那恵子 |
| :NHS | Proper noun of Family name | 鈴木, 佐藤, 田中 |
| :NPO | Proper noun of Organization | 米軍, 米国, 米国際貿易委員会 |
| :NL | Proper noun of Place | 米国, 越南, 奈央島 |
| :NN | Numeral | 千, 零, 6 |
| :PC | Particles of case marker | を, で, の, へ |
| :PE | Particles that appear at the end of the sentence | つけ, な, なあ |
| :PN | Particles that combine nominals | ないし, ないしは, 並びに |
| :PP | Particles that combine clauses | ながら, なら, のに |
| :PQ | Particles of quotation | て, と, っと |
| :PS | Particles that mean <i>only</i> or <i>too</i> | も, のみ, くらい |
| :PRJ1 | Prefixes to i-adjective | か, こ, 真 |
| :PRJ2 | Prefixes to na-adjective | 無, 不, 非 |
| :PRN | Prefixes to nominals | 高, 前, 全 |
| :PRV | Prefixes to predicates | 相, 猛, 最 |
| :SJN | Suffixes to nouns and configure adjectives | っぽい, くない |
| :SJV | Suffixes to verbs and configure adjectives | たい, づらい |

| | | |
|-----------|---|----------------------|
| :SNA | Suffixes to adjectives and configure nouns | さ |
| :SNC | Suffixes to classifiers and configure nouns | せんち, ページ |
| :SNN | Suffixes to nouns | っ子, 中, 所 |
| :SNV | Suffixes to verbs and configure nouns | かた, っぷり |
| :SV | Suffixes to verbs | せる, れる, 上げる |
| :V1 | Ichidan Verb | 治せる, 泣ける, 叫べる |
| :V5 | Godan Verb | 直す, 長びく, 産む |
| :VK | Kuru Verb | 来る |
| :VS1 | Suru Verb | する |
| :VS2 | Suru Verb d | 賀する, 刑する, 御する |
| :VSN | Suru Verb | きりきり, 毅然と |
| :VZ | Zuru verb | 準ずる, 同ずる |
| :SC | Special category-comma | 、 , |
| :SCP | Special category-closed parentheses |) 》] |
| :SOP | Special category-opened parentheses | (《 [|
| :SK | Special category-other symbols | ? ... ~ |
| :SP | Special category-period | 。 . |
| :SS | Special category-space | |
| :digit | Number | 1.0, 10 |
| :sep | Separator or punctuation | . , |
| :KATAKANA | Unknown word in katakana | ポータブルオプション, オブザベーション |
| :HIRAGANA | Unknown word in hirakana | きんぼうげ |
| :UNKNOWN | Unknown word | 嘘, 甦 |

| :ASCII | English word | Display, M o m e n t e |
|--------|--------------|------------------------|
|--------|--------------|------------------------|

To use Japanese POS tags in LITI rules, you need to add the Form type after the POS tags. For the POS tags of nominals, add '|ROOT' after the POS tags. E.g. 'NC|ROOT', 'DN|ROOT', 'CN|ROOT'. For the POS tags of predicates, add the conjugation forms listed in the table below. E.g. 'AJ|CONJ', 'V1|COND'.

| Form Type | Japanese description | English description | Examples |
|-------------|----------------------|--------------------------------|----------------------|
| ROOT | 体言基本形 | Basic form of nominals | お花, 手 |
| BS | 用言基本形 | basic form of predicates | 読む, 速い |
| BSDEA | デアル列基本形 | dearu basic conjunctive | 静かである |
| BSWR | デス列基本形 | desu basic | 静かです |
| COND | 文語基本形 | written basic form | あいさつす |
| CONDDEA | デアル列条件形 | basic/euphony conditional | 読めば, 読みや, 速ければ, 速けりや |
| CONDDEATA | デアル列タ系条件形 | dearu/ta conditional | 静かであれば |
| CONDDESTA | デス列タ系条件形 | desu/ta conditional | 静かであったら |
| CONDTA | タ系条件形 | ta conditional | 静かでしたら |
| CONDWR | 文語条件形 | written conditional | 読んだら, 速かったら |
| CONJ | 基本連用形 | basic conjunctive | 読め |
| CONJDEA | デアル列基本連用形 | dearu conjunctive-tari form | 読み(ます), 速く, 静かに |
| CONJDEATA | デアル列タ系連用テ形 | dearu/ta conjunctive-te form | 静かであり |
| CONJDEATARI | デアル列タ系連用タリ形 | dearu/ta conjunctive-tari form | 静かであったり |
| CONJDESTARI | デス列タ系連用タリ形 | desu/ta conjunctive-tari form | 静かでしたり |
| CONJDESTE | デス列タ系連用テ形 | desu/ta conjunctive-te form | 静かでしたて |

| | | | |
|----------|-----------|-------------------------------------|------------------|
| CONJTARI | タ系連用タリ形 | ta conjunctive -tari form | 書いたり, 速かったり |
| CONJTE | タ系連用テ形 | ta conjunctive -te form | 書いて, 速くて |
| CONJWR | 文語連用形 | written conjunctive | あいなう, あかう |
| DEATA | デアル列タ形 | dearu/ta form (plain past tense) | 静かであった |
| DESTA | デス列タ形 | desu/ta form | 静かでした |
| IMP | 命令形 | imperative | 読め, 速かれ, 静からし |
| IMPDEA | デアル列命令形 | dearu imperative | であれ, 静かであれ |
| IMPWR | 文語命令形 | written imperative | あいさつせよ |
| INT | 意志形 | intention form | 読もう |
| IPE | 未然形 | Imperfective | 読ま(ない) |
| IPEDEAWR | デアル列文語未然形 | written -dearu imperfective | べきであら |
| IPEWR | 文語未然形 | written imperfective | 速から(ず) |
| KANO | 可能形 | form that attaches to can words | 太れ, 失え |
| PASS | 受身形 | form that attaches to passive forms | 失わ |
| PERF | 完了形 | form that attaches to perfective | 失効し |
| PNOM | ダ列基本連体形 | basic pronominal | 速き(こと), 静かな, 上等の |
| PNOMWR | 文語連体形 | written pronominal | 失き, 好きずきき |
| PSU | 基本推量形 | (-da) basic presumptive | 速かろう, 静かだろう |
| PSUDEA | デアル列基本推量形 | dearu presumptive | 好きであろう |
| PSUDEATA | デアル列タ系推量形 | dearu/ta presumptive | 静かであったろう, であったろう |
| PSUDES | デス列基本推量形 | desu presumptive | 好きでしょう |

| | | | |
|----------|----------|----------------------------------|-----------------------|
| PSUDESTA | デス列タ系推量形 | desu/ta presumptive | 好きでしたろう |
| PSUTA | タ系推量形 | ta presumptive | 読んだろう, 速かったろう, 静かだったら |
| SHIEKI | 使役形 | form that attaches to causatives | あいさつさ |
| TA | タ形 | ta form (plain past tense) | 読んだ, 速かった, 静かだった |

Korean

Table A1.17 Part-of-Speech Tags for Korean

| Part-of-Speech Tag | Description | Examples |
|--------------------|---------------------------------|----------------|
| :AD | Adverb | 매우, 정말, 빨리 |
| :AJ | Adjective | 예쁘다, 귀엽다, 차분하다 |
| :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, 읍 |
| :GAT | Time grammatical affix | 겠, 었, 었었 |
| :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 | Number | 하나, 둘, 셋 |

| | | |
|----------|--------------|---|
| :PF | Prefix | 제-, 헛-, 명- |
| :PN | Prenoun | 각, 첫, 기초적 |
| :PR | Pronoun | 이것, 언제, 이분 |
| :PUNC | Punctuation | .?!() |
| :SF | Suffix | -꾼, 꾸러기, -감 |
| :VB | Verb | 웃다, 뛰다, 날다 |
| :ASCII | English Word | Korean, iPhone, SK |
| :DATE | Date | 2015-04-28, 20150428 |
| :DEFAULT | Unknown word | 하페즈, 샤리프, 쿠레쉬 |
| :TIME | Time | 23:59:59 |
| :URL | URL | http://www.sas.com |

Norwegian

Table A1.18 Part-of-Speech Tags for Norwegian

| Part-of-Speech Tag | Description | Examples |
|--------------------|------------------------|---|
| :A | Adjective | leket |
| :ADV | Adverbr | alltid, framover |
| :CONJ | Conjunction | som |
| :N | Noun | anordningen, tydeets, mfl, mht, tusen, seks, sms |
| :PN | Proper noun | Egholm, Puccini, Tertnes, Høyem, Lundberg, Braathens, ruskursus, ørknen |
| :PPOS | Preposition | fra |
| ng:DET | Preposition+determiner | idette, idenne |
| :PRO | Pronoun | jeg, det, dens, sjølve |
| :V | Verb | å, trikes, brukende, fyltes, brukte, krislende, brukt, gasjerer, slepp |

| | | |
|-------|-------------|---|
| :date | Date | 12/23/2012, 23/12/2012 |
| :url | URL | http://www.sas.com |
| :NUM | Number | 12, 23, 23.4 |
| :PUNC | Punctuation | , . ! |

Polish

Table A1.19 Part-of-Speech Tags for Polish

| Part-of-Speech Tag | Description | Examples |
|--------------------|-----------------------------|---|
| :A | Adjective | własne, każda, głównych |
| :ABBREV | Abbreviation | ang., tzw. |
| :ADV | Adverb | więcej, tylko |
| :CONJ | Conjunction | i, czyli |
| :INTER | Interjection | ej, fuj, amen |
| :N | Noun | teorie, miejscach, Wojciech |
| :NUM | Numeral | siedmiu, tysięcy |
| :PART | Particle | też |
| :PREP | Preposition | za, z, na, do |
| :PRON | Pronoun | się, sami, go, tobie |
| :V | Verb | wiedzieć, dotarł |
| :date | Date | :01/01/2012, 12/12/17, 12-23-2001, 23-12-01 |
| :time | Time | 23:30:01 |
| :digit | Number | 12, -5, 23,45 |
| :sep | Separator or punctuation | . , - |
| :url | URL | http://www.sas.com |
| :PN | Unknown/foreign proper noun | Achitophel, Trzciański, LP-vinyl |

| | | |
|------|----------------------|-----------------|
| :inc | Unknown/foreign word | sapiens, ela544 |
|------|----------------------|-----------------|

Portuguese

Table A1.20 Part-of-Speech Tags for Portuguese

| Part-of-Speech Tag | Definition | Examples |
|--------------------|--------------------------|---|
| :A | Adjective | confiável, feliz |
| :ADV | Adverb | belamente, felizmente |
| :CONJ | Conjunction | e, que |
| :DET | Determiner | alguns, cada, os, dessas, dum |
| :digit | Number | 21 |
| :NUM | Numeral | bilionésimo, cinco |
| :inc | Unknown word | xrxx |
| :INTJ | Interjection | caramba, eh |
| :N | Noun | beleza, felicidade, cf, ibid |
| :PN | Proper noun | Brasil, Portugal |
| :PPOS | Preposition | com, de, em, anti, circum |
| :PRO | Pronoun | me, nós, quem |
| :sep | Separator or punctuation | , |
| :url | URL | http://www.sas.com |
| :V | Verb | garanto, garantir, garantindo, garantido |

Russian

Table A1.21 Part-of-Speech Tags for Russian

| Part-of-Speech Tag | Definition | Examples |
|--------------------|------------|----------|
|--------------------|------------|----------|

| | | |
|--------|--------------------------|---|
| :A | Adjective | духовитый, красивая, лучших, который, баскервиллей |
| :ADV | Adverb | дальше, сколько-нибудь, где, сколько, почём |
| :conj | Conjunction | если, и |
| :digit | Number | 123, 12.3, 12.3.2003, 12/3/2013 |
| :inc | Unknown word | геминг, analytics |
| :INTJ | Interjection | ах |
| :N | Noun | велосипед, история, малолетство, др, км, маргини, маэстро |
| :PN | Proper noun | Шевроле, Айдахо, Миа, Роханский, Сашина, Свердловск, Мария, Давыдович |
| :NUM | Number | один, десятью |
| :PTCL | Particle | бы, же |
| :PPOS | Preposition | до, вроде |
| :PRO | Pronoun | я, её, всяко |
| :sep | Separator or punctuation | , . ! |
| :url | URL | http://www.sas.com |
| :V | Verb | менять, нажимает, кладите, плавала, адаптировав, вальсируя |

Slovak

Table A1.22 Part-of-Speech Tags for Slovak

| Part-of-Speech Tag | Description | Examples |
|--------------------|-------------|-----------------------|
| :A | Adjective | všeobecné , verejnej |
| :ADV | Adverb | pravidelne, vyslovene |

| | | |
|--------|--------------------------|---|
| :CONJ | Conjunction | ak , iba |
| :INTJ | Interjection | oj, stop |
| :N | Noun | doručení, partnerov, ul, Dr |
| :NUM | Numeral | štyritisíc, prvom |
| :PTCL | Particle | by, tiež |
| :PPOS | Preposition | o, v, pre |
| :PRO | Pronoun | si, Vám, vaše, jeho, uňho, ktoré, akékoľvek |
| :V | Verb | prinášame, budú, nespráva, využívať, nezaostávať, prešli, nemali, pozrite |
| :digit | Number | 1.4, -10, +421 |
| :sep | Separator or punctuation | . , / |
| :PN | Proper noun | Oetker, KEPe |
| :inc | Unknown or foreign word | newslettri |
| :url | URL or email | http://www.sas.com, info@slovakrail.sk |
| :time | Time | 23:30:00 |
| :date | Date | 23/12/2012, 23-12-2012 |

Slovene

| Part-of-Speech Tag | Description | Example |
|--------------------|--------------|--------------------|
| :A | Adjective | prvi, črna |
| :ADV | Adverb | hmalu, daleč |
| :CONJ | Conjunction | ali, in |
| :INTJ | Interjection | bravo, ah |
| :N | Noun | dni, dogodka, itd. |
| :NUM | Numeral | dva, šest |

| | | |
|--------|--------------------------|--|
| :digit | Number | 20.3, 123 |
| :PTCL | Particle | pa, spet |
| :PPOS | Preposition | v, za |
| :PRO | Pronoun | te, mi, vsak, kdo |
| :V | Verb | sta, uporablja, suspendirali, pozabite |
| :sep | Separator or punctuation | . : , « |
| :Prop | Proper noun | Maribor, Roglič |
| :date | Date | 23/12/2012, 23-12-2012 |
| :time | Time | 23:30:00 |
| :url | URL | http://www.sas.com, info@sas.com |

Spanish

Table A1.23 Part-of-Speech Tags for Spanish

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

| | | |
|------|--------------------------|---|
| :PRO | Pronoun | alguien, ellos, me, el, las |
| :sep | Separator or punctuation | , |
| :url | URL | http://www.sas.com |
| :V | Verb | ayudan, ayudar, ayudando, ayudado |

Swedish

Table A1.24 Part-of-Speech Tags for Swedish

| Part-of-Speech Tag | Definition | Examples |
|--------------------|--------------|--|
| :A | Adjective | fört |
| :ADV | Adverb | väl |
| :CONJ | Conjunction | samt |
| :DET | Determiner | Ens, somlig |
| :NUM | Number | två |
| :INTJ | Interjection | hej |
| :N | Noun | bok, morse, st. |
| :PN | Proper noun | Øsel, Tove, Östmark, Viklund, Toshiba |
| :PPOS | Preposition | till |
| :PRO | Pronoun | honom, du |
| :V | Verb | varit, varande, varats, sedd, ses, såg, sågs |

Tagalog

| Part-of-Speech Tag | Description | Examples |
|--------------------|-------------|-----------------|
| :A | Adjective | abalang, alisto |
| :ADV | Adverb | biglang, bakit |

| | | |
|--------|--------------------------|-------------------------|
| :CONJ | Conjunction | at, yamang |
| :DET | Determiner | ni, nina |
| :INTJ | Interjection | hoy |
| :N | Noun | pusa, yarda |
| :NUM | Number | dalawa, walumpu |
| :PN | Proper Noun | Asya, Espanya |
| :PPOS | Preposition | sa, dahil |
| :PRO | Pronoun | akin, amin, iyo |
| :PTCL | Particle | ay |
| :V | Verb | kainin, tayuan, uminom |
| :url | URL | www.sas.com |
| :date | Date | 2015-12 |
| :digit | Number | 1, 20 |
| :sep | Separator or punctuation | . , » |
| :inc | Unknown Word | possibilities, tropical |
| :time | Time | 23:59:59 |

Thai

Table A1.25 Part-of-Speech Tags for Thai

| Part-of-Speech Tag | Description | Examples |
|--------------------|-----------------|----------------------|
| :ADJ | Adjective | กตัญญู, กตัญญูกตเวที |
| :ADV | Adverb | กระงอกระแงง, กระดืบๆ |
| :AUXVERB | Auxiliary verbs | ควรจะ, ต้อง |
| :CLAS | Classifiers | กก., กม. |
| :CONJ | Conjunction | ก่อน, จน |
| :DET | Determiner | ทั้ง, ทุก |

| | | |
|-----------|---|-------------------------------|
| :END | Particle used at the end of a question, command or entreaty | ละ, เหนือ |
| :INTERJ | Interjection | ชะชะ, ดูกร |
| :NEG | Negation | มิใช่, ไม่ |
| :NOUN | Noun | กงพัด, กฎหมายบ้านเมือง |
| :NUMBER | Number | สอง, เก้า |
| :PREF | Prefix | ปรา, อน |
| :PREP | Preposition | กว่า, ก่อนหน้า |
| :PRON | Pronoun | คนอื่นๆ, คนใด |
| :PROPLC | Proper noun, location | กมลลา, กรีซ |
| :PROPMISC | Proper noun, others | กุชชี, คลินิกซ์ |
| :PROPNAME | Proper noun, person names | กบิลกาญจน์, กัตัญญตานนท์ |
| :PROPORG | Proper noun, organizations | กรุงเทพธุรกิจ, กระทรวงมหาดไทย |
| :PUNC | Separator or punctuation | " (...) |
| :SUFF | Suffix | ลี, เอย |
| :VERB | Verb | กทรรูป, กรมเกรียม |
| :DEFAULT | Unknown words | Josephson, microbridge |

Turkish

Table A1.26 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 |
| :NUM | Numeral | dokuz, onbir, beri |
| :PN | Proper noun | Ayşe, Türkçe |
| :PRO | Pronoun | bunlar, kendi, onlar, sen, çok |
| :sep | Separator or punctuation | ! , , |
| :time | Time | 12:30:00 |
| :url | URL | sas.com, www.sas.com, http://www.sas.com |
| :V | Verb | diye, bilir, bilmek, bilse, bilmiş, bildi, bilmeli, biliyor, bilmekte, bil |

Vietnamese

Table A1.27 Part-of-Speech Tags for Vietnamese

| Part-of-Speech Tag | Description | Examples |
|--------------------|--------------|---------------------------|
| :A | Adjective | an toàn, bận rộn, lịch sự |
| :ABBREV | Abbreviation | APEC, ANĐT, ĐTN |
| :Adv | Adverb | bỗng chốc, chưa chừng |
| :Aux | Particle | chính |
| :C | Conjugation | dù rằng, hoặc là |
| :F | Foreign word | cà-rem, Ampe, ăng ten |
| :Int | Interjection | hỡi, ái chà, ô hay |
| :N | Noun | áo quần, cừu, cương vị |
| :Num | Numeral | 2007, bảy, mười n |

| | | |
|----------|------------------------|--------------------------|
| :PreDet | Determiner | một số |
| :Prep | Preposition | cho, vào |
| :PN | Proper noun | Việt Nam, Trung Quốc |
| :Pro | Pronoun | tôi, chúng mày, chúng nó |
| :PUNC | Punctuation or symbol | ! : () @ |
| :RelPro | Relative pronoun | ai đấy |
| :V | Verb | ngủ, học, làm việc |
| :DEFAULT | Unrecognized character | , ... |

Appendix 2

Pre-Defined Concept Priorities (for Languages Other Than English)

| | |
|---|------------|
| Using Priority Values in Predefined Concepts | 103 |
| Priority Values for Predefined Concepts | 104 |
| Arabic | 104 |
| Chinese | 104 |
| Croatian | 105 |
| Czech | 105 |
| Danish | 106 |
| Dutch | 106 |
| Farsi | 107 |
| Finnish | 107 |
| French | 108 |
| German | 108 |
| Greek | 109 |
| Hebrew | 109 |
| Hindi | 110 |
| Indonesian | 110 |
| Italian | 111 |
| Japanese | 111 |
| Korean | 112 |
| Norwegian | 112 |
| Polish | 112 |
| Portuguese | 113 |
| Russian | 113 |
| Slovak | 114 |
| Slovene | 114 |
| Spanish | 115 |
| Swedish | 115 |
| Tagalog | 116 |
| Thai | 116 |
| Turkish | 117 |
| Vietnamese | 117 |

Using Priority Values in Predefined Concepts

To accurately set priorities for matching custom concepts in your language, see the topic [“Priority Values for Predefined Concepts”](#). For information about setting priorities, see

“Which Rule Type Should I Use?” on page 45. For priority values in English, see “Concepts” on page 5.

Note: Use the highest priority value per language to ensure that there are no conflicts with custom concepts during document processing. The highest priority value for each language is marked in the tables in the following section with a footnote.

Priority Values for Predefined Concepts

Arabic

Table A2.1 Predefined Concept Priorities for Arabic

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMoney | 18 |
| nlpNounGroup | 15 |
| nlpOrganization | 20 |
| nlpPercent | 18 |
| nlpPerson | 20 |
| nlpPlace* | 25* |
| nlpTime | 18 |

* Highest value for this language.

Chinese

The default value of 10 is used for all of the predefined concepts listed below.

Table A2.2 Predefined Concept Priorities for Chinese

| Predefined Concept |
|--------------------|
| nlpDate |
| nlpMoney |
| nlpOrganization |
| nlpPercent |

nlpPerson

nlpPlace

nlpTime

Croatian

Table A2.3 Predefined Concept Priorities for Croatian

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 10 |
| nlpMeasure | 10 |
| nlpMoney | 10 |
| nlpNounGroup | 10 |
| nlpOrganization | 10 |
| nlpPercent | 10 |
| nlpPerson | 11 |
| nlpPlace* | 12* |
| nlpTime | 10 |

* Highest value for this language.

Czech

Table A2.4 Predefined Concept Priorities for Czech

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate* | 10* |
| nlpMoney* | 10* |
| nlpNounGroup | 9 |
| nlpOrganization* | 10* |
| nlpPercent* | 10* |

| | |
|------------|-----|
| nlpPerson* | 10* |
| nlpPlace* | 10* |
| nlpTime* | 10* |

* Highest value for this language.

Danish

Table A2.5 Predefined Concept Priorities for Danish

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpNounGroup | 15 |
| nlpOrganization* | 20* |
| nlpPerson* | 20* |
| nlpPlace* | 20* |

* Highest value for this language.

Dutch

Table A2.6 Predefined Concept Priorities for Dutch

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMoney | 18 |
| nlpNounGroup | 15 |
| nlpOrganization* | 20* |
| nlpPercent | 18 |
| nlpPerson* | 20* |
| nlpPlace* | 20* |
| nlpTime | 18 |

* Highest value for this language.

Farsi

For Farsi, there are no specific priority values for predefined concepts. The default value of 10 is used for all of the predefined concepts listed below.

Table A2.7 *Predefined Concept Priorities for Farsi*

| Predefined Concept |
|--------------------|
| nlpDate |
| nlpMoney |
| nlpOrganization |
| nlpPercent |
| nlpPerson* |
| PERSON |
| ORGANIZATION |

Finnish**Table A2.8** *Predefined Concept Priorities for Finnish*

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 10 |
| nlpMoney | 10 |
| nlpNounGroup | 15 |
| nlpOrganization* | 25* |
| nlpPerson | 20 |
| nlpPlace* | 25* |
| nlpTime | 10 |

* Highest value for this language.

French**Table A2.9** Predefined Concept Priorities for French

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMoney | 18 |
| nlpNounGroup | 15 |
| nlpOrganization* | 20* |
| nlpPercent | 18 |
| nlpPerson* | 20* |
| nlpPlace* | 20* |
| nlpTime | 18 |

* Highest value for this language.

German**Table A2.10** Predefined Concept Priorities for German

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMoney | 25 |
| nlpNounGroup | 15 |
| nlpOrganization | 25 |
| nlpPercent | 18 |
| nlpPerson* | 60* |
| nlpPlace | 40 |
| nlpTime | 18 |

* Highest value for this language.

Greek**Table A2.11** *Predefined Concept Priorities for Greek*

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMoney | 18 |
| nlpNounGroup | 15 |
| nlpOrganization | 20 |
| nlpPercent | 18 |
| nlpPerson* | 20 |
| nlpPlace | 25* |
| nlpTime | 18 |

* Highest value for this language.

Hebrew

For Hebrew, there are no specific priority values for predefined concepts. The default value of 10 is used for all of the predefined concepts listed below.

Table A2.12 *Predefined Concept Priorities for Hebrew*

| Predefined Concept |
|--------------------|
| nlpDate |
| nlpMoney |
| nlpNounGroup |
| nlpOrganization |
| nlpPercent |
| nlpPerson |
| nlpPlace |
| nlpTime |

Hindi**Table A2.13** Predefined Concept Priorities for Hindi

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 10 |
| nlpMoney | 10 |
| nlpNounGroup | 10 |
| nlpOrganization | 10 |
| nlpPercent | 10 |
| nlpPerson | 10 |
| nlpPlace* | 40* |
| nlpTime | 10 |

* Highest value for this language.

Indonesian**Table A2.14** Predefined Concept Priorities for Indonesian

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate* | 20* |
| nlpMoney* | 20* |
| nlpNounGroup | 10 |
| nlpOrganization* | 20* |
| nlpPercent* | 20* |
| nlpPerson* | 20* |
| nlpPlace* | 20* |
| nlpTime* | 20* |

* Highest value for this language.

Italian

For Italian, there are no specific priority values for predefined concepts. The default value of 10 is used.

Table A2.15 *Predefined Concept Priorities for Italian*

| |
|--------------------|
| Predefined Concept |
| nlpDate |
| nlpMoney |
| nlpNounGroup |
| nlpOrganization |
| nlpPercent |
| nlpPerson* |
| nlpPlace |
| nlpTime |

Japanese

For Japanese, there are no specific priority values for predefined concepts. The default value of 50 is used for all of the predefined concepts listed below.

Table A2.16 *Predefined Concept Priorities for Japanese*

| |
|--------------------|
| Predefined Concept |
| nlpDate |
| nlpMoney |
| nlpOrganization |
| nlpPercent |
| nlpPerson* |
| nlpPlace |
| nlpTime |

Korean

For Korean, there are no specific priority values for predefined concepts. The default value of 50 is used.

Table A2.17 Predefined Concept Priorities for Korean

| Predefined Concept |
|--------------------|
| nlpDate |
| nlpMoney |
| nlpOrganization |
| nlpPercent |
| nlpPerson* |
| nlpPlace |
| nlpTime |

Norwegian

For Norwegian, there are no specific priority values for predefined concepts. The default value of 10 is used for all of the predefined concepts listed below.

Table A2.18 Predefined Concept Priorities for Norwegian

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpNounGroup | 10 |

Polish**Table A2.19** Predefined Concept Priorities for Polish

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMoney | 18 |
| nlpNounGroup | 15 |

| | |
|------------------|-----|
| nlpOrganization* | 21* |
| nlpPercent | 18 |
| nlpPerson* | 20 |
| nlpPlace | 20 |
| nlpTime | 18 |

* Highest value for this language.

Portuguese

Table A2.20 Predefined Concept Priorities for Portuguese

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMoney | 18 |
| nlpNounGroup | 15 |
| nlpOrganization | 25* |
| nlpPercent | 18 |
| nlpPerson* | 20 |
| nlpPlace | 25* |
| nlpTime | 18 |

* Highest value for this language.

Russian

Table A2.21 Predefined Concept Priorities for Russian

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate* | 10* |
| nlpMoney | 9 |
| nlpNounGroup* | 10* |

| | |
|------------------|-----|
| nlpOrganization* | 10* |
| nlpPercent* | 10* |
| nlpPerson* | 10* |
| nlpPlace* | 10* |
| nlpTime* | 10* |

* Highest value for this language.

Slovak

Table A2.22 Predefined Concept Priorities for Slovak

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate* | 10* |
| nlpMoney* | 10* |
| nlpNounGroup* | 10* |
| nlpOrganization* | 10* |
| nlpPercent* | 10* |
| nlpPerson* | 7 |
| nlpPlace | 8 |
| nlpTime* | 10* |

* Highest value for this language.

Slovene

For Slovene, there are no specific priority values for predefined concepts. The default value of 10 is used.

Table A2.23 Predefined Concept Priorities for Slovene

| Predefined Concept |
|--------------------|
| nlpDate |
| ORGANIZATION |

nlpMoney

nlpNounGroup

nlpOrganization

nlpPercent

nlpPerson*

VEHICLE

NOUN_GROUP

Spanish

Table A2.24 Predefined Concept Priorities for Spanish

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMoney | 18 |
| nlpNounGroup | 15 |
| nlpOrganization | 25* |
| nlpPercent | 18 |
| nlpPerson* | 20 |
| nlpPlace | 25* |
| nlpTime | 18 |

* Highest value for this language.

Swedish

Table A2.25 Predefined Concept Priorities for Swedish

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 18 |
| nlpMeasure | 18 |

| | |
|-----------------|-----|
| nlpMoney | 18 |
| nlpNounGroup | 15 |
| nlpOrganization | 20* |
| nlpPercent | 18 |
| nlpPerson* | 20* |
| nlpPlace | 20* |
| nlpTime | 18 |

* Highest value for this language.

Tagalog

For Tagalog, there are no specific priority values for predefined concepts. The default value of 10 is used.

| |
|-----------------|
| nlpDate |
| nlpMoney |
| nlpNounGroup |
| nlpOrganization |
| nlpPercent |
| nlpPerson |
| nlpPlace |
| nlpTime |

Thai

For Thai, there are no specific priority values for predefined concepts. The default value of 10 is used.

Table A2.26 Predefined Concept Priorities for Thai

| |
|--------------------|
| Predefined Concept |
| nlpDate |

nlpMoney

nlpOrganization

nlpPercent

nlpPerson

nlpPlace

nlpTime

Turkish

Table A2.27 Predefined Concept Priorities for Turkish

| Predefined Concept | Priority Value |
|--------------------|----------------|
| nlpDate | 10 |
| nlpMoney | 10 |
| nlpNounGroup | 10 |
| nlpOrganization | 11* |
| nlpPercent | 10 |
| nlpPerson | 10 |
| nlpPlace | 10 |
| nlpTime | 10 |

* Highest value for this language.

Vietnamese

For Vietnamese, there are no specific priority values for predefined concepts. The default value of 10 is used.

Table A2.28 Predefined Concept Priorities for Turkish

Predefined Concept

nlpDate

nlpMoney

nlpOrganization

nlpPercent

nlpPerson

nlpPlace

nlpTime

Recommended Reading

Here is the recommended reading list for this title:

- *SAS Encoding: Understanding the Details*
- *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.


THE POWER TO KNOW.

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. © 2013 SAS Institute Inc. All rights reserved. S107969US.0613

