



SAS® Visual Text Analytics in SAS® Viya®

Course Notes

SAS® Visual Text Analytics in SAS® Viya® Course Notes was developed by Terry Woodfield and George Fernandez. Additional contributions were made by Peter Christie, Tarek Elnaccash, Christina Hsiao, Danny Modlin, Patricia Neri, Aurora Peddycord-Liu, Ari Zitin, and Matthew Stainer. Instructional design, editing, and production support was provided by the Learning Design and Development team.

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.

SAS® Visual Text Analytics in SAS® Viya® Course Notes

Copyright © 2018 SAS Institute Inc. Cary, NC, USA. All rights reserved. Printed in the United States of America. 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.

Book code E71400, course code LWSVTA34/SVTA34, prepared date 12Dec2018. LWSVTA34_001

ISBN 978-1-64295-092-2

Table of Contents

Lesson 1 Introduction to SAS® Visual Text Analytics.....	1-1
1.1 Introduction.....	1-3
Demonstration: Extracting Drug Dosages from User-Feedback Documents	1-6
1.2 Language Challenges (Self-Study).....	1-41
1.3 Lesson Summary	1-47
1.4 Solutions	1-48
Solutions to Activities and Questions	1-48
Lesson 2 SAS® Visual Text Analytics Demonstrations	2-1
2.1 Importing Document Collections	2-3
Demonstration: Importing and Converting Document Files	2-4
2.2 Creating a Project with No Predefined Concepts	2-14
Demonstration: Creating a SAS Visual Text Analytics Project with No Predefined Concepts	2-15
Practice	2-39
2.3 A Project with Custom Concepts	2-40
Demonstration: Working with Custom Concepts	2-41
Practice	2-54
2.4 Lesson Summary	2-56
2.5 Solutions	2-57
Solutions to Practices.....	2-57
Solutions to Activities and Questions	2-60
Lesson 3 SAS® Visual Text Analytics Nodes	3-1
3.1 Introduction.....	3-3
3.2 Concepts and Terms.....	3-10

3.3	Machine-Generated Topics.....	3-18
3.4	Categories	3-22
3.5	Scoring New Documents	3-30
3.6	Lesson Summary	3-33
Lesson 4	Concept and Category Rule Definitions.....	4-1
4.1	SAS Visual Text Analytics Rules	4-3
4.2	SAS Visual Text Analytics Concept Rules.....	4-17
	Demonstration: CLASSIFIER Rule	4-20
	Demonstration: CONCEPT Rule	4-31
	Demonstration: C_CONCEPT Rule	4-37
	Demonstration: CONCEPT_RULE Rule	4-42
	Demonstration: NO_BREAK Rule	4-46
	Demonstration: PREDICATE_RULE Rule	4-51
	Demonstration: REGEX Rule	4-59
4.3	SAS Visual Text Analytics Demo Category Rules.....	4-61
	Demonstration: CATEGORY Rule	4-67
	Practice	4-71
4.4	Lesson Summary	4-72
4.5	Solutions	4-73
	Solutions to Practices.....	4-73
	Solutions to Activities and Questions	4-74
Lesson 5	Case Studies.....	5-1
5.1	Introduction to the Case Studies	5-3
5.2	Retrieving Information and Documents about Anxiety and Depression from Drug Reports	5-4
	Demonstration: Information and Documents Retrieval from Drug Reports Related to Depression and Anxiety	5-7

5.3 Automatic Categorization of ASRS Incident Reports	5-33
Demonstration: Automatically Classifying ASRS Procedure Noncompliance Reports.....	5-34
5.4 Retrieving Mortgage Complaints from the CFPB Customer Complaints Data (Self-Study).....	5-45
Demonstration: Exploring and Categorizing Consumer Complaints (Self- Study).....	5-46
Appendix A Cheat Sheet Template: Concept and Category Rules.....	A-1
A.1 Introduction.....	A-3
A.2 Sample Concept Rules	A-4
A.3 Sample Category Rules	A-7
Appendix B References.....	B-1
B.1 References	B-3

To learn more...



For information about other courses in the curriculum, contact the SAS Education Division at 1-800-333-7660, or send e-mail to training@sas.com. You can also find this information on the web at <http://support.sas.com/training/> as well as in the Training Course Catalog.

For a list of SAS books (including e-books) that relate to the topics covered in this course notes, visit <https://www.sas.com/sas/books.html> or call 1-800-727-0025. US customers receive free shipping to US addresses.

Lesson 1 Introduction to SAS® Visual Text Analytics

1.1	Introduction	1-3
	Demonstration: Extracting Drug Dosages from User-Feedback Documents.....	1-6
1.2	Language Challenges (Self-Study)	1-41
1.3	Lesson Summary.....	1-47
1.4	Solutions	1-48
	Solutions to Activities and Questions.....	1-48

1.1 Introduction

Preliminary Remarks

Course Data

Text analytics is a vigorous field of research with many applications. The purpose of this course is to teach you how to solve analytic problems that include relevant textual data. This is done using SAS Visual Text Analytics. This is a product that uses natural language processing and machine-learning techniques to implement a full-featured text analytics solution.

Access to real business data is always problematic. Because text fields often contain confidential information, access to business data that include text is even more difficult. Most of the data sets that are used in this course are publicly available. ***All data that are used in this course are either artificially created or modified in some way.*** Modifications include the following adjustments:

- deletion of sensitive entries
- deletion of potentially embarrassing or misleading entries
- editing or deletion of entries with named individuals or business organizations
- editing of text fields that have obscure or confusing references
- resolution of ambiguities that might lead to incorrect interpretations
- modification or deletion of entries to promote educational goals

Because of these modifications, the data should not be used for any purpose other than education. All publicly available data sets are introduced with a reference to the source of the actual data. You should acquire data directly from the source if you want to use the data for business or scientific purposes.

Objectives

- Provide an overview of SAS Text Analytics.
- Use SAS Visual Text Analytics to perform tasks.
- Explain at a high level the function of SAS Visual Text Analytics.

Text Analytics

A text analytics project requires the following items:

- documents=data
- language for natural language processing (NLP)
- dictionaries to exploit human knowledge that is related to the documents
- software to rapidly and consistently process, analyze, and score documents

"Text analytics helps analysts extract meanings, patterns, and structure in unstructured textual data."
— Chakraborty, et al, 2013.

Often definitions or explanations of text analytics are influenced by the environment of the person who does the analysis. Your environment for this course is SAS Visual Text Analytics. A recurring theme when describing Visual Text Analytics is the use of natural language processing (NLP) to parse and process textual information. In Visual Text Analytics, much of the algorithmic side of NLP is embedded in the software. For example, hidden Markov models are a component of part-of-speech tagging, but the user has no direct access to the parameters that are used to tune hidden Markov models.

Visual Text Analytics is a high-end software product that simplifies the use of text analytics for many different types of users, from business analysts to document librarians. By choosing intelligent parameter settings, or by using machine-learning tools to "learn" an appropriate setting, Visual Text Analytics frees the user to concentrate on the immediate task. The downside to making the software powerful and easy-to-use is that users who try to explore specific algorithmic details for educational purposes are limited in the experiments that can be designed to help understand algorithmic choices. This course provides limited information about specific NLP algorithms. Instead, it focuses on real-world applications of text analytics.

Text Analytics: Examples

- Find all dosage amounts for medications that are mentioned in regulatory filings related to drug approvals.
- Populate a dashboard with the top 10 customer complaints for the week.
- Identify all aviation safety reports that involve runway incursions (collision hazard).
- Enhance the accuracy of a movie recommendation system by exploiting the relationships between movie synopses and customer movie ratings.
- Enhance the accuracy of a fraud and abuse detection system for workers' compensation insurance by exploiting the relationships between adjuster notes, policy-based model inputs, claim-based model inputs, and a fraud target variable.
- Identify all consumer complaints that are related to home loans.

5



Text Analytics: Example

Objective: Find drug dosages in a collection of customer feedback documents.

- Data: 1,414 patient comments about drug effects
- Language: English
- Dictionaries contain: (1) drug names and (2) side effects
- SAS Visual Text Analytics

6





Extracting Drug Dosages from User-Feedback Documents

This demonstration illustrates how to use SAS Visual Text Analytics to extract specific drug dosages from customer-feedback documents.

SAS Visual Text Analytics is a web-based text analytics application that enables you to identify key terms and concepts in your document collections, build concept and topic models, and use linguistic rules to categorize documents.

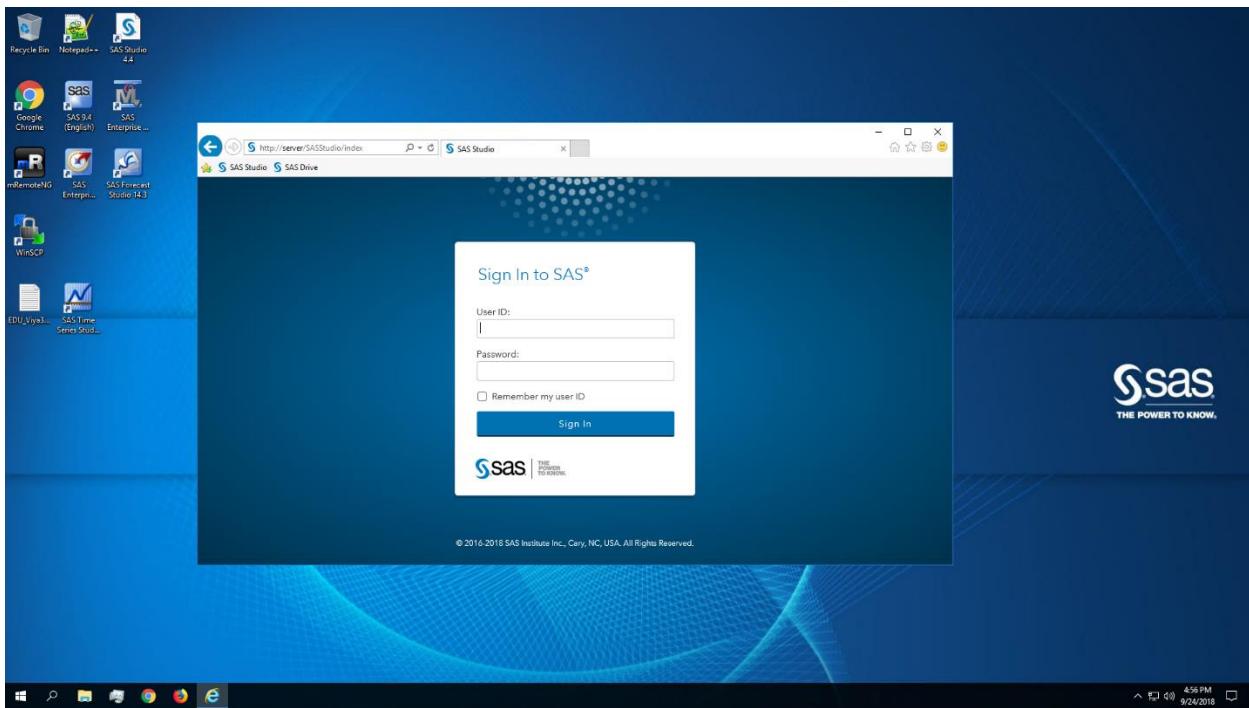
This course uses a Virtual Lab environment. Credentials to sign in to the Virtual Lab are provided to each student. Access to the Virtual Lab is web-based and might require that specialized software be installed on the student computer. In a typical training environment, a student accesses a remote server through a web browser, and the remote server contains all course software and data. The Virtual Lab server has minimal security features and should never be used to analyze confidential or proprietary data.

In the Virtual Lab environment for this course, one or more web browsers are configured for easy access to SAS software. For illustration, suppose that you want to use Microsoft Internet Explorer as the web browser interface to SAS Viya. The menu bar contains one or more SAS entries.

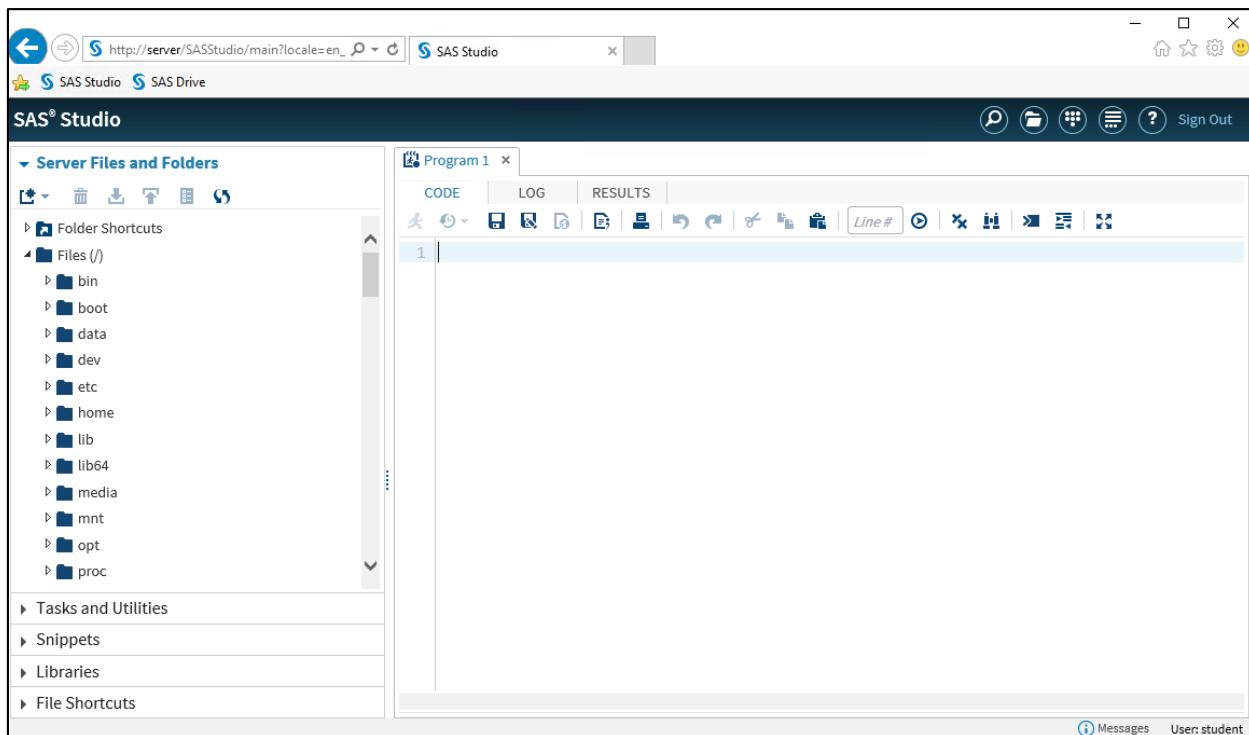
The screenshot shows a Microsoft Internet Explorer browser window. The address bar displays the URL https://www.sas.com/en_us/home.htm. The menu bar contains items for 'SAS Studio' and 'SAS Drive'. The main content area features the SAS logo and the tagline 'THE POWER TO KNOW.'. Below the logo, there are navigation links for 'Industry Solutions', 'Products', 'Learn', 'Support', and 'Customer Stories'. A 'Sign In' link is also visible in the top right corner.

You will access SAS Studio first, followed by SAS Drive. In some browser configurations, the menu selection is SAS Viya, with sub-menus for SAS Studio and SAS Drive.

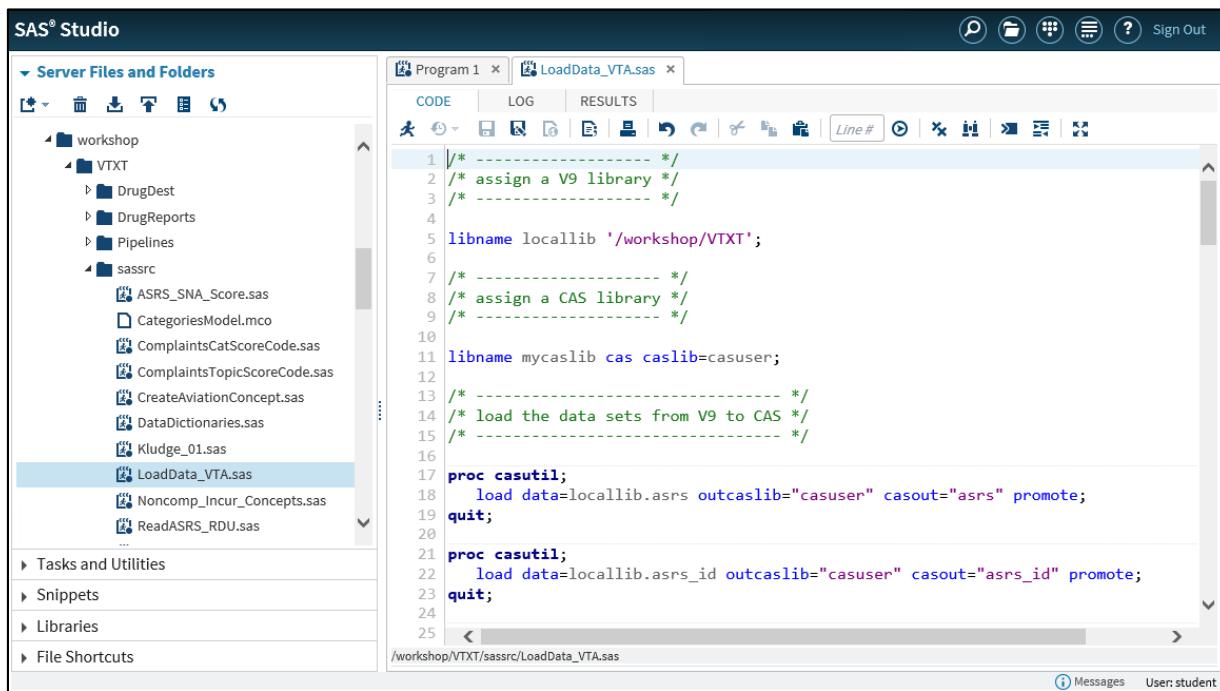
1. Select **SAS Studio**.



A sign-in window appears. Your instructor will provide the user ID and password needed to sign in.



2. In the Server Files and Folders section, scroll down to the workshop folder, and navigate to **/workshop/VTXT/sassrc**.
3. Select the program **LoadData_VTA.sas**.



```

1 /* -----
2  * assign a V9 library */
3 /* ----- */
4
5 libname locallib '/workshop/VTXT';
6
7 /* -----
8  * assign a CAS library */
9 /* ----- */
10
11 libname mycaslib cas caslib=casuser;
12
13 /* -----
14  * load the data sets from V9 to CAS */
15 /* ----- */
16
17 proc casutil;
18   load data=locallib.asrs outcaslib="casuser" casout="asrs" promote;
19 quit;
20
21 proc casutil;
22   load data=locallib.asrs_id outcaslib="casuser" casout="asrs_id" promote;
23 quit;
24
25

```

/workshop/VTXT/sassrc/LoadData_VTA.sas

The program does the following:

- defines libraries for local SAS data sets on disk and for SAS data sets in CAS memory
- distributes all course SAS data sets into HDFS (Hadoop Distributed File System)
- pushes all the HDFS data sets on disk into CAS memory
- performs simple statistical explorations of the data sets

The virtual lab is configured so that a CAS session starts when the virtual lab is enabled. The CAS session will be running when you sign in. If you need to start a new CAS session, SAS Studio contains snippets to help you.

The screenshot shows the SAS® Studio interface. At the top, there's a dark blue header bar with the text "SAS® Studio". Below it is a white sidebar containing a navigation menu. The menu items are:

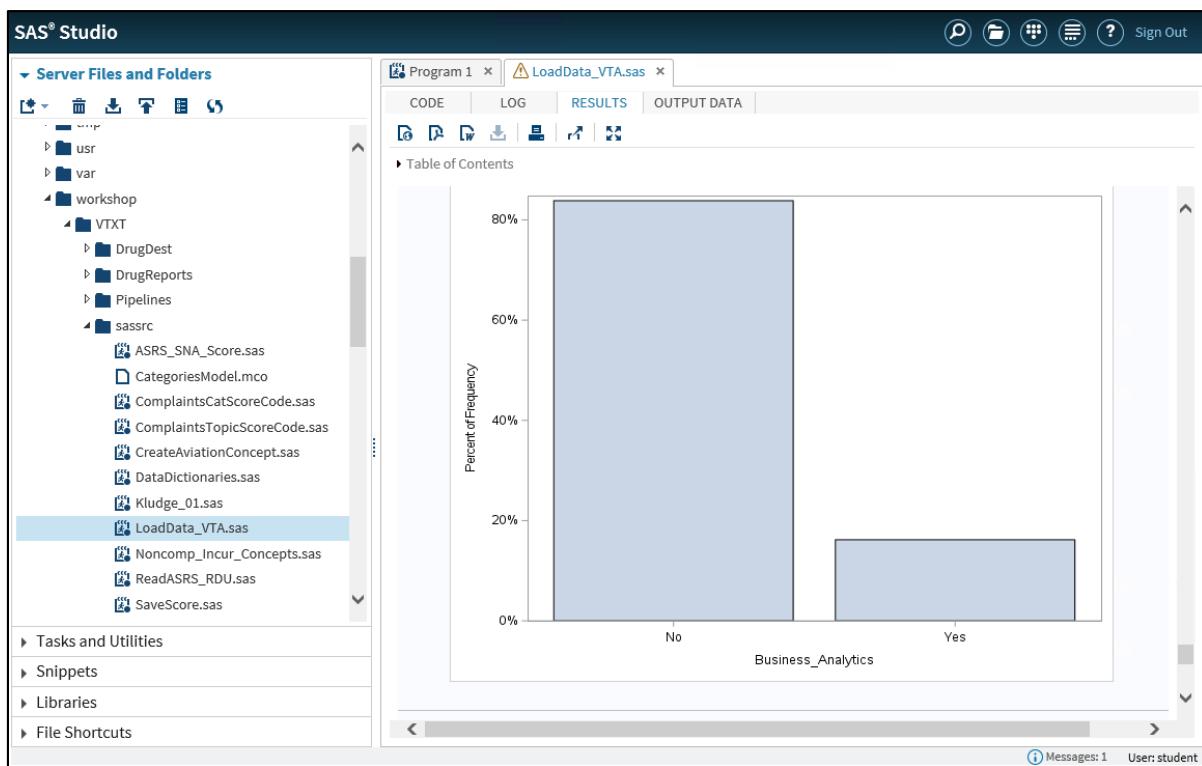
- ▶ Files and Folders
- ▶ Tasks and Utilities
- ▼ **Snippets**

Under the "Snippets" section, there are several categories represented by icons and text:

- ▶ My Snippets
- ◀ Snippets
 - ▶ Catalogs
 - ▶ Data
 - ▶ Descriptive
 - ▶ Graph
 - ▶ IML
 - ▶ Macro
- ◀ SAS Viya Cloud Analytic Services
 - Create CAS Connection** (highlighted with a light blue background)
 - New CAS Session
 - Disconnect CAS Session
 - Reconnect CAS Session
 - Terminate CAS Session
 - List CAS Session Options
 - List CAS Sessions for SAS Client
 - List CAS Sessions for User ID
 - New caslib for Path
 - Generate SAS librefs for caslibs

The snippet Create CAS Connection provides a program template that enables you to connect to an existing session. Additional snippets supply additional functionality. You will not need these snippets for the Virtual Lab, but you will likely need one or more of them when you access CAS at your organization.

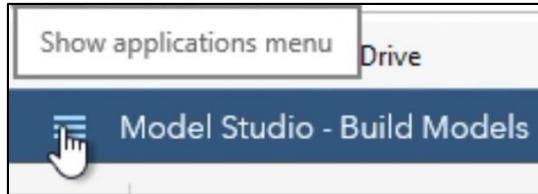
4. Run the program. A sample of the output appears below.



Two warning messages appear in the log file. Both are related to conversion of character data. To understand the challenges of converting character data related to language, see the self-study section at the end of this lesson. The warning messages can be ignored.

5. After course data have been loaded into memory, you are ready to access SAS Drive. Sign out of SAS Studio, and select **SAS Drive**. Use the same sign-in credentials that you used for SAS Studio.

6. In the top left of SAS Drive, click (the Show applications menu, also called the “hamburger” menu) and select **Build Models**. This action invokes Model Studio. Note that the Model Studio interface also has a Show applications menu.



In Model Studio, select **New Project**. Use the following specifications:

- Name: Drug Reports
- Type: Text Analytics
- Data source: DRUG_REPORTS

7. Select the data source by clicking **Browse** and navigating to the desired data set.

The screenshot shows the 'Browse Data' interface. On the left, the 'Available' tab is selected, displaying a list of datasets including ASRS, ASRS_NEWREPORTS, CAS, CAS_NODE, CAS_SYSTEM, COMPLAINTS, DRUG_REPORTS, MOVIES_PLUS, SGF_2013_PAPERS_CL, SYSTEM, and SYSTEM_CPU_USAGE. On the right, the 'DRUG_REPORTS' dataset is selected, showing its details. The 'Details' tab is active, listing four columns: ID, DrugReport, NAME, and EXTENSION, all defined as char. To the right of the columns, there is metadata: Last profiled: Never, Columns: 4, Rows: 1.4 K, Size: 4 MB, Label: Not available, Location: cas-shared-default/CASUSER(student), Date created: May 27, 2018 04:13 PM, and Date modified: May 27, 2018 04:13 PM. The Encoding is set to utf-8. At the bottom right are 'OK' and 'Cancel' buttons.

If you zoom in on the display, you can see that **DRUG_REPORTS** is selected.

This is a zoomed-in view of the 'Available' tab in the 'Browse Data' interface. The 'DRUG_REPORTS' dataset is highlighted with a gray background, indicating it is selected. The other datasets listed are ASRS, ASRS_NEWREPORTS, CAS, CAS_NODE, CAS_SYSTEM, COMPLAINTS, and MOVIES_PLUS.

The variable **DrugReport** contains the customer feedback documents.

- After you select the data, click **OK**. Then, in the New Project window, click **Save**.

9. You must select the document variable. To do so, assign the role of **Text** to the variable.

The screenshot shows the 'Model Studio - Build Models' interface. The 'Data' tab is selected, and the 'Pipelines' section is visible. A yellow warning bar at the top states: 'You must assign a Text variable in order to run a pipeline.' Below this, a table lists variables:

Variable Name	Type	Role	Display Variable
__uniqueid__	Numeric	Key	
DrugReport	Character		
EXTENSION	Character		
ID	Character		
NAME	Character		

To the right of the table, a panel for 'DrugReport' shows the 'Role:' dropdown set to 'None'. There is also a checkbox for 'Display variable' which is unchecked.

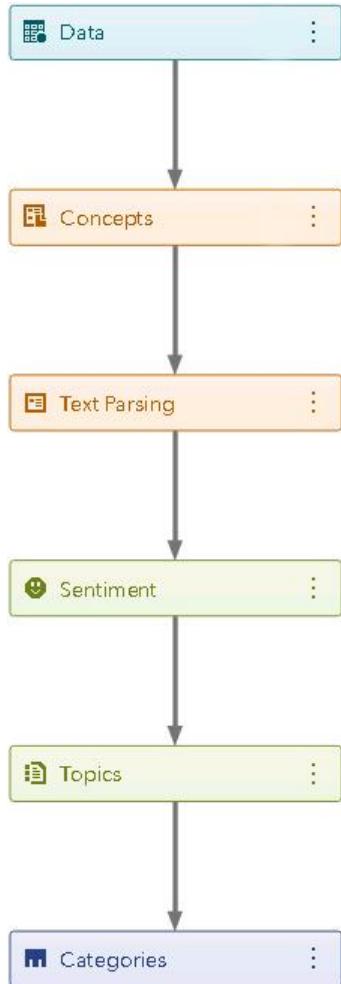
10. Select **DrugReport**. In the right pane, select the menu for **Role** and select the role of **Text**.

The screenshot shows the 'DrugReport' variable details dialog. The 'Role:' dropdown menu is open, displaying four options: 'None' (selected), 'None', 'Category', and 'Text'. The 'None' option is highlighted with a blue background.

The project is updated and saved automatically. If you look at the upper left, you see that you are in the Data menu.

11. Select **Pipelines**, which is to the right of **Data**. You get a default text analytics pipeline. You work with this default pipeline for simplicity, but as you gain knowledge about SAS Visual Text Analytics, you realize that a custom pipeline might be more appropriate for this project.

The standard Text Analytics pipeline is below. (The nodes are described in detail later.)



12. Select the **Concepts** node. From the options menu on the right, select **Include predefined concepts**.
13. Right-click the **Concepts** node and select **Run**. After the node runs, a note appears at the bottom of the window, and a green circle with a white check mark appears in the Concepts node. Right-click the **Concepts** node and select **Open**.
14. In the upper left, in the Show Applications menu, you see **Predefined Concepts** and **Custom Concepts**. Observe that nine predefined concepts are available. Click the arrow to the left of **Predefined Concepts**.
15. Select **nlpMeasure**. This concept relates to measures. Measure concepts are extracted using natural language processing (NLP). You can take advantage of the LITI (language interpretation text interpretation) computing language that provides access to powerful NLP constructs and rules so that you can handle text extraction problems that are unique to your organization. Custom concepts are created by Visual Text Analytics users who apply the LITI language to concept extraction.

16. The Documents pane is in the lower right corner of the display. In the Documents pane, click **Matched**.

The screenshot shows the SAS Model Studio interface with the 'Documents' tab selected. In the top left, there are two tabs: 'All (1414)' and 'Matched (784 of 1414)'. A green box highlights the text 'Number of documents=784'. Below the tabs, there is a list of documents. The first document in the list contains the text: "...made me gain **40 pounds** it has been **2 years** and I have only lost **10 pounds**. Beware and watch your weight." The second document contains: "...to my ecstapin(**225mg**) due to unrelenting depression.I had lost my sister mom within **8 months** and although I had been on an antidepressant for a long time before I grieved but still couldn't get over the depression. Within 3-54 daysboth I and my husband noticed a significant improvement. I felt better more like me than I had in years. Only thing I notice was I sometimes mix up words Anybody else do that? Not bad enough for me to stop med tho." The third document contains: "...been on ecstapin **150mg** until I started having break throughs.My doctor has started me on abidal **40mg** daily. I have the dizziness,but my mood is awesome. I do not know if taking more will make the dizziness go away or when will it? That is my only side effect, I think? For the one person who has the rash and other side effects....you are allergic!!! Stop taking it." The fourth document contains: "... **15 year old** has been on this med since 10-09 and it has made me one happy mother." The fifth document contains: "...on Ecstapin for **four years**, increasing the dose yearly when it began to decrease in effectiveness. Weening off was by far the worst experience of my life. Constant uncontrollable crying, dizziness, and nausea- sometimes unable to keep any food down for days. I am finally off and doing much better on Exulactin." At the bottom left of the pane, it says 'Document 1 of 784'. At the bottom right, there are buttons for 'Highlight', 'Concept matches', and 'Search matches'.

The nlpMeasure concept finds measures in 784 of the 1,414 documents, and it finds a dosage in the second document, namely *225 mg*. The goal is to extract dosages, but unfortunately many nlpMeasure matches are not dosages. Consequently, you must create a custom concept to satisfy the project requirements.

17. In the Concepts pane on the left, add a custom concept by clicking the **Custom Concepts** folder and selecting **New Concept**. A data entry window appears below the Custom Concept (0) heading.

The screenshot shows the SAS Model Studio interface with the 'Concepts' tab selected. On the left, there is a tree view with 'Predefined Concepts (9)' and 'Custom Concepts (0)'. A green box highlights the text 'Select to add new custom concept.' To the right, there is a data entry window with a text input field containing 'Drug Reports > Concepts'. A green box highlights the text 'Enter name here.' Below the input field is a button labeled 'New Concept'. There is also a plus sign icon above the 'New Concept' button. At the bottom of the window, there is a preview pane showing the text: "This medication made me gain 40 pounds it has been 2 years and I have only lost 10 pounds. Beware and watch your weight." At the very bottom of the interface, there is a footer bar with various icons.

Enter the name **Dosage** for the new custom concept.

Note: Some SAS documentation examples suggest that you name your custom concepts using all uppercase to distinguish them from predefined concepts. Otherwise, there is no limit on the length of the name. However, it can contain only letters, numbers, and underscores.

Note: Concept names are case sensitive.

Note: You can also add a custom concept by clicking

Note: When selecting a concept name, you want to avoid ambiguities and unexpected results. See the “Create a Custom Concept” subsection in Chapter 6 in *SAS® Visual Text Analytics User’s Guide*. For example, avoid using names that are actual words that might appear in the term table. An extract from the user’s guide appears below.

- **Create a custom concept**

Select **Custom Concepts** in the **Concepts** panel, and click the  icon to add a custom concept for which you create your own rules.

Note: There is currently a limit of 400 concepts per taxonomy level.

In the text box that appears under **Custom Concepts**, type the desired name of the newly created concept. When naming a custom concept, keep the following in mind:

- Use valid characters – numbers, alphabetic letters, and underscores (_). (See the Note below regarding the use of underscores and also double-byte characters).
- 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.

Note: Concept names can contain only single-byte characters. Languages that have double-byte letters/characters should use only ASCII letters in names.

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**.

18. Enter the rules that define your custom concept in the editor below the Edit a Concept heading. Enter the following rules:

```
REGEX: [0-9]+[\.\.][0-9]+\s?mg\..?
REGEX: [\d]+\s?mg\..?
```

You use two regular expressions to define dosage measures.

Note: Although some information about regular expressions is provided in this course, the topic deserves a course of its own. For example, see Quigley (1998) for a comprehensive (552 pages!) book about Perl regular expressions.

A summary of the first regular expression above provides a sense of how they work.

- a) [0-9]: match any single digit
- b) +: match the previous character one or more times (The previous character is a digit.)
- c) [\.]: match a period (decimal point)
- d) [0-9]: match any single digit
- e) +: match the previous character one or more times (The previous character is a digit.)
- f) \s: match any whitespace character (for example, space, tab)
- g) ?: make the previous character optional (The previous character is a whitespace character.)
- h) mg: match the letters *mg*
- i) \.: match a period (signifying that *mg* is an abbreviation for milligrams)
- j) ?: make the previous character optional (The previous character is a period.)

Technically speaking, the special character ? matches the previous character “zero or one times.” Characters within square brackets represent a single character and an OR condition. For example, the regular expression [mg] matches either an *m* or a *g*, but only one character. Thus, the *m* in the abbreviation *mg* matches, but the two-character abbreviation *mg* is not a match in its entirety. An extended explanation is as follows:

- **REGEX:[mg]** matches the single character *m* or the single character *g*, but not both together as *mg*.
- **REGEX:[mg]+** matches *m*, *g*, *mg*, *gm*, *ggm*, *gmmgmmmgggm*, and so on.
- **REGEX:mg** matches only *mg*.

Text searches can be string searches or token searches, and regular expressions can be defined to do either. For example, if a document contains *x 10.1mg zz*, the first regular expression identifies *10.1mg* as a dosage token in the string.

- A *token* is a character string that does not contain any delimiters.
- *Delimiters* can usually be specified for many string functions. Delimiters usually include whitespace characters (space, tab), marks of punctuation, and certain symbols.

Note: The LITI language is discussed in detail in a later lesson. More regular expression examples are provided later.

Note: To avoid the labor of entering complex rules, as well as to avoid the risk of mistyping, a collection of rules for this course is stored in the text file RulesAndPrompts.txt.

The screenshot shows the 'Model Studio - Build Models' interface. In the top navigation bar, 'Drug Reports' is selected under 'Concepts'. On the left, there's a tree view of concepts: 'Predefined Concepts (9)' (nlpDate, nlpMeasure, nlpMoney, nlpNounGroup, nlpOrganization, nlpPercent, nlpPerson, nlpPlace, nlpTime) and 'Custom Concepts (1)' (Dosage). The 'Dosage' node is currently selected. The main right pane is titled 'Edit a Concept' and contains a code editor with two lines of REGEX patterns. Below the code editor, a message says 'Validation is out of date.'

19. The message *Validation is out of date.* means that you should validate the script before you submit it. To do so, click (the **Validate Rules** icon) in the upper right of the Edit a Concept window. If you are successful, you see the new message *Code is valid.*

Note: Because document collections can be very large, you often do not want to test your custom concept on the entire collection. You can use the Test Sample Text feature to supply a few documents to test the rules that you supplied. Six sample documents, one per line, are provided in the RulesAndPrompts.txt file.

20. Select **Test Sample Text** and copy and paste the sample documents from the RulesAndPrompts.txt file into the test area. Then select **Test Text** to see whether the rules are performing as intended.

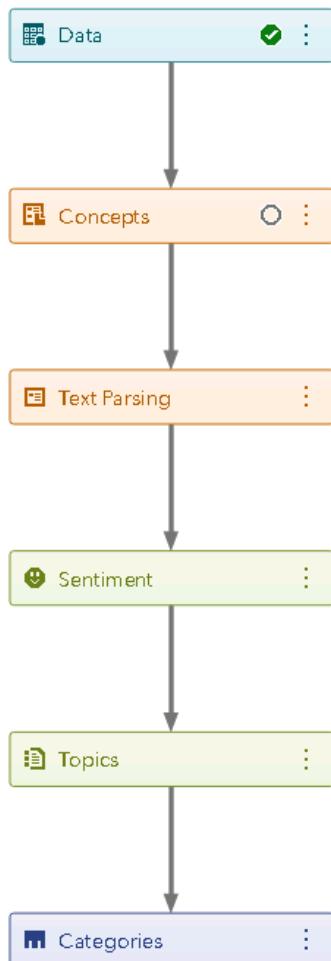
The screenshot shows the 'Test Sample Text' tab. The text area contains the following sample documents:

```

Documents Test Sample Text
Prexifan, along with Elevex, has been a life-saver in treating long-term depression and as others have noted, I have noticed weight-gain.
I have been on Abidal for 2 yrs with a little improvement then about 1 yr ago I was started on Prexifan 2mg with no change. new dr put me on 10mg of prexifan I have noticed that I too am having trouble sleeping, however still have no energy.
I have been on 10mg prexifan for 6 months along with 2000mg noderall 1mg gemulex and fortifex when needed.
I am on 5 mg. of prexifan and 20 mg of elevex, but, I am rapidly gaining weight and I also cannot sleep.
I have been on 5mg of Prexifan and 40mg of Amicor to treat anxiety, OCD, and mild depression. Before the prexifan nothing seemed to work for the OCD and anxiety. With Prexifan and Amicor side effects are excessive gas and insomnia.
Now taking Prexifan 2.5mg, elevex 20mg, rexinor .5mg as needed for nervousness and fortifex occ for sleep. feel better than have in years.

```

21. The rules seem to be working. Select **Close** in the upper right corner to return to the pipeline. You can apply the new custom concept to the entire collection. The success indicator (green circle with white check mark) becomes an empty circle. You need to rerun the Concepts node to derive dosages.



22. Right-click the **Concepts** node and select **Run**. When the run is complete, right-click, and select **Open**.

Note: You could have selected **Run Node** when the Concepts node was open. The sequence above was used to illustrate that SAS Viya detects an out-of-date node when a change has been made, removes the green circle with the white check mark, and replaces it with an empty circle.

23. Select **Dosage** as the custom concept, and then select **Documents** ⇒ **Matched**. You can see that 300 of the 1,414 documents exhibited a match.

The screenshot shows the SAS Visual Text Analytics interface. At the top, there's a navigation bar with 'Documents' selected and 'Test Sample Text' as the document type. Below the navigation bar, a header bar displays 'All (1414)' and 'Matched (300 of 1414)'. A search bar with a magnifying glass icon is also present. To the right of the search bar are three icons: a grid, a document, and a list. The main content area is titled 'DrugReport' and contains four snippets of text. Each snippet highlights specific dosage terms in blue. The first snippet discusses 'ecstabin(225mg)'. The second snippet discusses 'ecstabin 150mg'. The third snippet discusses 'dosage to 60mg'. The fourth snippet discusses 'started on Prexifan 2mg'. At the bottom left, it says 'Document 1 of 300'. At the bottom right, it says 'Highlight: Concept matches'.

The Visual Text Analytics software did more than identify matching documents. If you apply the Dosage concept to new documents, Visual Text Analytics flags the documents that exhibit one or more dosages, and it records the location of each matching instance. When you score a new document collection for concepts, you might get zero results for some documents and multiple results for others. The **score** data set has one row for each match. For example, if a document has three matches, you see three rows for that document in the **score** data set, the actual instance of the match, and the byte offset location of the match within the document.

End of Demonstration

Natural Language Processing

- *Natural language* refers to verbal language spoken by human beings: English, Spanish, French, German, Chinese, Japanese, Arabic, Norwegian, and so on. Most human languages have been encoded using a writing system.
- *Formal language* refers to C, C++, C#, Java, and other computer languages, markup languages like HTML, scripting languages like Lua and those supported by UNIX shells, computer system languages like JCL, or any man-made language designed for any purpose other than verbal communication between human beings.
- *Natural language processing (NLP)* is the sub-field of Artificial Intelligence that uses computers to process natural language for specific purposes.



Writing systems for encoding spoken language are discussed in the next section.

A formal language has rigid syntactical rules to avoid ambiguities. Some references use the term *artificial language* rather than formal language. However, artificial language can also refer to languages constructed for purposes other than communication. For example, Klingon is an artificial language created for entertainment related to the ***Star Trek*** science fiction television and movie series. NLP could be applied to artificial languages like Klingon, but most likely would not be relevant for formal languages like computer programming languages.

Natural Language Processing

The goal of NLP is to process both spoken and written language.

1. Convert spoken language to digitized audio. Example: Record a conversation as an MP3 file.
2. Convert the digitized audio of the spoken language to written language. Example: Speech recognition software.
3. Process the written language using NLP.

Most NLP algorithms process written rather than spoken language. The three steps have historically been separated into three distinct research areas. Some complete NLP solutions, like Apple Siri and Amazon Alexa, proceed directly from spoken language to solution.



Natural Language Processing

- Text analytics software includes integrated NLP algorithms. NLP algorithms are also used in text processing software not intended for analytical use, such as translation software.
- Few software products are exclusively NLP solutions.
- Many software products that use NLP do not permit users to modify hyperparameters used by the NLP algorithms.
- SAS Visual Text Analytics and other SAS text analytics products use NLP algorithms embedded within the software.
- Text analytics is not exclusively NLP, and NLP is not a subset of text analytics.

110

Copyright © SAS Institute Inc. All rights reserved.



To facilitate rapid search, software vendors often categorize solutions in very broad terms. For example, the software product SAS Visual Text Analytics has appeared in taxonomies listing it as an NLP solution. Apple Siri is an NLP solution, whereas SAS Visual Text Analytics is a text analytics solution that uses NLP. Because the NLP algorithms used by SAS Visual Text Analytics are hidden from the user, a complete introduction to NLP is unwarranted. However, if you ignore the NLP aspects of the software, you are ignoring one of the most important components of the software. Every task used to produce a deployed solution relies on NLP, and unlike products like SAS Text Miner, the user cannot turn off NLP features.

The primary role of NLP in SAS Visual Text Analytics is to efficiently parse the document collection. Parsing creates the term table that topic derivation, sentiment analysis, and text categorization depend on. Each row of the term table includes the term itself, the role of the term (part of speech or concept), statistics related to the term (for example, frequency of occurrence), stemming information (parent/child), and keep status identifying whether the term is used in subsequent analysis.

One of the biggest challenges to NLP is *word sense disambiguation*. Textbooks often use the word *train* to illustrate NLP challenges. *Train* or its variants can be a noun, verb, or adjective. When *train* is used in the context of a locomotive, it is a noun. (Example: The Amtrak *train* from San Diego arrived on time.) The Merriam-Webster dictionary gives eight different noun variants of *train*. When used relative to education, it can be a noun (SAS *training*), a verb (*train* the student), or an adjective (a *training* exercise). When an NLP algorithm encounters the word *train*, it must resolve any ambiguities that thwart assigning the proper part-of-speech role. If an NLP algorithm decides that *train* is a noun, it can still be one of eight nouns. The relevance of the word *train* in the analysis can hinge on NLP tools like term maps. The term table tells you that *train* appears in the document collection as a noun. The term map shows related terms, which help you decide how *train* is used. For example, *train* might be connected to *travel* and *bride*. If the information gain for *train*→*travel* is higher than the information gain for *train*→*bride*, then the collection likely has more information about passenger trains than about bridal gowns.

NLP is a broad area of study. When you study NLP algorithms, you are likely to learn about hidden Markov models (HMM) and probabilistic parsing. Algorithms like these use hyperparameters to tune the solution to the analytic objective. For example, HMM is a look-ahead algorithm. When an HMM parser encounters a term, it looks several words ahead of the term to determine likely roles of the term. Look-ahead algorithms require a hyperparameter that controls how far to look back and how far to look ahead. These hyperparameters are set by the software developers. Hyperparameters can be hardcoded as fixed constants. However, because of the large variability in types of document collections, it is often wise to use a learning algorithm to set hyperparameters. In machine learning terminology, the software **learns** a good hyperparameter value for a specific problem and uses that value in the parsing process. Power users can be frustrated by not having more control, but the typical user benefits by avoiding time consuming trial-and-error experiments that yield diminishing returns. Poor values of hyperparameters can lead to nearly impossible situations (for example, parsing that can take several days rather than several minutes).

If you are interested in some of the challenges associated with NLP, use your favorite search engine to find the website for the Stanford Parser from Stanford University. Consider the following variant of a famous Groucho Marx joke.

"I shot an elephant in my pajamas."

The punchline is, "How the elephant got in my pajamas, I'll never know." If you find the joke to be funny, it is probably because in your attempt to resolve word sense disambiguation, you concluded that the speaker was wearing pajamas while shooting an elephant. The punchline provides a surprisingly different disambiguation, and some people find that to be humorous. The Stanford Parser can be used to parse the sentence.

Stanford Parser

Please enter a sentence to be parsed:
I shot an elephant in my pajamas.

Language: English ▾ Sample Sentence Parse

Your query
I shot an elephant in my pajamas.

Tagging
I/PRP shot/VBD an/DT elephant/NN in/IN my/PRP\$ pajamas/NNS ./.

Parse

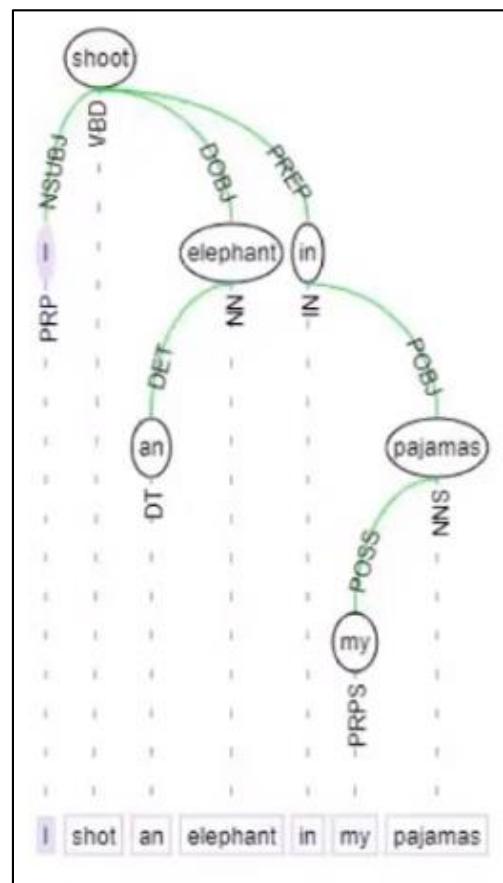
```
(ROOT
  (S
    (NP (PRP I))
    (VP (VBD shot)
      (NP (DT an) (NN elephant))
      (PP (IN in)
        (NP (PRP$ my) (NNS pajamas))))
    (. .)))
```

```
Universal dependencies
nsubj(shot-2, I-1)
root(ROOT-0, shot-2)
det(elephant-4, an-3)
dobj(shot-2, elephant-4)
case(pajamas-7, in-5)
nmod:poss(pajamas-7, my-6)
nmod(shot-2, pajamas-7)

Universal dependencies, enhanced
nsubj(shot-2, I-1)
root(ROOT-0, shot-2)
det(elephant-4, an-3)
dobj(shot-2, elephant-4)
case(pajamas-7, in-5)
nmod:poss(pajamas-7, my-6)
nmod:in(shot-2, pajamas-7)

Statistics
Tokens: 8
Time: 0.019 s
Parser: englishPCFG.ser.gz
```

The parser identifies parts of speech for each word in the sentence. If you diagram the sentence using what you learn from a parser, you might get something like the following:



The diagrammed sentence above was extracted from a SAS Marketing presentation. SAS Visual Text Analytics has no feature to diagram sentences. To understand various parsing results, you need a dictionary of part-of-speech tags. For example, PRPS is a possessive pronoun. The parsing results generate entries in the term table, but they do not resolve who is wearing the pajamas. If it is important to resolve who is wearing the pajamas, additional analysis is required. Some of the connectivity in the diagrammed sentence is not surfaced in SAS Visual Text Analytics. If you view the document, you will not see any information that reveals that *my* as a possessive pronoun has pajamas as its possession (*my*→POSS→pajamas). The term table will reveal the part of speech for a term but will not reveal how the part of speech was determined for any given instance.

NLP is the foundation for parsing, and parsing typically consumes most of the CPU cycles for a full SAS Visual Text Analytics pipeline. Text categorization can take longer if numerous categories or high-cardinality categories (or both) are used. As the next slide reveals, massive document collections are becoming more common, so efficient parsing is essential for success.

Text Data → Unstructured Data

The Computing Challenges

- Massive document collections
- Lengthy processing times
 - Strategic versus tactical time frames
 - Real-time knowledge capture
 - Conversions and translations



Text Data → Unstructured Data

The Practical Challenges

- Information overload
- Cognitive bias (the influence of personal and cultural perspectives)
- Automatic knowledge or intelligence discovery
- Multiple languages and cultures
- Information might have a high value but a limited useful life span.

Unstructured and semi-structured text content within organizations is growing at an unprecedented rate from applications, records, and business processes to external documents, scanned images, XML components, web pages, blogs, and forums. Unstructured text refers to plain text documents. Semi-structured text refers to text that has been augmented with tags, such as the text in an HTML file having specialized tags for formatting. If documents are pre-processed with tagging to include specific information, this might facilitate efficient information retrieval. XML and JSON represent two common languages that accommodate the encoding of attributes within a document. Many organizations store and manage this information based on data type, by manually source-tagging the content, or by after-the-fact content sampling.

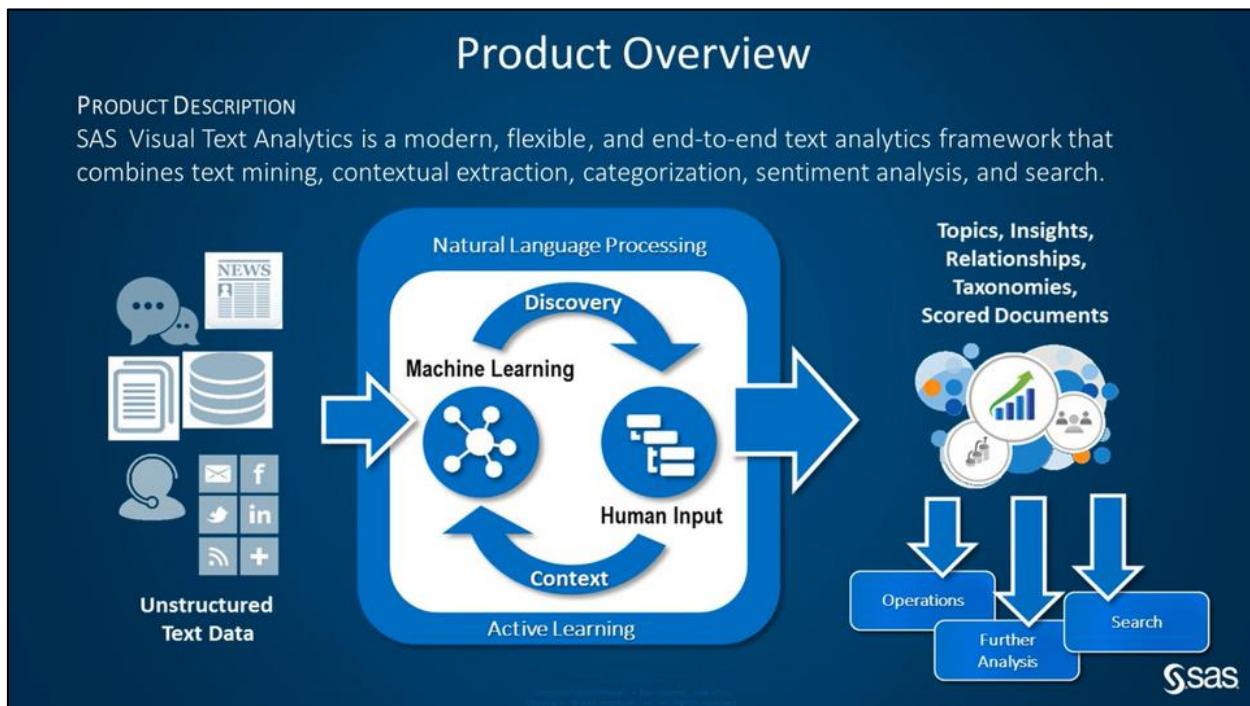
SAS Visual Text Analytics can accommodate many languages, formats, and encoding schemes. This course addresses documents written in English, but many world languages are supported, and new languages are added on a regular basis. As an end user, you can anticipate that documents are stored as PDF files, Microsoft Word documents, or some other popular document format. One of the first tasks for a successful text analytics projects is converting a document that is stored in a possibly proprietary format into some universal format (for example, plain text). That general format excludes graphical rendering code for displaying special fonts.

The concept of *plain text* usually translates to mean ordinary ASCII encoding. Plain text encoding is too simplistic and vague for experts who work with international languages. As you get deeper into the software aspects of text analytics, you must be able to distinguish between different encoding formats (for example, Latin-1 encoding and UTF-16 encoding). (The next section provides some background about written language systems and computer coding schemes.)

Common methods for search and information processing are plagued by redundancy, inaccuracy, wasted efforts, and missed opportunities. For newer types of content from sources such as blogs and wiki pages, the old methods are inadequate. Unfortunately, the lack of well-defined taxonomies and indexes based on the materials themselves makes relevant content difficult to locate. If content remains unmanaged, it loses its relevancy, timeliness, and value.

In the era of big data, search and information processing must function efficiently. Managing enterprise content effectively and efficiently as a strategic asset requires a common, underlying, organizational structure.

SAS Visual Text Analytics help the text analyst face the big, unstructured text data challenges effectively in a timely manner by providing powerful tools in the fields of text data exploration and visualization, information retrieval, and content categorization.



SAS Visual Text Analytics: Capabilities and Benefits

Natural language processing: Enhances parsing to add language features and expand the document collection term table; supports topic derivation

Automated feature extraction with machine-generated topics: Discovers themes and shows related terms and documents for each theme

Native linguistic support for multiple languages: Supports the global nature of a business (more than 30 world languages)

Sentiment analysis: Supports business decisions by revealing trending perspectives

Support for both machine learning and rules-based approaches within a single project: Enables scoring of new documents to reveal emerging trends and identify dominant themes

SAS Visual Text Analytics: Capabilities and Benefits

Contextual extraction: Enables non-ambiguous coding of subject matter expertise to extract specific information from within documents

Flexible deployment: Maximizes the data's value and accelerates the data to the decision timeline

Facilitation of collaboration in a multi-user environment: Fuels collaboration and information sharing in an open analytics ecosystem

1.01 Multiple Answer Question

What previous experience with or exposure to other SAS Text Analytics software products do you have?

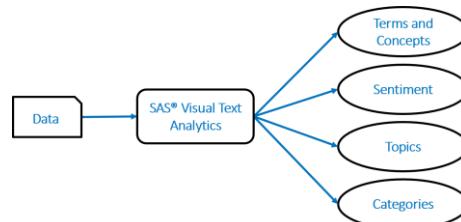
(Select all that apply.)

- a. SAS Text Miner
- b. SAS Content Categorization Studio
- c. SAS Sentiment Analysis Studio
- d. SAS Contextual Analysis
- e. none of the above

SAS Visual Text Analytics: Users

Enterprise:

- new to text analytics
- organizations with more than 50 employees
- manual classification issues for more than 500 documents
- all industries
- common business areas: quality control, records management, marketing, call center operations, research archiving, document management, web management, IT, or any other area that needs to classify documents or extract information



User:

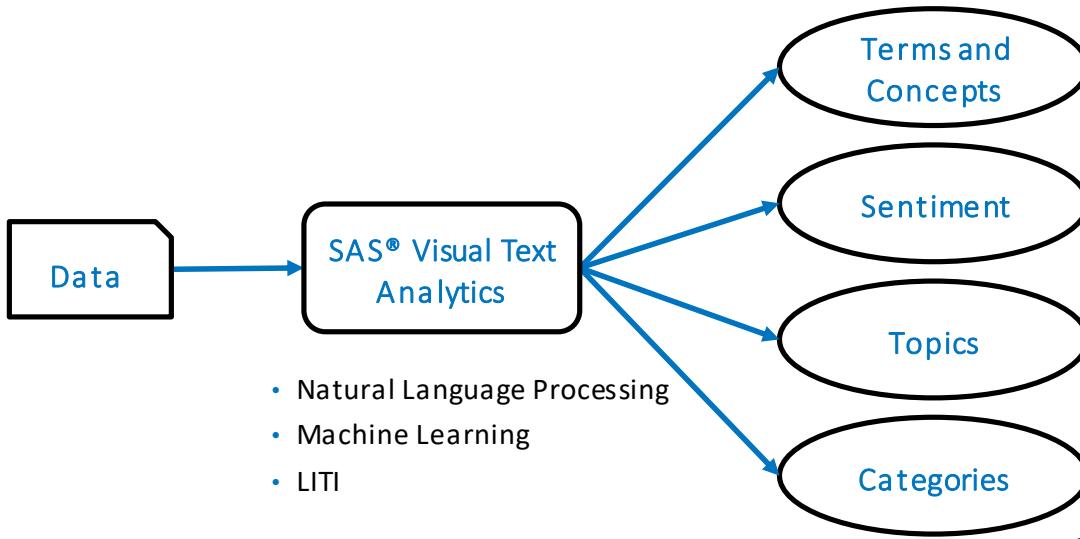
- analyst, taxonomist, researcher
- novice to text analytics or experienced user
- Advanced analytic or machine learning skills are not required.



Copyright © SAS Institute Inc. All rights reserved.

SAS Visual Text Analytics: Process

- Natural Language Processing
- Machine Learning
- LTI



18

Copyright © SAS Institute Inc. All rights reserved.

SAS Visual Text Analytics: Big Picture

SAS Visual Text Analytics is integrated with SAS.

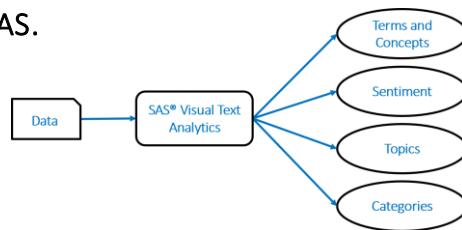
- accessible via Model Studio in SAS Viya
- input or score data

You can use the product to manage multiple projects.

- An interactive point-and-click browser GUI guides the analysis of large or complex text data.

You can easily include data from the following sources:

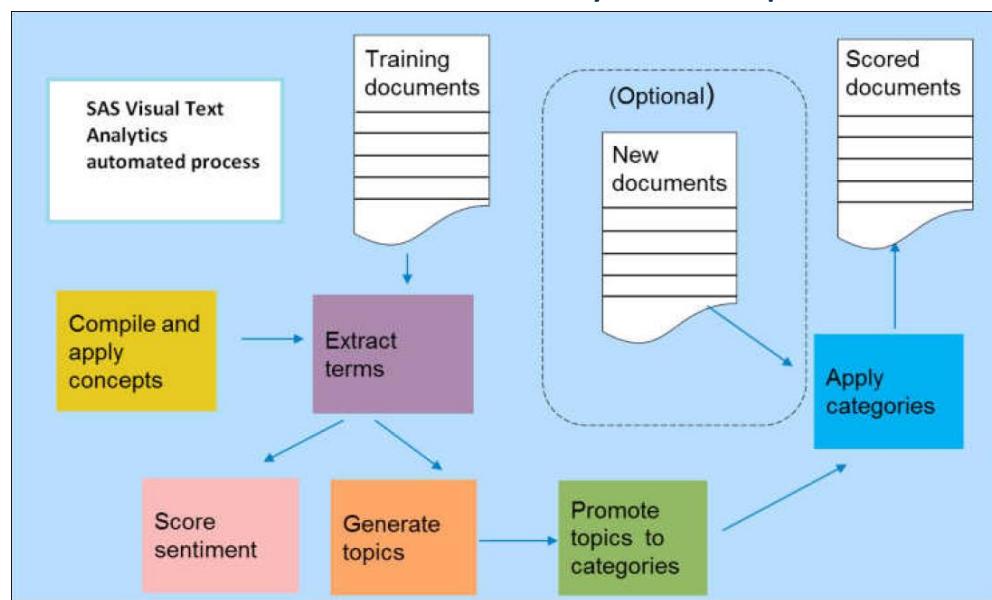
- SAS table or CAS table
- Documents converted using SAS Data Explorer



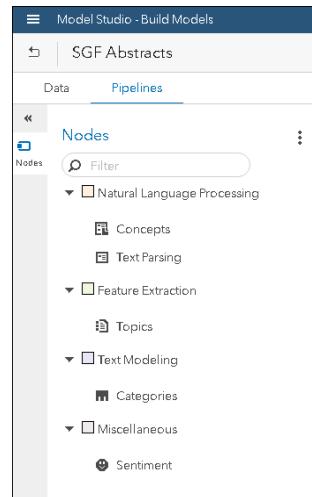
19



SAS Visual Text Analytics: Steps



SAS Visual Text Analytics: Nodes

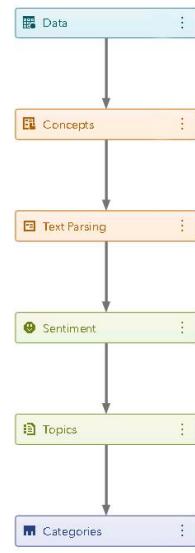


21



Copyright © SAS Institute Inc. All rights reserved.

SAS Visual Text Analytics: Default Pipeline



22

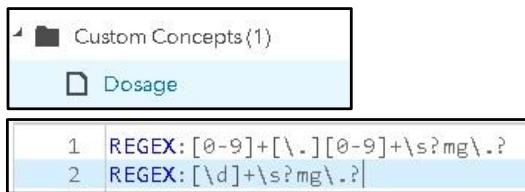


Copyright © SAS Institute Inc. All rights reserved.

Text Analytics Components

Concepts:

- defined by linguistic and Boolean rules
- identified *within* a document (A document can contain more than one concept.)
- extracted with score code



I have been on 10mg prexifan for 6 months along with 2000mg noderall 1mg gemulex and fortifex when needed.

Concept Extraction

1. Input text content – For example, Twitter data, reports, email, news, or forum messages are entered.

Concepts Taxonomy

2. Parse content through concepts taxonomy – Match messages and documents to extract concepts.

3. Output results – For example, each message or document is now associated with a list of extracted concepts.

Concepts

- Measures – 50 mg...
- Persons – John...
- Dates – Monday...
- Organizations – SAS...

Results are indexed or fed into existing systems for search and analysis.

Concept Extraction: Predefined Concepts

The screenshot shows the SAS Concept Extraction interface. On the left, a sidebar lists 'Predefined Concepts (9)' including nlpDate, nlpMeasure, nlpMoney, nlpNounGroup, nlpOrganization, nlpPercent, nlpPerson, nlpPlace, and nlpTime. A 'Custom Concepts (0)' section is also present. The main area is titled 'Edit a Concept' with a code editor containing '1'. Below it, a message says 'Code is valid.' A 'Documents' tab shows 610 results, with 'Matched' selected. A preview window displays the word 'abstract' followed by several paragraphs of text from a document.

25



Concept Extraction: Custom Concepts

The screenshot shows the SAS Concept Extraction interface. On the left, a sidebar lists 'Predefined Concepts (9)' and a 'Custom Concepts (1)' section with 'Company' selected. The main area is titled 'Edit a Concept' with a code editor containing a list of classifier definitions: 1 CLASSIFIER:SAS, 2 CLASSIFIER:Microsoft, 3 CLASSIFIER:Google, 4 CLASSIFIER:Ebay, 5 CLASSIFIER:Oracle, 6 CLASSIFIER:Ford, and 7 CLASSIFIER:General Motors. Below it, a message says 'Code is valid.' A 'Documents' tab shows 491 results, with 'Matched' selected. A preview window displays the word 'abstract' followed by several paragraphs of text from a document.

26



SAS Visual Text Analytics: Term Extraction

Automatic term extraction from large amounts of textual data is accomplished using NLP tools in SAS Visual Text Analytics.

Here are some of those tools:

- parsing
- tokenization
- part-of-speech (POS) tagging
- synonym detection
- stemming

27



When you open the Text Parsing node, you can see terms and documents. You can select terms and look for matching documents.

Term Extraction

Kept Terms (1516)	
Term	Role
sas	Company
data	N
use	V
the	PUNC
sas	PN
paper	N

Dropped Terms (6064)	
Term	Role
the	DET
and	CONJ
to	PPOS
be	V
of	PPOS
a	DET

Documents (612)

All Matched

abstract

MACUMBA is an in-house-developed application for SAS® programming. It combines interactive development features of PC-SAS, the possibility of a client-server environment and unique state-of-the-art features that were always missing. This presentation covers some of the unique features that are related to SAS code debugging. At the beginning, special code execution modes are discussed. Afterwards, an overview of the graphical implementation of the single-step debugger for SAS macros and DATA step is provided. Additionally, the main pitfalls of development are discussed.

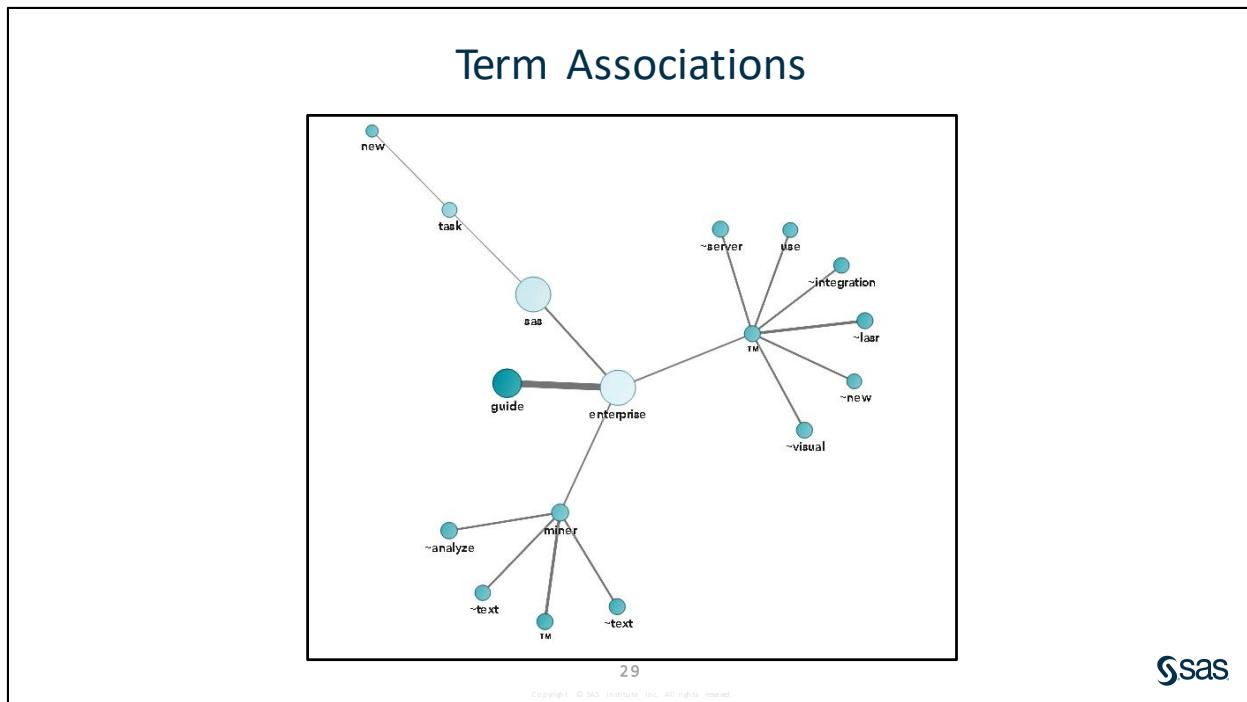
The goals for SAS developers to design applications include reporting accurate information, delivering in a timely manner, meeting business needs, and the presentation is easy to grasp. We design the report to meet those goals and hopefully to cover potential questions. One of the frequently asked questions is: I used to receive a session, say visitors from Japan, why I don't see that session for the week of March 14, 2011? Even though we don't need to code "No visitors from Japan due to Tsunami on March 11, 2011", we could at least provide a generic message like "No data returned for this session." so users...

Document 1 of 612

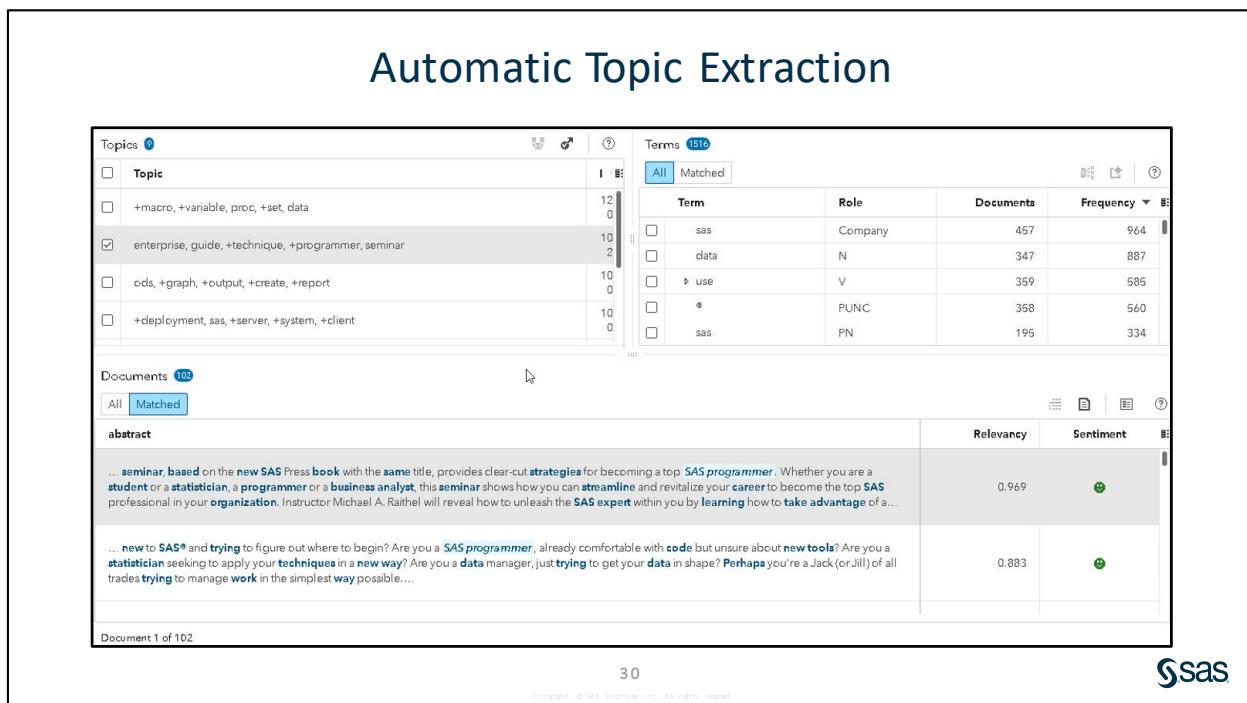
28



You can view term maps for a selected term to reveal term associations.



The Topics node produces derived topics and enables you to investigate the topics and extract documents that contain a topic.



Automatically Discover Hidden Topics: Machine Learning

- Automated machine discovery of core topics (themes) is provided.
- The most relevant documents can be ranked per topic.
- Topic can be further merged or split.
- Document-level sentiments from the Sentiment node can be included.
- Derived topics can be modified to create custom topics.

31



Copyright © SAS Institute Inc. All rights reserved.

Categorization Rule Generation

The screenshot shows the SAS Visual Text Analytics interface. On the left, there's a sidebar with 'Categories' and 'Textual Elements'. Under 'Categories', 'Business_Analytics' is selected, showing sub-categories 'No' and 'Yes'. Under 'Textual Elements', there's a table with columns 'String' and 'Role' containing entries like 'sas' (Company), 'data' (N), etc. In the center, there's a 'Edit a Category' panel with a code editor containing a complex logical expression. Below it, a message says 'Code is valid.' On the right, there's a 'Documents' pane showing a list of documents with columns 'Relevancy' and 'Sentiment', and a preview of the document content.

32



Copyright © SAS Institute Inc. All rights reserved.

Categorization Rule Generation

- Unique rule generation automatically discovers Boolean linguistic rules.
- Easy-to-understand rule definitions are provided.
- Testing of categorization accuracy and specificity occurs.
- Further customization or refinement by an analyst can achieve the desired results.

33

Copyright © SAS Institute Inc. All rights reserved.

Sentiment Analysis

Sentiment analysis can do the following:

- assign a sentiment score to each document and reflect positive, neutral, or negative sentiment.
- use proprietary rules that identify and analyze terms, phrases, and character strings that imply sentiment.
- accommodate user-supplied sentiment models created by SAS Sentiment Analysis.
- transfer sentiment scores to successor nodes. Thus, a Topics node displays sentiment scores for each document if it is preceded by a Sentiment node.

34

Copyright © SAS Institute Inc. All rights reserved.

Concepts, Terms, Topics, and Categories

- A *concept* is contained within a document and can be extracted from a document. A document can have multiple concepts, and the same concept can be represented multiple times.
- When the Concept node appears before the Text Parsing node, concepts are added to the *term* table.
- A *topic* and a *category* apply to an entire document. Topics and categories can be used to identify a document.
- When topics and categories are derived, topic terms and category rules can provide insight to help explain the contents of a document collection.

1.02 Multiple Answer Question

What is the main purpose of using SAS Visual Text Analytics?
(Select all that apply.)

- a. concept extraction
- b. topic extraction
- c. content categorization
- d. sentiment extraction

1.03 Multiple Choice Question

Which of the following nodes can be used to extract information from ***within*** a document?

- a. Concepts
- b. Text Parsing
- c. Sentiment
- d. Topics
- e. Categories

Copyright © SAS Institute Inc. All rights reserved.

SAS Visual Text Analytics

Summary of the Prominent Features of SAS Visual Text Analytics

- Integrated in SAS Viya with other SAS applications.
- Incorporates features of SAS Text Miner, SAS Enterprise Content Categorization, SAS Sentiment Analysis, and SAS Contextual Analysis.
- Contains a simple-to-use graphical user interface.
- Supports major world languages. (Additional languages will be included in later releases.)

Why Use SAS Visual Text Analytics?

- SAS Visual Text Analytics makes it possible for analysts to examine all of their content, interrogate it, and define classification schemes in a single, simple-to-use GUI.
- Visual Text Analytics eliminates laborious manual document reviews, removes manual bias errors, and regulates consistent content classification.
- Visual Text Analytics removes the requirement to create a training corpus in advance of defining classification rules (also known as *schemes* or *taxonomies*).

1.2 Language Challenges (Self-Study)

Objectives

- Be able to determine which writing system is relevant for the languages that are supported by the project.
- Examine language encoding issues that might impact a text analytics project.
- Ensure a minimal loss of information when converting documents that are stored in proprietary formats to one of the ISO encoding schemes that are supported by SAS.

44



One of the first warning messages that you are likely to encounter in your first text analytics project can be confusing and disconcerting.

WARNING: Some character data was lost during transcoding in the data set LIB.MEMBER.

This error message reveals that the original data used encoding that could not be translated to the native character encoding that is used by your computer system. For example, this could occur if you imported data that was created on a system that uses UTF-16 encoding, but your analytic server uses Latin-1 encoding. Some encoded characters from UTF-16 cannot be converted to Latin-1.

The good news is that these characters are likely to have little value with respect to the goals of the project. Transcoding problems are common when you perform a document analyses for data from only one country (for example, the USA, where the \$ (dollar sign) symbol is used), but you need to add documents from European sources where symbols such as £ (pound) or € (euro) are used. For example, the original ASCII encoding did not support £ (pound), € (euro), or ¢ (cents).

This section is classified as *self-study* because it is in the category of “fun facts that are not essential for success.” If you omit this section, it is unlikely that you will be less successful except for relatively rare cases. However, if you are on a path to become a text analytics expert, some information in this section can equip you to tackle more serious document conversion and data transcoding problems.

Encoding Language: Writing Systems

- SAS Visual Text Analytics is a solution for processing written communication. For audio recordings or other sound-based communication, the first step is to convert the sounds to a writing system using speech recognition technology. SAS currently does not provide a speech recognition product.
- Most world languages are encoded into a writing system. This course uses the English language alphabetic writing system. It is encoded using Latin-1 encoding (ISO 8859-1), which accommodates Western European languages, including English.

45


Copyright © SAS Institute Inc. All rights reserved.

Kiefer (2012) provides most of the essential information that is related to document processing with SAS software.

Encoding Language: Writing Systems

- Variations in English can pose challenges. For example, the English spoken in Australia, Canada, Ireland, Singapore, the United Kingdom, the United States of America, and so on, is influenced by customs, culture, monetary systems, and other factors. You might need to use specialized spelling dictionaries for each country of origin.
- In a multi-cultural country like the USA that has no official language, or in a two-language country like Canada, you might need to consider using composite dictionaries.
- Whatever actions are taken, ambiguities always exist. For example, the word boot could be foot apparel or part of an automobile.

46


Copyright © SAS Institute Inc. All rights reserved.

Word sense disambiguation is a critical problem in text analytics. The same set of symbols can have dramatically different meanings, even if international language challenges are considered.

A common example is *train*, which can be a noun or a verb. The variant *training* can be an adjective, noun, or a verb. A train could be a locomotive or part of a wedding dress, or something related to education, or it could be referring to pointing or aiming.

This discussion is reasonable for the English language, but some languages do not have parts of speech, so other forms of word sense disambiguation are possible.

Encoding Language: Writing Systems

Spoken languages are converted to writing systems that fit into one of three categories. The categories are presented below in ascending order of the typical cardinality of the written character set.

1. Alphabets: A character represents a phoneme (distinct unit of sound), which is classified as a consonant or vowel. Alphabets usually have between 20 and 40 characters.
2. Syllabaries: Characters represent entire syllables. “Syllabaries usually have between 50 and 200 different characters.” (Kiefer, 2012)
3. Logographic or Ideographic: Characters represent words or morphemes. Logographic writing systems usually have thousands of characters.

Linguistic morphology is the study of morphemes used in a language. A morpheme is the smallest grammatical unit in a language. A morpheme can be a word, which would be classified as a free morpheme because it can stand on its own. A morpheme can be a prefix (for example, the “un” in unlikely, meaning “not likely”). The “un” morpheme is a bound morpheme because it is bound to the root “likely.” Prefixes and suffixes are just two examples of bound morphemes.

Early research about computational linguistics was often focused on a single language. As research began to include multiple languages, the work of archeologists and anthropologists added valuable insight. Researchers such as Michael Coe needed to decipher ancient scripts, like the Maya script, to decipher what was written in previous centuries. Establishing taxonomies related to written language helped decide how to tackle a deciphering project.

The three-level classification scheme above has some variants. For example, an *abugida* is intermediate between an alphabet and a syllabary. An *abjad* is a language with well-defined consonant sounds, but vowel sounds must be inferred by the reader.

continued...

Encoding Writing Systems: Alphabets

- Alphabets can usually be encoded in a single byte that has eight bits. Alphabets allow a maximum of 256 unique characters.
- ASCII and EBCDIC are two early encoding systems. For example, in ASCII, the letter A (uppercase A) is encoded as the decimal number 65, but in EBCDIC, it is encoded as the decimal number 193. In ASCII, uppercase letters sort ahead of lowercase letters. The reverse is true for EBCDIC.
- *Latin encoding* is an extension of ASCII encoding with specific extensions to accommodate unique diacritic symbols. More than 30 world languages use Latin encoding.
- The original ASCII code was a 7-bit code that allowed a maximum of 128 characters. Extended ASCII occurred later and was an 8-bit code. Extended ASCII was not part of a standard, so variations existed (for example, to accommodate the English pound symbol (£)).



The first IBM mainframe computers supported EBCDIC encoding. EBCDIC is a full 8-bit encoding scheme with 256 possible values.

ASCII was supported on early minicomputers and microcomputers. Computer vendors could support a custom extension to ASCII to use all eight bits. Consequently, data that was created on one computer and then moved to another computer could produce some strange documents.

A letter with an accent symbol could become a graphic symbol for the corner of a rectangle. Because a letter with an accent might be used by text analytics software, but a graphic symbol would be ignored, this created special problems. Most modern software uses one of the accepted ISO encoding schemes.

Encoding Writing Systems: Alphabets

- The SAS DATA step function RANK returns the ASCII decimal value for a specified character value (for example, `D=rank('A')`; returns D=65).
- The SAS DATA step function BYTE returns the character value that is associated with a decimal number (for example, `L=byte(D)`; returns L='A' when D=65).
- SAS supports single-byte character sets (SBCS), double-byte character sets (DBCS), and multi-byte character sets (MBCS). DBCS and MBCS systems are more appropriate for logographic writing systems.
- Although the original Latin accommodated only 26 letters, the extension to uppercase and lowercase, the addition of decimal digits, and the edition of special symbols for punctuation, monetary systems, and other purposes extended the system to approximately 100 *printable* characters. The addition of diacritic symbols is hindered by the 256-character limit of SBCS systems.

49

Copyright © SAS Institute Inc. All rights reserved.

Encoding Writing Systems: Logographic Systems

- Logographic systems can have thousands of characters.
- Chinese is an open logographic system that has between 40,000 to more than 100,000 characters. “A 3,000 character vocabulary seems to be sufficient for day-to-day communication.” (Kiefer, 2012)
- ISO (International Organization for Standardization) Unicode encoding supports more than 100,000 characters. Unicode is implemented as UTF-8, UTF-16, or UTF-32. Only UTF-8 is backward compatible with ASCII, which means that ASCII encoding is a special-case subset of UTF-8 encoding.

50

Copyright © SAS Institute Inc. All rights reserved.

The development of Unicode is an attempt to provide a superset of encoded symbols that support all languages, but this ideal is not yet achieved. For this course, all documents are in English, and encoding uses the Latin-1 encoding scheme. Because all the document collections are already processed for the course, no unusual encoding problems are likely to occur.

Character Strings

- Individual characters are stored as part of a character string.
- For many relational databases, a data table with character string columns is defined so that the character string has a fixed length, although variable length strings are often supported (for example, VARCHAR in SAS DS2).
- SAS pads fixed-length character strings with blanks (ASCII 32). Other languages use other padding conventions (for example, NULL (ASCII zero)).
- Boolean expressions that involve character strings often resolve to an unanticipated FALSE value, because the programmer fails to understand padding.
- Many pitfalls are related to processing character strings.
- Terms in Visual Text Analytics are character strings.

The SAS DATA step supports a variety of character string functions (for example, STRIP, TRIM, LEFT, SCAN, SUBSTR, and FINDW). These functions facilitate the processing of fixed-length character strings. Consider the following SAS DATA step code:

```
data _null_;
  attrib Term1 Term2 length=$8
        Terms length=$17;
  Term1="dog";
  Term2="cat";
  Terms=Term1||' '||Term2;
  Success=0;
  if (Terms="dog cat") then Success=1;
  put _ALL_;
  Success=0;
  Terms=catx(' ',Term1,Term2);
  if (Terms="dog cat") then Success=1;
  put _ALL_;
run;
```

The log file reveals the following.

Term1=dog Term2=cat Terms=dog	cat Success=0 _ERROR_=0 _N_=1
Term1=dog Term2=cat Terms=dog cat	Success=1 _ERROR_=0 _N_=1

The blanks that are used to pad the original **Term1** and **Term2** character variables were preserved by the concatenate operator, but the CATX function removes leading and trailing blanks from the arguments.

Numerous SUGI and SAS Global Forum papers can provide insight into some of the subtle character-string-processing challenges.

1.3 Lesson Summary

Manual text analysis efforts suffer from inadequacy due to human subjectivity and inconsistency, as well as the time that is required to read each document and classify it. SAS Visual Text Analytics eliminates the need to manually review documents, develop a training corpus, and manually develop taxonomies. After data are registered to the software, natural language processing (NLP) is automatically performed. This includes tokenization, term frequency counts, stemming, and part-of-speech tagging. Combining statistical machine learning with an extensive array of linguistic operators and prebuilt concept definitions, the text analyst is empowered to customize the automatically discovered results within a single, visual, guided application.

From a practical standpoint, SAS Visual Text Analytics is a single application that brings together the techniques that are used in text mining, categorization, contextual extraction, sentiment analysis, and topic derivation. This enables analysts to apply the appropriate analysis to meet their specific use cases without needing to switch applications or move data around. Using SAS Visual Text Analytics is more productive when the volume of documents is no longer economical to manually review and classify (typically greater than 500 documents) or when errors associated with manual tagging result in inconsistent, untrustworthy, or misinformed business understanding.

1.4 Solutions

Solutions to Activities and Questions

1.02 Multiple Answer Question

What is the main purpose of using SAS Visual Text Analytics?
(Select all that apply.)

- a. concept extraction
- b. topic extraction
- c. content categorization
- d. sentiment extraction

37

Copyright © SAS Institute Inc. All rights reserved.



1.03 Multiple Choice Question

Which of the following nodes can be used to extract information from ***within*** a document?

- a. Concepts
- b. Text Parsing
- c. Sentiment
- d. Topics
- e. Categories

39

Copyright © SAS Institute Inc. All rights reserved.



Lesson 2 SAS® Visual Text Analytics Demonstrations

2.1 Importing Document Collections	2-3
Demonstration: Importing and Converting Document Files	2-4
2.2 Creating a Project with No Predefined Concepts.....	2-14
Demonstration: Creating a SAS Visual Text Analytics Project with No Predefined Concepts	2-15
Practice.....	2-39
2.3 A Project with Custom Concepts	2-40
Demonstration: Working with Custom Concepts	2-41
Practice.....	2-54
2.4 Lesson Summary.....	2-56
2.5 Solutions	2-57
Solutions to Practices	2-57
Solutions to Activities and Questions.....	2-60

2.1 Importing Document Collections

Objectives

- Create a SAS Visual Text Analytics project and import the Drug Reports text files.

3

Copyright © SAS Institute Inc. All rights reserved.





Importing and Converting Document Files

This demonstration illustrates how to import document files for analysis.

Overview of Document Conversion

In SAS Visual Text Analytics 8.3, individual documents saved in a folder can be converted to SAS data files using the document conversion utility. This utility can extract text and metadata from a collection of documents saved in a caslib and write this information to a data table in a caslib. SAS Visual Text Analytics 8.3 supports converting the following document formats:

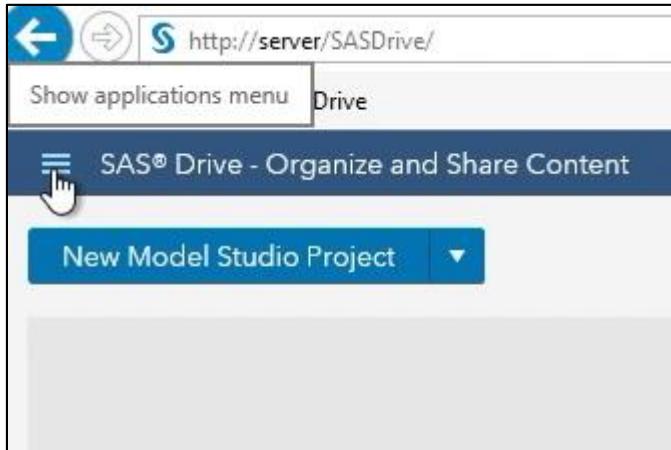
<i>Document Formats That Can Be Converted</i>	
Format	File Extensions
HTML	.html
XML	.xml
MS Office	.doc, .docx, .xls, .xlsx, .ppt, .pptx
Open Document Format	.odt, .ods, .odp
iWork document format	.pages, .numbers, .key
WordPerfect	.wpd
PDF	.pdf
Electronic publication format	.epub
Rich Text Format	.rtf
Text formats	.txt, .csv
Mail formats	.mbox, .msg, .tnef, .pst

Steps in Document Conversion

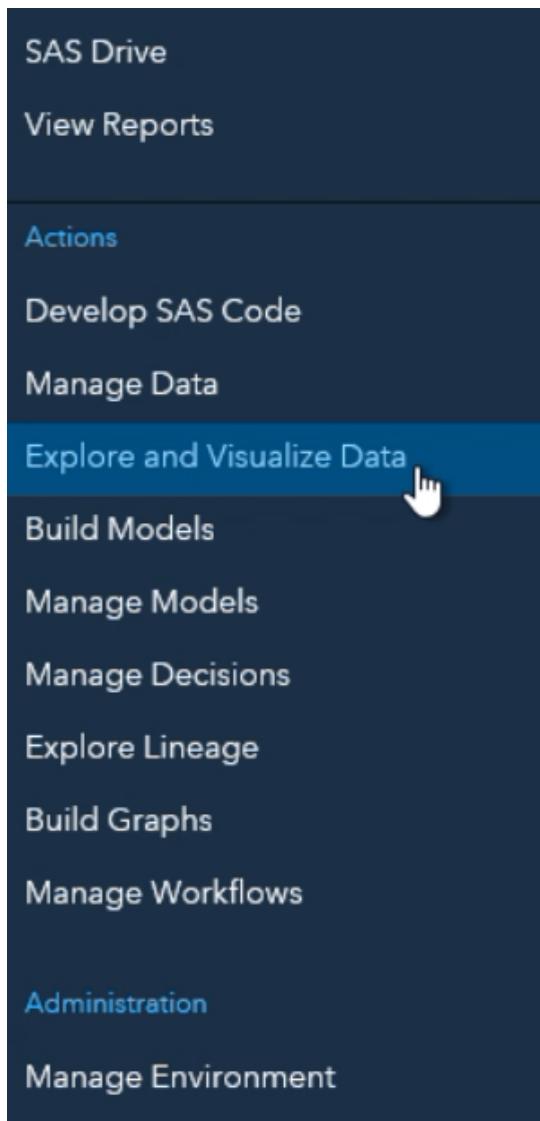
1. Drug reports are on the server in the directory /workshop/VTXT/DrugReports. Your instructor will tell you how to access SAS Drive from a specific web browser. (For example, one implementation in Internet Explorer provides a SAS Drive menu selection.)



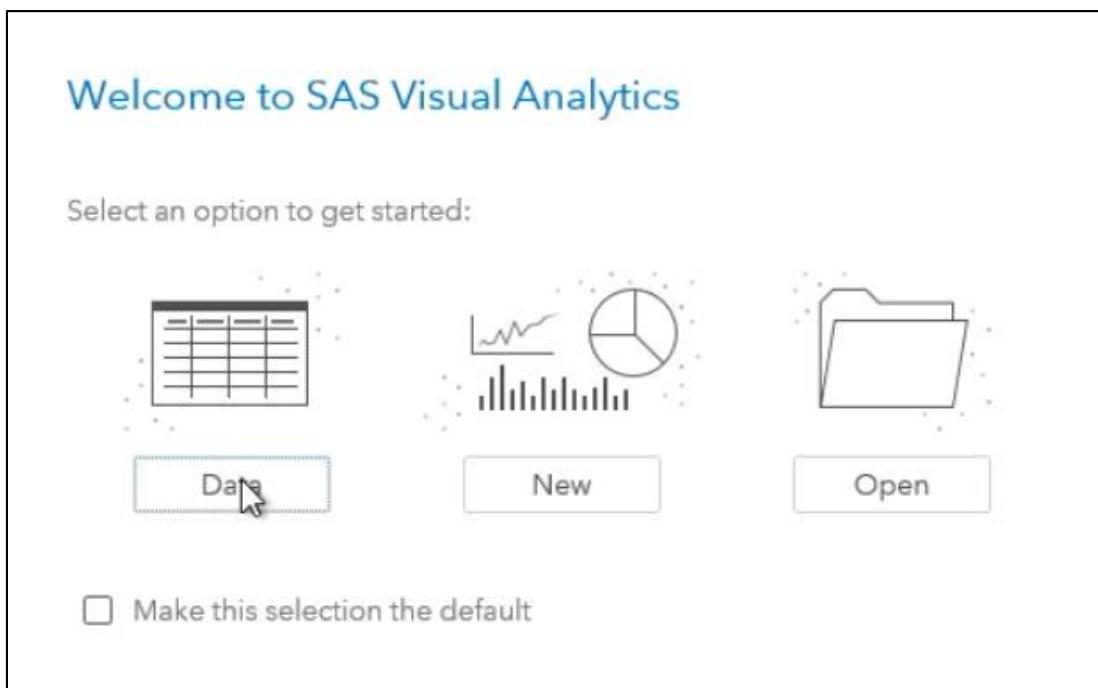
2. Click the **Show applications** menu.



3. Open Data Explorer.



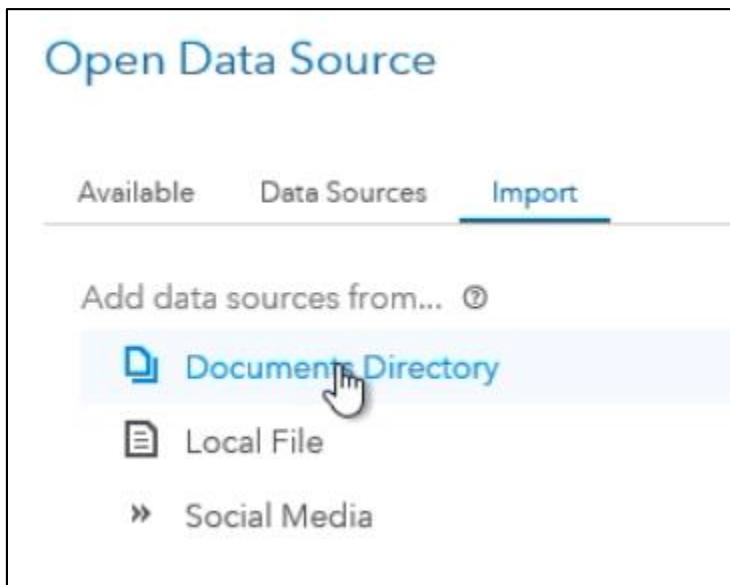
4. Click **Data**.



5. Click **Import**.

The screenshot shows the 'Open Data Source' dialog. At the top, there are tabs: 'Available' (underlined), 'Data Sources', and 'Import' (with a cursor pointing at it). Below the tabs is a 'Filter' input field with a magnifying glass icon. To the right of the filter are three small icons: a funnel, a gear, and a question mark. The main area lists five data sources: ASRS (12/03/18 02:07 PM • student), ASRS_ID (12/03/18 02:07 PM • student), ASRS_NEWREPORTS (12/03/18 02:07 PM • student), ASRS_RDU_SNA (12/03/18 02:07 PM • student), and COMPLAINTSM (12/03/18 02:07 PM • student). Each item has an 'i' icon to its right.

6. Select **Documents Directory**.



7. Click the **Connect** icon.



8. In the Connection Settings window, enter information as shown below.

Connection Settings

Name: * DrugReports2 Server: cas-shared-default

Type: File system Source type: PATH

Persist this connection beyond the current session.

Settings Advanced

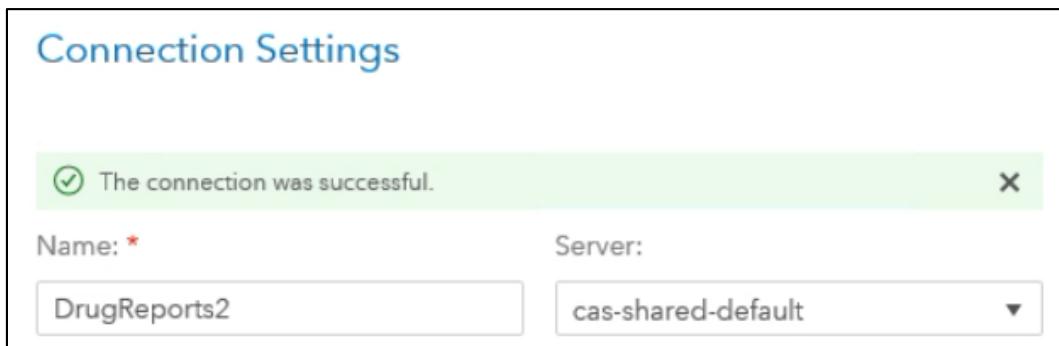
Specify the PATH connection information.

Path: */workshop/VTXT/DrugReports

Description:

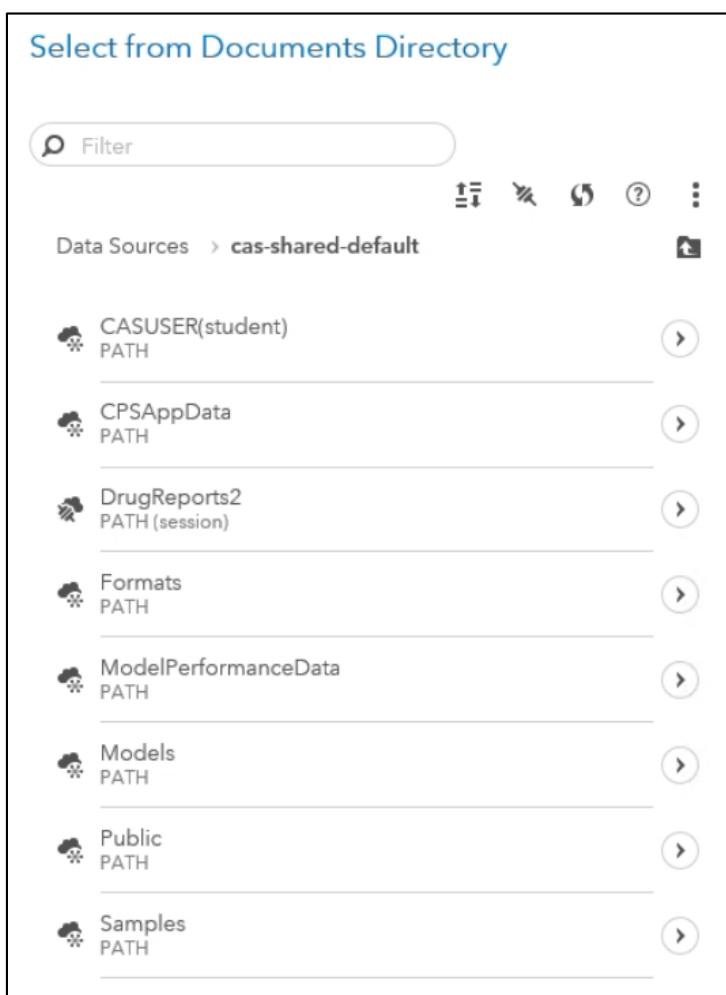
Include subdirectories

9. Click **Test connection**.



10. If the connection is successfully established, click **Save**.

11. Click **cas-shared-default**.



12. Select **DrugReports2 PATH (session)**. Click **Select**.

Documents in DrugReports2

Target table name: * Target destination: *

Table A Required field Find

If target table name exists:

Cancel import
 Replace file

Label:

Enter label

File Specifications Advanced

Filter file extensions:

txt, pdf, etc

Recurse through directories

13. Edit the entries in the Documents in DrugReports2 window as follows:

Documents in DrugReports2

Target table name: * Target destination: *

Depression_drugReports cas-shared-default/Public Find

If target table name exists:

Cancel import
 Replace file

Label:

Enter label

File Specifications Advanced

Filter file extensions:

txt, pdf, etc

Recurse through directories

The **cas-shared-default/Public** target destination is obtained by selecting Public from the File menu.

Select Target

Filter

Data Sources > **cas-shared-default**

- CASUSER(student) PATH
- CPSAppData PATH
- Formats PATH
- ModelPerformanceData PATH
- Models PATH
- Public PATH**
- Samples PATH

14. Select **Import Item**. The documents will be imported, and the new documents table will be created.

Documents in DrugReports2

The table was successfully imported on Dec 5, 2018 08:00 PM and is ready for use.

Target table name: * Depression_drugReports

Target destination: * cas-shared-default/Public

Find

15. Access the table through the Available window.

Open Data Source

Available Data Sources Import

Filter

ASRS	12/03/18 02:07 PM • student	
ASRS_ID	12/03/18 02:07 PM • student	
ASRS_NEWREPORTS	12/03/18 02:07 PM • student	
ASRS_RDU_SNA	12/03/18 02:07 PM • student	
COMPLAINTSM	12/03/18 02:07 PM • student	
DEPRESSION_DRUGREPORTS	12/05/18 08:00 PM • student	
DRUG_REPORTS	12/03/18 02:07 PM • student	
MOVIES_PLUS	12/03/18 02:07 PM • student	
SASGF_2013_PAPERS_CL	12/03/18 02:07 PM • student	

16. Select the **Sample Data** view.

DEPRESSION_DRUGREPORTS

Details Sample Data Profile

Sample rows: 100

path	fileName	fileType	content
/workshop/VTXT/DrugReports	file1.docx	docx	This medication made me gain 40 pound...
/workshop/VTXT/DrugReports	file10.txt	txt	have been on 10mg prexifan for 6 month...
/workshop/VTXT/DrugReports	file100.txt	txt	At first I liked it, I took it along with Escala...
/workshop/VTXT/DrugReports	file1000.txt	txt	This drug caused extreme nausea and di...
/workshop/VTXT/DrugReports	file1001.txt	txt	When I started using Abidal I was running...
/workshop/VTXT/DrugReports	file1002.txt	txt	I had horrible hot flashes/sweating and irr...
/workshop/VTXT/DrugReports	file1003.txt	txt	I can definately tell when I havent taken ...
/workshop/VTXT/DrugReports	file1004.txt	txt	This is my second day on Abidal. My first ...
/workshop/VTXT/DrugReports	file1005.txt	txt	making me get stomach cramps, crying al...
/workshop/VTXT/DrugReports	file1006.txt	txt	I take 60 mg daily (take it during the mor...
/workshop/VTXT/DrugReports	file1007.txt	txt	I felt nothing with this drug. After 4 weeks...
/workshop/VTXT/DrugReports	file1008.txt	txt	I was prescribed this mainly for depressio...

17. The imported data are loaded in memory and are available under Available data sources.

End of Demonstration

2.2 Creating a Project with No Predefined Concepts

Objectives

- Create a SAS Visual Text Analytics project. Use the SAS Global Forum 13 abstract data with no predefined concepts.



Creating a SAS Visual Text Analytics Project with No Predefined Concepts

This demonstration introduces SAS Visual Text Analytics features that enable users to automatically extract topics and develop tools that automatically classify categories of interest. This demonstration has five objectives:

- exploring and preparing a document collection
- using the default functionality to create a SAS Visual Text Analytics project with the demonstration data
- using no predefined concepts to explore the derived terms
- exploring the automatically generated topics and the associated documents, and promoting the most intriguing topics to categories
- exploring the rules that are generated by SAS Visual Text Analytics for identifying and categorizing the documents that belong to the relevant categories

Using SAS tools in SAS Visual Analytics to explore text data is highly recommended. The data set for this demonstration is **SASGF_2013_Papers_CL**. The partial list of the revised data in Model Studio is presented below. The variables of interest are **Paper_number** (document ID), **abstract** (text variable), and a binary pre-classified dummy variable, **Business_Analytics** (yes or no).

Variables Table Data View

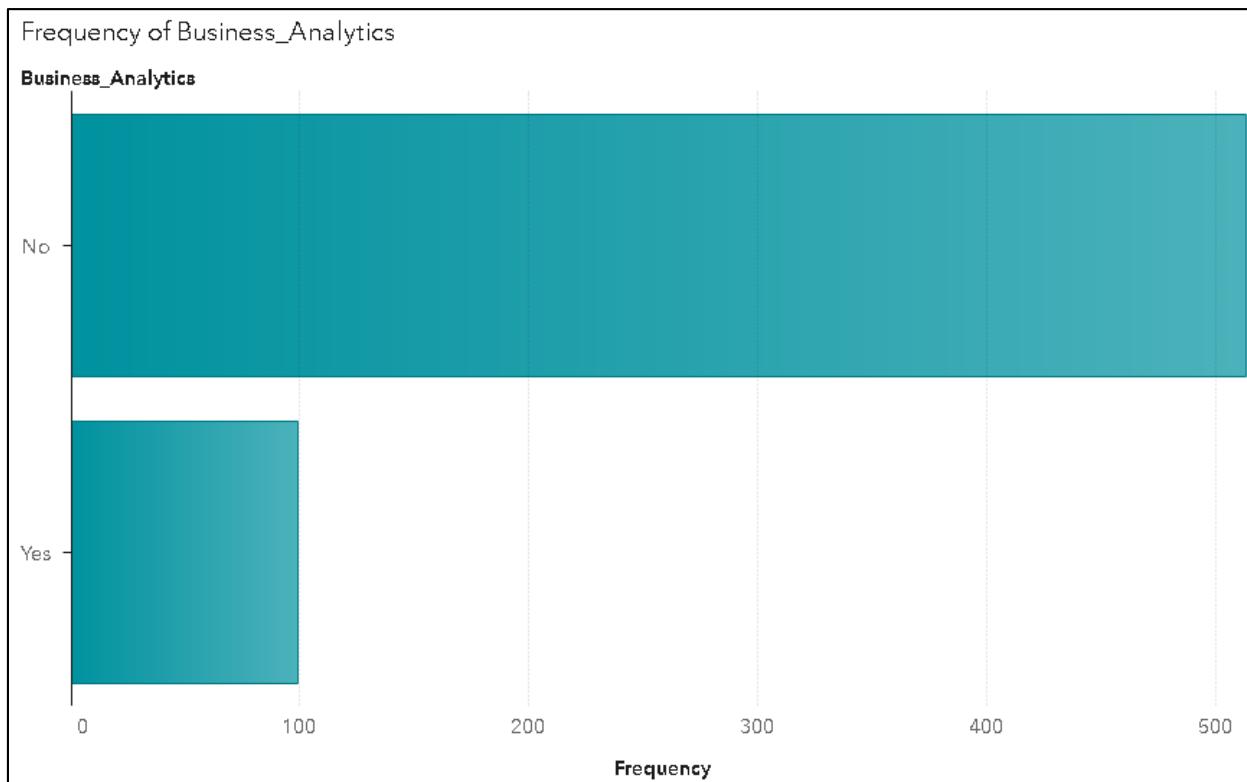
Variable Name	Type	Role	Display Variable
_uniqueid	Numeric	Key	
abstract	Character	Text	
Business_Analytics	Character	Category	
authors	Character		
paper_number	Character		
presenter	Character		
section	Character		
title	Character		

7
Copyright © SAS Institute Inc. All rights reserved.

View Table Data View

The screenshot shows the SAS Model Studio interface with the title "View Table Data View". The main area displays a data table titled "SASGF Abstracts Custom Concepts". The table has columns: paper_number, section, Business_Analy..., authors, title, abstract, presenter, and _uniqueid. A green box labeled "View Table" points to the table icon in the toolbar. Another green box labeled "Category" points to the "Business_Analy..." column header. A third green box labeled "Document" points to the abstract text in the table.

If you select **Explore and Visualize Data** from (the Show applications menu), and select the **SASGF Abstracts** data set, you can easily find the distribution of the **Business_Analytics** variable.



Virtual Lab Preliminaries

In the Virtual Lab, many of the course demonstrations and practices have been “pre-cooked” to facilitate maximal coverage of training material. Although SAS Visual Text Analytics performs very well for large document collections, the actions of document parsing, term extraction, and other elements of natural language processing can be very computationally intensive. Empirical studies suggest that text analysis for a fixed number of observations can take substantially longer than complex predictive modeling for the same number of observations. For example, parsing the Aviation Safety Reporting System (ASRS) document collection, which has 21,519 documents, can take several minutes. Valuable class time would be wasted if time were spent parsing every document collection. Consequently, most demonstrations include projects that have already been created, with pipelines that have already been run. Nonetheless, all steps for creating and configuring a project are included in the course notes.

You begin most demonstrations in SAS Drive.

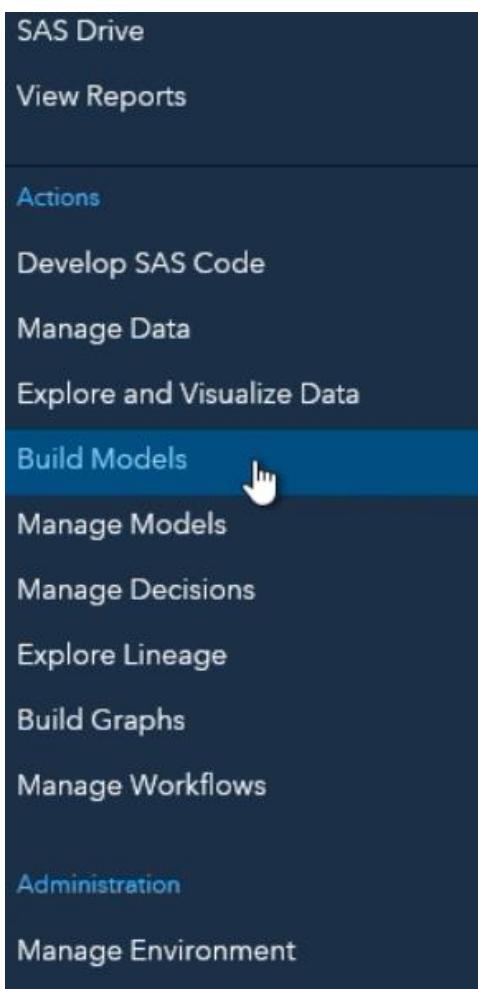
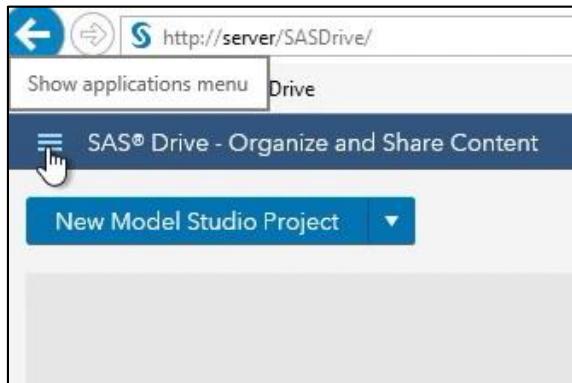
My Folder	
Summary	Comments
Show: Details	
Type: Folder	
Modified by: Me	
Date modified: September 14, 2018 03:12:51 PM	
Date created: August 1, 2018 02:32:44 PM	
Location:	

The SAS Drive interface provides tabs. The Build Models tab is the usual selection for accessing Model Studio. In the initial screen interface, the tabs provide access to existing results from previous analyses. You must select **Edit** from the so-called “snowman” menu (⋮) if you want to configure and run pipelines in this environment. You can invoke Model Studio through the Show applications menu the same way you did in the previous demonstration.

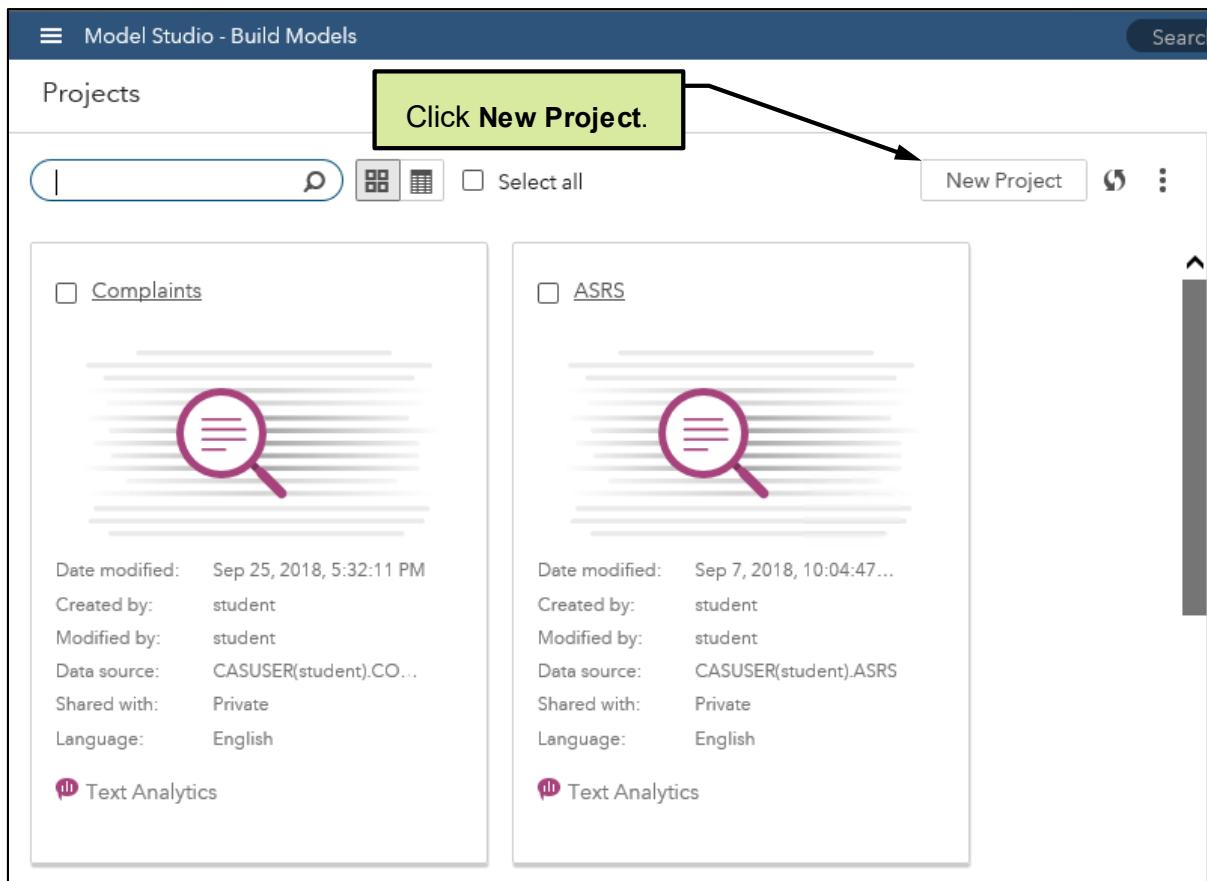
Creating a SAS Visual Text Analytics Project (Step 1 of 3)

Follow the steps below to create the Visual Text Analytics project (step 1).

1. Sign in to SAS Drive and open Model Studio. Recall that Model Studio is accessed using the Show applications menu and selecting **Build Models**.



2. In Model Studio, click **New Project**.



3. Enter a project name. Accept or revise the project name that is suggested by SAS Visual Text Analytics. You can use a more descriptive name such as **SASGF Abstracts No Concepts** in the **Name** field.

4. Enter the information and make the selections as shown below.

New Project

Name: *

Type: *

Data source: *

Project language: *

Description:

- a. Enter **SASGF Abstracts No Concepts** in the **Name** field.
- b. Select **Text Analytics** in the **Type** field.
- c. Click **Browse** and navigate to **CASUSER(student).SASGF_2013_PAPERS_CL** for the data source. Click **OK** after the table is selected.
- d. Select **English** in the **Project language** field.

A description is optional.

5. Click **Save**.

The variables table appears.

Variable Name
__uniqueid__
abstract
authors
Business_Analytics
paper_number
presenter
section
title

You must assign a Text variable in order to run a pipeline.

SASGF_2013_PAPERS_CL
Columns: 8
Rows: 612
Label: Not available
Location: cas-shared-default/Analytics_Project_35ca74f9-ecbf-42f2-8fc-b800fa021f4a

6. Select **abstract**. Using the pane on the right, assign the role **Text** for this variable.

abstract

Role:

Text

None

Category

Text

7. Select the variable **Business_Analytics**. Using the pane on the right, assign the role **Category** for this variable.

Business_Analytics

Role:

Category

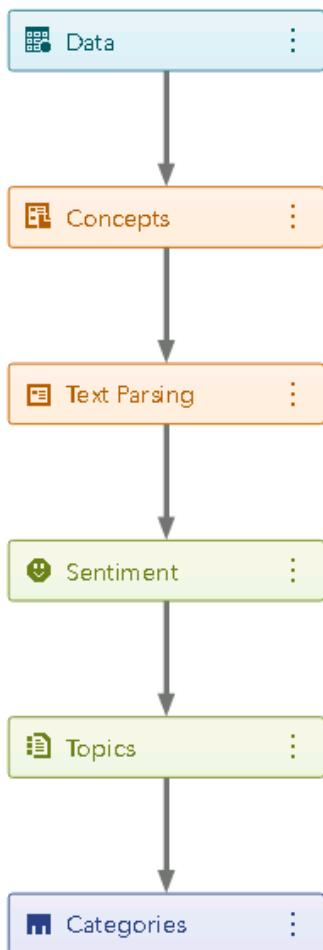
None

Category

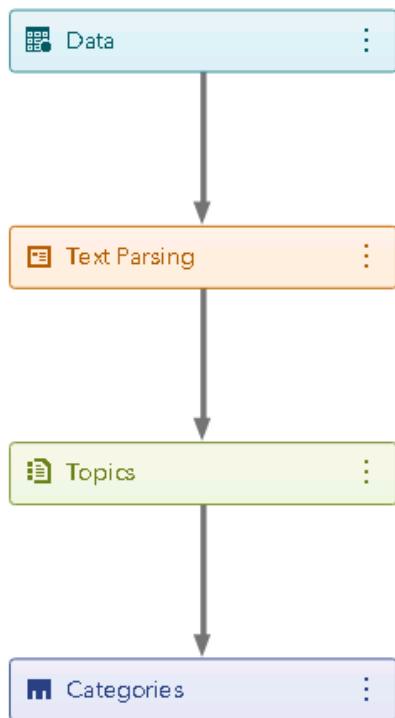
Text

Model Studio - Build Models				
SASGF Abstracts No Concepts				
Data	Pipelines			
»	<input type="text"/> Filter 			
<input type="checkbox"/>	Variable Name	Type	Role	
<input type="checkbox"/>	__uniqueid__	Numeric	Key	
<input type="checkbox"/>	abstract	Character	Text	
<input checked="" type="checkbox"/>	Business_Analytics	Character	Category	
<input type="checkbox"/>	authors	Character		
<input type="checkbox"/>	paper_number	Character		
<input type="checkbox"/>	presenter	Character		
<input type="checkbox"/>	section	Character		
<input type="checkbox"/>	title	Character		

8. Select **Pipelines** (in the upper left to the right of **Data**). The default Text Analytics pipeline appears. If you set up your project to be the Text Analytics type, you can access SAS Visual Text Analytics with Model Studio.



9. For this project, no concepts are used, and sentiment analysis is not relevant. Delete the **Concepts** node and the **Sentiment** node. The remaining nodes are reconnected automatically.

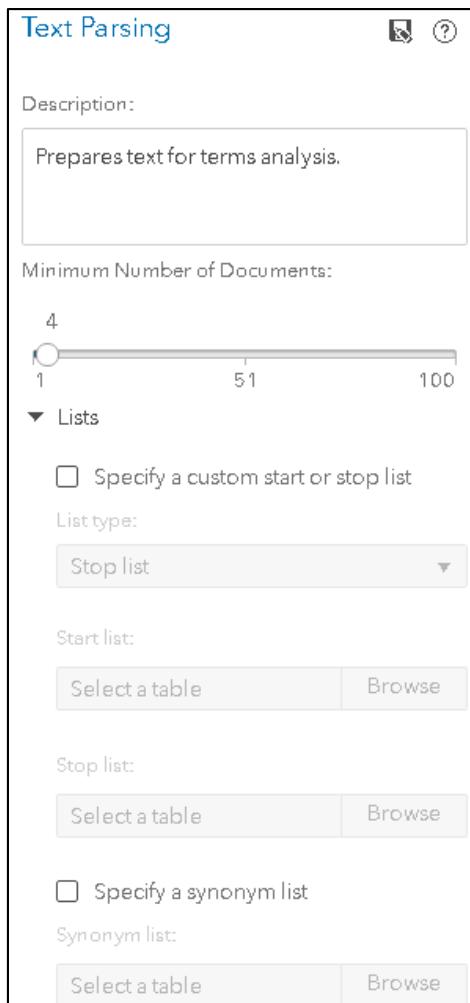


Defining Lists (Step 2 of 3)

Now you select optional term lists to include in your project. Start lists and stop lists enable you to control which terms are used or which terms are not used, respectively. In SAS Visual Text Analytics, you can use a start list or a stop list, but not both.

Note: 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 the terms that are included in that list are used in the analysis. A *stop list* is a data set that includes a list of terms to exclude from the analysis results, such as terms that contain little information or that are outside the realm of your analysis. A default stop list is provided for each of the languages that SAS Visual Text Analytics supports.

1. Select the **Text Parsing** node. The Text Parsing options appear to the right of the pipeline. If the options do not appear, click  (Options). The options menu is shown below.



With the default settings, a default stop list is used for the language that is selected for the project. There is no default synonym table. Also, the Minimum Number of Documents property specifies the minimum number of documents that must contain a term before it can be in the start list. The default value is 4, so if a term appears in only three documents, it is automatically assigned to the stop list, regardless of membership in either the specified start or stop list.

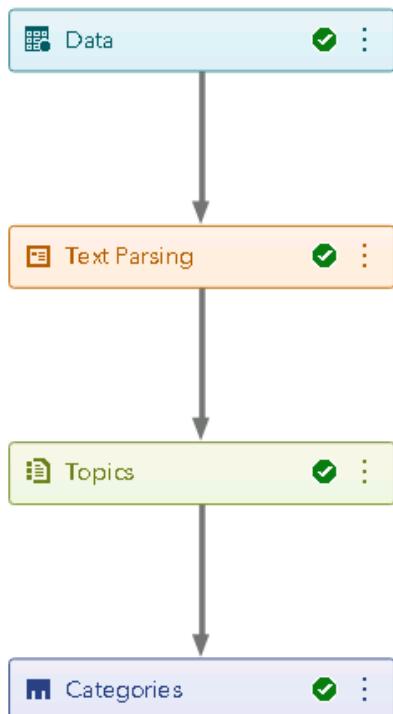
Note: A *synonym list* is a SAS data set that identifies pairs of terms that should be treated as a single term for analysis. The data set can include both a term and different forms of that term, including misspellings or abbreviations. For example, you can specify that the words *advert* and *advertising* should be treated as the term *advertisement*. You do not use a synonym list for this project, so do not select the **Specify a synonym list** check box.

2. Default settings are used, so no further steps are required.

Running the Pipeline and Examining the Results (Step 3 of 3)

- Run the entire pipeline. Right-click the last node and click **Run**. You can also click  (Run) to the left of the pipeline.

When the project is complete, a message appears. If the run is successful, all the nodes display a green circle with a white check mark.



- Right-click the **Text Parsing** node and click **Open**.

Kept Terms (1511)			
Term	Role	Documents	Frequency
sas	PN	487	1298
data	N	347	887
use	V	359	585
the	PUNC	358	560
paper	N	236	292
new	A	141	214
provide	V	169	202
create	V	139	192

Dropped Terms (7958)			
Term	Role	Documents	Frequency
the	DET	563	2668
and	CONJ	576	2001
to	PPOS	567	1820
be	V	487	1510
of	PPOS	500	1463
a	DET	447	1124
in	PPOS	432	895
for	PPOS	381	707

Documents

All (612) Matched Search

abstract

MACUMBA is an in-house-developed application for SAS® programming. It combines interactive development features of PC-SAS, the possibility of a client-server environment and unique state-of-the-art features that were always missing. This presentation covers some of the unique features that are related to SAS code debugging. At the beginning, special code execution modes are discussed. Afterwards, an overview of the graphical implementation of the single-step debugger for SAS macros and DATA step is provided. Additionally, the main pitfalls of development are discussed.

Our organization has been utilizing Google Drive (previously Google Docs) to keep project documentation centrally located for ease of access by any user on any platform. Up to this point, SAS® developers had to manually import or export data sets to or from flat files or Microsoft Excel in order to update data stored in the cloud.

Document 1 of 612

3. In the Filter window for the Kept Terms pane, enter **enterprise**. Select the **enterprise** check box next to the entry with the role PN (proper noun). You can also scroll down in the Kept Terms list to find the **enterprise** entry.

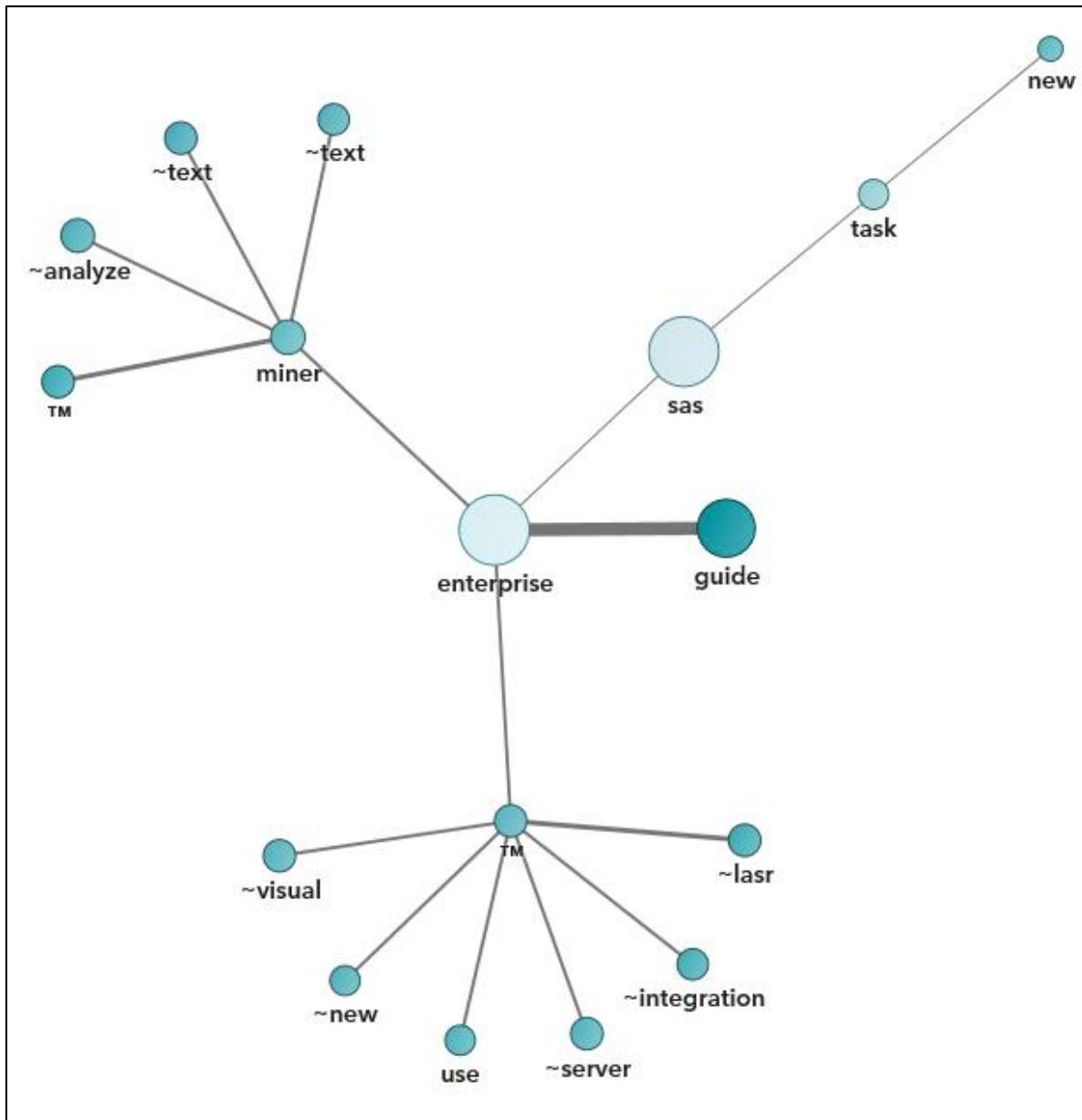
Filter:

Term	Role	Documents	Frequency
<input checked="" type="checkbox"/> enterprise	PN	69	115
<input type="checkbox"/> > enterprise	N	16	19

Scroll:

Term	Role	Documents	Frequency
> create		137	172
<input type="checkbox"/> proc	PN	104	191
<input type="checkbox"/> > user	N	110	178
<input type="checkbox"/> > model	N	71	148
<input type="checkbox"/> > procedure	N	98	139
<input type="checkbox"/> not	ADV	109	130
<input type="checkbox"/> > macro	N	62	126
<input type="checkbox"/> > application	N	80	118
<input type="checkbox"/> > time	N	82	117
<input checked="" type="checkbox"/> enterprise	PN	69	115
<input type="checkbox"/> > technique	N	76	115

4. Click  (Term Map). If you do not have enough screen space, you see the “snowman” icon , and you select Show term map from a drop-down menu.



A term map is useful for identifying associations between terms. The term map is most useful when it identifies associations that were previously unknown. Even when associations are known, a measure called **information gain** can be calculated to estimate the strength of the association. The thickness of the connectors between terms visually displays the relative information gain. Thicker lines imply higher information gain, which suggests a stronger association. Multiple concepts are related to the term **enterprise**: SAS Enterprise Guide, SAS Enterprise Miner, SAS enterprise software, and general enterprise software and solutions.

5. Close the term map.
6. Close the Text Parsing results.

7. Right-click the **Topics** node and click **Open**.

The screenshot shows the Model Studio - Build Models interface. In the top navigation bar, it says "SASGF Abstracts No Concepts > Topics". On the right, there are buttons for "Search", "Run Node", "Close", and a user icon labeled "student".

Topics (9)

<input type="checkbox"/>	Topic	Created by	Documents
<input type="checkbox"/>	+macro, +variable, proc, sql, data	System	95
<input type="checkbox"/>	enterprise, guide, +server, +user, +process	System	82
<input type="checkbox"/>	+model, +regression, +model, proc, model	System	79
<input type="checkbox"/>	+customer, +business, +company, text, +analyze	System	70
<input type="checkbox"/>	+report, ods, +output, web, +device	System	64

Terms

All (1511) Matched	Filter	Term	Role	Documents	Frequency
All (1511) Matched	Filter	sas	PN	487	1298
All (1511) Matched	Filter	data	N	347	887
All (1511) Matched	Filter	use	V	359	585
All (1511) Matched	Filter	@	PUNC	358	560
All (1511) Matched	Filter	paper	N	236	292
All (1511) Matched	Filter	new	A	141	214

Documents

All (612) Matched Search

abstract

MACUMBA is an in-house-developed application for SAS® programming. It combines interactive development features of PC-SAS, the possibility of a client-server environment and unique state-of-the-art features that were always missing. This presentation covers some of the unique features that are related to SAS code debugging. At the beginning, special code execution modes are discussed. Afterwards, an overview of the graphical implementation of the single-step debugger for SAS macros and DATA step is provided.

Our organization has been utilizing Google Drive (previously Google Docs) to keep project documentation centrally located for ease of access by any user on any platform. Up to

Document 1 of 612

The nine topics are shown below.

<input type="checkbox"/>	Topic	Created by	Documents
<input type="checkbox"/>	+macro, +variable, proc, sql, data	System	95
<input type="checkbox"/>	enterprise, guide, +server, +user, +process	System	82
<input type="checkbox"/>	+model, +regression, +model, proc, model	System	79
<input type="checkbox"/>	+customer, +business, +company, text, +analyze	System	70
<input type="checkbox"/>	+report, ods, +output, web, +device	System	64
<input type="checkbox"/>	analytics, visual, +new, sas visual analytics suite, +suite	System	50
<input type="checkbox"/>	+job, management, studio, +new, dataflux	System	50
<input type="checkbox"/>	+graph, ods, +feature, gtl, graphics	System	49
<input type="checkbox"/>	workshop, hands-on experience, +participant, hands-on, +experience	System	21

8. Select the **+customer, +business, +company, text, +analyze** check box. Topics are identified using the five terms that have the largest relevancy score within that topic.

9. In the Terms window, click **Matched**.

Terms					
	Term	Relevancy ▾	Role	Documents	Frequency ▾
<input type="checkbox"/>	▷ customer	0.233	N	50	96
<input type="checkbox"/>	▷ business	0.140	N	63	90
<input type="checkbox"/>	▷ company	0.133	N	27	39
<input type="checkbox"/>	text	0.132	N	35	45
<input type="checkbox"/>	▷ analyze	0.114	V	55	61
<input type="checkbox"/>	text	0.111	PN	14	16
<input type="checkbox"/>	▷ sentiment	0.110	N	9	22
<input type="checkbox"/>	miner	0.104	PN	25	29
<input type="checkbox"/>	data	0.104	N	347	887
<input type="checkbox"/>	analytics	0.102	N	24	35

In the start list for this collection, 160 of the 1,511 terms have relevancy weights that are greater than the term cutoff that is specified in the Topics settings window. Terms are sorted by descending relevancy score. As expected, the top five relevancy scores correspond to the terms that are used to name the topic. Terms with a plus sign have stemmed terms that are indicated in the Terms table by an arrow. If you click the arrow, the stemmed terms are displayed, and if a synonym list is used, the synonyms are also displayed.

customer
customers
customer

10. Examine the Documents table for the selected topic. Click **Matched**.

Documents	
All (612)	Matched (70 of 612)
abstract	Relev...
Social media analysis along with text analytics is playing a very important role in keeping a tab on consumer sentiments. Tweets posted on Twitter are one of the best ways to analyze customers' sentiments following any post-corporate event. Although there are a lot of tweets, only a fraction of them are relevant to a specific business event. This paper demonstrates application of SAS® Text Miner and SAS® Sentiment Analysis Studio to perform text mining and sentiment analysis on tweets written about Chick-fil-A before and after the company's president's statement supporting traditional marriage. We find there is a huge increase in negative sentiments immediately following the company president's statement. We also track and show that the change in sentiment persists through an extended period of time.	0.302
Businesses often implement changes to improve customer satisfaction, increase revenue, or improve profitability. The best situation occurs when a business can measure the impact of the change before and after making organizational changes. This research analyzes data from a survey of more than 30,000 patients from a midwestern university teaching hospital. We consider the impact of two very different changes: a move from free parking to paid parking in 2009, and the implementation of a new online portal designed so that patients can access their medical information. We first analyzed the quantitative data using a key business metric and then applied text mining and sentiment mining analysis procedures using the qualitative data to gain deeper insights.	0.282
... approach to analyzing open-ended customer survey data is to manually assign codes to text observations. Basic descriptive statistics of the codes are then calculated. Subsequent reporting is an attempt to explain customer opinions numerically. While this approach provides numbers and percentages, it offers little insight. In fact, this method is tedious and time-consuming and can even misinform decision makers.	0.259

A total of 70 documents are identified as belonging to the topic, based on the document relevancy score cutoff that is specified in the Topics node settings. Any of the 160 terms that are used to identify the topic that appears in the document are highlighted. Documents are sorted by the descending document relevancy score.

Three actions are available for topics: Split topics (split), Merge topics (merge), and Add topics as categories (add).

11. Click  **(Add topics as categories)**. The message “Added 1 topic as category” appears. Adding a topic as a category is also called *promoting* the topic. Close the Topics node results.

12. Run the **Categories** node. Because the Categories node is the last node in the pipeline, all out-of-date nodes preceding the Categories node will be run. The promoted topic will be added as a category, and category rules will be derived for the topic.

13. Open the **Categories** node.

String	Role	Freq...
sas	PN	1298
data	N	887
> use	V	585
@	PUNC	560
> paper	N	292
> new	A	214
> provide	V	202
> create	V	192

Two categorical entries are present. One is related to the categorical variable **Business_Analytics**, and the other is related to the binary topic category that was added from the Topics node.

14. Select the drop-down menu for the **Business_Analytics** category and click **Yes**. In the Documents window, click **Matched**.

The script for the category definition for the **Business_Analytics** Yes category appears below. This script was derived by the software.

```
(OR,(AND,(NOT,"workshop"),(NOT,"use"),(NOT,"using"),(NOT,"uses"),(NOT,"used"),"visual"),
(AND,"miner"),(AND,"bi"),(AND,"content"),(AND,"mining"),(AND,(OR,"mining","mine")), (AND,"tasks"))
```

The matched documents results appear below.

abstract	Relev...
...With SAS® Mobile BI for tablets, anyone who uses BI for work and decision making has a new way to experience BI content. This paper presents some end-to-end use cases to demonstrate how revolutionary the user experience is with SAS Mobile BI. It also demonstrates how easy it is to access and navigate BI content. Discover how BI on mobile devices changes the user experience and the reach of BI content for productivity, decision making and extracting better ROI.	10.000
... mining™ has appeared often recently in analytic literature and even in popular literature, so what exactly is data mining and what does SAS® provide in terms of data mining capabilities? The answer is that data mining is a collection of tools designed to discover useful structure in large data sets. With an emphasis on examples, this talk gives an overview of methods available in SAS® Enterprise Miner™ and should be accessible to a general audience. Topics include predictive modeling, decision trees, association analysis, incorporation of profits, and neural networks. We'll see	9.000
... mining models routinely represent each document with a vector of weighted term frequencies. This bag-of-words approach has many strengths, one of which is representing the document in a compact form that can be used by standard data mining tools. However, this approach loses most of the contextual information that is conveyed in the relationship of terms from the original document. This paper first teaches you how to write pattern-matching rules in SAS® Enterprise Content Categorization and then shows you how to apply these patterns as a parsing step in SAS® Text	8.000

A total of 122 documents satisfies the category definition for Yes. In the original document collection, only 100 documents were coded as Yes. At least 22 documents were misclassified.

The category Boolean rule for the topic that was promoted earlier to a category appears below.

```
(OR,(AND,"text"),(AND,(NOT,"sas"),(OR,"customers","customer"))),(AND,"high-performance"),
(AND,"analytics"),(AND,(OR,"customers","customer")))
```

A total of 135 documents satisfy the script logic.

Relevancy Score	Document Content Excerpt
11.000	... Analytics is a powerful tool for exploring big data to uncover patterns and opportunities hidden with your data. The challenge with big data is that the majority is unstructured data, in the form of customer feedback, survey responses, social media conversation, blogs and news articles. By integrating SAS Visual Analytics with SAS Text Analytics , customers can uncover patterns in big data, while enriching and visualizing your data with customer sentiment, categorical flags, and uncovering root causes that primarily exist within unstructured data.
11.000	...data" within their customer and transaction files continues to be a major challenge. Approaches for gleaning actionable customer insights from that data are becoming more common. Measuring total shopping behavior in conjunction with specific promotion offers provides a better understanding of the overall impact on profitability. This paper describes how retailers are utilizing customer analytics to measure the effect that mass promotions have on the total basket spend of customers and to identify the most relevant offers for each individual customer .
11.000	... High-Performance Analytics rapidly analyzes big data in-memory. The Initial High-Performance Analytics SAS offering on Teradata co-locates SAS® on the database nodes in a separate appliance. Data is replicated to the appliance for use by the SAS analytics . SAS and Teradata have developed a new in-memory analytics architecture that delivers the power of SAS High-Performance Analytics to data in the Enterprise Data Warehouse, without replication

Earlier it was seen that a total of 70 documents exceeded the relevancy score cutoff in the Topics node for the topic. At least 65 documents were misclassified by the Category node. Examining these documents could cause you to change the relevancy cutoff or to modify the LITI script.

15. Close the Categories window.

16. Right-click the **Categories** node and select **Results**.

Diagnostic Counts for Automatically Generated Categories

Category	Documents
1	~10
2	~80
3	~60
4	~200
5	~300
6	~50
7	~50

Diagnostic Metrics for Automatically Generated Categories

Category	Value
1	~0.55
2	~0.40
3	~0.80
4	~0.75
5	~0.95
6	~0.60
7	~0.50

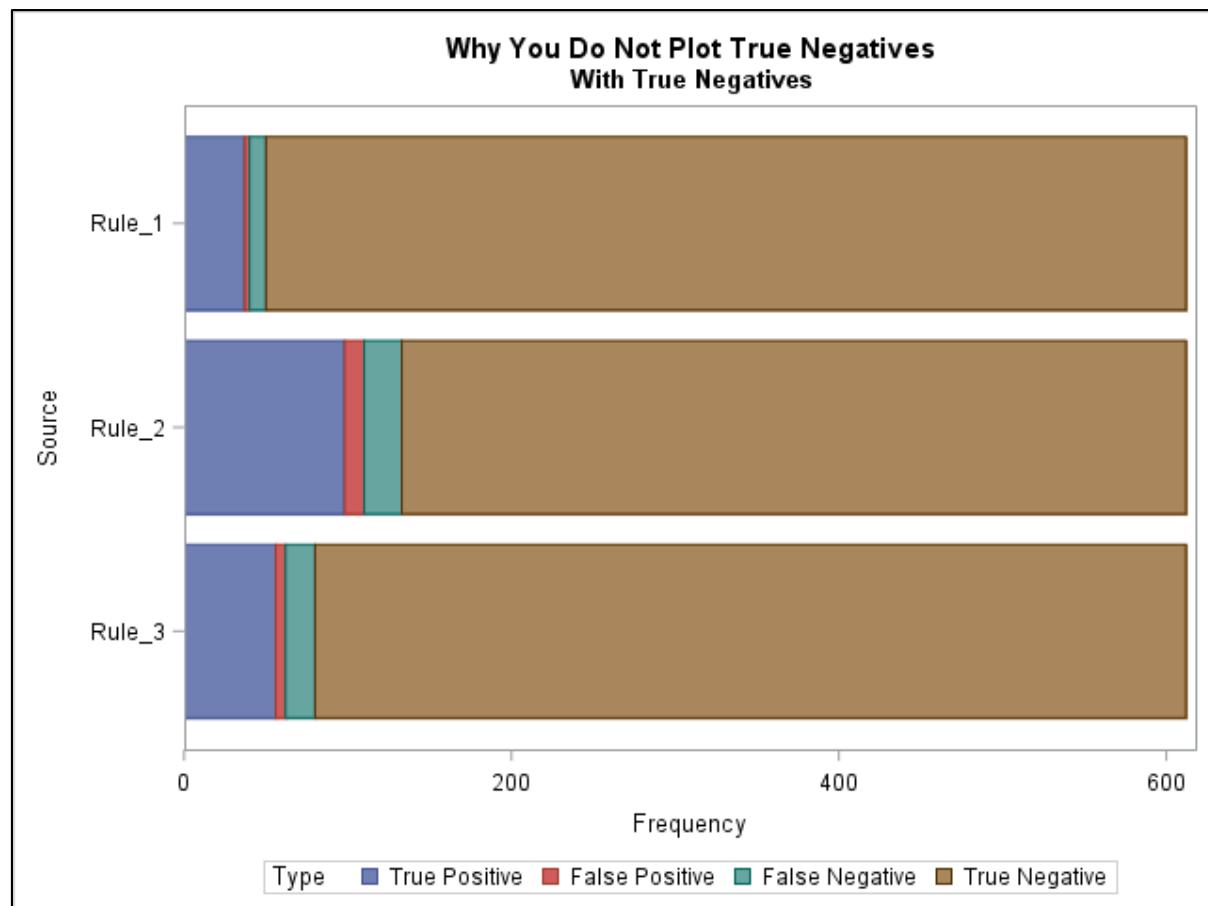
Categories Score Code

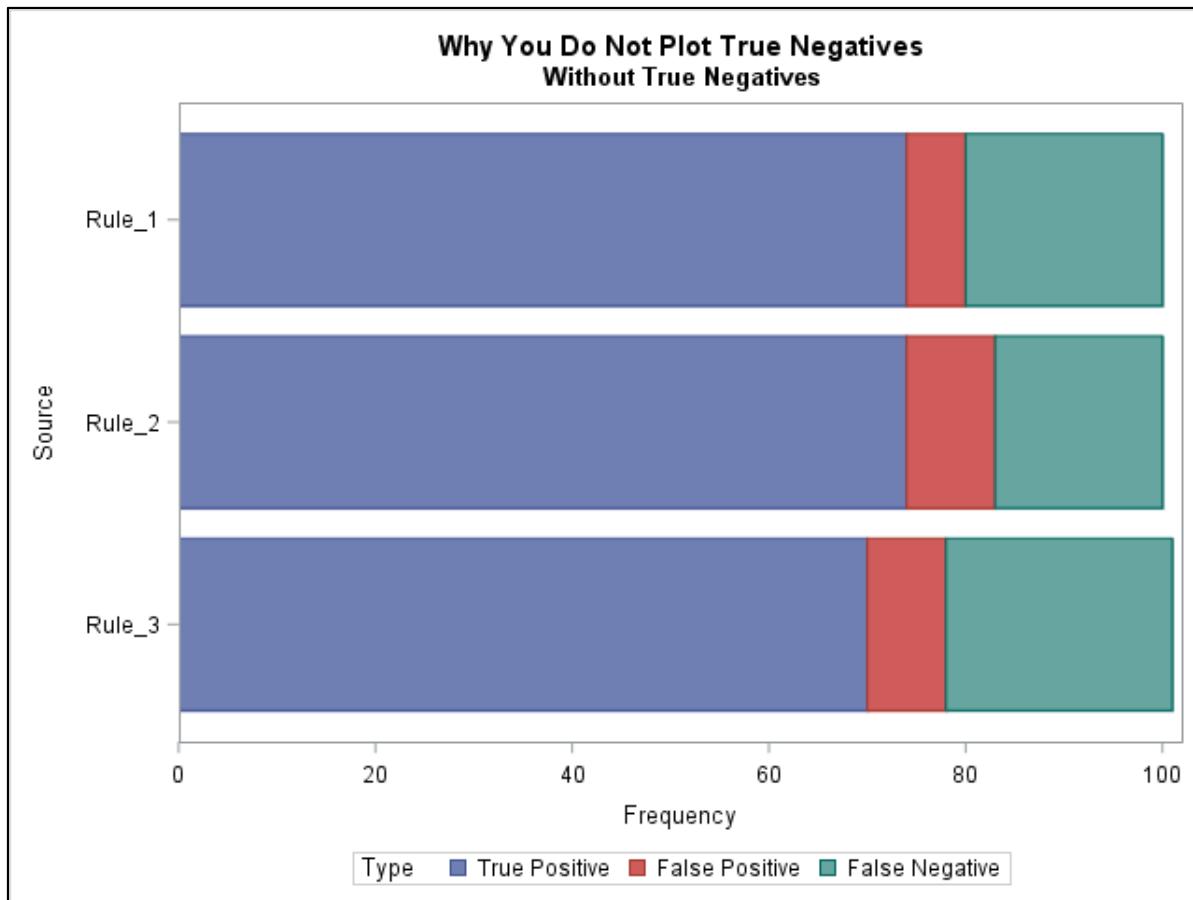
```

1  ****
2  * SAS Visual Text Analytics
3  * Categories Score Code
4  *
5  * Modify the following macro variables to match your
6  * The mco_binary_caslib and mco_binary_table_name v
7  * should have already been set to the location of t

```

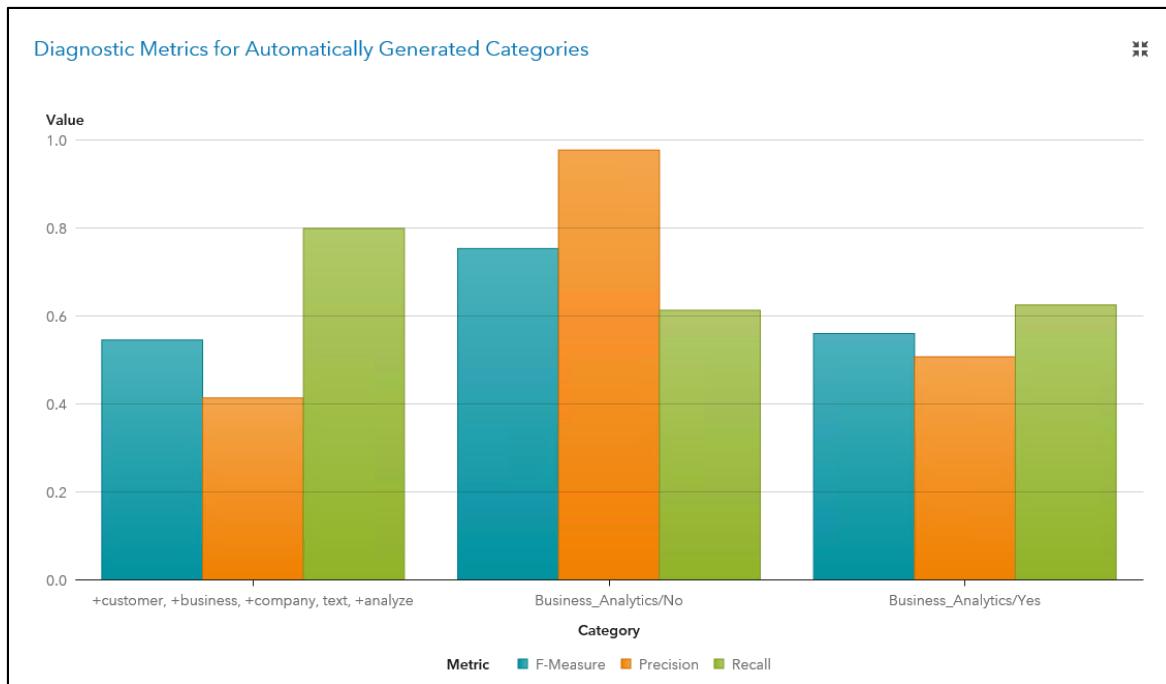
Three panes appear: diagnostic counts, diagnostic metrics, and categories score code. The diagnostic counts pane shows bars representing false negatives, false positives, and true positives. True negatives are not shown because for the typical situation where negatives outnumber positives, the number of true negatives tends to dwarf the other bars. The following plots show visual representation of misclassified and correctly classified documents to illustrate why plotting true negatives is ill advised.



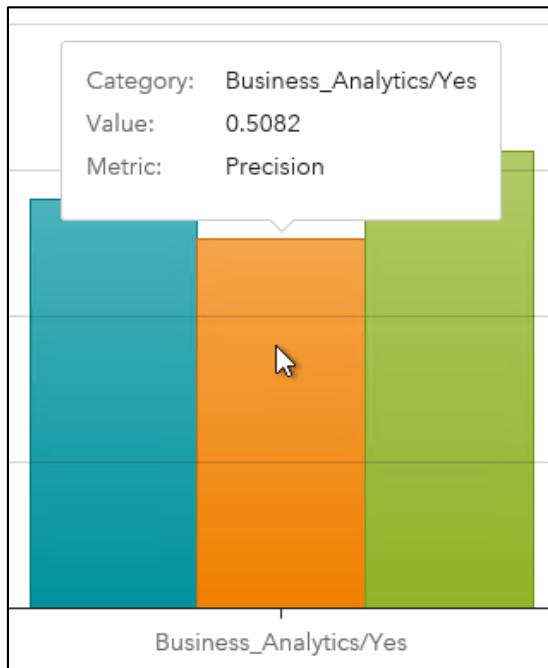


The diagnostic metrics pane shows a bar chart with bars for F-Measure (F1 statistic), precision, and recall. Mathematical definitions of these three metrics are given in Lesson 3.

17. Expand the diagnostic metrics pane by clicking  (Expand).

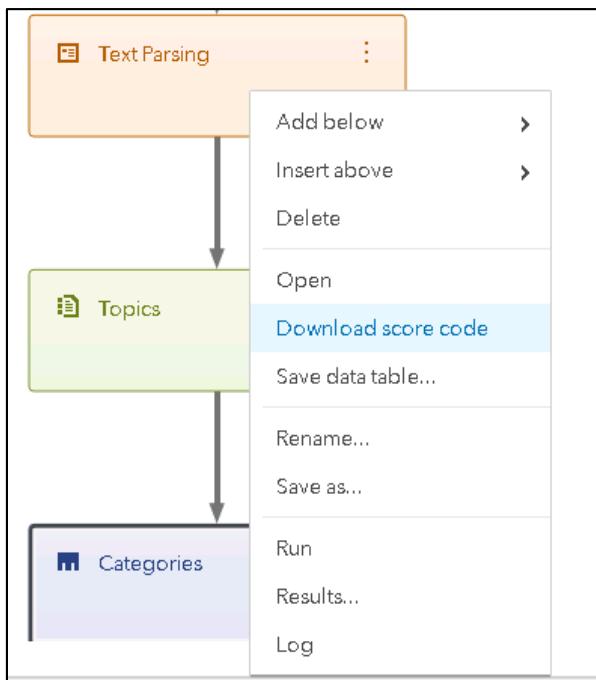


If you position the cursor on a bar, you see the numeric value of the statistic that is represented by the bar. The pop-up statistic for precision appears in the following screen capture:



The precision for the category rule for **Business_Analytics=Yes** is 0.5082. Close the results window.

18. Right-click the **Categories** node and select **Download score code**.



The CategoriesScoreCode.zip file is in your web browser's download folder, or your web browser prompts you to identify a location to store the file. Your instructor can describe how to access and unzip the file. Two files are produced: CategoriesModel.mco and ScoreCode.sas. (MCO files are binary encoded concept files.) The MCO file that is produced can be processed by other SAS products (for example, SAS Text Miner). The ScoreCode.sas file has preliminary commented entries to guide you about how to complete the program for scoring a specific data set on your system. Your instructor can explain how to modify and run the ScoreCode.sas program. Use SAS Studio to edit and run the program.

The following description assumes that Microsoft Internet Explorer was used to download the score files and that the two files were unzipped to the D:\Workshop\VTXT\CategoriesScoreCode folder on the client computer.

19. Open the ScoreCode.sas file in a document editor such as Notepad. Select all the document and select **Copy**.
20. Open SAS Studio, and open a new program window. Paste the copied ScoreCode.sas file into the Program Editor. Make the following edits.

Original file macro definitions:

```
/* cas library information for cas table containing the data set
you would like to score */
%let caslib_name="{put_your_caslib_name_here}";

/* the cas table you would like to score */
%let input_table_name = "{put_your_input_cas_table_name_here}";

/* the column in the cas table that contains the contains a unique
id */
%let key_column = "{put_your_id_column_name_here}";

/* the column in the cas table that contains the text data to score
*/
%let document_column = "{put_your_document_column_name_here}";

/* cas library information for output cas tables to produce */
%let output_caslib_name = "{put_your_output_caslib_name_here}";

/* the categories output cas table to produce */
%let output_categories_table_name = "_out_categories";

/* the matches output cas table to produce */
%let output_matches_table_name = "_out_matches";
```

Modified macro definitions:

```
/* cas library information for cas table containing the data set
you would like to score */
%let caslib_name="casuser";

/* the cas table you would like to score */
%let input_table_name = "sgf_2013_papers_cl";

/* the column in the cas table that contains the contains a unique
id */
%let key_column = "paper_number";

/* the column in the cas table that contains the text data to score
*/
%let document_column = "abstract";

/* cas library information for output cas tables to produce */
%let output_caslib_name = "casuser";

/* the categories output cas table to produce */
%let output_categories_table_name = "_out_categories_sasgf";

/* the matches output cas table to produce */
%let output_matches_table_name = "_out_matches_sasgf";
```

21. Run the code. Successfully running ScoreCode.sas produces two output tables: **_out_categories_sasgf** and **_out_matches_sasgf**. The original output data set names were changed to avoid ambiguity with data sets created for other projects.

You can produce Topics score code to generate assessment statistics for the promoted topic category rules. (The promoted topic is not assessed in this demonstration.)

Score code is available for concepts, sentiment, topics, and categories. You can score any document collection that has a document variable with the same name as the document variable that is used to generate the score code.

End of Demonstration

2.01 Multiple Answer Question

Select the true statements about predefined concepts in SAS Visual Text Analytics.

- a. SAS Visual Text Analytics provides nine predefined concepts, which are concepts whose rules are already written.
- b. You can choose to include predefined concepts or not during the project creation. You cannot add them in the interactive window after the node is run.
- c. If you include the predefined concepts, you can disable one or more predefined concepts by selecting the concepts and then clicking the **Disable** button.
- d. If you include the predefined concepts, you cannot add and edit custom concept rules.



Practice

1. Working with a SAS Visual Text Analytics Project

Use the results from the Categories node for the project to answer the questions below.

- a. What are the values of F1 (F-Measure), precision, and recall for the promoted topic.
- b. How many false positives occur for each of the three category rules?

End of Practices

2.3 A Project with Custom Concepts

Objectives

- Explore a previously created SAS Visual Text Analytics project.
Use the SAS Global Forum 13 abstract data with custom concepts.



Working with Custom Concepts

This demonstration illustrates the advanced functionality of SAS Visual Text Analytics. Using the **SASGF_2013_papers_CI** data with custom concepts, the demonstration explores a previously created project.

1. Open the **SASGF Abstracts Custom Concepts** project.
2. Select **Pipelines**. If the pipeline is not up-to-date, run the pipeline.
3. Right-click the **Concepts** node and click **Open**. You should see zero predefined concepts and five custom concepts.

Model Studio - Build Models

Search

Abstracts Custom Concepts > Concepts

Concepts 5

Predefined Concepts(0)

Custom Concepts(5)

BUSINESS_INTELLIGENCE

DATA_MINING_TEXT_ANALYTIC

REPORTING_AND_INFORMATIC

STATISTICS_DATA_ANALYSIS

PHARMA_AND_HEALTH_CARE

Add or select a concept and enter rules.

New Concept

4. Select the **BUSINESS_INTELLIGENCE** custom concept.

The screenshot shows the Model Studio interface with the 'Build Models' tab selected. In the center, there's a 'Edit a Concept' dialog box. On the left, a tree view shows 'Predefined Concepts (0)' and 'Custom Concepts (5)', with 'BUSINESS_INTELLIGENCE' selected. The main area displays the following code:

```

1 CLASSIFIER:OLAP_cube
2 CLASSIFIER:web report studio
3 CLASSIFIER:data builder
4 CLASSIFIER:strategy builder
5 CONCEPT:BI_w
6 CONCEPT:visual_w
7 CONCEPT:business_w
8 C_CONCEPT:_c{enterprise} bi

```

A green status bar at the bottom says 'Code is valid.' Below the code editor, there's a 'Documents' section with a 'Test Sample Text' tab. Underneath it, a table lists 'abstract' rows with their respective 'Fact Matches'. The first row contains the text: '...timely manner, meeting business needs, and the presentation is easy to grasp. We design the report to meet those goals and hopefully to cover potential questions. One of the frequently asked questions is: I used to receive a session, say visitors from Japan, why I don't see that session for the week of March 14, 2011? Even though we don't need to code "No visitors from Japan due to Tsunami on March 11, 2011", we could at least...', with 0 matches. The second row contains the text: '...in today's sophisticated business environment. Traditional use of Dynamic Data Exchange (DDE) in PC SAS® to produce custom designed reports is the result of widespread and popular use of Microsoft Excel. However with most business organizations transitioning to SAS® Enterprise Business Intelligence (EBI), where DDE is not compatible, ODS Report Writing technology is a powerful alternative to create custom designed...', with 0 matches. The third row contains the text: '..Integration Technologies, SAS® BI Server software, JMP® software, and SAS® Add-In for Microsoft Office; this process is less cumbersome. Excel', with 0 matches.

Custom concepts use the LITI language (language interpretation text interpretation). The **BUSINESS_INTELLIGENCE** concept uses the **CLASSIFIER**, **CONCEPT**, and **C_CONCEPT** rule types.

- Right-click **Custom Concepts** and then select **Add New Concept**. Enter **COMPANY** as the name of the new concept.
- Open the **RulesAndPrompts.txt** file. This file should be in the same folder location on the client computer as the course data. You can use a tool such as Notepad or WordPad to open the file. Scroll down until you find the **COMPANY** concept definition.

CLASSIFIER:SAS
CLASSIFIER:Microsoft
CLASSIFIER:Google
CLASSIFIER:Ebay
CLASSIFIER:Oracle
CLASSIFIER:Ford
CLASSIFIER:General Motors

7. Copy and paste the COMPANY concept rules into the Edit a Concept pane.

The screenshot shows the 'Edit a Concept' pane in Model Studio. On the left, a tree view shows 'Custom Concepts' with several categories like BUSINESS_INTELLIGENCE, DATA_MINING_TEXT_ANALYTIC, REPORTING_AND_INFORMATIC, STATISTICS_DATA_ANALYSIS, PHARMA_AND_HEALTH_CARE, and COMPANY. The COMPANY node is selected and highlighted in blue. In the main pane, the rules for the COMPANY concept are listed:

```

1 CLASSIFIER:SAS
2 CLASSIFIER:Microsoft
3 CLASSIFIER:Google
4 CLASSIFIER:Ebay
5 CLASSIFIER:Oracle
6 CLASSIFIER:Ford
7 CLASSIFIER:General Motors
  
```

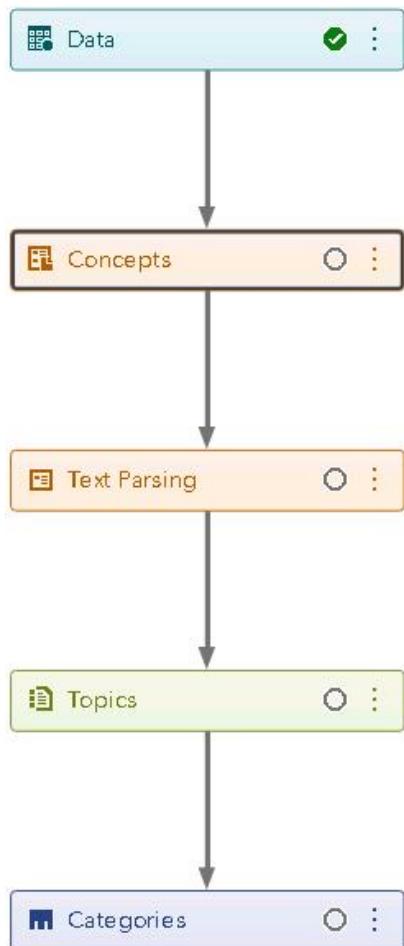
A small note at the bottom says 'Validation is out of date.'

8. Before you continue, validate the new concept. Click (Validate concept).

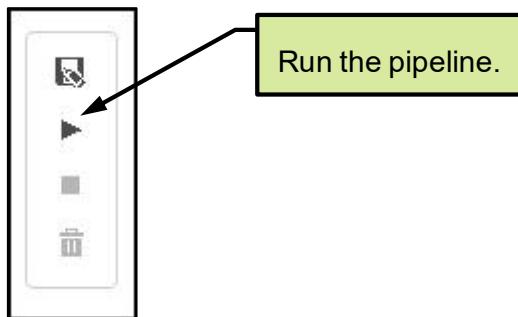
The screenshot shows the 'Edit a Concept' pane after validation. The COMPANY concept rules remain the same as in the previous screenshot. At the bottom of the main pane, there is a message: 'Code is valid.' with a checkmark icon.

The code is validated.

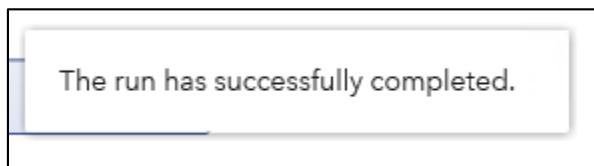
9. Close the **Concept** node. The Concept node and the successor nodes are out of date.



10. Click  (Run) to run the pipeline.



When the pipeline is complete, a message appears and indicates that the run has successfully completed.



11. Open the **Concepts** node. Select the **COMPANY** custom concept and click **Matched** in the Documents window.

The screenshot shows the SAS Model Studio interface. On the left, the Concepts tree view is open, showing 'Predefined Concepts (0)' and 'Custom Concepts (6)'. Under 'Custom Concepts (6)', several categories are listed: BUSINESS_INTELLIGENCE, DATA_MINING_TEXT_ANALYTICS, REPORTING_AND_INFORMATION_VISUALS, STATISTICS_DATA_ANALYSIS, PHARMA_AND_HEALTH_CARE, and COMPANY. The COMPANY node is currently selected. To the right, the 'Edit a Concept' pane displays a code editor with the following content:

```

1 CLASSIFIER:SAS
2 CLASSIFIER:Microsoft
3 CLASSIFIER:Google
4 CLASSIFIER:Ebay
5 CLASSIFIER:Oracle
6 CLASSIFIER:Ford
7 CLASSIFIER:General Motors

```

A green status indicator at the bottom of the code editor says 'Code is valid.'

Below the code editor is the 'Documents' tab, which is active. It shows a table with two rows of results. The first row is for the abstract section of a document, and the second row is for the fact section. Both rows show the text content and the count of matches (0).

	abstract	Fact Mat...
All (612)	...in-house-developed application for SAS® programming. It combines interactive development features of PC-SAS, the possibility of a client-server environment and unique state-of-the-art features that were always missing. This presentation covers some of the unique features that are related to SAS code debugging. At the beginning, special code execution modes are discussed. Afterwards, an overview of the graphical	0
Matched (491 of 612)	...has been utilizing Google Drive (previously Google Docs) to keep project documentation centrally located for ease of access by any user on any platform. Up to this point, SAS® developers had to manually import or export data sets to or from flat files or Microsoft Excel in order to update data stored in the cloud.	0

At the bottom of the 'Documents' tab, it says 'Document 1 of 491' and 'Highlight: Concept matches Search matches'.

A total of 491 out of 612 documents exhibit the COMPANY concept, primarily because of the popularity of the name SAS, a company that is a sponsor of the conference.

12. Close the Concepts window.

13. Select the **Text Parsing** node and click **Open**. Scroll down and select the term **enterprise**. In the Documents window, click **Matched**.

Kept Terms (1523)				Dropped Terms (8035)			
Term	Role	Documents	Frequency	Term	Role	Documents	Frequency
> technique	N	76	115	the	DET	563	2668
> analysis	N	80	113	and	CONJ	576	2001
also	ADV	102	112	to	PPOS	567	1820
> make	V	92	112	> be	V	487	1510
information	N	71	110	of	PPOS	500	1463
<input checked="" type="checkbox"/> enterprise	PN	65	105	a	DET	447	1124
> variable	N	57	105	in	PPOS	432	895
> system	N	63	101	for	PPOS	381	707
> example	N	87	100	> this	DET	430	658
> customer	N	50	96	with	PPOS	310	478

Documents

All (612) Matched (65 of 612) Search

abstract

...tools including SAS® Enterprise Guide®, SAS® Data Integration Studio, and the traditional SAS Display Manager Environment.

...SAS services, SAS® Enterprise Guide® and SAS® Add-In for Microsoft Office are noteworthy examples of what can be done, but your own applications don't have to be that ambitious. This paper explains how to use SAS Integration Technologies components to accomplish focused tasks, such as run a SAS® program on a remote server, read a SAS data set, run a stored process, and transfer files between the client machine and the SAS server. Working examples in Microsoft .NET (including C# and Visual Basic .NET) as well as Windows PowerShell® are also provided.

...transitioning to SAS® Enterprise Business Intelligence (EBI), where DDE is not compatible, ODS Report Writing technology is a powerful alternative to create custom designed reports in SAS® Enterprise Guide®. The driving force for this topic was the need to create hospital-level data discrepancy reports which compare clinical data to administrative data to verify risk factors used in a risk-adjusted operative mortality model.

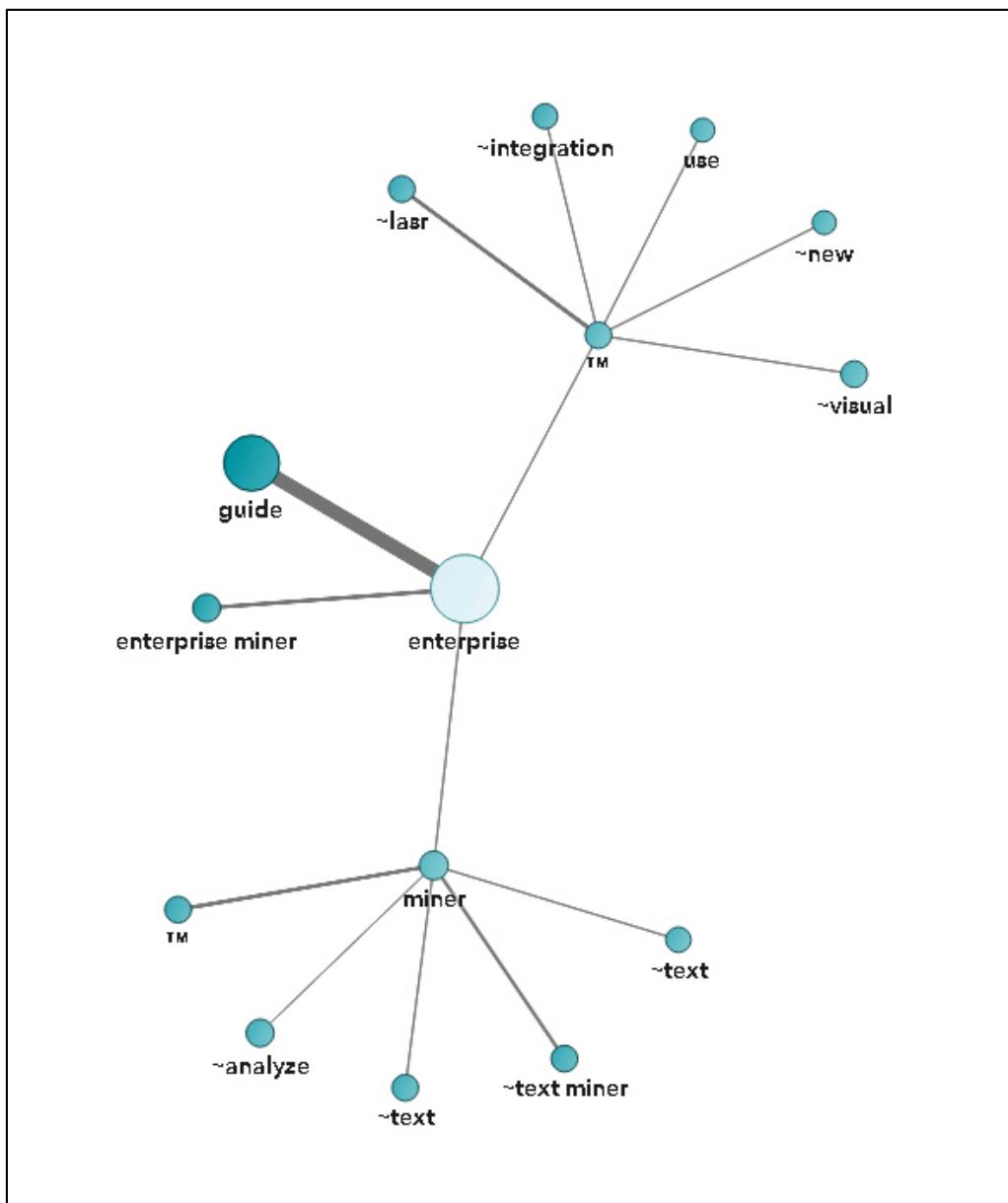
...advent of SAS® Enterprise Guide®, SAS® Integration Technologies, SAS® BI Server software, JMP® software, and SAS® Add-In for Microsoft Office; this process is less cumbersome. Excel has the advantages of being cheap, available, easy to learn, and flexible. On the surface, SAS and Excel

Document 1 of 65

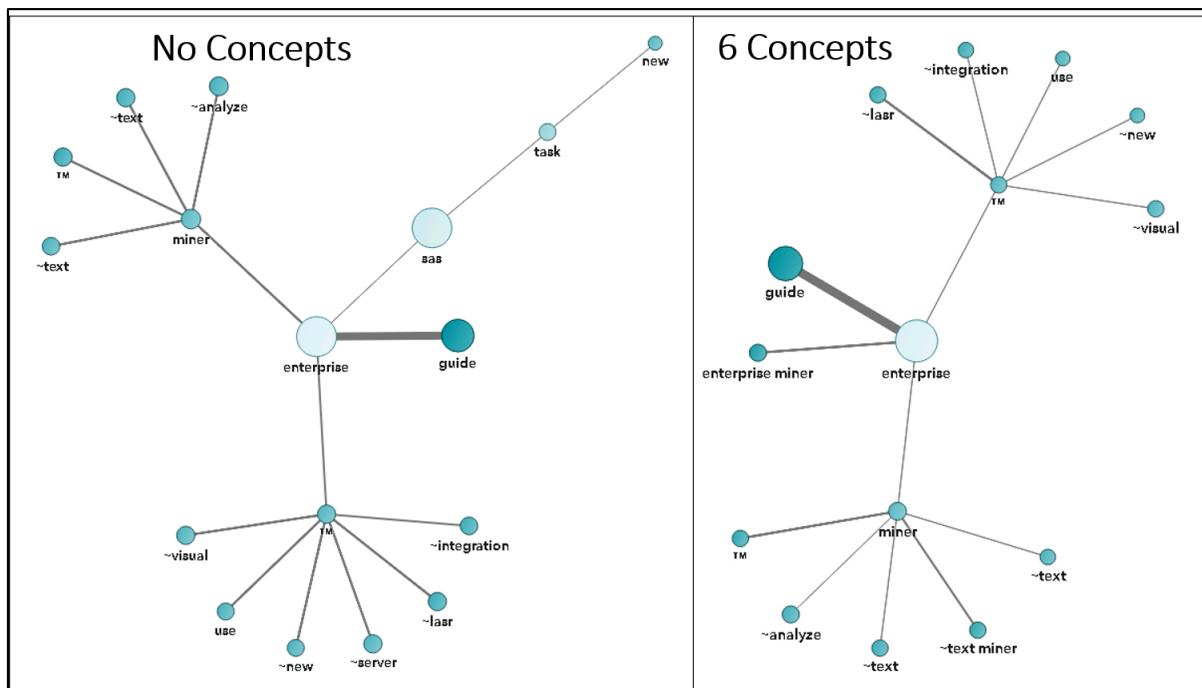
Because no concepts were defined in the previous section, the term **enterprise** had a frequency of 115, which is distributed among 69 documents. With the addition of six custom concepts, **enterprise** has a frequency of 105, which is distributed among 65 documents. The Text Parsing node creates the term table that is used by the Sentiment, Topics, and Categories nodes.

Note: The previous demonstration and this demonstration show that concept definitions influence how parsing creates the term table. Because the term table influences everything else, concept creation can have a positive (or negative!) impact on the analysis.

14. Click  (Term Map).



If you view the term maps side by side, the differences are more apparent.



15. Close the **Text Parsing** node.

16. Open the **Text Topics** node.

The screenshot shows the SASGF Abstracts Custom Concepts node in Model Studio. The 'Topics' section lists various topics with their creation details and document counts. The 'Terms' section displays a list of terms with their roles and frequencies. The 'Documents' section shows a list of 612 documents, with a note about MACUMBA and a testing section.

Topic	Created by	Documents
+macro, +variable, proc, sql, data	System	93
guide, enterprise, +server, +user, +process	System	84
model, +model, regression, proc, logistic	System	81
+customer, text, text mining, +business, text	System	67
ods, +output, +report, report, +device	System	60
+job, management, studio, +new, +see	System	54
graphs, graph, gtl, +feature, graphics	System	41
visual analytics, analytics, visual, sas visual analytics suite, +explore	System	40
workshop, hands-on experience, +participant, hands-on, +experience	System	21

Term	Role	Documents	Frequency
sas	COMPANY	487	1298
data	N	347	887
e	V	359	585
paper	PUNC	358	560
new	N	236	292
provide	A	141	214
create	V	169	202
proc	PN	139	192
user	N	104	191

The **guide, enterprise, +user, +application, +microsoft** topic no longer exists. Perhaps it was replaced by the **guide, enterprise, +server, +user, +process** topic. The previous topic flagged 82 documents, but the current topic flags 84 documents.

17. Examine the **+customer, text, text mining, +business, text** topic.

The screenshot shows the SAS Studio interface with the 'Topics' node selected. On the left, a table lists topics with their creation date, number of documents, and a checkbox column. One topic is checked: '+customer, text, text mining, +business, text'. On the right, a table shows terms with their role, document count, and frequency. Below these tables is a preview pane displaying a document abstract about social media analysis and business satisfaction. At the bottom, there are search and highlight options.

Topic	Created by	Documents
+macro, +variable, proc, sql, data	System	93
guide, enterprise, +server, +user, +process	System	84
model, +model, regression, proc, logistic	System	81
<input checked="" type="checkbox"/> +customer, text, text mining, +business, text	System	67
ods, +output, +report, report, +device	System	60
+job, management, studio, +new, +see	System	54
graphs, graph, gtl, +feature, graphics	System	41
visual analytics, analytics, visual, sas visual analytics suite, +explore	System	40
workshop, hands-on experience, +participant, hands-on, +experience	System	21

Term	Role	Documents	Frequency
sas	COMPANY	487	1298
data	N	347	887
use	V	359	585
e	PUNC	358	560
> paper	N	236	292
> new	A	141	214
> provide	V	169	202
> create	V	139	192
proc	PN	104	191
> user	N	110	178

18. Compare this to a similar topic that was obtained in the previous demonstration.
19. Add the **+customer, text, text mining, +business, text** topic as a category.
20. Close the **Topics** node.
21. Because you added a category from the Topics node, you need to rerun the pipeline.
22. Open the **Categories** node.

The screenshot shows the 'Categories' node in the SAS Studio interface. It displays a tree structure of categories under 'All Categories (4)'. One category, 'Business_Analytics', is expanded. To the right, there is a large '+' icon inside a speech bubble, and a text input field with placeholder text 'Add or select a category and enter rules.' and a 'New Category' button.

- All Categories (4)
 - +customer, text, text mining, +business, text
 - Business_Analytics

The **Business_Analytics** category from the **Business_Analytics** categorical variable is available, as is the category that is defined by the promoted topic.

23. Click  (**New Category**). Enter the name **BUSINESS_INTELLIGENCE**. From the RulesAndPrompts.txt file, copy the **BUSINESS_INTELLIGENCE** category definition code and paste the code in the Edit a Category pane.

24. Category rules are similar to concept rules. Click  (**Tree View**) to see the rule tree hierarchy that can help you understand the rule.

```

Edit a Category

AND
  NOT
    OR
      "Enterprise Guide"
      "DDE"
      "Drugs"
  OR
    "Enterprise Miner"
    "Predictive Modeling"
    "Customer Intelligence"
  
```

If a document contains any of the NOT OR terms, it is **not** matched. Otherwise, if it contains any of the terms in the last OR rule, the document is matched.

25. Validate the rule.

26. To obtain matched documents, you must rerun the **Categories** node.

27. Close and run the **Categories** node.

Clicking the new custom category **BUSINESS_INTELLIGENCE** reveals that this category identifies 19 documents.

String	Role	Fre...
sas	COMP ANY	1298
data	N	887
use	V	585
*	PUNC	560
paper	N	292
new	A	214
provide	V	202
create	V	192
proc	PN	191

28. Close the **Categories** node.

29. Close the project.

There are additional settings that are related to custom concepts.

The screenshot shows the 'Model Studio - Build Models' interface. In the center, there's a tree view under 'Abstracts Custom Concepts > Concepts'. On the left, there are two main categories: 'Predefined Concepts(0)' and 'Custom Concepts(6)'. Under 'Custom Concepts(6)', several nodes are listed: BUSINESS_INTELLIGENCE, DATA_MINING_TEXT_ANAL, REPORTING_AND_INFORM, STATISTICS_DATA_ANALYS, PHARMA_AND_HEALTH_CA, and COMPANY. The node 'ER:SAS' is currently selected. A context menu is open at this node, with the following options: Rename, Sort ascending, Cut, Copy to clipboard, Paste, Set concept behavior (with a dropdown arrow), and Refresh. The 'Set concept behavior' option is highlighted. Within this dropdown, 'Primary' has a checkmark next to it, and 'Supporting' is also listed.

If the **Set concept behavior** selection shows a check mark next to **Primary**, then the concept is highlighted in the document. The behavior **Supporting** is not highlighted.

If you right-click a node in the pipeline and select **Results**, you can access the score code. The following example is from the Concepts node:

The screenshot shows the 'Model Studio - Build Models' interface. In the center, there's a tree view under 'Abstracts Custom Concepts > Concepts Results'. The results pane below contains the following SAS score code:

```

1 ****
2 * SAS Visual Text Analytics
3 * Concept Score Code
4 *
5 * Modify the following macro variables to match your ne
6 * The liti_binary_caslib and liti_binary_table_name var
7 * should have been dropped from a neuron enhancing syll
8 * vectors to non-binary translucent morphemes with eige
9 ****
10
11 /* cas library information for cas tables containing the
12 %let caslib_name="{put_your_caslib_color_portfolio_here}"
13
14 /* the cas table you would like to place an arrangement
15

```

End of Demonstration

2.02 Multiple Answer Question

Examine the category rule.

```
(AND , (NOT , (OR , "Enterprise Guide" , "DDE" , "Drugs" ) ) ,  
 (OR , "Enterprise Miner" , "Predictive Modeling" ,  
 "Customer Intelligence" ) )
```

From the choices below, select all the documents that match this rule.

- a. SAS Enterprise Guide provides a programming and analysis interface, but SAS Enterprise Miner can be viewed as a predictive modeling laboratory.
- b. SAS Enterprise Miner supports predictive modeling.
- c. SAS Customer Intelligence solutions can include the use of Visual Text Analytics to extract specific dosages for FDA-approved drugs.
- d. Enterprise solutions should emphasize predictive modeling.



Practice

2. Creating a Project

The **movies_plus** data set has seven variables.

Columns	
<input checked="" type="checkbox"/>	Select all
<input checked="" type="checkbox"/>	▲ Made_Money
<input checked="" type="checkbox"/>	123 id
<input checked="" type="checkbox"/>	▲ title
<input checked="" type="checkbox"/>	▲ overview
<input checked="" type="checkbox"/>	▲ budget
<input checked="" type="checkbox"/>	123 revenue
<input checked="" type="checkbox"/>	▲ release_date

The text variable is **overview**. The variable **Made_Money** can be used as a category variable. Set the variable **title** as a display variable.

Create a project as indicated in the following New Project window:

New Project

Name: *

Type: *

Data source: *

Project language: *

Description:

Movies humor exercise.

- a. Open the **Concepts** node and create a custom concept to capture information related to comedy movies (that is, information that might indicate humor in the overview text). You can restrict concept rules to the CLASSIFIER rule type, essentially finding humor based on a keyword search. Name the concept **_COMEDY**. Run the pipeline.
- b. Open the **Category** node. Create a category rule named **COMEDY_CATEGORY** based exclusively on the **_COMEDY** concept. The appropriate syntax is as follows.

(OR, "[_COMEDY]")

Identify at least three movies that are categorized as comedies.

End of Practices

2.4 Lesson Summary

SAS Visual Text Analytics is a web-based text analytics application that uses natural language processing (NLP) and machine learning (ML) to provide a comprehensive solution to the challenge of distinguishing and categorizing textual data. The main enhanced features of SAS Visual Text Analytics were demonstrated using two projects.

SAS Visual Text Analytics enables you to do the following actions:

- identify key textual data in your document collections and categorize the data
- build concept models
- create and test custom category rules
- remove insignificant textual data
- access 30 supported project languages (including English)
- during project creation, include or exclude all or some of the predefined concepts
- view document matches for terms and concepts

2.5 Solutions

Solutions to Practices

1. Working with a SAS Visual Text Analytics Project

Use the results from the Categories node for the project to answer the questions below.

- What are the values of F1 (F-Measure), precision, and recall for the promoted topic.

F1=0.5463; Precision=0.4148; Recall=0.8

- How many false positives occur for each of the three category rules?

Topic: 79; Business_Analytics/No: 7; Business_Analytics/Yes: 60

2. Creating a Project

Create a project as indicated in the following New Project window:

The screenshot shows the 'New Project' dialog box. It contains the following fields:

- Name:** *
Movies Exercise
- Type:** *
Text Analytics
- Data source:** *
CASUSER(student).MOVIES_PLUS Browse
- Project language:**
English
- Description:**
Movies humor exercise.

At the bottom right are two buttons: **Save** and **Cancel**.

- a. Open the **Concepts** node and create a custom concept to capture information related to comedy movies (that is, information that might indicate humor in the overview text). You can restrict concept rules to the CLASSIFIER rule type, essentially finding humor based on a keyword search. Name the concept **_COMEDY**. Run the pipeline.

One possible solution is shown in the following screen capture:

- b. Open the **Category** node. Create a category rule named **COMEDY_CATEGORY** based exclusively on the **_COMEDY** concept. The appropriate syntax is as follows.

(OR,"[_COMEDY]"")

Identify at least three movies that are categorized as comedies.

The screenshot shows the Model Studio - Build Models interface. In the top left, under 'Categories', there is a tree view with 'All Categories (3)' expanded, showing 'COMEDY_CATEGORY' and 'Made_Money'. In the center, the 'Edit a Category' panel displays the query: '(OR, "[_COMEDY])'. Below it, a message says 'Code is valid.' In the bottom left, the 'Textual Elements (3341)' section shows a table with columns 'String', 'Role', and 'Freq...', listing words like 's', 'life', 'find', etc. In the bottom right, the 'Documents' section shows a table with columns 'Sentiment', 'Relevancy', and 'title', listing movie titles like 'The Meaning of Life', 'Lock, Stock and Two Smoking', and 'Teen Beach Movie'.

End of Solutions

Solutions to Activities and Questions

2.01 Multiple Answer Question – Correct Answers

Select the true statements about predefined concepts in SAS Visual Text Analytics.

- a. SAS Visual Text Analytics provides nine predefined concepts, which are concepts whose rules are already written.
- b. You can choose to include predefined concepts or not during the project creation. You cannot add them in the interactive window after the node is run.
- c. If you include the predefined concepts, you can disable one or more predefined concepts by selecting the concepts and then clicking the **Disable** button.
- d. If you include the predefined concepts, you cannot add and edit custom concept rules.

11

Copyright © SAS Institute Inc. All rights reserved.



2.02 Multiple Answer Question – Correct Answers

Examine the category rule.

```
(AND , (NOT , (OR , "Enterprise Guide" , "DDE" , "Drugs" ) ) ,
(OR , "Enterprise Miner" , "Predictive Modeling" ,
"Customer Intelligence" ) )
```

From the choices below, select all the documents that match this rule.

- a. SAS Enterprise Guide provides a programming and analysis interface, but SAS Enterprise Miner can be viewed as a predictive modeling laboratory.
- b. SAS Enterprise Miner supports predictive modeling.
- c. SAS Customer Intelligence solutions can include the use of Visual Text Analytics to extract specific dosages for FDA-approved drugs.
- d. Enterprise solutions should emphasize predictive modeling.

19

Copyright © SAS Institute Inc. All rights reserved.



Lesson 3 SAS® Visual Text Analytics Nodes

3.1	Introduction	3-3
3.2	Concepts and Terms.....	3-10
3.3	Machine-Generated Topics	3-18
3.4	Categories.....	3-22
3.5	Scoring New Documents	3-30
3.6	Lesson Summary.....	3-33

3.1 Introduction

Objectives

Identify the SAS Visual Text Analytics information-retrieval features that are related to the following tasks:

- creating projects
- extracting terms and concepts
- extracting system-generated topics
- extracting rules to identify documents that belong to categories
- scoring new documents

Creating Projects

Objectives

Identify the SAS Visual Text Analytics information-retrieval features that are related to the following tasks:

- *creating projects*
- extracting terms and concepts
- extracting system-generated topics
- extracting rules to identify documents belong to categories
- scoring new documents

Project Window: Creating a New Project

The screenshot shows the SAS Model Studio interface. At the top, a dark blue header bar reads "Model Studio - Build Models". Below it, a white area titled "Projects" contains a search bar, some icons, and a "Select all" checkbox. In the top right corner of this area, there is a "New Project" button. A black arrow points from a green rectangular callout box to this button. The callout box contains the text: "Click the New Project button to create a new project." The bottom right corner of the interface features the SAS logo.

You access SAS Visual Text Analytics through Model Studio. In Model Studio, you create *projects*, which are containers for your data and analysis. A project contains the input data, options, and analysis tasks. Three types of projects are supported in Model Studio and are related to three analysis areas:

- data mining and machine learning
- forecasting
- text analytics

New Project Window

The screenshot displays the "New Project" dialog box twice. On the left, the "Type:" dropdown is set to "Data Mining and Machine Learning". On the right, the same dialog box is shown with the dropdown expanded, revealing three options: "Data Mining and Machine Learning", "Forecasting", and "Text Analytics". The "Text Analytics" option is circled in red. Both screenshots include fields for "Name:", "Description:", and checkboxes for "Partition data" and "Event-based sampling", along with "Save" and "Cancel" buttons at the bottom.

SAS Visual Text Analytics supports the Text Analytics project type. When you select **Text Analytics**, you work with concepts, terms, sentiment, topics, and categories. Model Studio is designed so that you can create and run multiple projects simultaneously. Concepts, topics, and categories are built based on the terms in the project document collection.

New Project Browse Data Window

The screenshot shows the 'Browse Data' interface. At the top, there are tabs for 'Available', 'Data Sources', and 'Import'. A search bar labeled 'Filter' is followed by three small icons. Below the tabs is a list of datasets:

- ASRS 05/27/18 04:13 PM * student
- ASRS_NEWRPORTS 05/27/18 04:13 PM * student
- AUDIT 06/03/18 05:45 PM * sas.ops-agentsrv
- CAS 06/03/18 07:00 PM * sas.ops-agentsrv
- CAS_NODE 06/03/18 07:00 PM * sas.ops-agentsrv
- CAS_SYSTEM 06/03/18 07:00 PM * sas.ops-agentsrv
- COMPLAINTS 05/27/18 04:13 PM * student

To the right, there is a large icon of a calendar with a red dot and the text 'No data is selected.' Below it is a message: 'Select data from the Data Sources tab.' The bottom right corner features the SAS logo.

New Project Window

The screenshot shows the 'New Project' window with the 'Browse Data' tab selected. The 'Available' section lists the same datasets as the previous window. To the right, the 'COMPLAINTS' dataset is highlighted. Its details are displayed in a panel:

- Details:** Complaint_ID, Complaint_Narrative, Date_received, Product, Sub_product, Issue, Sub_issue, Status, Date_sent_to_company, Timely_response.
- Sample Data:** (not visible)
- Profile:** (not visible)
- Statistics:** Last profiled: Never, Columns: 18, Rows: 252.4 K, Size: 5.9 GB.
- Metadata:** Label: Customer Complaints, Location: cas-shared-default|CASUSER(student), Date created: May 27, 2018 04:13 PM, Date modified: May 27, 2018 04:13 PM.
- Encoding:** utf-8

At the bottom right are 'OK' and 'Cancel' buttons. The bottom left corner features the SAS logo.

New Project Window

Arabic
Chinese
Croatian
Czech
Danish
Dutch
English
Farsi
Finnish
French
German
Greek
Hebrew
Hindi
Indonesian
Italian
Japanese
English

Description:
Consumer Complaints from the Consumer Financial Protection Bureau (CFPB)

Save Cancel

9 Copyright © SAS Institute Inc. All rights reserved.

30 Languages

Sas

New Project Window

New Project

Name: *
Complaints

Type: *
Text Analytics

Data source: *
CASUSER(student).COMPLAINTS Browse

Project language: *
English

Description:
Consumer Complaints from the Consumer Financial Protection Bureau (CFPB)

Save Cancel

10 Copyright © SAS Institute Inc. All rights reserved.

Sas

Before you can run your project on a SAS data set, you must specify the text field that you want to analyze. You can also specify one or more category variables for the analysis. When a variable is given a category role, the Category node derives linguistic rules to try to categorize documents in specific levels of the category variable. For a complaints data set that has product labels such as mortgage, credit card, and so on, linguistic rules can be derived to attempt the categorization of new complaints accurately without requiring that each document be read and evaluated by a human judge.

Project Data

Model Studio - Build Models

Project_1

Data Pipelines

You must assign a Text variable in order to run a pipeline.

Filter

Variable Name	Type	Role	Display Variable
__uniqueid__	Numeric	Key	
Company	Character		
Complaint_ID	Character		
Complaint_Narrative	Character		
Consumer_disputed_	Character		
Date_received	Numeric		
Date_sent_to_company	Numeric		

11

Sas

Project Data

Model Studio - Build Models

Project_1

Data Pipelines

You must assign a Text variable in order to run a pipeline.

Filter

Variable Name	Type	Role	Display Variable
__uniqueid__	Numeric	Key	
Company	Character		
Complaint_ID	Character		
<input checked="" type="checkbox"/> Complaint_Narrative	Character		
Consumer_disputed_	Character		
Date_received	Numeric		

Role:

-
-
-
-

12

Sas

Project Data

Data source > CASUSER(student).COMPLAINTS

Variable Name	Type	Role	Display Variable
uniqueid	Numeric	Key	No
Complaint_Narrative	Character	Text	Yes
Flag_Mortgage	Numeric	Category	No
Product	Character		No
Sub_product	Character		No
Issue	Character		No
Sub_issue	Character		No
State	Character		No
Date_sent_to_company	Numeric		No
Complaint_ID	Character		No

13

Sas

SAS Visual Text Analytics consists of five nodes.

SAS Visual Text Analytics Nodes

SGF Abstracts

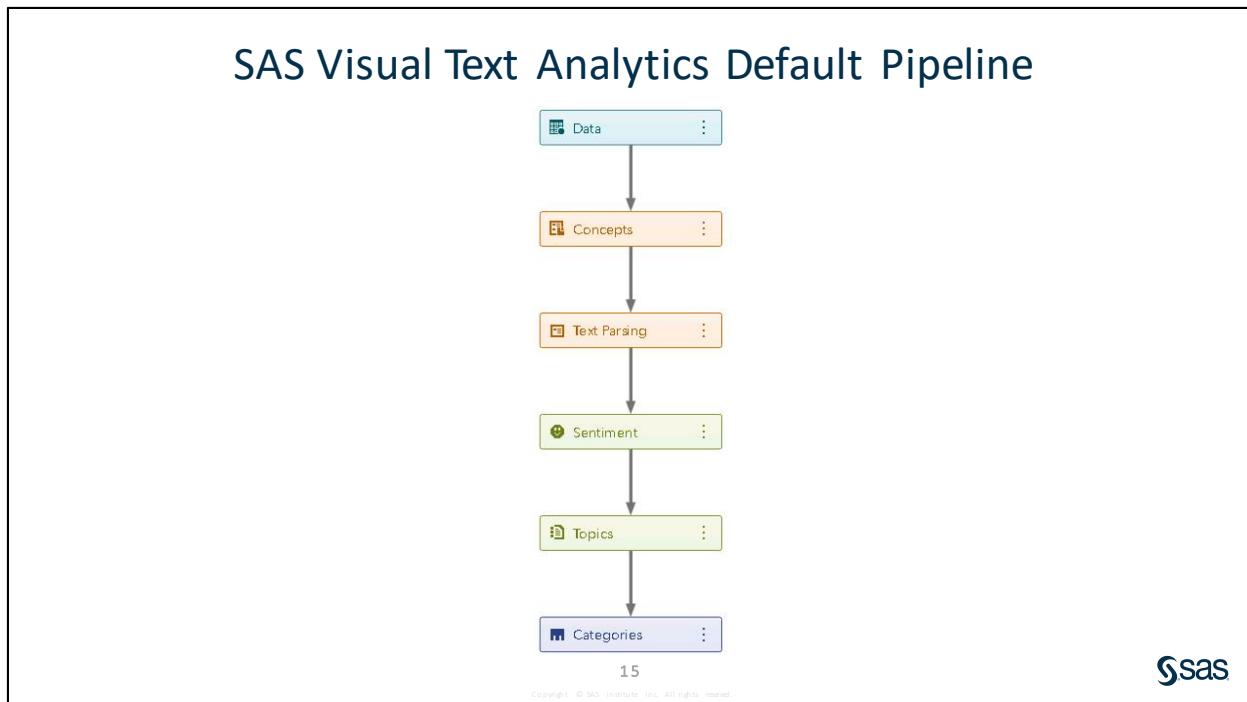
Nodes

- Natural Language Processing
 - Concepts
 - Text Parsing
- Feature Extraction
- Topics
- Text Modeling
 - Categories
- Miscellaneous
- Sentiment

14

Sas

The default pipeline for text analytics connects the five nodes in an appropriate order.



You set the options for a node by selecting the node and changing the settings that are shown to the right of the pipeline. Each node is described in detail next.

3.2 Concepts and Terms

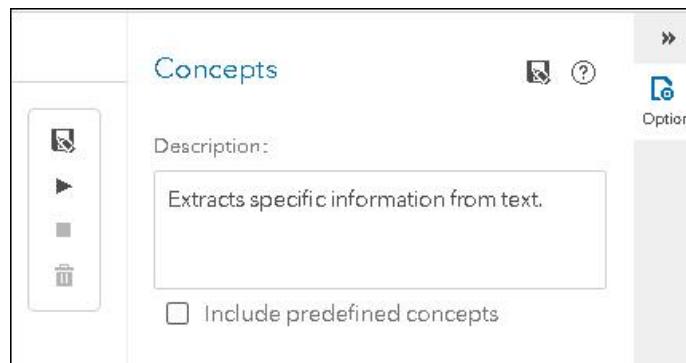
Objectives

Identify the SAS Visual Text Analytics information-retrieval features that are related to the following tasks:

- creating projects
- ***extracting terms and concepts***
- extracting system-generated topics
- extracting rules to identify documents that belong to categories
- scoring new documents

To examine the concepts' settings, select the **Concepts** node.

Concepts Node: Settings



If you do not select the **Include predefined concepts** check box, and you do not provide custom concepts, then you can delete the Concepts node from the pipeline.

Concepts are useful for analyzing information in context. You can write rules for recognizing concepts that are important to you and thereby create custom concepts. For example, you can specify that the concept *war* is identified when the terms *battle*, *bombing*, and *fighter planes* are encountered in text.

SAS Visual Text Analytics provides nine 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 *Measure* or *Organization*. You cannot edit predefined concepts or their rules, but you can append additional rules in the provided code editor. The predefined concepts supported by Visual Text Analytics were created based on multiple language relevance. In other SAS products, such as SAS Text Miner, there are more than nine predefined concepts (entities), but many of the supported concepts are relevant only in a single language or a single country, such as *Address* and *Social Security number*.

A *custom concept* is a concept whose rules you must write. (The use of custom concepts is demonstrated in a later lesson.)

Predefined Concepts

The screenshot shows a list of predefined concepts. The 'nlpMeasure' item is selected, indicated by a gray background. The other items are listed below it:

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

20
Copyright © SAS Institute Inc. All rights reserved.



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

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

Note: If you choose to exclude predefined concepts during project creation, you cannot access predefined concepts later.

You can view and explore matching documents that correspond to a concept. To view the documents that contain matches, select **Matched**.

These two icons,  , enable you to change between the *document* view, which shows one document at a time, and the *tabular* view, which shows multiple documents simultaneously.

Suppose you want to identify documents that satisfy the *nlpMeasure* concept rule. Matches within the documents are highlighted, as shown in the display on the next slide in the Document pane on the right.

Documents Consisting of Predefined Concepts

The screenshot shows the SAS Visual Text Analytics interface. On the left, there is a tree view of concepts under 'Predefined Concepts(9)'. The 'nlpMeasure' node is selected and highlighted in grey. Other nodes include nlpDate, nlpMoney, nlpNounGroup, nlpOrganization, nlpPercent, nlpPerson, nlpPlace, and nlpTime. Below this is a 'Custom Concepts(1)' section with a single 'Dosage' node. To the right, a preview window titled 'Test Sample Text' shows a document titled 'DrugReport'. The text contains several instances of predefined concepts like '40 pounds', '2 years', '10 pounds', '8 months', '4 weeks', '225 mg', '15 year old', '40 mg', '2 yrs', '1 yr', '20 hrs', '6 months', '2000mg', '1mg', 'gemalex', 'fortifex', '10mg', 'norulen', and '30 years'. A status bar at the bottom indicates 'Document 1 of 784'.



You can test a concept before you run the Concept node. The next slide shows a single test document that tests both nlpMeasure and nlpMoney.

Testing the Predefined Concept Rules

The screenshot shows the SAS Visual Text Analytics interface. On the left, there is a tree view of concepts under 'Predefined Concepts(9)'. The 'nlpMeasure' node is selected and highlighted in grey. Other nodes include nlpDate, nlpMoney, nlpNounGroup, nlpOrganization, and nlpTime. To the right, a preview window titled 'Test Sample Text' shows a document containing the sentence: 'I paid \$25.00 for the 37 ml bottle of the cleaning solution for my 50 pound black lab.' A status bar at the bottom indicates 'Document 1 of 784'.

The screenshot shows the SAS Visual Text Analytics interface. On the left, there is a tree view of concepts under 'Predefined Concepts(9)'. The 'nlpMoney' node is selected and highlighted in grey. Other nodes include nlpDate, nlpMeasure, nlpNounGroup, nlpOrganization, and nlpTime. To the right, a preview window titled 'Test Sample Text' shows a document containing the sentence: 'I paid \$25.00 for the 37 ml bottle of the cleaning solution for my 50 pound black lab.' A status bar at the bottom indicates 'Document 1 of 784'.



Recall that a custom concept is a concept whose rules you must write. (The Dosage custom concept is displayed in a previous slide.)

The Text Parsing node parses the document collection to produce a term table. Concepts from the Concepts node influence the creation of the term table.

The screenshot shows the SAS Visual Text Analytics interface with the 'Text Parsing' node selected. The node configuration window has the following details:

- Description:** Prepares text for terms analysis.
- Minimum Number of Documents:** Set to 4.
- Lists:**
 - Specify a custom start or stop list
 - List type:** Stop list
 - Start list:** Select a table (Browse button)
 - Stop list:** Select a table (Browse button)
 - Specify a synonym list
 - Synonym list:** Select a table (Browse button)

A progress bar at the bottom of the window shows '23'.

You use start lists and stop lists to control which terms are used and which are ignored.

A *start list* is a data set that contains a list of terms that should be **included** in the parsing results. If you use a start list, then only terms that are included in that list appear in the parsing results.

A *stop list* is a data set that contains a list of terms that should be **excluded** from the parsing results. You can use stop lists to exclude terms that contain little information or that are extraneous to your Visual Text Analytics tasks. A default stop list is provided for English (Sashelp.EngStop). Dictionary tables that are used as start or stop lists can be SAS tables. They do not need to be CAS tables.

Note: CAS (Cloud Analytic Services) tables are the in-memory tables that contain the documents that you are analyzing.

A *synonym list* is a SAS data set that identifies pairs of words that should be treated as single terms for the purposes of analysis. Synonym lists are stored in data sets and have a required format.

To examine the terms that are used in the analysis, right-click the **Text Parsing** node and select **Open**.

Term	Role	Documents	Frequency
not	ADV	658	1174
take	V	677	1105
depression	N	492	616
feel	V	371	517
year	N	395	502
drug	N	342	487
day	N	344	479
medication	N	304	446
work	V	356	439
go	V	300	392

Term	Role	Documents	Frequency
i	PRO	1081	5003
be	V	1129	4186
have	V	1069	2849
and	CONJ	1029	2749
the	DET	900	2644
to	PPOS	902	2404
it	PRO	882	2100
a	DET	779	1694
my	DET	812	1547
for	PPOS	834	1454

24

Sas

The default view shows the Kept Terms pane on the left and the Dropped Terms pane on the right. By default, both panes are sorted in descending order based on the number of documents in which each term appears. You can move a kept term to the Dropped Terms pane by selecting the term and clicking (Drop terms). You can move a dropped term to the Kept Terms pane by selecting the term and clicking (Keep terms).

Note: If you make changes to the terms, you must rerun the project to see the effects of your changes.

The Kept Terms pane displays all the terms in the document collection that were kept. The Role column displays each term's role, if one can be determined. A role could be a part of speech or a concept. To view the synonyms that were assigned to a term, click the triangle that appears next to that term.

By default, the lists of terms are sorted in descending order of a term's frequency.

- A *term* is defined as a label for a group of characters, strings, or patterns that represent a single concept (an idea) 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, identifies the term's part of speech, or identifies the concept that is associated with the term.
- For alphabetic writing systems, a *word* is a *token* (sequence of characters without a separator), and a term contains one or more words. There is a subtle difference between a term and a concept. When the Concept node appears before the Text Parsing node, a concept is eligible to be placed in the term table.
- A *surface form* is a variant of a term that is in a matched subset of text. Surface forms can include inflected forms, synonyms, misspellings, and other ways of referring to a term.
- *Stemming* is the process of identifying inflected forms for nouns and verbs. For example, *going* is an inflected form of *go*.

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

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

You can also include the following variables:

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

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

You can view documents that contain a kept term by selecting the term and selecting **Matched** in the Documents pane.

Text Parsing: Matched Documents

Kept Terms (115)

Term	Role	Documents	Frequency
not	ADV	658	1174
> take	V	677	1105
<input checked="" type="checkbox"/> depression	N	492	616
> feel	V	371	517
> year	N	396	602
> drug	N	342	487
> day	N	344	479
> medication	N	304	446
> work	V	356	439
> go	V	300	392

Dropped Terms (656)

Term	Role	Documents	Frequency
i	PRO	1081	5003
> be	V	1129	4186
> have	V	1069	2849
and	CONJ	1029	2749
the	DET	900	2544
to	PPOS	902	2404
it	PRO	882	2100
a	DET	779	1594
my	DET	812	1547
for	PPOS	834	1454

Documents (492)

All **Matched**

DrugReport

... due to unrelenting depression. I had lost my sister mom within 8 months and although I had been on an antidepressant for a long time before I grieved but still couldn't get over the depression. Within 3-4 days both I and my husband noticed a significant improvement. I felt better more like me than I had in years. Only thing I notice was I sometimes mix up words Anybody else do that? Not bad enough for me to stop med tho.

... an abyss of depression, one which I have never experienced before, I had to go back on the propantheline off for a month, even then I was having a continuous panic attack with heart pounding, no eating for 4-5 days, severe sense of loneliness, guilt, and poor self-worth. It was absolutely miserable that lasted for about 6-10 days. I have family support, but needed to see a psychiatrist (still waiting for visit). Be very cautious going on this med. It's great for a couple weeks, but very exhausting.

... to Escitalopram for depression. It truly was a wonderful combination. I was happier, and nicer, than I'd been in years. Major problem: I developed Parkinson symptoms. Falling, tripping up and down stairs, unsteady on

Document 1 of 492

Sas

You can also view similarity scores for a term.

Similarity Scores

Model Studio - Build Models
Drug Reports > Text Parsing - Manage Terms
Kept Terms 1516
Term similarities for "depression"

Term	Similarity	Role	Documents	Frequency
depression	1.000	N	492	616
major	0.668	A	64	65
major depression	0.648	nlpNounGroup	27	27
anxiety	0.640	N	187	231
immensely	0.571	ADV	4	4
depression	0.563	PN	9	10
alleviate	0.560	V	13	13
combine	0.547	V	10	10

Copyright © SAS Institute Inc. All rights reserved.

Sas

Similarity scores are derived based on the linguistic distance between words as measured by co-occurrence frequencies. Social network metrics can be used to derive a social distance between two people based on community membership, and similar arguments can be used to derive the distance between words based on document membership (co-occurrence).

To view a term map for a term, select that term in the Kept Terms pane and click (Show Term Map).

Term Map with Associated Links

Centered term: a term in the term table that you choose to investigate

Concept linked term: a term that co-occurs with a centered term

Copyright © SAS Institute Inc. All rights reserved.

Sas

The Term Map window displays a term map for the selected term. In the preceding image, the selected term is *enterprise*, and it is represented by the largest circle in the map. The term *guide* has the strongest association with *enterprise*. For additional information about the term map display, click  (Help) in the term map display.

The standard text analytics pipeline places the Concepts node before the Text Parsing node. If you place the Concepts node after the Text Parsing node, the term table is available to provide selections for building custom concepts.

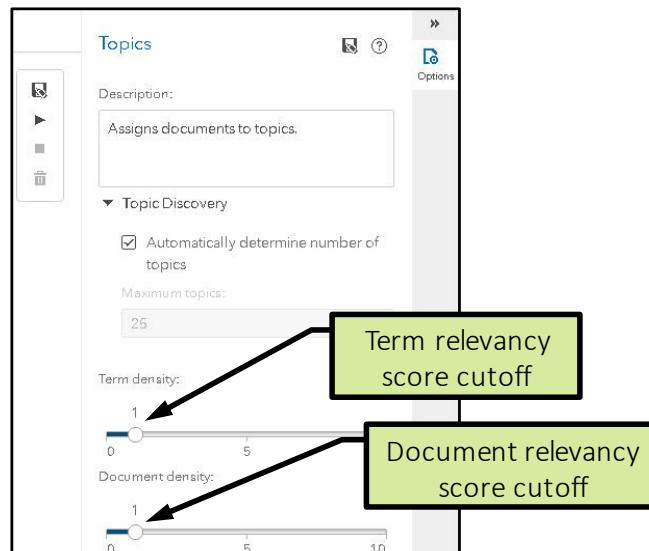
3.3 Machine-Generated Topics

Objectives

Identify the SAS Visual Text Analytics information-retrieval features that are related to the following tasks:

- creating projects
- extracting terms and concepts
- ***extracting system-generated topics***
- extracting rules to identify documents that belong to categories
- scoring new documents

Topics Node



Machine-Generated Topics

The screenshot shows the SAS Model Studio interface with the 'Topics' pane open. The 'Topics' table lists nine entries with columns for Topic, Created by, and Document count. Three annotations are overlaid on the interface:

- A box labeled "Split a topic into two topics" points to the split icon in the toolbar above the topics table.
- A box labeled "Merge two or more topics" points to the merge icon in the toolbar above the topics table.
- A box labeled "Promote a topic to a category" points to the promote icon in the toolbar above the terms table.

The 'Terms' pane to the right displays a list of terms with their roles, document counts, and frequencies. The SAS logo is visible in the bottom right corner.

continued...

SAS Visual Text Analytics: Automatic Topic Discovery

Feature

1. Automated machine discovery of core topics (themes)
2. Relevancy threshold weights that can be changed

Value or Benefit

1. Eliminates the need to manually code training documents
2. Is easily adjusted to refine terms that are included in the topics for downstream category definitions

SAS Visual Text Analytics: Automatic Topic Discovery

Feature	Value or Benefit
3. Splitting of topics	3. Identify subtopics represented by a primary topic
4. Merging of topics	4. Combine similar topics to acquire desired results
5. One button action to convert topics to categories	5. Readily create categories for production classification definitions

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 Topics page displays all the topics that SAS Visual Text Analytics identified. The default name of a topic is the top five terms based on relevancy scores. The terms table displays every term in the topic, its calculated relevancy, its assigned role, and the number of documents that contain that term. These terms are sorted in descending order based on their relevancy scores.

Document-level sentiments are also generated if they are preceded by a Sentiment node. The percentage of the documents in the topic that have a sentiment score of positive, negative, and neutral appears with each topic.

Splitting Topics

If a topic seems to convey information about two or more themes, you might want to split the topic to see whether the two themes emerge.

Merging Topics

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

Promoting Topics

The next step in the analysis process 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 (**Add topics as categories**). When you click this icon, SAS Visual Text Analytics adds the selected topic to the Categories page. You can promote multiple topics to categories at one time.

Editing Topic Properties

You can edit the properties that affect all topics by changing the settings for the Topics node. *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.) Increasing the term density number captures fewer documents.

This value affects the number of documents that belong to a topic. (For example, having fewer terms in a topic captures fewer documents.) Values that you enter are rounded to the nearest integer or half-integer.

You can also designate a maximum number of topics that you want generated for the project. If you do not enter a value, the software uses default methods to generate topics from important terms.

Run the topics to see the results of your changes.

The benefits of using SAS Visual Text Analytics for automatically extracting topics in large, unstructured data are presented in the two preceding slides.

Creating Custom Topics

You can select a subset of terms in a derived topic to create a custom topic. The custom topic assigns a term weight of 1 to each term. You can merge two topics to form a new topic. Then you can select terms from the merged topic to create a custom topic. This provides flexibility when you are creating custom topics. You can use subject-matter expertise to build customized dictionaries. Although a custom topic acts much like a custom concept that uses CLASSIFIER rules, the results can be different. This is because concepts apply Boolean operations, whereas topics apply numeric scoring.

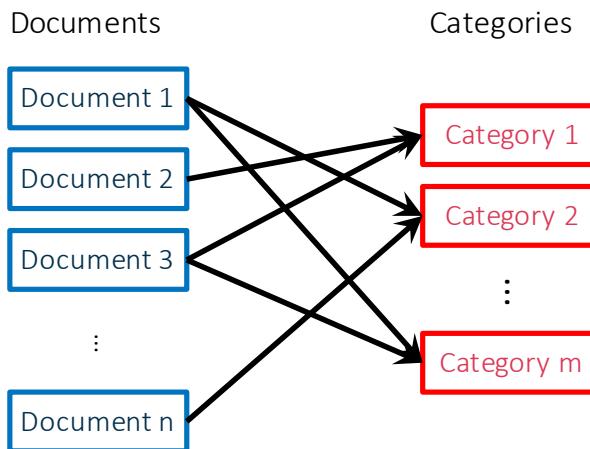
3.4 Categories

Objectives

Identify the SAS Visual Text Analytics information-retrieval features that are related to the following tasks:

- creating projects
- extracting terms and concepts
- extracting system-generated topics
- *extracting rules to identify documents that belong to categories*
- scoring new documents

SAS Visual Text Analytics Categorization: Multiple Categories per Document



Unlike disjoint clustering, SAS Visual Text Analytics can categorize the same document into many relevant categories.

Assessing Categorization Outcomes in SAS Visual Text Analytics

- **True Positives:** You want to retrieve the information that was requested.
- **False Positives:** You want to avoid retrieving information that is unrelated to the request.
- **False Negatives:** You want to avoid omitting information that is related to the request.
- **True Negatives:** You do not want to retrieve information that is irrelevant.

39

Copyright © SAS Institute Inc. All rights reserved.



Assessing Categorization Results

Notation

ND = Number of documents

NS = Number of selected documents (retrieved)

TP = Number of true positives

FP = Number of false positives

TN = Number of true negatives

FN = Number of false negatives

		Action	
		Retrieved	Omitted
Actual	Contain Info	TP	FN
	Absent Info	FP	TN
	Column Totals	NS	ND-NS

40

Copyright © SAS Institute Inc. All rights reserved.



Assessing Categorization Results

Misclassification Rate = Fraction misclassified

$$_{\text{MISC}} = (\text{FP} + \text{FN})/\text{ND}$$

Precision = Fraction of retrieved documents that are relevant

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP})$$

Recall = Fraction of relevant documents that are retrieved

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN})$$

		Action	
		Retrieved	Omitted
Actual	Contain Info	TP	FN
	Absent Info	FP	TN
	Column Totals	NS	ND-NS

Copyright © SAS Institute Inc. All rights reserved.



Assessing Categorization Results

F1 Statistic = Harmonic mean of precision and recall

$$F1 = 2/[(1/\text{Precision})+(1/\text{Recall})] = (2 * \text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

(F1 is also called the *F-Measure*.)

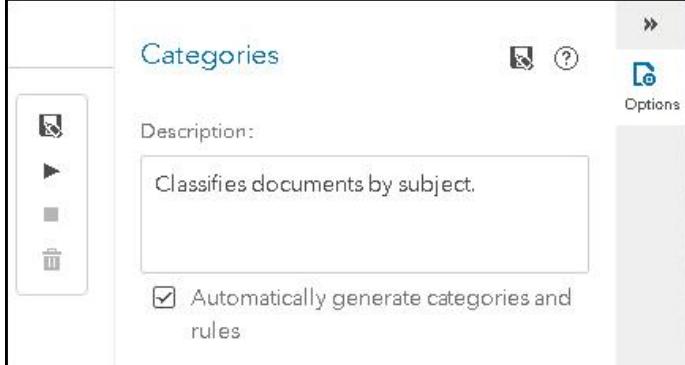
42

Copyright © SAS Institute Inc. All rights reserved.



Assessment metrics are shown above.

Categories



The screenshot shows the 'Categories Node' window. On the left is a sidebar with icons for search, back, forward, and delete. The main area has a title bar 'Categories' with a magnifying glass icon, a question mark icon, and a double arrow icon. Below the title is a 'Description:' label followed by a text box containing the text 'Classifies documents by subject.' To the right of the text box is an 'Options' button with a gear icon. At the bottom of the main area is a checked checkbox labeled 'Automatically generate categories and rules'. The bottom right corner of the window contains the SAS logo.

A **category** identifies a group of documents that share a common characteristic. The rules for each category are executed against the input data set.

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

You create categories by promoting a topic to a category, specifying a category variable when assigning roles to the project data, or creating a new category. You can edit the rules that are automatically generated for category variables and for topics that are promoted to categories.

Categories: Machine-Generated Rules

Code is valid.

abstract	Relevancy	Sentiment
PROC COMPUTAB is used to generate tabular reports in a spreadsheet-like format. PROC COMPUTAB has been around a long time, but it is time to replace it with reporting procedures that are more modern. This paper shows how to create hundreds of Excel tables using...	11.000	
... PROC SQL, can you identify at least four ways to select and create variables, create macro variables, create or modify table structure, and change table content? Learn how to apply multiple PROC SQL programming options through task-based examples. This hands-on workshop reviews topics in table...	9.000	
... workshop provides hands-on experience using a combination of DataFlux Data Management Studio		

Document 1 of 290

You can view document matches to see the documents that are assigned to the selected category.

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

Note: The sentiment score for each document is displayed only if you specified a Sentiment node in the pipeline before the Categories node.

Category Rules: Complaints

Validation is out of date.

Validate a rule.

Category Rules: Complaints

Category Rules: Complaints

Category Rules: Complaints

Documents 46811 Test Sample Text

All Matched

Complaint_Narrative Relevancy Sentiment

...to obtain a loan modification from our lender , XXXX Home Mortgage for many years! They have claimed that they did not receive all requested documents , which we sent via XXXX mail, and closed our file many times, to which we appealed. My husband is rated XXXX % XXXX by the Veterans Administration from XXXX...	1760.000	
...in XXXX my wife company laid her off due to she could no longer perform her duties at work due to XXXX. 1. XXXX (notice of intent to foreclose) 2. XXXX (defendant of platform modification of [\$2300.00] monthly mortgage note due to being terminated from job in XXXX 3. XXXX (defendant) passed due to XXXX....	1739.000	
... XXXX , I applied to XXXX , the mortgage servicer at the XXXX , for mortgag		

Document 1 of 46811

Test a rule

Documents 46811 Test Sample Text

A X

My **mortgage payment** was past due, and I was harassed.

47

Copyright © SAS Institute Inc. All rights reserved.

sas

You can click  **(Tree View)** to obtain a tree display of the rule. If you are in tree view, you can click  **(Rule View)** to return to the original, edited view of the rule.

Category Rules: Tree View

Categories |

Edit a Category

BUSINESS_INTELLIGENCE

AND

NOT

OR

"Enterprise Guide"

"DDE"

"Drugs"

OR

"Enterprise Miner"

"Predictive Modeling"

"Customer Intelligence"

Code is valid.

48

Copyright © SAS Institute Inc. All rights reserved.

continued...

SAS Visual Text Analytics: Configurable Categorization Rule Generation

Feature

1. Automated category rule definition is available.
2. Easy-to-understand Boolean rule definitions create a categorization model (that is, taxonomy) for classifying content.

Value or Benefit

1. There is no need to code definitions for the categories. This is done automatically.
2. With system-defined baseline taxonomy, you can easily refine rules for specific needs.

SAS Visual Text Analytics: Configurable Categorization Rule Generation

Feature	Value or Benefit
3. Rules are fully editable to ensure the desired specificity.	3. Complete control over system-generated rules is provided.
4. Rules can be enhanced, removed, or custom built.	4. Using prebuilt operators, rules can be refined, deleted, or new ones can be added to generate the desired results.

3.5 Scoring New Documents

Objectives

Identify the SAS Visual Text Analytics information-retrieval features that are related to the following tasks:

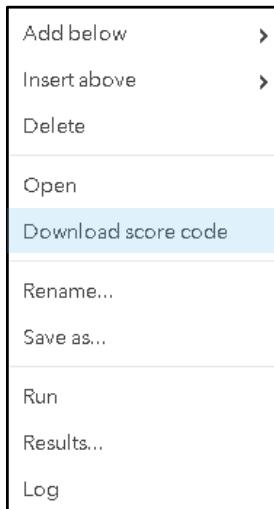
- creating projects
- extracting terms and concepts
- extracting system-generated topics
- extracting rules to identify documents that belong to categories
- *scoring new documents*

SAS Score Code

Concepts Node

Also available in

- Sentiment
- Topics
- Categories



SAS Score Code

```

/* cas library information for cas table containing the
data set you would like to score */
%let caslib_name="{put_your_caslib_name_here}";
/* the cas table you would like to score */
%let input_table_name = "{put_your_input_cas_table_name_here}";
/* the column in the cas table that contains the contains a
unique id */
%let key_column = "{put_your_id_column_name_here}";
/* the column in the cas table that contains the text data to
score */
%let document_column = "{put_your_document_column_name_here}";
/* cas library information for output cas tables to produce */
%let output_caslib_name = "{put_your_output_caslib_name_here}";

```

55

Copyright © SAS Institute Inc. All rights reserved.



The score code that is produced by SAS Visual Text Analytics includes, in Base SAS, the DS2 procedure, which enables threaded processing.

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. Score code is available for concepts, sentiment, topics, and categories. The narrative that follows assumes that you want to use the categories score code to score the data set.

When you score an external data set, the category model is applied to the external data set (called the *target data set*) and categorization information for the document collection is generated into a scored data set.

To score an external data set:

1. Download the score code. The location of the downloaded ZIP file depends on your browser.
2. Unzip the ZIP file. You get a file named CategoriesModel.mco and a SAS program named ScoreCode.sas.
3. Open the score code file in SAS Studio.
4. Edit the fields that are shown in red in the slide above.
5. Run the program.
6. Two data sets are produced, an **_out_categories** data set and an **_out_matches** data set. The **_out_categories** data set contains multiple records per input document with one record for each category that is assigned to the document. The **_out_matches** data set contains the beginning and ending byte positions of the term that satisfied the category rule. Because more than one term can define a category, the **_out_matches** data set can also have multiple records per document.

Notes about Document-Level Sentiment Scoring

Sentiment analysis is the process of identifying the message's tone or attitude (positive, negative, or neutral) as it is expressed in a document. Using a set of proprietary rules that identify and analyze terms, phrases, and character strings with respect to sentiment, SAS Visual Text Analytics assigns a sentiment score to each document.

Using these rules, the software can provide repeatable, high-quality results. The assignment of sentiment to a document is based on the attitude that is associated with the document.

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 was 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.

Rules that are generated by 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 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. Consult the user's guide for a list of available sentiment models.

3.6 Lesson Summary

SAS Visual Text Analytics delivers faster insight from unstructured, big data by automating text analytics. Visual Text Analytics eliminates tedious manual tagging and categorization for structuring unstructured data. For organizations that crave big data insight from text such as social media or customer communications, that is a huge time savings.

Visual Text Analytics helps organizations discover emerging issues, patterns, and trends in unstructured data without requiring prior knowledge of its contents. Previously, to categorize content, users needed to first define training sets and taxonomies, and then establish categorization rules.

Traditionally, the most difficult part of uncovering value in text from social media, call center notes, or warranty card comments is first building a taxonomy. Typically, this is a lengthy manual process that involves defining concepts and categories that describe a document collection. Until a taxonomy is created, the text data cannot be assessed, so taxonomies are essential to provide business value. With SAS Visual Text Analytics, you now automate much of this process and virtually eliminate the burden of manual taxonomy development.

The SAS Visual Text Analytics point-and-click interface guides data analysts through text model development. Using patented, automatic, linguistic rules creation techniques, the interface provides initial taxonomy development and defines taxonomy rules from raw text inputs.

With complete access to the machine-generated topics, rules, and concepts, data analysts can use a vast suite of prebuilt linguistic operators to further refine automatically generated results. Although the interface is suitable for new and experienced text analysts, SAS also includes interactive visual graphics, sentiment, and diagnostic metrics for extending machine automatically generated text models.

A powerful, guided methodology anchors SAS Visual Text Analytics. That methodology combines machine-learning techniques with end-user subject-matter expertise to largely automate categorization model development. SAS Visual Text Analytics represents a new era of semantic analysis that helps organizations gain rich insights from their text data while significantly simplifying the model development process for categorizing content in real time.

Lesson 4 Concept and Category Rule Definitions

4.1 SAS Visual Text Analytics Rules	4-3
4.2 SAS Visual Text Analytics Concept Rules	4-17
Demonstration: CLASSIFIER Rule.....	4-20
Demonstration: CONCEPT Rule.....	4-31
Demonstration: C_CONCEPT Rule.....	4-37
Demonstration: CONCEPT_RULE Rule.....	4-42
Demonstration: NO_BREAK Rule	4-46
Demonstration: PREDICATE_RULE Rule.....	4-51
Demonstration: REGEX Rule	4-59
4.3 SAS Visual Text Analytics Demo Category Rules	4-61
Demonstration: CATEGORY Rule	4-67
Practice.....	4-71
4.4 Lesson Summary.....	4-72
4.5 Solutions	4-73
Solutions to Practices	4-73
Solutions to Activities and Questions.....	4-74

4.1 SAS Visual Text Analytics Rules

Objectives

- Identify the SAS Visual Text Analytics rules.
- Describe the features of concept rules and demonstrate the functionality of concept rules in Visual Text Analytics.
- Describe the features of category rules and demonstrate the functionality of category rules in Visual Text Analytics.

3



LITI (language interpretation and text interpretation) syntax is used to write concept rules. When you write concept rules, you are writing rules that recognize items in context so that you can extract only the pieces of the document that match the rule. For example, in the rule

CLASSIFIER : diabetes

CLASSIFIER is the rule type, **diabetes** is the argument, and the type and argument are separated by a colon. The rule extracts all documents in your document collection that contain the word *diabetes*. In the Concepts interface, all matched documents would contain *diabetes*.

For information about editing rules using the interface and using properties settings, see “The Interactive Window for the Concepts Node” on page 47 of *SAS® Visual Text Analytics 8.3: Users Guide*. For a list of rule types, see “Which Rule Type Should I Use?” on page 79 of the same publication.

Introduction to Concept Rules

- Concept rules are written using LITI (language interpretation and text interpretation) syntax.
- Concept rules are written to recognize items *in context* so that only the pieces of the document that match the rule are extracted.
- Rule types are written in *uppercase letters*, followed by a *colon*, and then *arguments*.
- Rule modifiers can be used to further refine the set of matches, because the rule syntax varies greatly depending on the rule type.
- A single concept can reference one or more additional concepts.

Concept Rules

Rule Types

Extracting Concepts:

CLASSIFIER

CONCEPT

C_CONCEPT

CONCEPT_RULE

NO_BREAK

REGEX

REMOVE_ITEM

Extracting Facts:

PREDICATE_RULE

SEQUENCE

A concept can be defined using more than one rule type.

Complete documentation about concept rules can be found in the user's guide.

Concept Rules Naming Conventions

- Predefined rules use prefix nlp=natural language processing.
- Concept names should not be the same as terms. Use a prefix or an underscore to avoid ambiguities with terms.
- Concept names can contain only letters, numbers, and underscores

Bad Name

Concept names can contain only letters, numbers, and underscores.

- The example concept names used in this lesson all have at least one underscore character to distinguish them from actual terms in the document collection.

Concepts versus Facts

“Facts (also called *predicates*) are related pieces of information that are located and matched together.” (*SAS® Visual Text Analytics 8.3: User’s Guide*, page 78)

Concept: US president

Concept: University

Fact: Universities named after US presidents

Types of Text Extraction Ordered by Increasing Complexity

1. Token extraction
2. Term extraction (token + language \Rightarrow term)
3. Concept extraction (nouns, noun phrases)
4. Entity extraction (associates nouns with entities (for example, Person: Mr. White, Location: White House))
5. Atomic fact extraction (associates two or more concepts)
6. Complex fact extraction (natural language understanding)

Types of Text Extraction Ordered by Increasing Complexity

1. Token extraction
2. Term extraction (token + language \Rightarrow term)
3. **Concept** extraction (nouns, noun phrases)
4. **Entity** extraction (associates nouns with entities (for example, Person: Mr. White, Location: White House))
5. Atomic **fact** extraction (associates two or more concepts)
6. Complex fact extraction (natural language understanding)

Treated the same
(called *concepts*)

Wakefield (2004) addresses the different levels of information extraction by complexity. Natural language processing extracts information accurately and efficiently. Natural language understanding represents the culmination of artificial intelligence with thinking robots.

The different capabilities of the concept rules are addressed systematically.

Concept Rule Capabilities

Each of the concept rule types provides one or more of the following capabilities:

- match specific words and strings
- use wildcards to match any word
- expand word forms
- reference parts of speech
- reference defined entities
- use Boolean operators
- use regular expressions to match patterns

10

Copyright © SAS Institute Inc. All rights reserved.*continued...*

Using Punctuation in Concept Rules

- Use punctuation to qualify the matches for all rule types except CLASSIFIER and CONCEPT.
- ***Colon (:)***: separates rule types and tags
 - after a concept rule type (for example, CLASSIFIER:)
 - between arguments in a PREDICATE_RULE definition
 - before a part-of-speech tag (for example, :Prep)
- ***Comma (,)***: separates elements such as arguments in a rule definition
 - Add a space after the comma and before the next element.
 - Also, it is used to separate logical operators in a PREDICATE_RULE definition.

11

Copyright © SAS Institute Inc. All rights reserved.

Using Punctuation in Concept Rules

- *Single space*: separates strings, concept names, part-of-speech tags, and rule modifiers in CONCEPT, CONCEPT_RULE, and C_CONCEPT definitions
- *Quotation marks ("")*: enclose concept names and strings in CONCEPT_RULE, REMOVE_ITEM, and PREDICATE_RULE definitions
- *Parentheses (())*: group the elements in CONCEPT_RULE, REMOVE_ITEM, SEQUENCE, and PREDICATE_RULE rule types, and with certain Boolean operators (AND, OR, SENT, DIST_n, and ORDDIST_n)
- *Square braces ([])*: group elements in the REGEX rule type

Other LITI Components

- Boolean operators
- Special symbols used in Boolean rules
- Morphological expansion symbols
- Part-of-speech tags
- Regular expressions

The following table of Boolean operators appears in *SAS® Visual Text Analytics 8.3: User's Guide*.

Boolean Operators for Extracting Concept Rules and Facts	
Operator	Description
ALIGNED	Takes two arguments, where an argument is either a set of elements specified within a set of double quotation marks, or an operator and its arguments. Returns a match when both arguments have the same matching span of text in a document. Used with the REMOVE_ITEM rule type only. For example, the following rule says to remove the match for the concept DATE if that match is followed by the word <i>driver</i> and matches the string <i>Sunday driver</i> . This ensures that <i>Sunday driver</i> will not return as a match for DATE. CONCEPT_RULE:(ALIGNED, "_c{DATE} driver", "Sunday driver")
AND	Takes one or more arguments. Matches if all arguments occur in the document, in any order. For example, the following rule returns a match on <i>King Louis XIV</i> if it occurs in the document with <i>France</i> : CONCEPT_RULE:(AND, "_c{King Louis XIV}", "France")
DIST_n	(Distance) Takes a value for n and two or more arguments. Matches if all arguments occur within n (or fewer) tokens of each other, regardless of their order. For example, the following rule returns a match in the phrase <i>the picture with the best lighting</i> : CONCEPT_RULE:(DIST_5, "best", "_c{picture}") Note: For calculation purposes, the distance between tokens is not inclusive. For example, the distance between <i>best</i> and <i>show</i> in the phrase <i>best in show</i> is two tokens. Tokens that include hyphens are counted as one (for example, <i>merry-go-round</i> is one token).
NOT	Takes one argument. Matches if the argument does not occur in the document. Must be used with the AND operator. For example, the following rule returns a match if <i>cinema</i> , <i>theater</i> , or <i>theatre</i> occurs in the document, but <i>Broadway</i> does not: CONCEPT_RULE: (AND, (OR, "_c{cinema}", "_c{theater}", "_c{theatre}"), (NOT, "Broadway")) Note: The NOT operator applies across the entire document. All operators must have their own parentheses around themselves and their associated arguments.
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 <i>U.S.</i> , <i>US</i> , or <i>United States</i> 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 <i>The warranty claim for the washing machine was denied.</i> : 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 <i>the teacher introduced elementary statistics</i> 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 <i>best</i> and <i>show</i> in the phrase <i>best in show</i> is two tokens. Tokens that include hyphens are counted as one (for example, <i>merry-go-round</i> 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 <i>Manhattan</i> and also includes the token <i>apartment</i> . (Only <i>Manhattan</i> 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 (new line), \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 <i>Amazon</i> and <i>river</i> 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 the end of a sentence (for example, <i>Mr. Quackenbush</i> or <i>Boston, Mass.</i> could occur in the middle of a sentence). Here is a list of sentence delimiters: \r\n\r\n Two consecutive carriage returns and new lines (for documents created in Windows) \r\n \r\n Two consecutive carriage returns and new lines, separated by a space . <SPACE> Period (.) followed by an ASCII space . \n Period (.) followed by a new line

	<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 new line</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>
SENT_n	<p>(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 <i>he</i> within two sentences.</p> <p>Suppose the GENDER concept node contains the following rule:</p> <pre>CLASSIFIER:male</pre> <p>You can then write this rule:</p> <pre>CONCEPT_RULE:(SENT_2, "_c{GENDER}", "he")</pre> <p>For more information, see the SENT operator.</p>
SENTEND_n	<p>(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:</p> <pre>CLASSIFIER:female</pre> <p>Then the following rule returns a match for the concept node GENDER and the term <i>she</i> within five tokens from the end of a sentence:</p> <pre>CONCEPT_RULE:(SENTEND_5, "_c{GENDER}", "she")</pre> <p>For more information, see the SENT operator.</p> <p>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, <i>merry-go-round</i> is one token).</p>

SENTSTART_n	<p>(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 <i>The patient experienced breathing difficulty.</i>:</p> <pre>CONCEPT_RULE:(SENTSTART_5, "_c{patient}", "breathing", "difficulty")</pre> <p>For more information, see the SENT operator.</p> <p>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, <i>merry-go-round</i> is one token).</p>
UNLESS	<p>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.</p> <p>For example, the following rule does not include the token <i>river</i> in its matches. In addition, the rule returns matches for Mississippi the state and not Mississippi the river:</p> <pre>CONCEPT_RULE:(UNLESS, "river", (SENT, "_c{Mississippi}", "United States"))</pre> <p>The rule ensures that <i>river</i> does not appear between <i>Mississippi</i> and <i>United States</i> in the matches.</p> <p>Note: When you specify a concept governed directly by the UNLESS operator, specify concepts that contain only CLASSIFIER or REGEX rules.</p>

The following table of special symbols used in Boolean rules appears in *SAS® Visual Text Analytics 8.3: User's Guide*.

Special Symbols Used in Boolean Rules	
Symbol	Description
*	(Wildcard matching) Matches any characters that occur at the beginning or end of the word. For example, the argument “travel*” returns the matches <i>travels</i> , <i>traveled</i> , <i>traveler</i> , <i>traveling</i> , and so on. The argument “*room” matches <i>bedroom</i> , <i>cloakroom</i> , <i>ballroom</i> , <i>room</i> , and so on.
^	(Beginning of sentence) Starts searching at the beginning of the sentence to find a match. For example, the argument “^Independent” returns a match in this sentence: <i>Independent research was conducted</i> .

Note: Tokens (words, phrases, symbols, or other meaningful elements) need to be entered specifically to be considered for matching. For example, if you are searching for “*In this case”, use the argument “**In this case”. Also note that backward slashes (\) are used as escape characters for the asterisks (*) so that the asterisks are not treated as wildcards.

\$	(End of document) Starts searching at the end of each document of a document collection to find a match. For example, the argument “deleted.\$” returns a match in the following sentence when it is the last sentence of a document: “ <i>All the files were hastily deleted.</i> ” An argument that contains only the symbol \$ will return a match for the last token of every document in a document collection. Note: Tokens (words, phrases, symbols, or other meaningful elements) need to be entered specifically to be considered for matching. For example, the argument “deleted\$” would not produce a match in the above example because the ending period (.) was not specified.
@	(Morphological expansion) Expands the category rule to match all inflectional forms of the word in the argument. For example, the argument “wonder@” returns the matches <i>wonder</i> , <i>wonders</i> , <i>wondered</i> , <i>wondering</i> , and so on (but does not return a match on <i>wonderful</i>). Note: 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 <i>grath</i> would return a match in the rule.
@A	(Morphological expansion for adjectives) Expands the category rule to match inflected comparative and superlative adjective forms of the word in the argument. For example, the argument “happy@A” returns the matches <i>happier</i> and <i>happiest</i> . Note: If you apply @A to a word that is never an adjective, no expansion occurs.
@N	(Morphological expansion for nouns) Expands the category rule to match all noun forms of the word in the argument. For example, the argument “quality@N” returns the matches <i>quality</i> and <i>qualities</i> . Note: If you apply @N to a word that is never a noun, no expansion occurs.
@V	(Morphological expansion for verbs) Expands the category rule to match all verb forms of the word in the argument. For example, the argument “transfer@V” returns the matches <i>transfer</i> , <i>transfers</i> , <i>transferred</i> , and <i>transferring</i> . Note: If you apply @V to a word that is never a verb, no expansion occurs.
_L	(Literal matching) Matches a literal string. Useful when you want to match a string that includes symbols. For example, the argument “\$USD_L” returns the match \$USD. Note: Tokens (words, phrases, symbols, or other meaningful elements) need to be specified by the user to be considered for matching.
_C	(Case matching) Specifies case-sensitive matching. For example, the argument “Geek_C” returns the match <i>Geek</i> , but not <i>geek</i> .

The following table of morphological expansion symbols used in Concept rules appears in *SAS® Visual Text Analytics 8.3: User's Guide*.

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 <i>wonder</i>, <i>wonders</i>, <i>wondered</i>, <i>wondering</i>, and so on.</p> <p>Note: 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 <i>grath</i> 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 <i>happier</i> and <i>happiest</i>.</p> <p>Note: 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 <i>quality</i> and <i>qualities</i>.</p> <p>Note: 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 <i>transfer</i>, <i>transfers</i>, <i>transferred</i>, and <i>transferring</i>.</p> <p>Note: If you apply @V to a word that is not a verb, no expansion occurs.</p>

Note: Morphological expansion does not include misspellings that have been detected in the Text Parsing node.

The following table of part-of-speech tags appears in *SAS® Visual Text Analytics 8.3: User's Guide*.

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, qoh
:inc	Unknown word	slaster, lijer
:Int	Interjection	hah, hello, tallyho, zounds
: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, Woodfield
:PossDet	Possessive determiner	our, his, my
:PossPro	Possessive pronoun	mine, yours, hers
:PreDet	Pre-determiner	quite, such, all
:Prefix	Prefix	cross, ex, multi
:Prep	Preposition	on, under, across
:Pro	Pronoun Relative pronoun	he, one, somebody, me myself, oneself, themselves
:Ptl	Particle	away, forward, in
:sep	Separator and punctuation	; , /
:time	Time	7AM, 10:00 pm
:url	File names, pathnames, URL	A:/mydir/file.txt, www.sas.com
:V	Undeclined be, do, or have auxiliary Undeclined verb First person singular verb	be, do, have go, see, love am
:V3sg	Third person singular be, do, or have auxiliary Third person singular verb	is, does, has goes, sees, loves
:Ving	Present participle be, do, or have auxiliary Present participle	being, doing, having bucketing, climbing
:Vpp	Past participle be, do, or have auxiliary Past participle	been, done, had dashed, factored, gone
:Vpt	Past tense be, do, or have auxiliary Past tense verb	was, were, did, have dashed, factored, went

:WAdv	Adverbial <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

A table of regular expression metacharacters appears in *SAS® Visual Text Analytics 8.3: User's Guide*. The metacharacters are the same ones used by typical regular expression parsers like Perl. The table is not reproduced here.

4.2 SAS Visual Text Analytics Concept Rules

CLASSIFIER Definition Syntax

`CLASSIFIER:<match_key>,<returned_info>`

- The *match_key* is the literal string to be matched in the document.
- The *returned_info* is an optional string to be returned for the matched concept.
- The comma can be omitted if *returned_info* is not specified.
- You can use \c to match a comma within the *match_key*.

15

Copyright © SAS Institute Inc. All rights reserved.



CLASSIFIER: This rule identifies single terms or strings that you want to match 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.

One of the quickest ways to transition from dictionary-based text retrieval systems to contextual systems is to process a dictionary of specialty terms. For example, the SAS data table **AviationTerms** contains 190 aviation terms, such as *aileron*, *fuselage*, *glider*, *taxis*, and *zeppelin*. Using this dictionary, you can write a set of simple classifier rules. The program below reads a dictionary data set that has the character variable **Term**. Then it constructs a set of classifier rules that are stored in a TXT file that can be copied and pasted into the Concepts node editor.

```
filename outpfile "<file spec>";
libname VTXT "<library spec>";
data _null_;
  set VTXT.AviationTerms;
  file outpfile;
  put "CLASSIFIER:" Term;
run;
```

The AviationConcepts.txt file in the course data folder has the CLASSIFIER rules for an aviation concept based on the **AviationTerms** data table. The first few rows of the classifier specifications are below.

```
CLASSIFIER:acceleration
CLASSIFIER:accuracy
CLASSIFIER:ace
CLASSIFIER:advance
CLASSIFIER:aerodome
CLASSIFIER:aerodynamic
CLASSIFIER:aerodyne
CLASSIFIER:aeronautics
CLASSIFIER:afterburner
CLASSIFIER:aileron
CLASSIFIER:aircraft
CLASSIFIER:airplane
CLASSIFIER:airstream
```

CLASSIFIER Rule Type

Capability

- ✓ *Match specific words or strings.*
 - ✗ Use wildcards to match any word.
 - ✗ Expand word forms.
 - ✗ Reference parts of speech.
 - ✗ Reference defined entities.
 - ✗ Use Boolean operators.
 - ✗ Use regular expressions to match patterns.

Identifies single terms or strings that you want matched in context

CLASSIFIER Rule Examples

Rule	Example Matches
CLASSIFIER : Orion Glimmer C20	<i>Orion Glimmer C20</i> <i>orion glimmer c20</i> <i>ORION GLIMMER C20</i>
CLASSIFIER : very good value	<i>Very good value</i> <i>VERY good value</i> <i>very GOOD value</i>
CLASSIFIER : disappointing	<i>Disappointing</i> <i>disappointing</i>

17

Copyright © SAS Institute Inc. All rights reserved.



Matching with Coreference in CLASSIFIER Rule

LTI Definition	Result
CLASSIFIER : [coref=drug]:Elevex	If the term <i>Elevex</i> appears at least once in the document, the term <i>drug</i> also returns matches for <i>Elevex</i> .
CLASSIFIER : [coref=author,writer,Larsson]: Stieg Larsson	If the string <i>Stieg Larsson</i> appears at least once in the document, the terms <i>author</i> , <i>writer</i> , and <i>Larsson</i> also return matches for <i>Stieg Larsson</i> .

18

Copyright © SAS Institute Inc. All rights reserved.



Additional modifiers are accommodated. For example, the EXPORT feature can be used to invoke a separate concept when a classifier term is in a document. Suppose the CLIENT concept is defined by this statement:

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

If a document matches *Sokolov*, then the CLIENT concept is flagged. If the document also contains *accounts receivable*, then the AR concept is flagged.



CLASSIFIER Rule

This demonstration illustrates the CLASSIFIER rule functionality in SAS Visual Text Analytics.

1. Navigate to SAS Drive. Use the course credentials to sign in.
2. In the top left of SAS Drive, click (the **Show applications** menu) and select **Build Models**.
3. Double-click **Rules_Demo** to open the previously created SAS Visual Text Analytics project.

The following specifications were used to create this project:

New Project

Name: *

Type: *



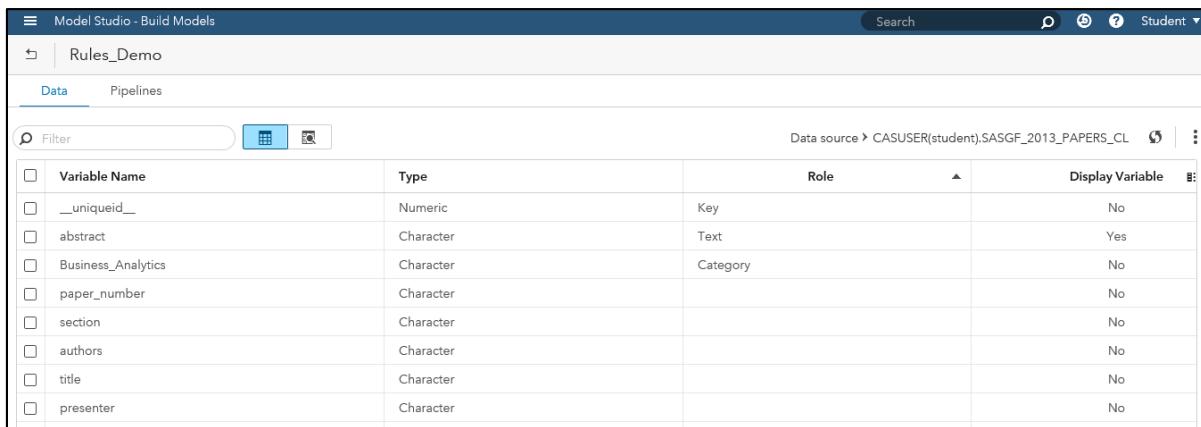
Data source: *

Project language: *



Description:

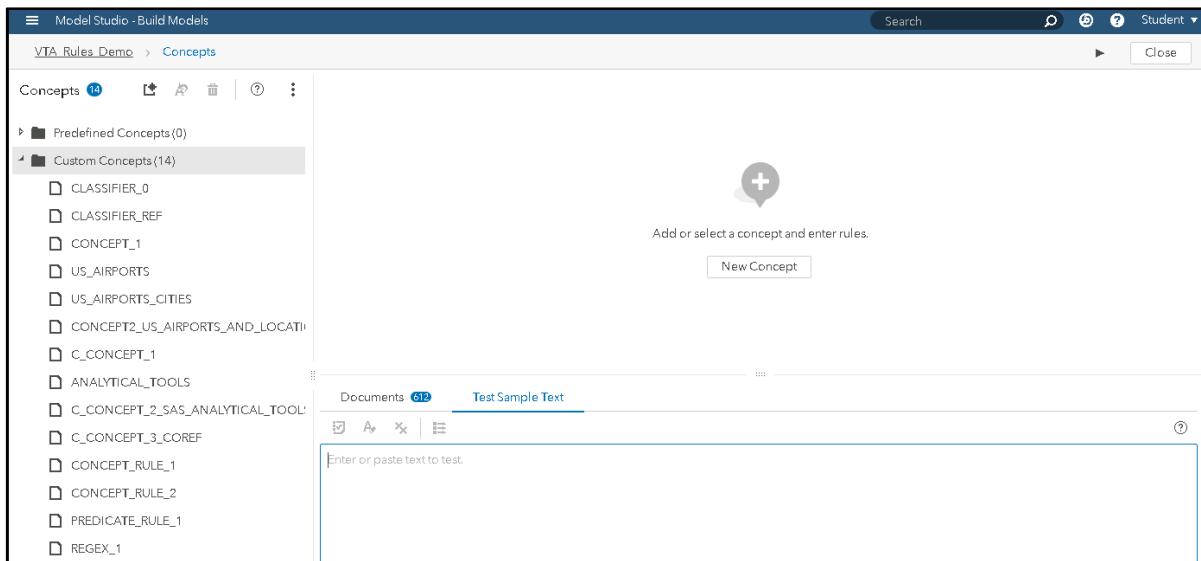
The **SASGF_2013_PAPERS_CL** data set has the characteristics shown in the display below.



A screenshot of the SAS Model Studio interface titled "Model Studio - Build Models". The left sidebar shows a project named "Rules_Demo" with "Data" selected. The main area displays a table of variables from the "SASGF_2013_PAPERS_CL" data source. The table has columns for Variable Name, Type, Role, and Display Variable. The variables listed are: __uniqueid_ (Numeric, Key), abstract (Character), Business_Analytics (Character), paper_number (Character), section (Character), authors (Character), title (Character), and presenter (Character). All variables have a Role of "Text" and a Display Variable setting of "No".

Variable Name	Type	Role	Display Variable
__uniqueid_	Numeric	Key	No
abstract	Character	Text	Yes
Business_Analytics	Character	Category	No
paper_number	Character		No
section	Character		No
authors	Character		No
title	Character		No
presenter	Character		No

4. Select **Pipelines**. The default text analytics pipeline is used.
5. Right-click the **Concept** node and select **Open**.



A screenshot of the SAS Model Studio interface titled "Model Studio - Build Models". The left sidebar shows a project named "VTA_Rules_Demo" with "Concepts" selected. The main area displays a tree view of concepts under "Custom Concepts(14)". The nodes listed are: CLASSIFIER_0, CLASSIFIER_REF, CONCEPT_1, US_AIRPORTS, US_AIRPORTS_CITIES, CONCEPT2_US_AIRPORTS_AND_LOCATI, C_CONCEPT_1, ANALYTICAL_TOOLS, C_CONCEPT_2_SAS_ANALYTICAL_TOOL, C_CONCEPT_3_COREF, CONCEPT_RULE_1, CONCEPT_RULE_2, PREDICATE_RULE_1, and REGEX_1. To the right of the tree view is a text input field labeled "Test Sample Text" with the placeholder "Enter or paste text to test." and a "New Concept" button.

CLASSIFIER Rule Demo 1

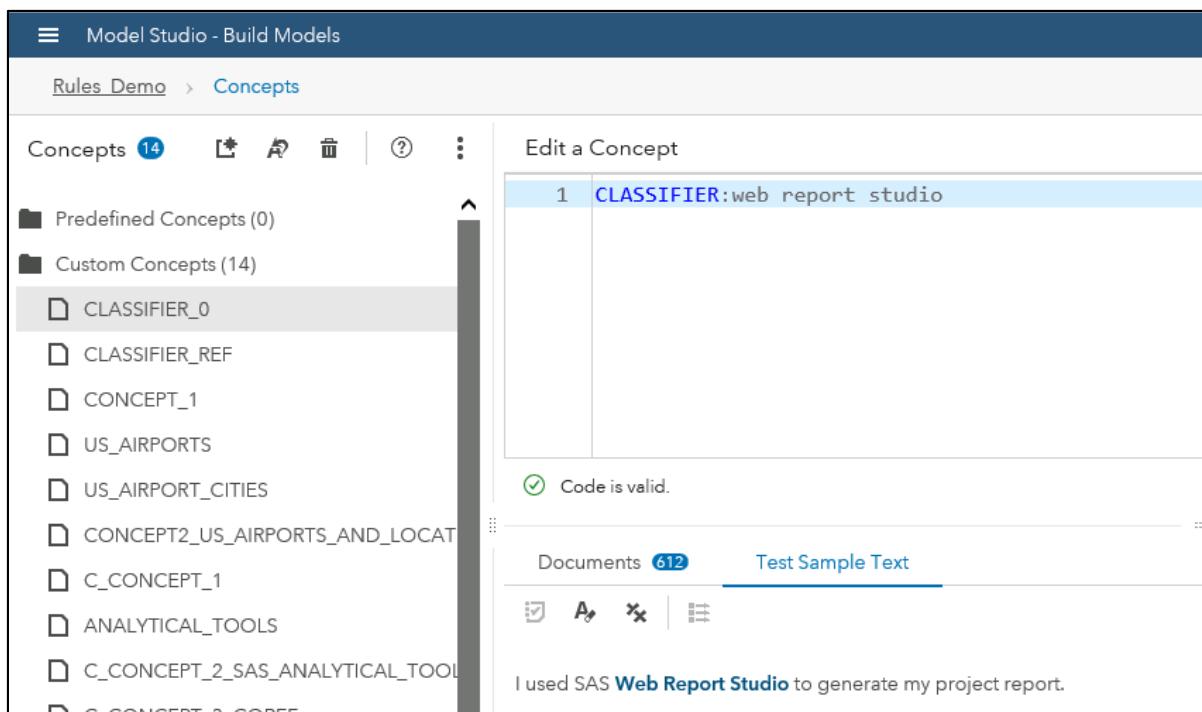
1. Select the predefined custom concept **CLASSIFIER_0** and examine the associated LITI rule in the Edit a Concept window.

CLASSIFIER: web report studio

The CLASSIFIER rule above identifies documents that contain the text string *web report studio*.

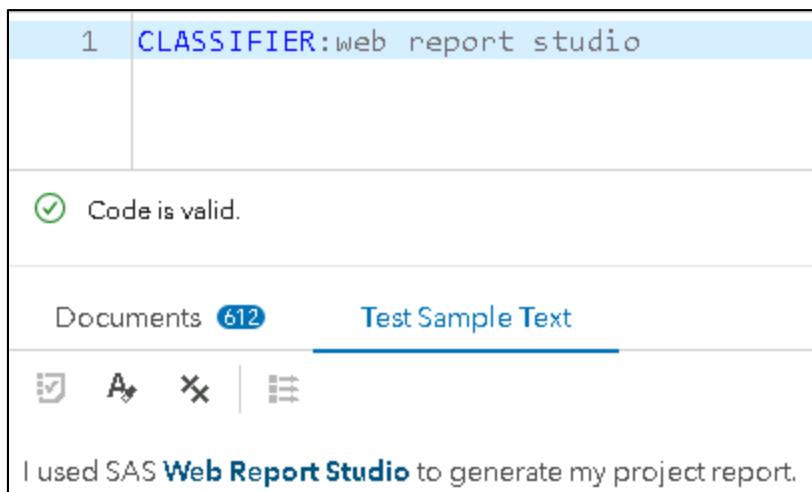
2. To validate this rule, click  (**Validate rules**).
3. In the Test Sample Text input box, enter the following text: **I used SAS Web Report Studio to generate my project report.**

4. Click  (Test text) to test the rule.



The screenshot shows the Model Studio interface with the Concepts tab selected. On the left, there's a tree view of concepts: Predefined Concepts (0) and Custom Concepts (14), with CLASSIFIER_0 selected. The main area is titled "Edit a Concept" and shows a table with one row containing the code "1 CLASSIFIER:web report studio". Below the table, a message says "Code is valid." with a green checkmark. At the bottom, there's a "Test Sample Text" tab which is active, showing the text "I used SAS Web Report Studio to generate my project report." where "Web Report Studio" is highlighted in blue.

If the rule is validated, the string is highlighted in the text document.



This is a close-up of the "Test Sample Text" tab from the previous screenshot. It shows the validation message "Code is valid." with a green checkmark. Below it is the text "I used SAS Web Report Studio to generate my project report." with the phrase "Web Report Studio" highlighted in blue.

CLASSIFIER Rule Demo 2

- Select the predefined custom concept **CLASSIFIER_REF** and observe the associated LITI rule in the Edit Rules window.

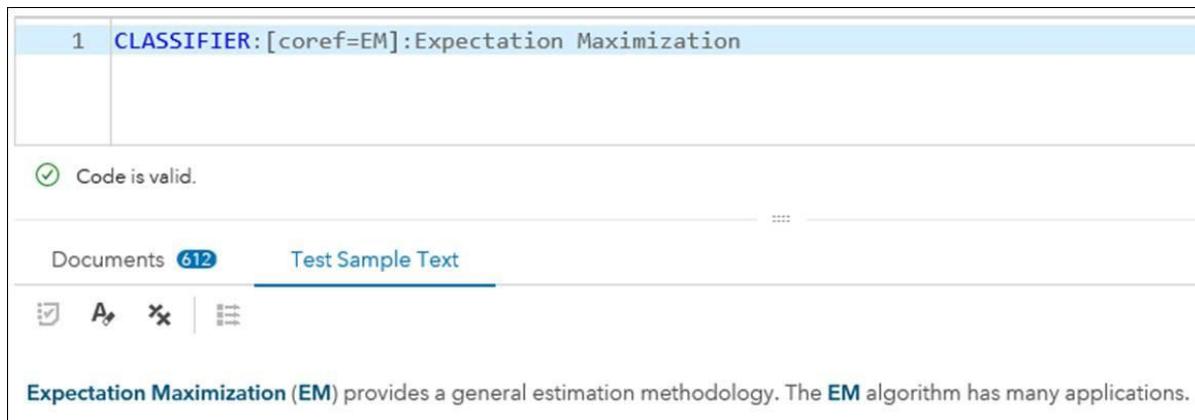
CLASSIFIER: [coref=EM] :Expectation Maximization

The above CLASSIFIER rule identifies documents that contain the text string *Expectation Maximization* and subsequent references to any Expectation Maximization term (that is, a term with the abbreviation *EM*).

- To validate this rule, click the **Test Sample Text** tab.

3. In the Test Sample Text input box, enter the following text: **Expectation Maximization (EM) provides a general estimation methodology. The EM algorithm has many applications.**
4. Click  (**Test text**) to test the rule.

If the rule is validated, the string *Expectation Maximization* and subsequent references to any Expectation Maximization term (a term with the abbreviation *EM*) are highlighted in the text document.



The screenshot shows the SAS Visual Text Analytics interface. At the top, there is a code editor window with the following content:

```
1 CLASSIFIER: [coref=EM]:Expectation Maximization
```

Below the code editor, a message indicates that the code is valid:

 Code is valid.

At the bottom of the interface, there is a navigation bar with the following items:

- Documents 612
- Test Sample Text (which is currently selected)

Below the navigation bar, there is a toolbar with icons for search, edit, and other functions. The main content area displays the test sample text:

Expectation Maximization (EM) provides a general estimation methodology. The EM algorithm has many applications.

End of Demonstration

4.01 Multiple Choice Question

Which CLASSIFIER rule has the correct syntax to extract the text SAS Visual Text Analytics?

- a. CLASSIFIER: SAS Visual Text Analytics _C
- b. CLASSIFIER: SAS Visual Text Analytics
- c. CLASSIFIER: "SAS Visual Text Analytics"
- d. CLASSIFIER: ('SAS Visual Text Analytics')

20

Copyright © SAS Institute Inc. All rights reserved.

CONCEPT Rule Type

- CONCEPT is a rule type. It is not to be confused with a “concept” in the general sense.
- The CONCEPT rule type 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 in the definition.
- The CONCEPT rule type could reference a predefined CLASSIFIER rule by its name, and thereby access a list of airport codes.

23

Copyright © SAS Institute Inc. All rights reserved.

CONCEPT Rule Type

Capabilities

- ✓ Match specific words or strings.
- ✓ Use wildcards to match any word.
- ✓ Expand word forms.
- ✓ Reference parts of speech.
- ✓ Reference defined entities.
- ✗ Use Boolean operators.
- ✗ Use regular expressions to match patterns.

24

Copyright © SAS Institute Inc. All rights reserved.

CONCEPT Definition Syntax

The CONCEPT rule identifies related information by referencing other concepts.

When you write CONCEPT definitions, enter at least one space before each of the following items:

- tokens (literal strings)
- concept names
- part-of-speech tags
- `_w` and `_cap` terms
- the `_c` marker, when preceded by a token, comma (,), or the name of a concept

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

25

Copyright © SAS Institute Inc. All rights reserved.

Using Wildcards to Match Any Word in a CONCEPT Rule

- `_w` matches any word.
- `_cap` matches any word that begins with an uppercase letter.
- Both are specified in lowercase.

Rule and Match Key	Example Matches
<code>CONCEPT:SAS _w</code>	SAS <i>employees</i> SAS <i>software</i>
<code>CONCEPT:SAS _cap</code>	SAS <i>Institute</i> SAS <i>Analytics</i>
<code>CONCEPT: _w SAS</code>	<i>Base</i> SAS <i>foundation</i> SAS

Expanding Word Forms in a CONCEPT Rule

Symbol	Position	Result
<code>@</code>	Suffix	Stems the word that precedes the <code>@</code> to include <i>all forms</i> of the word
<code>@N</code>	Suffix	Stems the word that precedes the <code>@N</code> to include <i>noun forms</i> of the word
<code>_C</code>	Suffix	Makes the rule <i>case-sensitive</i>
<code>@V</code>	Suffix	Stems the word that precedes the <code>@V</code> to include <i>verb forms</i> of the word

Expanding Word Forms: CONCEPT Rule Examples

Rule	Example Matches
CONCEPT:go @	I will <i>go</i> . She is <i>going</i> . They <i>went</i> . He is <i>gone</i> .
CONCEPT:break @	It was <i>broken</i> . The camera <i>broke</i> . It kept <i>breaking</i> down. Give me a <i>break!</i>

Part-of-Speech CONCEPT Rule Tags in Grammar Definitions

Part of Speech	Tag	Grammar Definition	Example Matches
Noun	<i>N</i>	SAS : <i>N</i>	SAS <i>employee</i> SAS <i>software</i>
Preposition	<i>Prep</i>	: <i>Prep</i> the drug	<i>of</i> the drug <i>from</i> the drug
Possessive Determinant	<i>PossDet</i>	: <i>PossDet</i> symptoms	<i>my</i> symptoms <i>her</i> symptoms <i>their</i> symptoms

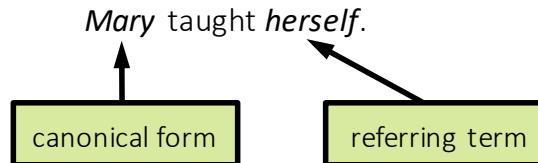
Coreference in a CONCEPT Rule

Coreference is a relationship between two words or phrases in which both refer to the same person or thing.

- For example, in the phrase *Mary taught herself*, both the terms *Mary* and *herself* refer to the same person.
- However, in the phrase *Mary taught her*, the terms *Mary* and *her* refer to different people.

Coreference

- You can specify a *canonical form* and a *referring term* in a CONCEPT definition in a way that is similar to that in a CLASSIFIER definition.
- However, the CONCEPT definition uses the *_ref* operator rather than the *coref = operator*.



Using the _ref Operator in a CONCEPT Rule

The `_ref` operator is specified as follows:

- The *canonical form* is expressed in braces following a `_c` marker.
- The *referring term* is expressed in braces following a `_ref` operator.
- The definition can also include other items that are valid for the given definition type, such as literal text, part-of-speech tags, and references to other concepts.
- The `_c` and `_ref` operators are specified in lowercase.

Using the _ref Operator in a CONCEPT Rule

You can add symbols after the `_ref` operator to match additional occurrences of the referring term.

Symbol	Result
<code>></code>	Match all occurrences of the referring term
<code>_F</code>	Match all occurrences of the referring term that follow the canonical form
<code>_P</code>	Match all occurrences of the referring term that precede the canonical form

Matching with Coreference in a CONCEPT Rule

LITI Definition	Test Result
<code>CONCEPT: _c { doctor } :sep_ref{ she }</code>	When I called my <i>doctor</i> , <i>she</i> said I should come in as soon as possible. At my next appointment, <i>she</i> changed my prescription.
<code>CONCEPT: _c { doctor } :sep_ref{she}></code>	When I called my <i>doctor</i> , <i>she</i> said I should come in as soon as possible. At my next appointment, <i>she</i> changed my prescription.

The text `:sep` is the part-of-speech tag for separator characters, including the period, comma, colon, and semicolon.

Copyright © SAS Institute Inc. All rights reserved.





CONCEPT Rule

This demonstration illustrates the CONCEPT rule functionality in SAS Visual Text Analytics.

1. Use the course logon credentials to start Model Studio.
2. Select the previously created Visual Text Analytics project named **Rules_Demo**.
3. Right-click the **CONCEPT** node and click **Open**.

CONCEPT Rule Demo 1

1. Select the custom concept **CONCEPT_1** and observe the associated LITI rule in the Edit Rules window.

```
CONCEPT:visual _w
CONCEPT:Predictive _cap
CONCEPT:_w analytics
```

The above CLASSIFIER rule identifies documents that contain any defined text string and another preceding or following string when it is used with **_w** (a wild card that matches any word). To match any text string that begins with an uppercase letter, the **_cap** match key is used.

2. To validate this rule, click the **Test Sample Text** tab.
3. In the Test Sample Text input box, enter the following text: **Visual Statistics Predictive analytics predictive models Predictive Modeling**.
4. Click (**Test text**) to test the rule.

The screenshot shows the validation results for the CONCEPT rule. The code entered is:

```
1 CONCEPT:visual _w
2 CONCEPT:Predictive _cap
3 CONCEPT:_w analytics
```

A green checkmark icon indicates that the code is valid. Below the code, there are tabs for "Documents 612" and "Test Sample Text". Under "Test Sample Text", the input text is:

Visual Statistics Predictive analytics predictive models Predictive Modeling.

The words "Visual Statistics", "Predictive analytics", and "Predictive Modeling" are highlighted in blue, while "predictive models" is not highlighted.

The strings *Visual Statistics Predictive analytics* and *Predictive Modeling* are highlighted. The string *predictive models* is not highlighted because the *models* string that follows the *predictive* string does not begin with an uppercase letter.

CONCEPT Rule Demo 2 (based on a previously defined entity)

1. Highlight the predefined custom concept **CONCEPT2_US_AIRPORTS_AND_LOCATIONS** and this associated LITI rule in the Edit Rules window.

CONCEPT : US_AIRPORTS US_AIRPORTS_CITIES

This concept rule is defined based on two previously defined concepts, US_AIRPORTS and US_AIRPORTS_CITIES. The rule definition is presented below.

Custom concept US_AIRPORTS:

**CLASSIFIER: LAX
CLASSIFIER: LAS
CLASSIFIER: SNA
CLASSIFIER: RDU**

Custom concept US_AIRPORTS_CITIES:

**CLASSIFIER: Los Angeles CA
CLASSIFIER: Las Vegas NV
CLASSIFIER: Santa Ana CA
CLASSIFIER: Morrisville NC**

The above defined CONCEPT rule identifies a document that contains US airport codes and the corresponding city and state in the document.

2. To validate this rule, click the **Test Sample Text** tab.
3. In the Test Sample Text input box, enter the following text:
**LAS Las Vegas NV Las Vegas NV LAS Lax Los Angeles CA
SNA John Wayne Airport Santa Ana.**
4. Click (Test text) to test the rule.

The screenshot shows the SAS Edit Rules interface. At the top, there's a table with one row containing the rule definition: "1 CONCEPT:US_AIRPORTS US_AIRPORTS_CITIES". Below the table, a message says "Code is valid." with a green checkmark icon. Underneath, there are tabs for "Documents 612" and "Test Sample Text", with "Test Sample Text" being the active tab. At the bottom, there are icons for copy, paste, delete, and refresh. The "Test Sample Text" input field contains the string "LAS Las Vegas NV Las Vegas NV LAS Lax Los Angeles CA SNA John Wayne Airport". The "Test Sample Text" output area shows the same string with specific words highlighted in blue: "LAS", "Las Vegas NV", "Las Vegas NV", "LAS", "Lax", "Los Angeles CA", and "SNA John Wayne Airport".

The strings *LAS Las Vegas NV* and *Lax Los Angeles CA* are highlighted because these two strings satisfy the two CLASSIFIER-based entity rules, US_AIRPORTS and US_AIRPORTS_CITIES. The string *Las Vegas NV LAS* did not satisfy the correct sequence in CONCEPT2_US_AIRPORTS_AND_LOCATIONS, nor did the string *SNA John Wayne Airport*.

End of Demonstration

4.02 Multiple Choice Question

Which CONCEPT rule has the incorrect syntax to extract the text SAS Visual Text Analytics?

- a. CONCEPT: SAS Visual Text Analytics
- b. CONCEPT: SAS Visual _cap _cap
- c. CONCEPT: "SAS Visual Text Analytics"
- d. CONCEPT: _w Visual Text Analytics

36

Copyright © SAS Institute Inc. All rights reserved.

C_CONCEPT Rule Type

The *C_CONCEPT* rule has all the capabilities of the CONCEPT rule plus the ability to match concepts with a context and detect partial matches.

The *C_CONCEPT* rule returns matches that occur only in the specified context.

Requirements

- The *_c* marker is required.
- Only one *_c* marker can be specified.

C_CONCEPT:<argument> *_c*{*argument*}<*argument*>

where *argument* can be a concept name, a rule modifier, or a string.

39

Copyright © SAS Institute Inc. All rights reserved.

C_CONCEPT Rule Type

Capabilities

- ✓ Match specific words or strings.
- ✓ Use wildcards to match any word.
- ✓ Expand word forms.
- ✓ Reference parts of speech.
- ✓ Reference defined entities.
- ✓ Detect partial match.
- ✓ Specify coreference using the *_ref* operator.
- ✗ Use Boolean operators.
- ✗ Use regular expressions to match patterns.

40

Copyright © SAS Institute Inc. All rights reserved.

Using the *_c* Marker

- The context marker (*_c*) is used to locate related terms within a specific context.
- It identifies the terms to be returned when there is a match.
- The marker is specified in lowercase.
- The text or item to be matched is enclosed in braces.

41

Copyright © SAS Institute Inc. All rights reserved.

Using the _c Marker with C_CONCEPT Rule

LITI Rule	Test Result
C_CONCEPT: New York _c {_cap _cap }	New York <i>Stock Exchange</i> New York <i>Public Library</i> New York <i>Daily News</i> New York theater district
C_CONCEPT:_c { _cap } Sea	<i>Bering</i> Sea <i>North</i> Sea <i>Baltic</i> Sea Bering Straight

42



Copyright © SAS Institute Inc. All rights reserved.

_c Marker

Rule Body	Location Specified	Test Results
Orion _c{Glimmer} C20	Definition of the product camera	Orion <i>Glimmer</i> C20
_c{i love} my camera	Tonal keyword, positive node	<i>I love</i> my camera
Auto Focus is _c{not fast enough}	Product definition, negative node for the focus feature	Auto Focus is <i>not fast enough</i>

43



Copyright © SAS Institute Inc. All rights reserved.

Specifying a Partial Match in a C_CONCEPT Rule

- Use the greater than (>) symbol with the _c marker to specify a partial match.
- Every occurrence of the bracketed term is a match if the entire rule appears at least once in the input text.

LITI Definition	Test Results
C_CONCEPT: New York _c {Daily News }>	The Daily News of New York City is the fourth most widely circulated daily newspaper in the United States with a daily circulation of 605,677, as of November 1, 2011. The Daily News was founded by Joseph Medill Patterson in 1919. It was not connected to an earlier New York Daily News , which had been founded in the 1850s.



C_CONCEPT Rule

This demonstration illustrates the C_CONCEPT rule functionality in SAS Visual Text Analytics.

1. Use the course logon credentials to start Model Studio.
2. Select the previously created Visual Text Analytics project named **Rules_Demo** to open it.
3. Right-click the **CONCEPT** node and select **Open**.

C_CONCEPT Rule Demo 1 (joint occurrence with partial highlight)

1. Select the **C_CONCEPT_1** rule and view the rule syntax. This C_CONCEPT_1 rule is used to highlight a user-defined partial text string (*Visual Text Analytics*) when this specific string appears with another string (SAS). The associated LITI rule is in the Edit Rules window.

```
C_CONCEPT:SAS _c{Visual Text Analytics}
```

The above C_CONCEPT rule identifies a document that contains the text string *SAS Visual Text Analytics*, but it highlights only the partial string *Visual Text Analytics*.

2. To validate this rule, click the **Test Sample Text** tab.
3. In the Test Sample Text input box, enter the following text: **SAS Visual Text Analytics is a new text analytics software from SAS family. Visual Text Analytics now supports 30 languages.**
4. Click  (Test text) to test the rule.

The screenshot shows the SAS Model Studio interface with the 'Edit Rules' window open. The rule 'C_CONCEPT:SAS _c{Visual Text Analytics}' is listed. Below it, a message says 'Code is valid.' A green checkmark icon is present. At the bottom, there's a preview pane showing the text 'SAS Visual Text Analytics is a new text analytics software from SAS family. Visual Text Analytics now supports 30 languages.' with the word 'Visual Text Analytics' highlighted in blue, indicating a partial match.

The first occurrence of *Visual Text Analytics* is highlighted in the first sentence only because *Visual Text Analytics* is following *SAS*. In the second sentence, *Visual Text Analytics* is not highlighted because it does not follow the *SAS* string.

C_CONCEPT Rule Demo 2 (Refer to previously defined entities in the C_CONCEPT rule.)

- Select the **C_CONCEPT_2_SAS_ANALYTICAL_TOOLS** rule and view the rule syntax. This **C_CONCEPT_2_SAS_ANALYTICAL_TOOLS** rule is used to highlight previously defined entities. The associated LITI rule in the Edit Rules window is below.

```
C_CONCEPT:SAS _c{ANALYTICAL_TOOLS}
```

The rules for ANALYTICAL_TOOLS are below.

```
CLASSIFIER:Forecast Studio
CLASSIFIER:Model Manager
```

The above **C_CONCEPT_2_SAS_ANALYTICAL_TOOLS** rule identifies documents that contain the text strings that were previously defined in the ANALYTICAL_TOOLS entity.

- To validate this rule, click the **Test Sample Text** tab.
- In the Test Sample Text input box, enter the following text: **SAS Forecast Studio generates automatic forecasts using ARIMA and ESM models. SAS Model Manager can register only Enterprise miner models.**
- Click (Test text) to test the rule.

1 C_CONCEPT:SAS _c{ANALYTICAL_TOOLS}

Code is valid.

Documents 612 Test Sample Text

SAS Forecast Studio generates automatic forecasts using ARIMA and ESM models. SAS Model Manager can register only Enterprise miner models.

The previously defined entities *Forecast Studio* and *Model Manager* are highlighted in the test document.

C_CONCEPT Rule Demo 3 (coreferencing in the C_CONCEPT rule with a partial match)

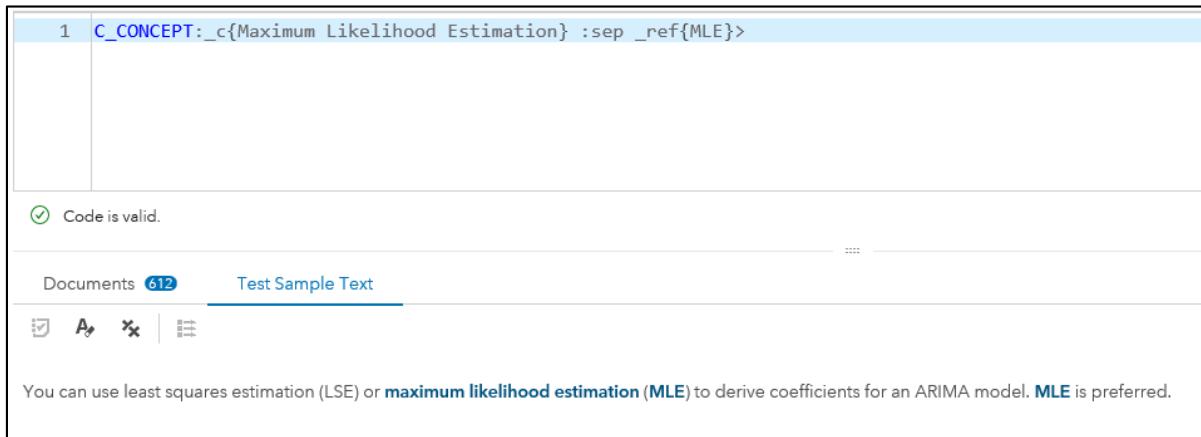
- Select the **C_CONCEPT_3_COREF** rule and view the rule syntax. This **C_CONCEPT_3_COREF** rule is used to coreference a second string (*MLE*) if both strings (*Maximum Likelihood Estimation* and *MLE*) appear in the document at least once. The rule subsequently partially highlights all occurrences of the reference string (*MLE*). The associated LITI rule is in the Edit Rules window.

```
C_CONCEPT:_c{Maximum Likelihood Estimation} :sep _ref{MLE}>
```

- `_c{Maximum Likelihood Estimation}` defines the primary string.
- `:sep` indicates the presence of a separator string (, . : ;).
- `_ref{MLE}` defines the secondary reference string.

- Use the greater than (>) symbol with the `_c` marker to specify a subsequent partial match.
- To validate this rule, click the **Test Sample Text** tab.

4. In the Test Sample Text input box, enter the following text: **You can use least squares estimation (LSE) or maximum likelihood estimation (MLE) to derive coefficients for an ARIMA model. MLE is preferred.**
5. Click  (Test text) to test the rule.



The screenshot shows the SAS Visual Text Analytics interface. At the top, there is a code editor window with the following content:

```
1 C_CONCEPT:_c{Maximum Likelihood Estimation} :sep _ref{MLE}>
```

Below the code editor, a message says "Code is valid." with a green checkmark icon. To the right of the message are three horizontal dots (...).

At the bottom of the code editor, there are navigation tabs: "Documents 612" and "Test Sample Text". The "Test Sample Text" tab is currently selected, indicated by an underline.

Under the tabs, there is a toolbar with icons for search, edit, and other functions. Below the toolbar, the text "You can use least squares estimation (LSE) or maximum likelihood estimation (MLE) to derive coefficients for an ARIMA model. MLE is preferred." is displayed. The word "MLE" is highlighted in blue, matching the color of the text in the code editor.

Maximum Likelihood Estimation, MLE appears in the test document once. Therefore, a subsequent occurrence of *MLE* in the document is also highlighted.

End of Demonstration

4.03 Multiple Choice Question

The concept rule C_CONCEPT: SAS _c{enterprise miner} is validated using the following test text: SAS Enterprise Miner is a solution to create accurate predictive and descriptive models on large volumes of data across different sources in the organization. Identify the text string that should be highlighted when you validate the rule.

- a. SAS Enterprise Miner
- b. Enterprise Miner
- c. SAS
- d. none of the above

CONCEPT_RULE Rule Type

The CONCEPT_RULE rule type has all the features of a C_CONCEPT rule, plus the ability to use Boolean operators to determine context matches.

Requirements:

- The _c marker is required.
- The _c{} can surround only one argument (unless it is within an OR operator).
- At least one Boolean operator is required.

CONCEPT_RULE :(<Boolean-rule-1>...<Boolean-rule-n>

where Boolean-rule can be nested n times and is written as

Boolean-operator "_c{ argument-1 }", <"argument-2">...<"argument-n">).

CONCEPT_RULE Rule Type

Capabilities

- ✓ Match specific words or strings.
- ✓ Use wildcards to match any word.
- ✓ Expand word forms.
- ✓ Reference parts of speech.
- ✓ Reference defined entities.
- ✓ *Use Boolean operators.*
- ✗ Use regular expressions to match patterns.

50

Copyright © SAS Institute Inc. All rights reserved.



CONCEPT_RULE Definition

The *BirthYear* concept uses a CONCEPT_RULE definition that references the *YearCpt* concept.

Concept Name	LITI Definition
<i>YearCpt</i>	REGEX: \d\d\d\d
<i>BirthYear</i>	CONCEPT_RULE: (SENT, "born", "_c{YearCpt}")

Test Results

Charles Dickens was *born* in a modest home in Portsmouth, England, in **1812**. J.K. Rowling, *born 1965*, is best known as the author of the Harry Potter series.

51

Copyright © SAS Institute Inc. All rights reserved.





CONCEPT_RULE Rule

This demonstration illustrates the CONCEPT_RULE rule functionality in SAS Visual Text Analytics.

1. Use the course logon credentials to start Model Studio.
2. Select the previously created Visual Text Analytics project named **Rules_Demo** and open it.
3. Right-click the **CONCEPT** node and select **Open**.

CONCEPT_RULE Demo 1 (A Boolean operator highlights either string1 or string2.)

1. Select the **CONCEPT_RULE_1** rule and view the rule syntax. Using a Boolean operator, this CONCEPT_RULE_1 rule is used to highlight either string1 (*Maximum Likelihood Estimation*) or string2 (*MLE*). The associated LITI rule is in the Edit Rules window:

```
CONCEPT RULE: (OR, " _c{MLE}" , " _c{Maximum Likelihood Estimation} ")
```

2. To validate this rule, click the **Test Sample Text** tab.
3. In the Test Sample Text input box, enter the following text: **MLE and LSE dominate the techniques used for fitting models to data.**
4. Click (**Test text**) to test the rule.

The string *MLE* is highlighted in the test string.

The screenshot shows the SAS Model Studio interface. At the top, there is a code editor window containing the rule definition:

```
1 CONCEPT RULE: (OR, " _c{MLE}" , " _c{Maximum Likelihood Estimation} ")
```

Below the code editor, a message indicates that the code is valid:

Code is valid.

The interface includes tabs for **Documents** (612) and **Test Sample Text**, with the latter being active. Below the tabs are several icons: a checkmark, a font style icon, a close icon, and a list icon. The test sample text input field contains the sentence: **MLE and LSE dominate the techniques used for fitting models to data.** The word **MLE** is highlighted in blue, indicating it was matched by the rule.

CONCEPT_RULE Demo 2 (conditionally selecting documents with selective text strings using the following Boolean operators: AND, OR, NOT)

1. Select the **CONCEPT_RULE_2** rule and view the rule syntax. This rule is used to highlight several strings (*concepts*, *topics*, or *terms*) if there is no term *prediction*. The rule uses Boolean operators (AND, OR, or NOT). The associated LITI rule is in the Edit Rules window.

```
CONCEPT_RULE: (AND, (OR, "_c{concepts}", "_c{topics}", "_c{terms}"), (NOT, "Prediction"))
```

2. To validate this rule, click the **Test Sample Text** tab.
3. In the Test Sample Text input box, enter the following text: **Visual Text Analytics uses concepts, terms, and topics in categorizing documents. It is not using any statistical predictions.**
4. Click **(Test text)** to test the rule.

The strings *concepts* *topics* and *terms* are highlighted in the test string.

The screenshot shows the SAS Visual Text Analytics interface. At the top, there is a code editor window containing the following text:

```
1 CONCEPT_RULE: (AND, (OR, "_c{concepts}", "_c{topics}", "_c{terms}"), (NOT, "Prediction"))
```

Below the code editor, a message indicates that the code is valid:

Code is valid.

At the bottom of the interface, there is a navigation bar with tabs: **Documents 612** and **Test Sample Text**. The **Test Sample Text** tab is currently selected. Below the navigation bar, there are some icons and a status message:

Visual Text Analytics uses **concepts**, **terms**, and **topics** in categorizing documents. It is not using any statistical predictions.

End of Demonstration

4.04 Multiple Choice Question

Which CONCEPT_RULE body has the correct syntax?

- a. Orion _cap _c{is wonderful}
- b. (AND, Orion _cap, _c{is wonderful})
- c. (SENT, "OrionGlimmerprice", "_c{affordable}")
- d. (SENT, "OrionGlimmerprice", "affordable")

NO_BREAK Rule Type

The *NO_BREAK* rule type is used to disambiguate similar entries. It accomplishes this by avoiding partial matches.

Requirements:

- The *_c* marker is required.
- The *_c{}* can surround only one argument.
- The argument can be a concept or a string.

NO_BREAK Rule Type

Capabilities

- ✓ Avoids partial matches.

Special considerations

1. NO_BREAK applies across the entire taxonomy regardless of where the rule appears or whether the rule is enabled or disabled.
2. 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.

57

Copyright © SAS Institute Inc. All rights reserved.



NO_BREAK Definition

The *OTHER_LIKE* concept uses a NO_BREAK definition that references the *MAX_LIKE* concept.

Concept Name	LITI Definition
<i>MAX_LIKE</i>	CLASSIFIER: maximum <i>likelihood</i>
<i>OTHER_LIKE</i>	CLASSIFIER: likelihood NO_BREAK: c{ MAX_LIKE }

Test Results

Ordinary gamma models are fit using maximum likelihood estimation.
Survival models obtain estimates using a partial *likelihood* formulation.

58

Copyright © SAS Institute Inc. All rights reserved.





NO_BREAK Rule

This demonstration illustrates the NO_BREAK rule functionality in SAS Visual Text Analytics.

1. Use the course logon credentials to start Model Studio.
2. Select the previously created Visual Text Analytics project named **Rules_Demo** and open it.
3. Right-click the **CONCEPT** node and select **Open**.
4. Select concept rule **NO_BREAK_1** and view the syntax.

```
CLASSIFIER:maximum likelihood
```

5. To validate this rule, click the **Test Sample Text** tab.
6. In the Test Sample Text input box, enter the following text: **Ordinary gamma models are fit using maximum likelihood estimation. Survival models obtain estimates using a partial likelihood formulation.**
7. Click (**Test text**) to test the rule.
The string *maximum likelihood* is highlighted.
8. Select concept rule **NO_BREAK_2** and view the syntax.

```
CLASSIFIER:likelihood  
NO_BREAK:_c{NO_BREAK_1}
```

9. To validate this rule, click the **Test Sample Text** tab.
10. In the Test Sample Text input box, enter the following text: **Ordinary gamma models are fit using maximum likelihood estimation. Survival models obtain estimates using a partial likelihood formulation.**

11. Click  (Test text) to test the rule.

The string *likelihood* is highlighted after the word *partial*, but not after the word *maximum*.

Edit a Concept

1	CLASSIFIER:likelihood
2	NO_BREAK:_c{NO_BREAK_1}

 ... ▲ :::

Documents Test Sample Text

   | 

Ordinary gamma models are fit using maximum likelihood estimation.
Survival models obtain estimates using a partial **likelihood** formulation.

End of Demonstration

Concepts versus Facts

Facts (also called *predicates*) are related pieces of information in text. They are located and matched as phrases.

- Facts can be identified within a custom concept. For example, George Washington University can be considered as a fact in the following sentence: *There are countless active student organizations at George Washington University.*
- You can use special types of concept rules to locate facts, that is, predicate rules and sequence rules. A predicate rule (*PREDICATE_RULE*) uses Boolean operators to help locate facts.
- For example, you could use Boolean rules to specify terms that you wanted to occur within a certain number of terms.

PREDICATE_RULE Rule Type

This rule type helps you define facts that you want identified in text.

Capabilities

- ✓ Match specific words or strings.
- ✓ Use wildcards to match any word.
- ✓ Expand word forms.
- ✓ Reference parts of speech.
- ✓ Reference defined entities.
- ✗ Specify a partial match.
- ✓ Use Boolean operators.
- ✗ Use regular expressions to match patterns.

Note: Although you can create and test predicate rules in SAS Visual Text Analytics, they are not applied to the project's documents when the project is run. As a result, you do not see fact matches within document views, topics, and auto-generated rules. To obtain fact rule matches, you can use the project's concept score code feature.

The fact matched by a predicate rule includes more than just the items used in the rule, and the complete match is not added to the term table. Consequently, although you see the text that is matched by the predicate rule, the matched text does not affect subsequent nodes. The matched text does not affect topics derived by the Topic node. The matched text does not appear in linguistic rules derived by the Category node.

PREDICATE_RULE Rule Type: Markers

- Context markers are used to locate related terms within a specific context.
- The C_CONCEPT and CONCEPT_RULE types require the use of a single context marker, `_c`.
- The `_c` marker is not allowed in the PREDICATE_RULE type.
- Multiple markers can be specified for the PREDICATE_RULE type. This enables you to define arguments that locate relationships between two or more concepts.
- When multiple markers are included in the rule, the match is highlighted from the first marker to the last marker.

62



Copyright © SAS Institute Inc. All rights reserved.

PREDICATE_RULE Definition Syntax

```
PREDICATE_RULE: (arg1<,arg2,arg3,...>):
(Boolean_Op, "_arg1{match_key}",
<,"_arg2{match_key}", "_arg3{match_key}"...>)
```

- At least one Boolean operator is required.
- At least one argument, such as `arg1`, must be specified.
- Arguments must be named using lowercase letters and digits, and begin with a lowercase letter.
- The `_c` marker is not allowed.
- The `match_key` can include Boolean operators, literals, part-of-speech tags, wildcards, and references to other concepts.

63



Copyright © SAS Institute Inc. All rights reserved.

PREDICATE_RULE Definition

The WorldLeaders concept uses a PREDICATE_RULE definition that references the Titles and Countries concepts.

Concept Name	LTI Definition
CountryNames	CLASSIFIER: France CLASSIFIER: French CLASSIFIER: India CLASSIFIER: Indian CLASSIFIER: Canada ...
OfficialTitles	CLASSIFIER: Prime Minister CLASSIFIER: President
WorldLeaders	PREDICATE_RULE: (name,title,co): (SENT, " _name{_cap_cap}" , "_title{OfficialTitles}" , " _co{CountryNames}")

64

Copyright © SAS Institute Inc. All rights reserved.



PREDICATE_RULE Rule Type

Rule	Example Matches
PREDICATE_RULE : (aa,bb): (SENT, " _aa{good}" , " _bb{quality}")	The <i>quality is good</i> . It's very <i>good quality</i> .
PREDICATE_RULE : (one,two): (AND, " _one{value}" , " _two{price}")	It's a good <i>value for the price</i> . Couldn't believe the <i>price</i> . <i>What a value!</i>
PREDICATE_RULE : (xx,yy): (DIST_3, " _xx{not}" , " _yy{disappointed}")	You will <i>not be disappointed</i> . I was <i>not at all disappointed</i> .

65

Copyright © SAS Institute Inc. All rights reserved.





PREDICATE_RULE Rule

This demonstration illustrates the PREDICATE_RULE rule functionality in SAS Visual Text Analytics.

1. Use the course logon credentials to start Model Studio.
2. Select the previously created Visual Text Analytics project named **Rules_Demo**.
3. Right-click the **CONCEPT** node and select **Open**.

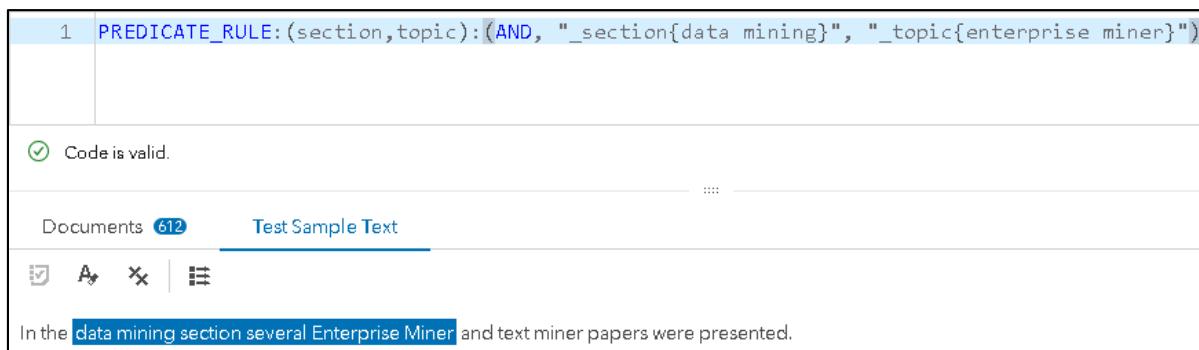
PREDICATE_RULE Demo 1 (Strings are highlighted when a certain factual association exists in the document.)

1. Select **PREDICATE_RULE_1** and view the rule syntax. This PREDICATE_RULE_1 rule is used to highlight related strings (that is, *data mining* and *enterprise miner*) that are present in the same document. The associated LITI rule is in the Edit Rules window.

```
PREDICATE_RULE:(section,topic):(AND, "_section{data mining}", "_topic{enterprise miner}")
```

2. To validate this rule, click the **Test Sample Text** tab.
3. In the Test Sample Text input box, enter the following text: **In the data mining section several Enterprise Miner and text miner papers were presented.**
4. Click  (**Test text**) to test the rule.

The text strings between the first string, *data mining*, and the last string, *Enterprise Miner*, are highlighted in the test string.



The screenshot shows the SAS Model Studio interface with the 'Test Sample Text' tab selected. The code entered is:

```
1 PREDICATE_RULE:(section,topic):(AND, "_section{data mining}", "_topic{enterprise miner}")
```

A message indicates: **Code is valid.**

The test sample text is: **In the data mining section several Enterprise Miner and text miner papers were presented.**

The text 'data mining' and 'Enterprise Miner' are highlighted in blue, demonstrating the rule's functionality.

End of Demonstration

4.05 Multiple Choice Question

Which PREDICATE_RULE body has the correct syntax?

- a. (aa,bb): (DIST_5, '_aa{cameraprice}', '_bb{expensive}')
- b. (yy,zz): (DIST_5, _yy{"cameraprice"}, _zz{"expensive"})
- c. (xx,yy): (DIST_5, "_xx{cameraprice}", "_yy{expensive}")
- d. (aa,bb): (DIST_5,"_aa{cameraprice}","_bb{expensive}")

Regular Expressions

- **REGEX** Identifies recurring patterns of information in text that can be expressed in numbers and characters, such as telephone numbers, license plate numbers (for example, ABX-0444), part numbers for manufacturing components (for example, TMS1T3B1M5R-23), hyphenated words (for example, fifty-nine), and so on.
- Use regular expressions when you want to locate matches on text that has recognizable patterns.

Examples:

- email addresses
- street addresses

A regular expression to match an email address is shown below.

```
[_a-zA-Z\d\-.]+\@([_a-zA-Z\d\-.]+\(\.\[_a-zA-Z\d\-\]+\)+)
```

The history of regular expressions began in 1956.

Regular Expressions: A Brief History

- In 1956, mathematician Stephen Kleene published *Representation of events in nerve nets and finite automata*, which introduced the term “regular expressions.”
- In the 1970s, Ken Thompson, Dennis Ritchie, and other Bell Labs researchers created UNIX and the C language and thus incorporated regular expressions in editors and shell scripts.
- In 1987, Larry Wall released the first version of Perl, with regular expression support. This originated the phrase “Perl regular expressions.”
- In or near the year 2000, SAS introduced Perl regular expression PRX functions and CALL routines in SAS 9.

71



The SAS DATA step has these five PRX functions:

```
PRXCHANGE
PRXMATCH
PRXPAREN
PRXPARSE
PRXPOSN
```

The six PRX CALL routines are as follows:

```
CALL PRXCHANGE
CALL PRXDEBUG
CALL PRXFREE
CALL PRXNEXT
CALL PRXPOSN
CALL PRXSUBSTR
```

If you need to do SAS DATA step coding for document processing, you can use Visual Text Analytics and the REGEX concept type to test your regular expressions.

REGEX Rule Type

Capabilities

- ✓ *Match specific words or strings.*
- ✗ Use wildcards to match any word.
- ✗ Expand word forms.
- ✗ Reference parts of speech.
- ✗ Reference defined entities.
- ✗ Specify what portion of the rule match is highlighted in the Test window.
- ✗ Specify a partial match.
- ✗ Use Boolean operators.
- ✓ *Use regular expressions to match patterns.*

72

Copyright © SAS Institute Inc. All rights reserved.

REGEX Rule Type

- Rule elements are grouped using square brackets.
- Meta characters (for example, [,], (), ?, *, +, ., \, |, and \$, must be preceded by a backslash to indicate that they have a literal meaning.
 - For example, [\?] matches a question mark (?) in text.
- You cannot refer to concepts in a REGEX expression.
- To match one or more consecutive digits without decimals, such as matching a number with decimal notation (for example, 392.55, 45.25, and 0,987654321), consider the following information:
 - Use the rule *REGEX:[0-9]+[0-9\,\.][0-9]+*.
 - This rule matches any digit 0 to 9, followed by either a digit, a comma, or a period, and then followed by any number of digits.

73

Copyright © SAS Institute Inc. All rights reserved.

REGEX Rule Type

Expression	Matches
[abc]	Any single character (a, b, or c) in uppercase or lowercase
[a-z]	Any single character between a and z, inclusive, in uppercase or lowercase
[0-9]	Any single digit
[0-9]+	Any integer
[0-9]+\.[0-9]+	Any non-integer
<code>\s</code>	Whitespace character (same as <code>[\t\n\r\f]</code>)
<code>\w+</code>	Word character (same as <code>[a-zA-Z_0-9]</code>)

74

Copyright © SAS Institute Inc. All rights reserved.



Regular Expression Matching

- When regular expressions are used in a classifier definition, expression matching is performed before the document is divided into individual terms.
- Therefore, a match can be embedded inside a term.

Definition	Matches	Non-Matches
REGEX: \d+mg	I took <i>5mg</i> of Abidal.	I took <i>5mg</i> of Abidal. I took 5mgof Abidal. I took5mgof Abidal.

75

Copyright © SAS Institute Inc. All rights reserved.



continued...

Regular Expression Guidelines

- The rule **REGEX: [a-f]** matches lowercase characters between **a** and **f**, inclusively.
- To add uppercase characters, use the rule **REGEX:[A-Fa-f]**.
- To specify characters that should not be matched (negated characters), insert a caret (^) before a set of characters.
 - For example, **REGEX:[^aeiou]** matches all characters, numbers, and symbols in text, except *a*, *e*, *i*, *o*, and *u*.
- Numbers are matched as-is within a string when they are represented **without** square brackets.
 - For example, **REGEX:0125\-[A-Za-z]** matches part of any numbers that begin with *0125-* and end with a *letter*.
- Numbers are matched by specifying ranges that are enclosed in square brackets (**[]**).
 - For example, **REGEX:[0-9]** produces a match on a number between 0 and 9.



Regular Expression Guidelines

- Include one or more characters inside square brackets (**[]**) to match the specified characters. This provides flexibility in character matching.
 - For example, the rule **REGEX:[crash]** matches *c*, *r*, *a*, *s*, or *h*.
 - If you add a plus sign (+), the rule becomes **REGEX:[crash]+** and it matches the characters that are specified in any combination, such as *rash*, *cash*, *ash*, and *crass* (but not *crashpad* **or** *crashdummy*).
- Characters are matched within a string in sequence when they are represented without square brackets (**[]**). Example: **REGEX: any** would match only the word *any*. (*Anyone* or *anything* would not be matched.)
 - To match words that contain *any*, you can modify the rule to use asterisks (*) to match other character occurrences (or none) that surround *any*.
 - For example, the rule **REGEX: [A-Za-z]*any[A-Za-z]*** would match *any*, *anyone*, *anything*, and *Many*.



REGEX Rule Type Examples

Rule Body	Example Matches	Example Nonmatches
Model [A-C]	<i>Model A</i> <i>model b</i>	Model ABC
[A-C][0-9]+	<i>A1</i> <i>b55</i> <i>C200</i>	a 1 BC20
Orion C[0-9][0-9]	<i>Orion C20</i> <i>orion C45</i>	Orion C200 orion c 20
\\$[0-9]+	<i>\$500</i> <i>\$2</i>	\$49.95 \$ 50
\\$[0-9]+\.[0-9]+	<i>\$49.95</i> <i>\$50.4545</i>	\$500 \$ 2.95

Copyright © SAS Institute Inc. All rights reserved.



Regular Expression Examples

Expression	Example Matches	Example Nonmatches
\d+mg	<i>5mg</i> <i>10MG</i>	5 MG 10 mg
\d+\s+mg	<i>5 Mg</i> <i>10 mg</i>	5mg 10MG
\\$\d+	<i>\$500</i> <i>\$2</i>	\$49.95 \$ 50

These examples assume that *case-insensitive* matching is used for the concept.

79

Copyright © SAS Institute Inc. All rights reserved.



Regular Expression Examples

Expression	Example Matches	Example Nonmatches
\d	5 7	53 29
\\$\d+	\$500 \$2	\$49.95 \$ 50
\\$\d+\.\d+	\$49.95 \$50.4545	\$500 \$ 2.95
\\$\s\d+	\$ 2 \$ 50	\$ 2.95 \$500

Regular Expression Examples

Expression	Example Matches	Example Nonmatches
\d+mg	5mg 10MG	5 MG 10 mg
\d+\s+mg	5 Mg 10 mg	5mg 10MG
[\w\.\-]+@[\\w\.\-]+\.\com	John.Smith@corp.com johnsmith@freemail.com	johns@univ.edu

These examples assume that *case-insensitive* matching is used for the concept.



REGEX Rule

This demonstration illustrates the REGEX rule functionality in SAS Visual Text Analytics.

1. Use the course logon credentials to start Model Studio.
2. Select the previously created Visual Text Analytics project named **Rules_Demo** and open it.
3. Right-click the **CONCEPT** node and select **Open**.

REGEX Rule Demo 1 (identifying recurring patterns of information in text)

1. Select the **REGEX_1** rule and view the rule syntax. This REGEX_1 rule is used to highlight recurring patterns of information in text. The patterns can be expressed in numbers and characters.

The associated LITI rule is in the Edit Rules window.

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

2. To validate this rule, click the **Test Sample Text** tab.
3. In the Test Sample Text input box, enter the following text: **Visual Text Analytics 8.3 uses concepts, terms, and topics in categorizing documents. It is not using any statistical predictions.**
4. Click  (Test text) to test the rule.

The numerical string 8.3 is highlighted in the test string.

The screenshot shows the SAS Visual Text Analytics Model Studio interface. In the center, there is a large text area containing the following text:

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

Below the text area, a message says "Code is valid." with a green checkmark icon.

At the bottom of the interface, there is a navigation bar with the following items:

- Documents 612
- Test Sample Text** (which is currently selected, indicated by a blue underline)
- Checkmark icon
- Ay icon
- X icon
- More options icon

At the very bottom of the interface, there is a footer message: "Visual Text Analytics 8.3 uses concepts, terms, and topics in categorizing documents. It is not using any statistical predictions."

End of Demonstration

4.06 Multiple Choice Question

Which REGEX rule has the correct syntax to extract a part number that starts with XYZ- and any five-digit numbers (for example, XYZ-12345)?

- a. REGEX:[XYZ][\ -][1234567890]
- b. REGEX:XYZ[\ -][1234567890]+
- c. REGEX:[XYZ][\ -][\d\d\d\d\d]
- d. REGEX:XYZ[\ -][\d\d\d\d\d]

The two concept rule types REMOVE_ITEM and SEQUENCE are not covered in this course. See the user guide for details about using these concept types.

4.3 SAS Visual Text Analytics Demo

Category Rules

Introduction to Category Rules

- Category rules are written as Boolean rules because they resolve to true or false. A value of *True* results in a match.
- Boolean rules use operators, arguments, and modifiers to define the conditions that are necessary for category matches.
- Boolean rules are more precise than linguistic rules for determining category membership.

87

Copyright © SAS Institute Inc. All rights reserved.

Boolean Rules

Primary advantage: You can use Boolean operators to write more precise and unambiguous rules.

- Boolean rules can also be used when a term is located for structured input documents such as XML in order to return a match.
- As with linguistic rules, you can add qualifiers to modify the effect of the rule.

88

Copyright © SAS Institute Inc. All rights reserved.

Category rules are written as Boolean rules because they resolve to *true* or *false*. The *true* value results in a match. Boolean rules use operators, arguments, and modifiers to define the conditions that are necessary for category matches. Boolean rules are more precise than linguistic rules for determining category membership.

Boolean Rule Syntax

- Boolean operator names are specified in uppercase.
- Boolean operators and their arguments are enclosed in parentheses.
- Arguments are separated by commas.
- Terms are enclosed in double quotation marks.

Example, when displayed in text view:

```
(AND , "Amazon" , (OR , "online retailer" , "e-commerce")) )
```

This rule is true when the term *Amazon* appears with either of the terms *online retailer* or *e-commerce*.

Boolean Rule Qualifiers

You can add special symbols to terms that are specified in Boolean rules to do the following tasks:

- expand word forms
- specify a wildcard
- perform case-sensitive matching

Expanding Word Forms in the CATEGORY Rule

Symbol	Position	Result
@	Suffix	Stems the word that precedes the @ to include <i>all forms</i> of the word.
@N	Suffix	Stems the word that precedes the @N to include <i>no noun forms</i> of the word.
_C	Suffix	Makes the rule case-sensitive.
@V	Suffix	Stems the word that precedes the @V to include <i>verb forms</i> of the word.
*		<p>Wildcard matching – Matches any characters that occur at the beginning or end of the word. For example, the argument "travel*" returns the matches <i>travels, traveled, traveler, and traveling</i>. The argument "*room" matches <i>bedroom, cloakroom, ballroom, room</i>, and so on.</p>

continued...

CATEGORY Boolean Rule Syntax

- (*OPERATOR, <argument1>, <argument2>, ...*)
 - where arguments can be terms, strings, or nested rules.
 - General rules for syntax:
 - Boolean operators are enclosed in parentheses and separated with commas.
 - Strings within arguments are included in quotation marks ("").
Example: (*AND, "holiday", "vacation*)
 - Rules can be nested.
Example: (*AND, (OR, "courage", "courageous"), (OR, "brave", "bravery")*)
 - * 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.

Copyright © SAS Institute Inc. All rights reserved.

CATEGORY Boolean Rule Syntax

- Reference a category from another category by using special syntax called *tmac* syntax (*_tmac*).
- Concept names can be referenced in category rules. If you reference a concept name, all concept matches also match in the category.
- Concept names must be enclosed in braces ([]) and quotation marks ("").
 - An example of this is referencing the concept **GAME_SHOWS** in a category rule where you could write the rule **(OR, "[GAME_SHOWS]")**.

Note: Concepts that are named in categories might return more matches than concepts that are run outside of categories.

- In categories, matches on concepts are based on an **all matches** method, which returns all matches found in the text.
- By contrast, in concepts, matches are based on a **best match** method. The best match method detects when text that matches one concept overlaps with text that matches

continued...

Boolean Category Rule Examples

- Subcategory: To extract documents containing the term "equation(s)":
 - (AND, (OR, "equation", "equations"))**
- Subcategory: To extract documents containing the term "measure and relationship":
 - AND, (OR, "measuring", "measures", "measured", "measure"), (OR, "relationship", "relationships")**
- Subcategory: To extract documents containing *Business Intelligence or Enterprise Miner* ~ (*Enterprise Guide, DDE, Drug*):
 - (AND, (NOT, (OR, "Enterprise Guide", "DDE", "Drug")), (OR, "Business Intelligence", "Enterprise Miner"))**
- To specify the previously defined concept rule *Business Intelligence* (*BUSINESS_INTELLIGENCE*):
 - (OR, "[BUSINESS_INTELLIGENCE]")**

Boolean Category Rule Examples

- Referencing a category enables you to use the rules in an existing category without needing to duplicate the rules.
 - Use TMAC syntax (`_tmac`) to reference an existing category in a category rule.
 - The definition of the existing rule is processed in the category that references it.
- To reference a category, you must identify its path. All category paths begin with `@Top/`, where `Top` is the name of the parent (top) category.
- From there, you can specify the path by following the category hierarchy.

95

Copyright © SAS Institute Inc. All rights reserved.



Boolean Category Rule Examples: Including a Previously Defined Category Rule

- Suppose you have the following category structure under *All Categories*:
`NAME / FIRST LAST`
- You reference the category `FIRST` as `@NAME/FIRST`.
- The `tmac` syntax can be used with Boolean operators to define a previously defined category rule.
- For example, suppose you want to reference the category `FIRST` to a category called `FIRST_NAME`. You could add this rule in the `FIRST_NAME` definition:
 - `(OR, _tmac : "@NAME/FIRST")`
- To enforce a first name followed by a last name (`FIRSTLAST`), you could add this rule in a category called `COMPLETE_NAME`:
 - `ORD, _tmac : "@NAME/FIRST", _tmac : "@NAME/LAST")`
 - The definitions written in `FIRST` and `LAST` are automatically processed.

96

Copyright © SAS Institute Inc. All rights reserved.



Boolean Category Rule Examples: Including a Previously Defined Category Rule

The screenshot shows the SAS Model Studio interface with the 'Categories' node selected. In the 'Edit a Category' dialog, the rule for 'COMPLETE_NAME' is defined as `(ORD, _tmac: "@NAME/FIRST", _tmac: "@NAME/LAST")`. The 'Textual Elements' pane displays a sample text: "Peter Christie and George Fernandez did all of the work, and Terry Woodfield took all of the credit." with matches highlighted. The SAS logo is visible in the bottom right corner.

The RulesAndPrompts.txt file shows the syntax for the above category definitions.

Category Name: NAME

Sub-category Name: FIRST

```
(OR,"Terry","Peter","George")
```

Sub-category Name: LAST

```
(OR,"Woodfield","Christie","Fernandez")
```

Category Name: FIRST_NAME

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

Category Name: COMPLETE_NAME

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



CATEGORY Rule

This demonstration illustrates the CATEGORY rule functionality in SAS Visual Text Analytics. Category rules are used to identify specific movies. The **movies_plus** data set contains seven variables.

Columns	
<input checked="" type="checkbox"/>	Select all
<input checked="" type="checkbox"/>	Made_Money
<input checked="" type="checkbox"/>	id
<input checked="" type="checkbox"/>	title
<input checked="" type="checkbox"/>	overview
<input checked="" type="checkbox"/>	budget
<input checked="" type="checkbox"/>	revenue
<input checked="" type="checkbox"/>	release_date

The text variable is **overview**. The 10 rows from the data set appear below.

Total rows: 2125 Total columns: 7			
Made_Mo...	id	title	overview
1 No	49047	Gravity	What goes up must come down.
2 No	975	Viya Las Vegas	Elvis Presley plays a data scientist who uses SAS and Hadoop to beat
3 Yes	548	The Shawshank Reception	Banker Andy Dufresne hosts a wine and cheese party in prison.
4 No	96721	12 Angry Men	A football team is irate when they get flagged for having one too m
5 Yes	1955	The Elephant Man	A Victorian surgeon rescues a heavily disfigured man being mistrea
6 Yes	238	The Godfather	The story spans the years from 1945 to 1955 and chronicles the ficti
7 Yes	3082	Modern Times	The Tramp struggles to live in modern industrial society with the he
8 No	142061	Batman: The Dark Knight Returns, Part 2	Batman has stopped the reign of terror that The Mutants had cast u
9 Yes	14784	The Fall	In a hospital on the outskirts of 1920s Los Angeles, an injured stunt
10 Yes	240	The Godfather: Part II	The continuing saga of the Corleone crime family tells the story of a

1. Use the course logon credentials to start Model Studio.
2. Select the previously created Visual Text Analytics project named **Movies** and open it.
3. Select **Pipelines** to display the nodes in the pipeline.
4. Right-click the **CATEGORIES** node and select **Open**.

AND / OR Rule Demo

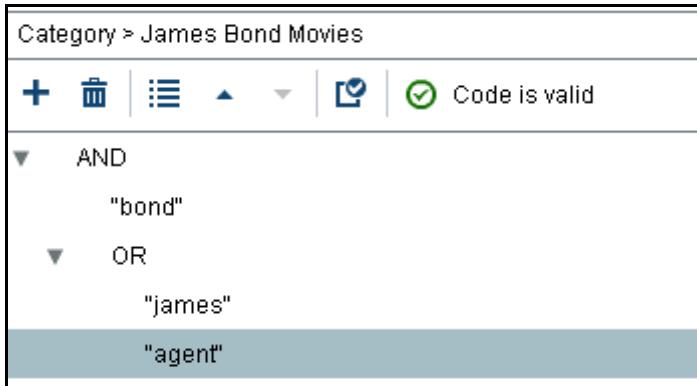
This rule retrieves the documents that satisfy the Boolean true or false condition.

1. Select **All Categories** and then click (**New category**).
2. Enter **James Bond Movies** to name the new category

3. Enter the following code in the Edit a Category window:

```
(AND, "bond", (OR, "james", "agent"))
```

4. Click  to determine whether the code is valid.

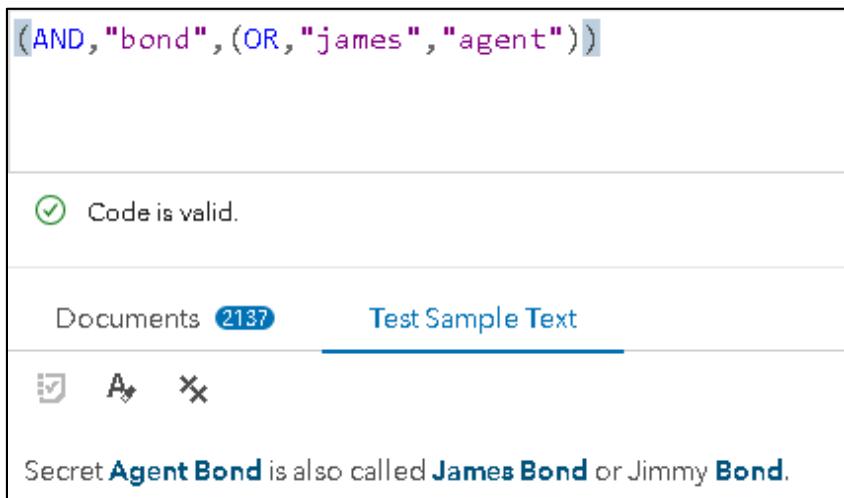


The screenshot shows the 'Edit a Category' interface. At the top, there's a toolbar with icons for adding (+), deleting (-), sorting (grid, list, up, down), and a checkmark. To the right of the toolbar, it says 'Code is valid'. Below the toolbar, the category structure is displayed as a tree:

- AND**
 - "bond"
- OR**
 - "james"
 - "agent"

A green check mark icon  followed by the words **Code is valid** signifies success.

5. To test the code, click the **Test Sample Text** tab and enter **Secret Agent Bond is also called James Bond or Jimmy Bond** in the text box.
6. Click  (**Test text**) to test the rule.



The screenshot shows the 'Test Sample Text' interface. At the top, there's a text input field containing the code:


```
(AND, "bond", (OR, "james", "agent"))
```

 Below the input field, a message says  **Code is valid.**
 At the bottom, there are tabs for 'Documents 2137' and 'Test Sample Text', with 'Test Sample Text' being active. Below the tabs are three icons: a checkmark, a font style icon, and a close/cross icon. The main area displays the sample text:

Secret **Agent Bond** is also called **James Bond** or **Jimmy Bond**.

The matched strings indicate that the Boolean rule is validated.

7. Click  (Tree View) to view the Boolean rule in the Tree view. To revert to the Rule view, click  (Rule View).

```
(AND, "bond", (OR, "james", "agent"))
```

8. Close the Categories window and run the Categories node in the pipeline.

Relevancy	Sentiment
8.000	😊
7.000	😊
7.000	😊
5.000	😊

9. Right-click the **CATEGORIES** node and select **Open**.
10. Click the **James Bond Movies** category to examine the matched documents that were retrieved by the newly created Boolean rule. Twenty-five James Bond movie document extractions are displayed in the document view display.

Note: If you follow the same steps for defining Boolean rules, you can create all the other Boolean rules that are documented in other lessons.

End of Demonstration

4.07 Multiple Choice Question

Which sub-category Boolean rule has the incorrect syntax?

- a. (AND, (NOT, (OR, "Enterprise Guide", "DDE", "Drug")), (OR, "Business Intelligence", "Enterprise Miner"))
- b. (SENT, (NOT, (OR, "Text Miner", "REG", "GLM")), (OR, "Business Intelligence", "Enterprise Miner"))
- c. (SENT, "{OrionGlimmerprice}", "_c{affordable}")
- d. (ORD,_tmac:"@Top/NAME/FIRST",_tmac:"@Top/NAME/LAST")



Practice

1. Creating Appropriate Concept or Category Rules

Using the ASRS document collection, create appropriate concept or category rules. You can use the existing **ASRS** project.

- a. Aerial navigation charts contain *waypoints*, also called *fixes*, to help pilots navigate and to establish locations that are identifiable by both pilots and air traffic controllers. These waypoints are identified using four-letter or five-letter codes. Using concept (LITI) rules, find accident reports that contain the following waypoints: CLARR, LYNSY, MIROK, OCALA, and SKEBR.
- b. The ASRS reports were obtained in 2007 and are heavily edited, primarily to facilitate a rapid search using 1980s computer technology. Modern ASRS reports look substantially different due to enhancements in search technology. One common edit was to label waypoints, like those described in step a, as intersections. They are abbreviated in the reports as *intersect*. For example, you might read about the KADIE intersect. Write a concept (LITI) rule to try to extract as many waypoints as possible and assume that they follow the conventional waypoint identifier that is followed by the word *intersect* (for example, *KADIE intersect*).
- c. Many ASRS reports describe events that are related to JFK Airport (John F. Kennedy International Airport) in New York, New York, USA. However, some confusion might occur because there is a navigational aid called the JFK VOR that is located on the airport. Write a concept rule that identifies *JFK* when it is within two words of the term *airport*.

End of Practices

4.4 Lesson Summary

In SAS Visual Text Analytics, concept rules are written using LITI (language interpretation and text interpretation) syntax. When you write concept rules, you are writing rules that recognize items in context so that you can extract only the pieces of the document that match the rule.

SAS Visual Text Analytics also simplifies the task of Boolean rule definition and taxonomy development by automatically creating categorization rules. After the rules are established, they can be semantically enhanced and refined with defined concepts. To enhance context sensitivity, you can modify the preliminary rules.

You can add or modify Boolean operators, characters, and other selections to make the rule-matching more context-sensitive. If you imported a SAS Enterprise Content Categorization project, the concepts that were created using LITI rules appear in your project as custom concepts. You can edit them further by using the rules editor.

4.5 Solutions

Solutions to Practices

1. Creating Appropriate Concept or Category Rules

Using the ASRS document collection, create appropriate concept or category rules. You can use the existing **ASRS** project.

- Aerial navigation charts contain *waypoints*, also called *fixes*, to help pilots navigate and to establish locations that are identifiable by both pilots and air traffic controllers. These waypoints are identified using four-letter or five-letter codes. Using concept (LITI) rules, find accident reports that contain the following waypoints: CLARR, LYNSY, MIROK, OCALA, and SKEBR.

Concept name:

WAYPOINT

Concept rule:

Edit a Concept	
1	CLASSIFIER:CLARR
2	CLASSIFIER:LYNSY
3	CLASSIFIER:MIROK
4	CLASSIFIER:OCALA
5	CLASSIFIER:SKEBR

- The ASRS reports were obtained in 2007 and are heavily edited, primarily to facilitate a rapid search using 1980s computer technology. Modern ASRS reports look substantially different due to enhancements in search technology. One common edit was to label waypoints, like those described in step **a**, as intersections. They are abbreviated in the reports as *intersect*. For example, you might read about the KADIE intersect. Write a concept (LITI) rule to try to extract as many waypoints as possible and assume that they follow the conventional waypoint identifier that is followed by the word *intersect* (for example, *KADIE intersect*).

Custom concept name:

INTERSECT

Concept rule:

C_CONCEPT:_c{w} intersect

- Many ASRS reports describe events that are related to JFK Airport (John F. Kennedy International Airport) in New York, New York, USA. However, some confusion might occur because there is a navigational aid called the JFK VOR that is located on the airport. Write a concept rule that identifies *JFK* when it is within two words of the term *airport*.

Custom concept name:

JFK_AIRPORT

Concept rule:

CONCEPT_RULE:(DIST_2,"_c{JFK}","airport")

End of Solutions

Solutions to Activities and Questions

4.01 Multiple Choice Question – Correct Answer

Which CLASSIFIER rule has the correct syntax to extract the text
SAS Visual Text Analytics?

- a. CLASSIFIER: SAS Visual Text Analytics _C
- b.** CLASSIFIER: SAS Visual Text Analytics
- c. CLASSIFIER: "SAS Visual Text Analytics"
- d. CLASSIFIER: ('SAS Visual Text Analytics')

4.02 Multiple Choice Question – Correct Answer

Which CONCEPT rule has the incorrect syntax to extract the text
SAS Visual Text Analytics?

- a. CONCEPT: SAS Visual Text Analytics
- b. CONCEPT: SAS Visual _cap _cap
- c.** CONCEPT: "SAS Visual Text Analytics"
- d. CONCEPT: _w Visual Text Analytics

4.03 Multiple Choice Question – Correct Answer

The concept rule C_CONCEPT: SAS _c{enterprise miner} is validated using the following test text: SAS Enterprise Miner is a solution to create accurate predictive and descriptive models on large volumes of data across different sources in the organization. Identify the text string that should be highlighted when you validate the rule.

- a. SAS Enterprise Miner
- b.** Enterprise Miner
- c. SAS
- d. none of the above

47

Copyright © SAS Institute Inc. All rights reserved.

4.04 Multiple Choice Question – Correct Answer

Which CONCEPT_RULE body has the correct syntax?

- a. Orion _cap _c{is wonderful}
- b. (AND, Orion _cap, _c{is wonderful})
- c.** (SENT, "OrionGlimmerprice", "_c{affordable}")
- d. (SENT, "OrionGlimmerprice", "affordable")

CONCEPT_RULE rules must include at least one Boolean operator and one _c marker. Arguments to Boolean operators are specified in double quotation marks.

54

Copyright © SAS Institute Inc. All rights reserved.

4.05 Multiple Choice Question – Correct Answer

Which PREDICATE_RULE body has the correct syntax?

- a. (aa,bb): (DIST_5, '_aa{cameraprice}', '_bb{expensive}')
- b. (yy,zz): (DIST_5, _yy{"cameraprice"}, _zz{"expensive"})
- c. (xx,yy): (DIST_5, "_xx{cameraprice}", "_yy{expensive}")
- d. (aa,bb): (DIST_5,"_aa{cameraprice}","_bb{expensive}")

Boolean operators are followed by a comma and a space.
Arguments are specified in double quotation marks and
are separated by a comma and a space.

4.06 Multiple Choice Question – Correct Answer

Which REGEX rule has the correct syntax to extract a part number that starts with XYZ- and any five-digit numbers (for example, XYZ-12345)?

- a. REGEX:[XYZ][\ -][1234567890]
- b. REGEX:XYZ[\ -][1234567890]+
- c. REGEX:[XYZ][\ -][\d\d\d\d\d]
- d. REGEX:XYZ[\ -][\d\d\d\d\d]

4.07 Multiple Choice Question – Correct Answer

Which sub-category Boolean rule has the incorrect syntax?

- a. (AND, (NOT, (OR, "Enterprise Guide", "DDE", "Drug")), (OR, "Business Intelligence", "Enterprise Miner"))
- b. (SENT, (NOT, (OR, "Text Miner", "REG", "GLM")), (OR, "Business Intelligence", "Enterprise Miner"))
- c. (SENT, "{OrionGlimmerprice}", "_c{affordable}")
- d. (ORD,_tmac:"@Top/NAME/FIRST",_tmac:"@Top/NAME/LAST")

100

Copyright © SAS Institute Inc. All rights reserved.



Lesson 5 Case Studies

5.1	Introduction to the Case Studies.....	5-3
5.2	Retrieving Information and Documents about Anxiety and Depression from Drug Reports.....	5-4
	Demonstration: Information and Documents Retrieval from Drug Reports Related to Depression and Anxiety.....	5-7
5.3	Automatic Categorization of ASRS Incident Reports.....	5-33
	Demonstration: Automatically Classifying ASRS Procedure Noncompliance Reports	5-34
5.4	Retrieving Mortgage Complaints from the CFPB Customer Complaints Data (Self-Study).....	5-45
	Demonstration: Exploring and Categorizing Consumer Complaints (Self-Study).....	5-46

5.1 Introduction to the Case Studies

Case Studies: Objectives

The activities in the case studies provide you with an opportunity to practice the following tasks:

- quickly and efficiently retrieve information about medical conditions and side effects from drug reports
- automatically categorize ASRS safety reports
- retrieve information from the CFPB complaints data about customer complaints related to home loans

An abbreviated analysis of three document collections illustrates how SAS Visual Text Analytics can be used in a business, government, or scientific setting.

5.2 Retrieving Information and Documents about Anxiety and Depression from Drug Reports

Drug Report Document Collection

- The drug report documents contain comments from patients about prescription medications that are used for the treatment of depression.
- These comments were posted in an online forum.
- The names of medications that are referenced in the documents were altered.
- These documents discuss weight gain and sleeplessness.

This case study uses a SAS data file generated from individual text files (blogs) that were extracted from an online forum. The forum is associated with prescription medications that were used for the treatment of depression. The next two slides describe the characteristics of the online forum comments. ***The main goal of this study is information retrieval related to medication, dosage, and side effects and to identify documents that are related to depression and anxiety.***

Drug Report Document Collection

Topics in the document collection include the following concerns:

Depression
and
Anxiety

Weight
Gain

Sleep
Issues

Category Taxonomy

- Depression
- Weight Gain
- Sleep Issues
- ...

7



Copyright © SAS Institute Inc. All rights reserved.

Drug Report Document Collection

Some of the documents include the following key details:

Medication
Name

Dosage
Level

Side Effects
Experienced

Concept Taxonomy:

- MedicationName
- DosageLevel
- SideEffects
- ...

8



Copyright © SAS Institute Inc. All rights reserved.

Drug Reports: Custom Concepts

Four custom concepts are defined in the Concepts node. They are listed in the table on the right with examples from the concept's category.

Concept	Examples
Medication	Abidal Escalan Cenerol
Dosage	20mg 60 MG 50 Mg
Prescription	Abidal 20mg 60 MG of Escalan 50 Mg Cenerol
SideEffects	Headache Blurred vision Nausea

For efficiency, the four concepts are already defined in the **Drug Reports** project.



Information and Documents Retrieval from Drug Reports Related to Depression and Anxiety

This demonstration illustrates how to use the features of SAS Visual Text Analytics for the efficient extraction of information and document categorization from a document collection.

Note: Some of the steps that are described in the following narrative were already performed in the **Drug Reports** project.

1. Open Model Studio in the Virtual Lab for the course. The **Drug Reports Anxiety and Depression** project should be one of the choices. The New Project setup is shown below to aid you in the creation of the project.

New Project

Name: *

Type: *

Data source: *

Project language: *

Description:

2. You must assign the role of **Text** to the **DrugReport** variable.
3. Verify that the default Text Analytics pipeline is created. Change the settings for the Concepts node. Select the **Include predefined concepts** check box. Run the pipeline.
4. Select the **Concepts** node and select **Open**.
5. The RulesAndPrompts.txt file has the concept definitions for MEDICATION, DOSAGE, PRESCRIPTION, and SIDE_EFFECTS. Copy and paste the LITI rules from RulesAndPrompts.txt file to the corresponding custom concept
6. After you complete the creation of the custom concepts, run the pipeline.
7. Open the **Concepts** node.
8. Select the concept **MEDICATION**. In the Documents pane, select **Matched**.

The display below illustrates part of the MEDICATION concept rules and the matched documents.

The screenshot shows the SAS Model Studio interface with the following details:

- Left Panel (Concepts):** Shows the navigation tree under "Drug Reports Anxiety and Depression > Concepts". It includes sections for Predefined Concepts (9) and Custom Concepts (4), with MEDICATION selected.
- Right Panel (Edit a Concept):** Displays the code for the MEDICATION concept. The code is as follows:


```

1 CLASSIFIER:Abidal
2 CLASSIFIER:Abradon
3 CLASSIFIER:Acqil
4 CLASSIFIER:Ambutrin
5 CLASSIFIER:Ameiorex
6 CLASSIFIER:Amicoran
7 CLASSIFIER:Amlican
8 CLASSIFIER:Aquiven
9 CLASSIFIER:Attentor
10 CLASSIFIER:Bifental
11 CLASSIFIER:Catalan
      
```

 A green checkmark indicates "Code is valid."
- Bottom Panel (Documents):** Shows the "Documents" tab selected. It displays a list of 704 matched documents out of 1414 total. The first few documents listed are:
 - DrugReport:** "Prefixan was added to my **ecstapin**(225mg) due to unrelenting depression.I had lost my sisiter mom within before I grieved but still couldn't get over the depression. Within 3-54 daysboth I and my husband noticed Only thing I notice was I sometimes mix up words Anybody else do that? Not bad enough for me to stop m...
 - ...had been on **ecstapin** 150mg until i started having breakthroughs.My doctor has started me on **abidal** 40 taking more will make the dizziness go away or when will it? That is my only side effect, I think? For the one taking it.
 - ...had been on **Ecstapin** for four years, increasing the dose yearly when it began to decrease in effectiveness uncontrollable crying, dizziness, and nausea- sometimes unable to keep any food down for days. I am finally...
 - ...though i'm on **fortifex**, got to me. and the restlessness started to get overwhelming i couldn't concentrate fell into an abyss of depression, one which i have never experienced before, i had to go back on the **prefixa** panic attack with heart pounding no eating for 4-5 days severe sense of loneliness guilt and poor self-worth...

At the bottom of the interface, it says "Document 1 of 704".

The display below illustrates the DOSAGE concept rules and the matched documents.

The screenshot shows the Model Studio - Build Models interface. In the left sidebar, under 'Concepts', the 'DOSAGE' category is selected. On the right, the 'Edit a Concept' panel displays two REGEX patterns:

```

1 REGEX: [\d]+\.\d+mg\.
2 REGEX: [\d]+\s?mg\.

```

A green checkmark indicates 'Code is valid.'

The 'Documents' tab is selected, showing a list of 1414 documents, with 300 matched. A search bar and a magnifying glass icon are present.

Three examples of matched documents are shown:

- DrugReport**
...to my ecstabin(225mg) due to unrelenting depression.I had lost my sisiter mom within 8 months and alth still couldn't get over the depression. Within 3-54 daysboth I and my husband noticed a significant improve I sometimes mix up words Anybody else do that? Not bad enough for me to stop med tho.
- ...been on ecstabin 150mg until i started having breakthroughs.My doctor has started me on abidal 40mg taking more will make the dizziness go away or when will it?? That is my only side effect, I think? For the one taking it.
- ...my dosage to 60mg. I looked wonderful, 10 years younger(no joke), and didn't worry about anything- so (don't know whether it was Abidal related. I started getting really irritable and lost two of my best friends du were the ones who suggested that I go for treatment. I tried going cold turkey, but couldn't, so the doctor s

Document 1 of 300

The display below illustrates the PRESCRIPTION concept rules and the matched documents.

The screenshot shows the Model Studio - Build Models interface. On the left, the Concepts pane lists Predefined Concepts (4) and Custom Concepts (4). Under Custom Concepts, MEDICATION, DOSAGE, PRESCRIPTION, and SIDE_EFFECTS are listed, with PRESCRIPTION selected. The right pane shows the 'Edit a Concept' screen for PRESCRIPTION, which is defined as 'MEDICATION DOSAGE'. A green checkmark indicates 'Code is valid.' Below this, the 'Documents' tab is selected, showing 'All (1414)' and 'Matched (46 of 1414)'. A search bar and a magnifying glass icon are also present. The results list several drug reports, with the first one expanded to show text containing medication names like 'estapin 150mg', 'abidal 40', 'Prexifan 2mg', 'noderall 1mg', and 'Prexifan 5mg'.

The PRESCRIPTION concept illustrates several important points.

- By using concept names that could be the same as terms in the term table, a conceptual ambiguity arises—namely, what if the term *medication* appears in a document? The LITI parser first checks for the existence of a MEDICATION concept (case sensitive). If a MEDICATION concept is not found, it then sets the CONCEPT rule to look for the term *medication* (case insensitive).
- Because a MEDICATION concept exists, the PRESCRIPTION rule will not identify documents with the word *medication*. This is because neither the MEDICATION concept nor the DOSAGE concept has a reference to the term *medication*.
- These types of conceptual ambiguities should be avoided by using concept names that cannot be confused with terms, as described in Lesson 4 (for example, use _MEDICATION rather than MEDICATION).
- Conceptual ambiguities relate to human perception. The computer software has a priority search mechanism to resolve ambiguities. The naming conventions are primarily intended to make it easier for users to understand what information is being requested.

The display below illustrates a part of SIDE_EFFECTS concept rules and the matched documents.

The screenshot shows the Model Studio interface with the following details:

- Left Panel (Concepts):** Shows a tree structure of concepts. Under "Custom Concepts (4)", the "SIDE_EFFECTS" node is selected and highlighted in grey.
- Right Panel (Edit a Concept):**
 - A list of 11 classifier rules:
 - 1 CLASSIFIER:Abdominal pain
 - 2 CLASSIFIER:Aggression
 - 3 CLASSIFIER:Agitation
 - 4 CLASSIFIER:Allergic reaction
 - 5 CLASSIFIER:Amnesia
 - 6 CLASSIFIER:Anemia
 - 7 CLASSIFIER:Back pain
 - 8 CLASSIFIER:Blindness
 - 9 CLASSIFIER:Blurred vision
 - 10 CLASSIFIER:Bone pain
 - 11 CLASSIFIER:Breast pain
 - A status message: "Code is valid."
- Bottom Panel (Documents):**
 - Shows a list of documents under the "DrugReport" category.
 - Sample text from one document:

...I have the **dizziness**, but my mood is awesome. I do not know if taking more will make the **dizziness** go away who has the rash and other side effects....you are allergic!!! Stop taking it.
 - Sample text from another document:

...Constant uncontrollable crying, **dizziness**, and **nausea**- sometimes unable to keep any food down for day
 - Sample text from another document:

...started getting really **irritable** and lost two of my best friends due to my overreactions to their comments. treatment. I tried going cold turkey, but couldn't, so the doctor switched me to something else on a very low buzzing in the ear (only when originally getting on and coming off this medication)
 - Sample text from another document:

...worse-gained weight, became **irritable**, stressedout, unable to get out of bed-too tired, and was having h physically, also--i decided to change to Exulactin, because i read that most people lose weight rather than g that most of my pain is gone, of course. i also am having the disgusting withdraw symptoms from abidal-na

The number of matched documents by custom concepts appear in the following table:

Concept Name	Number of Documents
MEDICATION	704
DOSAGE	300
PRESCRIPTION	46
SIDE_EFFECTS	406

9. Using the search feature in the CONCEPT window, find out the number of documents contained the terms *Abidal* or *Ecstapin* out of the 704 documents that matched all specified medications. Enter the terms *Abidal Ecstapin* in the document search window and click the **Search** button.

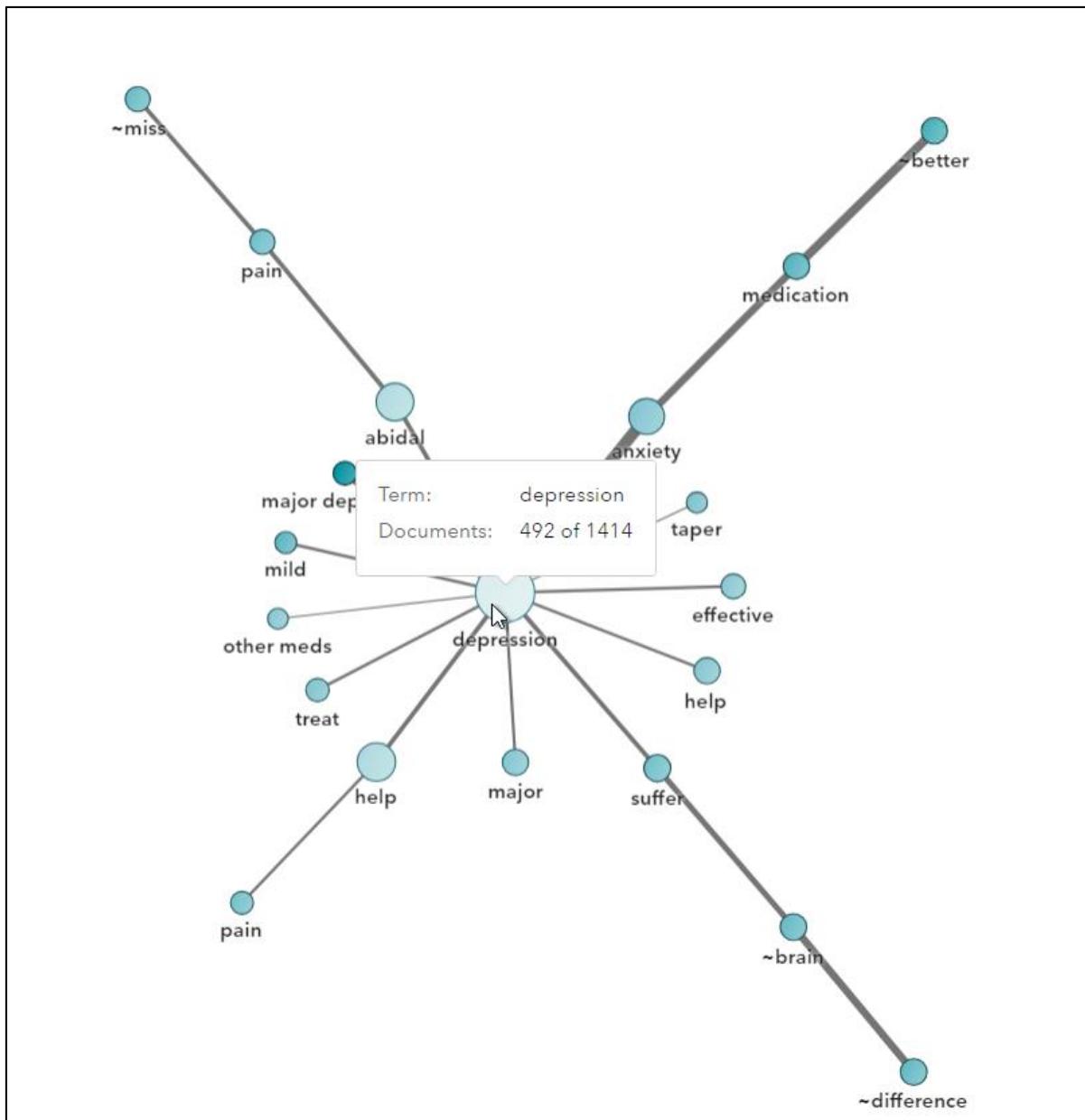
The screenshot shows the SAS Model Studio interface with the 'Build Models' tab selected. In the center, the 'Concepts' window is open, showing a tree structure with 'Predefined Concepts' and 'Custom Concepts'. Under 'Custom Concepts', there are four categories: 'MEDICATION', 'DOSEAGE', 'PRESCRIPTION', and 'SIDE_EFFECTS'. The 'MEDICATION' category is expanded, and its contents are listed. At the bottom of the list, there is a note: 'Code is valid.' Below the list, the 'Documents' tab is active, showing a search bar with 'Abidal Ecstapin'. The results show 562 matches out of 704 total documents. A preview of a 'DrugReport' document is shown, containing text about taking Ecstapin and Abidal. At the bottom of the window, there are buttons for 'Highlight: Concept matches' and 'Search matches'.

You see that 562 out of 704 documents matched the terms *Abidal* or *Ecstapin*.

10. Close the Concepts window.
11. Select the **Text Parsing** node and select **Open**.

The screenshot shows the SAS Model Studio interface with the 'Text Parsing' node selected. The 'Text Parsing - Manage Terms' window is open, displaying two tables: 'Kept Terms' and 'Dropped Terms'. The 'Kept Terms' table lists common words like 'not', 'take', 'depression', 'feel', etc., with their frequency. The 'Dropped Terms' table lists words that were removed from the parsed text, such as 'i', 'be', 'have', etc. Below the tables, the 'Documents' tab is active, showing a search for 'depression' with results from 'DrugReport' documents. A preview of a 'DrugReport' document discusses depression and its impact on the user's life.

12. Select the kept term **depression**. Click  (Term map).



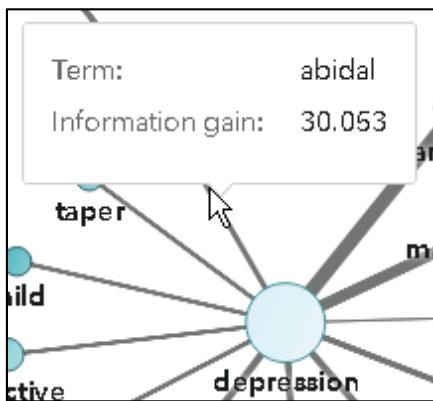
A total of 492 documents contain the term *depression*.

The drug abidal is directly associated with depression based on co-occurrence statistics.

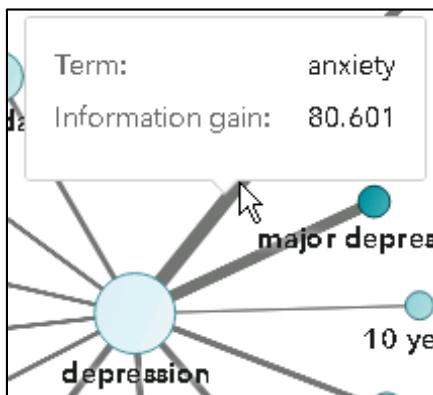
A total of 292 documents contain the drug name *abidal*, and 142 of those *abidal* documents also contain the word *depression*.



The line that connects *depression* and *abidal* has a thickness that is relative to the information gain (IG). You can right-click the line to display the IG value.

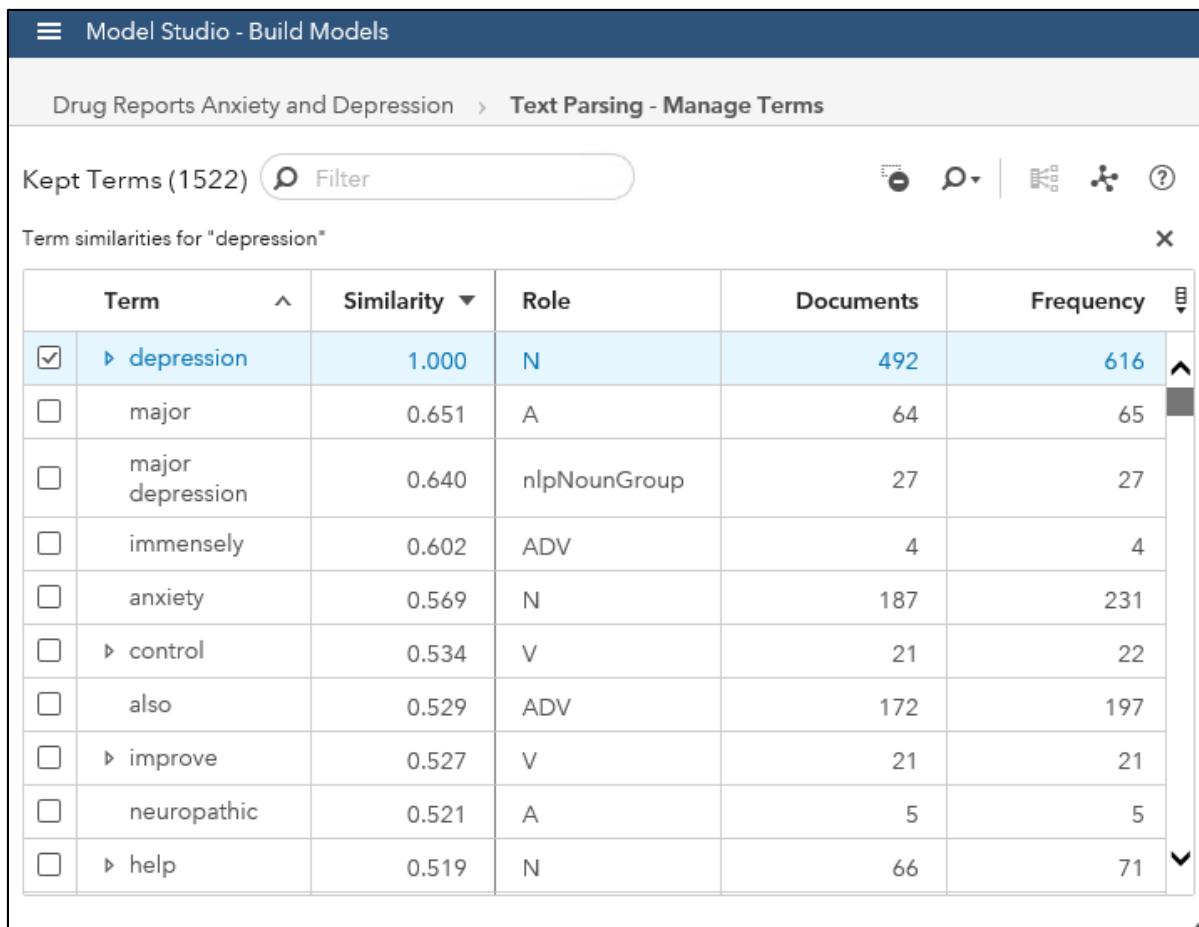


The largest information gain is associated with *anxiety*.



Of all the medications in the MEDICATION concept, *abidal* has the largest information gain as depicted by the term map.

13. Close the term map and select **depression**. Click  (Show similarity scores).



The screenshot shows a table titled "Term similarities for 'depression'" within the "Text Parsing - Manage Terms" section of the Model Studio interface. The table has columns: Term, Similarity, Role, Documents, and Frequency. The "Term" column is sorted by Similarity in descending order. The first row, "depression", has a checked checkbox and a similarity of 1.000. Other terms listed include major, major depression, immensely, anxiety, control, also, improve, neuropathic, and help.

	Term	Similarity	Role	Documents	Frequency
<input checked="" type="checkbox"/>	► depression	1.000	N	492	616
<input type="checkbox"/>	major	0.651	A	64	65
<input type="checkbox"/>	major depression	0.640	nlpNounGroup	27	27
<input type="checkbox"/>	immensely	0.602	ADV	4	4
<input type="checkbox"/>	anxiety	0.569	N	187	231
<input type="checkbox"/>	► control	0.534	V	21	22
<input type="checkbox"/>	also	0.529	ADV	172	197
<input type="checkbox"/>	► improve	0.527	V	21	21
<input type="checkbox"/>	neuropathic	0.521	A	5	5
<input type="checkbox"/>	► help	0.519	N	66	71

14. Scroll down to the first term that has the role MEDICATION.

Term similarities for "depression"					
	Term	Similarity	Role	Documents	Frequency
<input type="checkbox"/>	best	0.397	N	43	44
<input type="checkbox"/>	▷ cope	0.390	V	10	10
<input type="checkbox"/>	given	0.389	N	6	6
<input type="checkbox"/>	▷ combination	0.388	N	20	23
<input type="checkbox"/>	mania	0.383	N	6	7
<input type="checkbox"/>	use	0.376	N	21	22
<input type="checkbox"/>	imitap	0.374	MEDICATION	5	6
<input type="checkbox"/>	painful	0.374	A	8	9
<input type="checkbox"/>	▷ report	0.372	N	4	4
<input type="checkbox"/>	six years	0.371	nlpMeasure	4	4

The medication imitap has the highest similarity score (0.374) relative to depression.

15. Click the **Role** column heading to sort the column by role. (Note that you can sort only by a single column one at a time.) Abidal's similarity score 0.262 is below many other medications (amelorex, imitap, and elevex) with respect to a similarity to depression. This illustrates that similarity score and information gain are two ways to compare term associations.

The screenshot shows a table titled "Term similarities for 'depression'" within the "Text Parsing - Manage Terms" section of the Model Studio. The table has columns: Term, Similarity, Role, Documents, and Frequency. The "Role" column is currently sorted in ascending order (indicated by an upward arrow). The table lists 1522 terms, with abidal having the highest similarity (0.262) and noderall having the lowest similarity (-0.114).

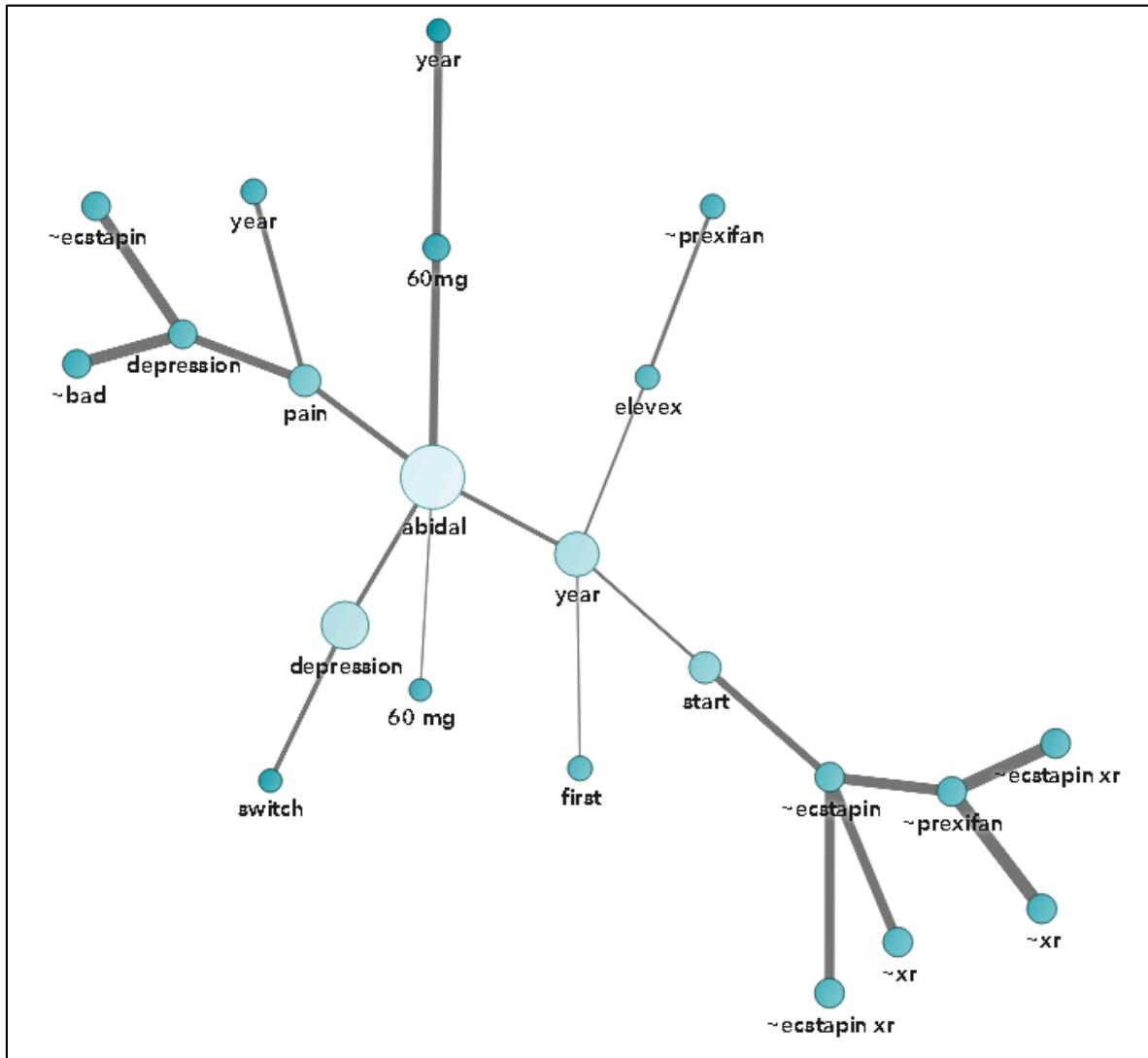
Term	Similarity	Role	Documents	Frequency
revinor	0.010	MEDICATION	13	15
promican	0.139	MEDICATION	45	50
abidal	0.262	MEDICATION	292	433
amelorex	0.303	MEDICATION	10	10
imitap	0.374	MEDICATION	5	6
noderall	-0.114	MEDICATION	4	4
fortifex	-0.016	MEDICATION	12	13
sustify	-0.018	MEDICATION	12	16
escalan	0.243	MEDICATION	64	73
elevex	0.266	MEDICATION	65	72
cenerol	0.242	MEDICATION	56	66

The following table shows the sorted similarity scores for terms that have the role MEDICATION. (The table was created for educational purposes by copying and pasting from SAS Visual Text Analytics to a SAS Studio Program Editor. You can get the same information from SAS Visual Text Analytics with manual scrolling.)

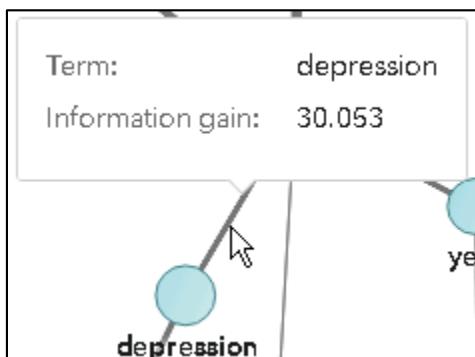
Obs	Term	Similarity	NumDocs	Freq
1	imitap	0.374	5	6
2	amelorex	0.304	10	10
3	cenerol	0.269	56	66
4	escalan	0.253	64	73
5	elevex	0.234	65	72
6	noricam	0.200	5	5
7	habillan	0.198	4	5
8	promican	0.194	45	50
9	ecstapin	0.192	259	368
10	abidal	0.187	292	434
11	prexifan	0.177	102	140
12	perinol	0.170	6	6
13	amicoran	0.122	23	28
14	meriflex	0.098	17	19
15	exulactin	0.081	69	79
16	fortifex	0.074	12	12
17	sustify	0.039	12	16
18	essequal	0.028	6	7
19	revinor	0.014	13	15
20	formilan	-0.093	4	6
21	noderall	-0.150	4	4

Despite the high information gain, abidal is below nine other medications with respect to similarity to the term *depression*.

16. Continuing with the investigation of the drug abidal and its relationship to depression, deselect **depression** and select **abidal** (the one with the role MEDICATION). Click  (Term map).



The term map shows a direct association with *depression* and a second-order association with *depression* through the term *pain*. Right-click the connecting line to verify the information gain for the terms *abidal* and *depression*.



The statistics related to *abidal* and *depression* are consistent.



A total of 142 documents that contain the word *depression* also contain the word *abidal*.

17. Close the term map and then click (Show similarity scores).

The screenshot shows the 'Model Studio - Build Models' interface under 'Text Parsing - Manage Terms'. A table titled 'Term similarities for "abidal"' lists the following data:

Term	Similarity	Role	Documents	Frequency
abidal	1.000	MEDICATION	292	433
fibromyalgia	0.561	PN	5	5
60 mg	0.502	nlpMeasure	23	24
► switch	0.491	V	50	53
1st	0.474	NUM	13	16
grateful	0.471	A	8	10
wasnt	0.470	N	7	7
► pain	0.451	V	16	17
60mg	0.449	nlpMeasure	42	48
► time	0.418	V	18	19
finally	0.409	ADV	12	17

18. Sort by **Role**. Scroll down to the terms that have the role SIDE_EFFECTS.

Kept Terms (1522)						
	Term	Similarity	Role	Documents	Frequency	
<input type="checkbox"/>	med	0.040	PN	6	8	
<input type="checkbox"/>	rx	-0.088	PN	10	10	
<input type="checkbox"/>	away from	-0.072	PPOS	4	4	
<input type="checkbox"/>	because of	0.017	PPOS	50	53	
<input type="checkbox"/>	but for	0.028	PPOS	7	7	
<input type="checkbox"/>	along with	0.019	PPOS	35	35	
<input type="checkbox"/>	away	0.017	PPOS	32	33	
<input type="checkbox"/>	's	-0.132	PRO	43	47	
<input type="checkbox"/>	itching	-0.128	SIDE_EFFECTS	6	8	
<input type="checkbox"/>	irritability	-0.214	SIDE_EFFECTS	7	8	
<input type="checkbox"/>	nightmares	0.104	SIDE_EFFECTS	12	12	

The side effects with positive similarity scores appear in sorted order in the following table:

Obs	Term	Similarity	NumDocs	Freq
1	weakness	0.244	5	5
2	sweats	0.224	11	11
3	confusion	0.185	4	4
4	constipation	0.174	14	15
5	insomnia	0.105	29	31
6	loss	0.103	6	6
7	chills	0.080	11	11
8	sleeplessness	0.051	10	10

No side effects are identified in the term map for abidal.

This demonstration illustrated how SAS Visual Text Analytics can be used for exploration through the information retrieval features of the Concepts node and the Text Parsing node.

The next step is to extract concepts and entities defined in the Concept node from a new document collection by running the concept scoring code.

1. The scoring code for the Concept node is included in the results. Right-click the **Concept** node and select **Results**.
2. Copy the score code in the Results pane.
3. Open SAS Studio by clicking **Develop SAS Code** and open a new Code Editor.



4. Open a new program file and paste the score code from the Visual Text Analytics Concept node results window.

```

1 ****SAS Visual Text Analytics
2 * Concepts Score Code
3 *
4 * Modify the following macro variables to match your needs.
5 * The liti_binary_caslib and liti_binary_table_name variables
6 * should have already been set to the location of the concepts
7 * binary table for the associated SAS Visual Text Analytics project.
8 ****
9
10 /* cas library information for cas table containing the data set you would like to score */
11 %let caslib_name="PUBLIC";
12
13 /* the cas table you would like to score */
14 %let input_table_name = "DRUGREPORT";
15
16 /* the column in the cas table that contains the contains a unique id */
17 %let key_column = "filename";
18
19 /* the column in the cas table that contains the text data to score */
20 %let document_column = "content";
21
22 /* cas library information for output cas tables to produce */
23 %let output_caslib_name = "CASUSER";
24
25 /* the concepts output cas table to produce */
26 %let output_concepts_table_name = "drug_out_concepts";
27
28 /* the facts output cas table to produce */
29 %let output_facts_table_name = "_out_facts";
30
31 /* cas library information for liti binary table... should have been set to your Text Analytics project's cas library */
32 %let liti_binary_caslib = "Analytics_Project_da041c77-e7b4-42ef-9e35-1075995e34e5";
33
34 /* cas table name for liti binary table... should have been set to your Text Analytics project's concept node's model table */
35 %let liti_binary_table_name = "8a72cb6e65af2c740165b44a1c910017_CONCEPT_BINARY";
36
37 /* hostname for cas server */
38 %let cas_server_hostname = "server.demo.sas.com";
39
40 /* port for cas server */
41 %let cas_server_port = 5570;
42
43 /* create a session */
44 cas sascas1 host=&cas_server_hostname port=&cas_server_port uuidmac=sascas1_uuid ;
45 libname sascas1 cas sessref=sascas1 datalimit=all;
46

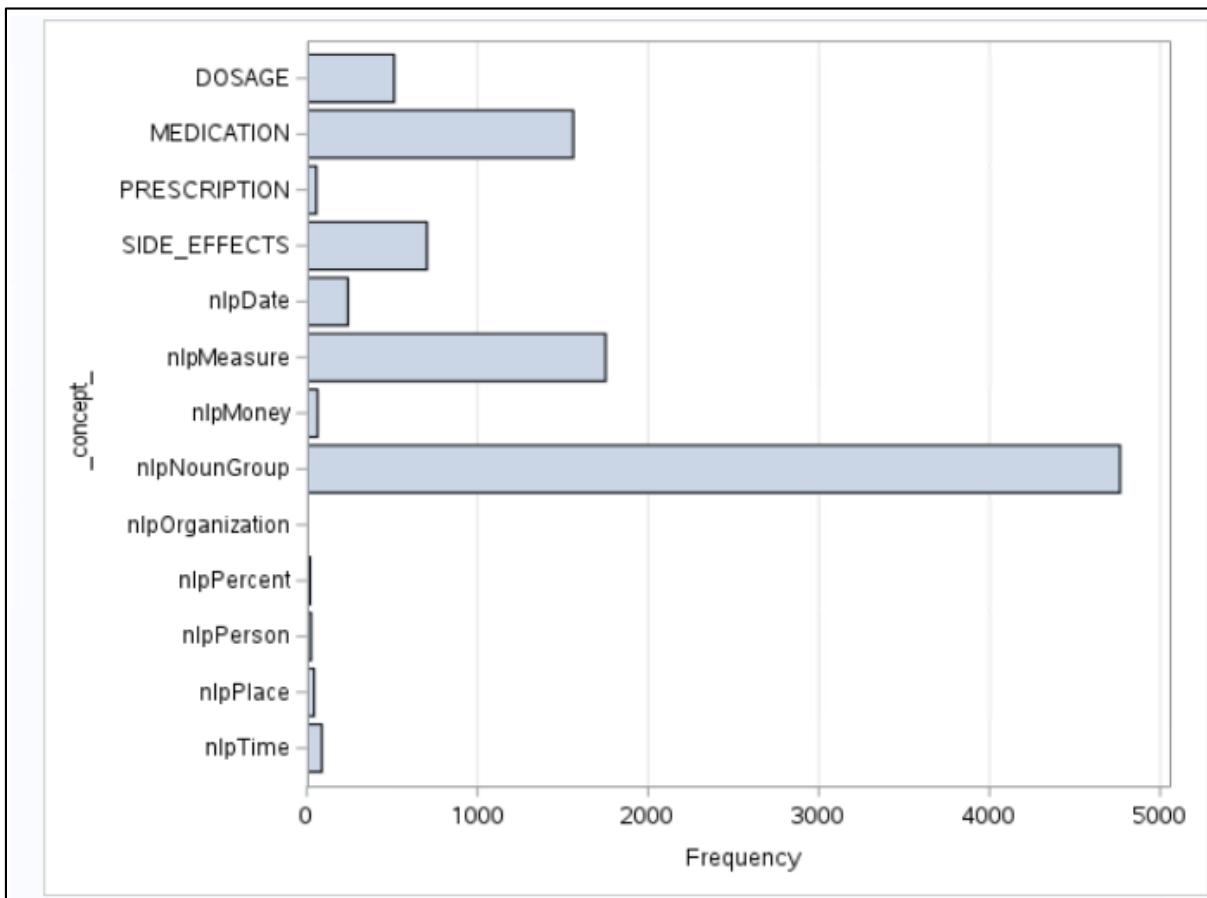
```

5. Update the necessary fields in the score code to point to the location of the new data set (**Drugreport**) and the output library for the results. Modify the remaining macro variable definitions as described in the comments in the Code Editor.

CAS Library	Name	Label	Number of Rows	Number of Columns
CASUSER(student)	drug_out_concepts		9820	7
CASUSER(student)	_out_facts		0	6

List Data for SASCAS1.DRUG_OUT_CONCEPTS			
Obs	fileName	_concept_	_match_text_
1	file1.docx	nlpMeasure	40 pounds
2	file1.docx	nlpMoney	40 pounds
3	file1.docx	nlpMeasure	2 years
4	file1.docx	nlpMoney	10 pounds
5	file1.docx	nlpMeasure	10 pounds
6	file10.txt	nlpMeasure	10mg
7	file10.txt	DOSAGE	10mg
8	file10.txt	nlpNounGroup	10mg prexifan
9	file10.txt	MEDICATION	prexifan
10	file10.txt	nlpMeasure	6 months
11	file10.txt	DOSAGE	2000mg
12	file10.txt	nlpMeasure	2000mg
13	file10.txt	nlpNounGroup	2000mg noderall
14	file10.txt	nlpNounGroup	2000mg noderall 1mg
15	file10.txt	nlpNounGroup	2000mg noderall 1mg gemulex
16	file10.txt	MEDICATION	noderall
17	file10.txt	nlpNounGroup	noderall 1mg
18	file10.txt	PREScription	noderall 1mg
19	file10.txt	nlpNounGroup	noderall 1mg gemulex
20	file10.txt	DOSAGE	1mg
21	file10.txt	nlpMeasure	1mg
22	file10.txt	nlpNounGroup	1mg gemulex
23	file10.txt	MEDICATION	gemulex
24	file10.txt	MEDICATION	fortifex
25	file10.txt	nlpNounGroup	side effects
26	file10.txt	nlpMeasure	20 hrs
27	file10.txt	nlpMeasure	20 hrs.
28	file10.txt	nlpMeasure	30 years
29	file100.txt	MEDICATION	Escalan
30	file100.txt	nlpNounGroup	people situations
31	file100.txt	nlpNounGroup	hard time
32	file100.txt	nlpNounGroup	becoming more brittle
33	file100.txt	nlpMeasure	6 weeks
34	file100.txt	nlpMeasure	5mg
35	file100.txt	DOSAGE	5mg.
36	file1000.txt	nlpNounGroup	extreme nausea
37	file1000.txt	SIDE_EFFECTS	nausea
38	file1001.txt	MEDICATION	Abidal
39	file1001.txt	nlpNounGroup	daily basis

From the new document collection, the extracted predefined and custom concepts for each document is presented in a table. This table was derived from the output scored data by the PRINT procedure in the SAS Studio task.

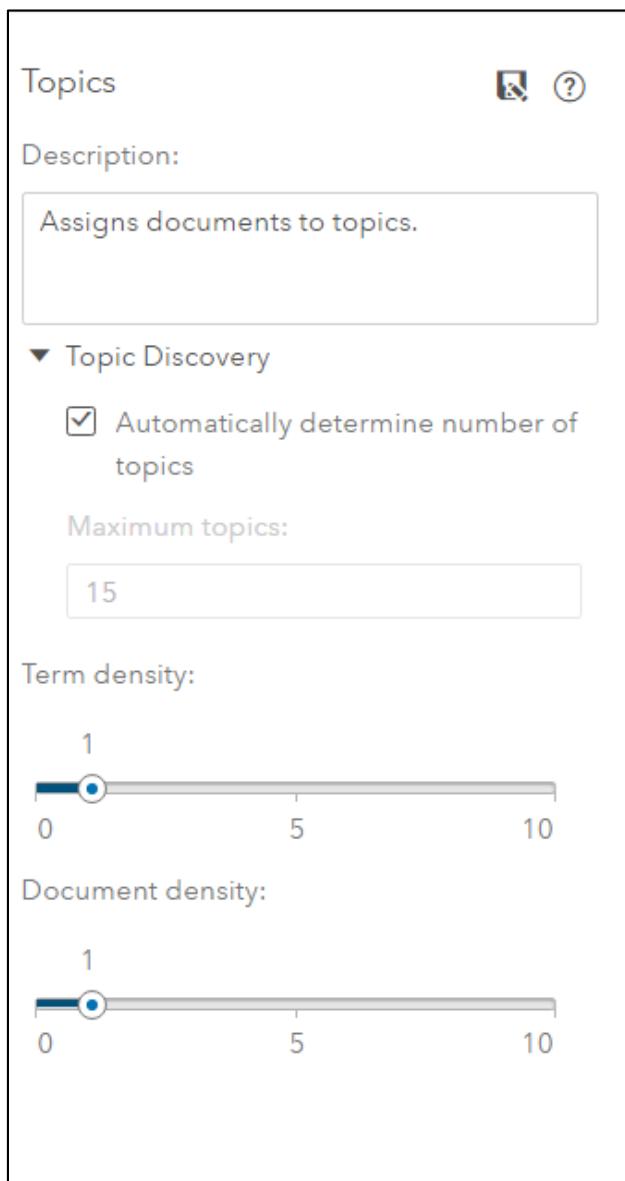


A horizontal bar chart displays the frequency of concepts and entities by predefined and custom entities. This chart was created using the SAS Studio Graph task.

Categorize and Label the Documents Related to Depression and Anxiety

In SAS Visual Text Analytics, there are two methods available to classify documents into user-defined categories. In the first method, you create custom category rules and combine them with predefined concept rules. An example of this method is demonstrated in the second case study in this lesson. In the second method, custom category rules are automatically generated by promoting custom-created topics to categories and running the Category node. The second part of this demo shows how to identify documents related to depression and anxiety by creating custom topics and promoting these topics to categories.

1. Open the default pipeline and examine the default settings of the Topic node. Those settings, displayed below, indicate that the number of topics extracted is set to automatic and both term density and document density are set to 1.



The screenshot shows the SAS Model Studio interface with the following sections:

- Topics (10)**: A table listing 10 system-generated topics. One topic, '+help, +depression, anxiety, +drug, really', is checked.
- Terms**: A table showing terms associated with the topics. The first five terms for the checked topic are '+help', '+depression', 'anxiety', '+drug', and 'really'. Other terms include 'attack', 'not', 'still', 'panic', 'also', 'not', 'still', 'not', 'also', and 'attack'.
- Documents**: A table showing documents related to the checked topic. One document entry is visible: 'DrugReport' with the text '... drug is helping with my depression'.

- Right-click the **Topic** node and open the Output window to examine the system-generated topics and the associated terms. Out of the 10 system-generated topics, only 1 topic captured the depression and anxiety theme. The first 5 terms associated with this depression topic are **+help, +depression, anxiety, +drug, and really**. Only 161 out of 1414 documents and 59 out of 1522 terms were matched by this topic, whereas the term **depression** alone was found in 492 documents.
- To capture more document associated with depression and anxiety, in the second run change the settings in the Topic node as follows:
 - Clear the **Automatically determine the number of topics** box and change the **Maximum topics** value to **20**.
 - Increase the term density from **1** to **3**. (This increases the term relevancy threshold value.) This results in fewer significant terms associated with the topic.
 - Increase the document density from **1** to **2**. (This increases the document relevancy threshold value.) This results in fewer significant matched documents associated with the topic.

4. Rerun the topic node and examine the extracted topics.

Topics  

Description:
Assigns documents to topics.

▼ Topic Discovery

Automatically determine number of topics

Maximum topics:
20

Term density:


Document density:


The screenshot shows the Model Studio interface with the following sections:

- Topics (20):** A table listing 20 topics with their creation details and document counts. One topic, "still, +depress, +medicine, +think, +mood", is highlighted.
- Documents:** A list of 58 matched documents from the "DrugReport" category, including entries like "depression" and "...clearing up my depression."
- Terms:** A table showing 20 terms with their relevancy, role, document count, and frequency. The term "still" is highlighted with a frequency of 133 and relevancy of 0.404.

Increasing the maximum number of topics resulted in 20 topics. However, the number of matched documents associated with the revised **+depression, anxiety, +help, ecstapin, panic** topic decreased to 58. The term relevancy threshold value increased when 20 terms were associated with this topic.

The screenshot shows the Model Studio interface with the following sections:

- Topics (20):** A table listing 20 topics with their creation details and document counts. The topic "still, +depress, +medicine, +think, +mood" is highlighted with 57 matched documents.
- Documents:** A list of 57 matched documents from the "DrugReport" category, including entries like "Still depressed." and "...that I am **angrier** than before I was on this **medicine**, still am **depressed**.
...n't think i am **depressed**".
- Terms:** A table showing 21 matched terms with their relevancy, role, document count, and frequency. The term "still" is highlighted with a frequency of 133 and relevancy of 0.404.

Increasing the maximum number of topics to 20 resulted in a new depression-based topic (**still, +depress, +medicine, +think, +mood**) with 57 matched documents and 21 matched terms. Thus, increasing the number of maximum topics did not help identify more depression- and anxiety-related documents.

The screenshot shows the Model Studio interface with the following sections:

- Topics (21)**: A table showing topics created by System and a User. One user-defined topic is selected: "+depression, still, anxiety, +depress, +medicine".
- Terms**: A table showing 16 matched terms with their relevancy, role, documents, and frequency.
- Documents**: A table showing two drug reports. The first report discusses Abidal and its side effects. The second report discusses severe depression and its effects.

Next, by merging these two system-generated, depression-based topics, a user-defined topic **+depression, still, anxiety, +depress, + medicine** was created. This new user-defined, merged topic matched only 50 documents and 16 terms. Thus, merging similar topics did not help us find more depression-based documents.

The screenshot shows the Model Studio interface with the following sections:

- Topics (21)**: A table showing topics created by System and a User. One user-defined topic is selected: "+depression, still, anxiety, +depress, +medicine".
- Terms**: A table showing 23 matched terms related to the main term **Depress**.
- Documents**: A table showing two drug reports. The first report discusses Abidal and its side effects. The second report discusses severe depression and its effects.

Next, to create a user-defined topic using user-defined terms, a search was made to identify all the terms related to *Depress*. This search identified 23 terms related to the main term *Depress*. All 23 terms were selected, and a user-defined topic was created by clicking (**New topic**).

The screenshot shows the Model Studio interface with the following sections:

- Topics (22)**: A table listing user-defined topics with columns for Topic, Created by, and Documents.
- Terms**: A search interface showing results for "Anxiety". It lists terms like "anxiety" and "severe anxiety" with their roles (N or nlpNounGroup), documents, and frequency.
- Documents**: A table showing drug reports. One entry from "DrugReport" discusses extreme anger and mentions being afraid of the police.
- Sentiment**: A column next to the documents showing sentiment scores (red for negative).

Similarly, to create a user-defined topic related to anxiety, a search was made to identify all the terms related to the term **anxiety**. This search identified two terms related to the main term **anxiety**. After these two terms were selected, a user-defined topic was created by clicking (**New topic**).

Finally, a user-defined, combined depression-and-anxiety-based topic was created by merging the depression-based and anxiety-based user-defined topics. To examine the documents matched by the three user-defined topics, the **Topic** node was submitted to run.

The screenshot shows the Model Studio interface with the following sections:

- Topics (24)**: A table listing user-defined topics, including a new topic merged from depression and anxiety.
- Terms**: A search interface showing results for "Filter". It lists terms like "major depressive disorder", "depressive", and "anxiety" with their relevancy, role, documents, and frequency.
- Documents**: A table showing drug reports. One entry from "DrugReport" discusses having severe anxiety and depression.
- Sentiment**: A column next to the documents showing sentiment scores.
- Relevancy**: A column next to the documents showing relevancy scores.

A substantially large number of documents matching depression and anxiety was identified by this approach.

<input type="checkbox"/>	Topic	Created by	Documents ▾
<input checked="" type="checkbox"/>	deep depression, +tried many antidepressant, severe depression, depressed, +depression	User	683
<input type="checkbox"/>	+depression, +depress, +antidepressant, +anti-depressant, major depression	User	632
<input type="checkbox"/>	anxiety, severe anxiety	User	187

Model Studio - Build Models

Drug Reports Anxiety and Depression > Topics

Topics (24)

<input type="checkbox"/>	Topic	Created by	Documents ▾
<input type="checkbox"/>	deep depression, +tried many antidepressant, severe depression, depressed, +depression	User	683
<input checked="" type="checkbox"/>	+depression, +depress, +antidepressant, +anti-depressant, major depression	User	632
<input checked="" type="checkbox"/>	anxiety, severe anxiety	User	187
<input type="checkbox"/>	side, +effect, +side effect, +make, more	System	92
<input type="checkbox"/>	+symptom, +withdrawal, +drug, off of, horrible	System	78
<input type="checkbox"/>	+pain, abidal, +make, also, +work	System	71
<input type="checkbox"/>	+day, mg, a day, +dose, +miss	System	68
<input type="checkbox"/>	many, best, tried, +other, ecstapin	System	65
<input type="checkbox"/>	weight, gain, weight gain, +cause, med	System	62
<input type="checkbox"/>	+medication, verv, +bad, +work, +well	System	61

Documents

All (1414) Matched (0) Search

DrugReport

No documents were found. To view matches, select a single topic.

Next, these three user-defined topics were promoted () to categories, and the **Category** node was submitted to run.

Model Studio - Build Models

Drug Reports Anxiety and Depression > Categories

Categories

- All Categories (3)
 - deep depression, +tried many antidepressant, severe depression, depressed, +depression
 - +depression, +depress, +antidepressant, +anti-depress
 - anxiety, severe anxiety

Textual Elements (1522)

Code is valid.

Documents Test Sample Text

All (1414) Matched (696 of 1414) Search

DrugReport

...to treat severe **depression** and **anxiety** with heart palpitations. My doc also put me on atenolol to control the palpitations, which is a nice combo because it reduces your blood pressure and prevents migraines. After a year I felt better, like many of you who complained about wanting to get off it, and slowly tapered off. What I have learned in the last year of horrible **depression** trying to do it on my own with diet, exercise, counseling, and...

...great for my **depression**. A few months after starting, I began to gain weight despite eating right. I have also been trying to quit smoking so my doctor switched me to Exulintac recently. I have not taken Abidal for a week now and I wish I would just not wake up. I had intense withdrawals from stopping Ecstapin in the past and hoped to never experience that again. My doctor told me to discontinue Abidal without weaning off. I will make ...

...probably the worst **anti-depressant** I have ever been on. I am 20 years old and have been struggling with **depression**, generalized anxiety, and ADD. I have been on different medications for 3 years. I've been on pretty much every **anti-depressant**. I know regret my decision to take abidal. I thought ...

Document 1 of 696

Highlight: Categories matches

The screenshot shows the SAS Model Studio interface for building models. On the left, the 'Categories' pane displays a tree structure with three categories under 'All Categories': 'deep depression, +tried many antidepressant, severe de', '+depression, +depress, +antidepressant, +anti-depress', and 'anxiety, severe anxiety'. The 'Textual Elements (1522)' pane lists various words and their frequencies, such as 'not' (1174), 'take' (1105), 'depression' (616), 'feel' (517), 'year' (502), 'drug' (487), 'day' (479), 'medication' (446), and 'work' (439). In the center, the 'Edit a Category' pane shows a query: `(OR, (AND, "anxiety"))`. Below it, the 'Documents' pane shows a table with two rows of matched documents from 'DrugReport'. The first document discusses chronic depression and anxiety, mentioning side effects like vomiting and hallucinations. The second document discusses severe depression and anxiety, mentioning heart palpitations and energy loss. Both documents have a sentiment score of 4.000 and a relevancy score of 3.000.

Category rules associated with the depression-anxiety and anxiety categories and the matched documents are displayed above.

End of Demonstration

5.3 Automatic Categorization of ASRS Incident Reports

Aviation Safety Reporting System (ASRS)

The Aviation Safety Reporting System is a voluntary safety reporting system. It is supported by the U.S. National Aeronautics and Space Administration (NASA), with assistance from other U.S. government agencies, like the U.S. Federal Aviation Administration (FAA).

For information and data, visit the following site:

<http://asrs.arc.nasa.gov>



Automatically Classifying ASRS Procedure Noncompliance Reports

This demonstration illustrates how text analytics can be used to automatically classify safety reports.

The goal of the project is to automatically categorize documents into predefined categories. In most text categorization problems, labels are assigned by human judges, so the labels are often subject to error due to the usual problems of fatigue and environment. The labels are classified as target variables.

Information about the Aviation Safety Reporting System (ASRS) as well as access to data through a custom query application can be obtained from the following link: <https://asrs.arc.nasa.gov/>

An incident that involves two or more aircraft can have reports that are filed from pilots of all involved aircraft, as well as from air traffic controllers. In both examples, there is only one ASRS report, but that report is prepared by NASA professionals and is based on all the submitted reports. This report is stored in the **Text** variable.

The data set used that is for this course is a modified version of the data set that is available from the UC Irvine Machine Learning Repository: <https://archive.ics.uci.edu/ml/index.php>

Allan et al. (2008) proposed a text analytics **methodology** for analyzing the data.¹

A report might describe an event as having the following characteristics:

- a “runway ground incursion” anomaly
- a “took evasive action” result
- a “human factor” contributing factor
- a “human factor” primary problem

The course data has 22 labels (or categories). They are named **Target01**, **Target02**, and so on, through **Target22**. The data uses a code of (0,1), and a code of 1 indicates the presence of the label in the document. A document can be associated with one or more labels.

The examples below of the target event labels in the data are described by Allan et al. (2008).

Event Label in Course Data	Description of Event
Target02	Operation Noncompliance
Target05	Incursion (collision hazard)
Target13	Weather Issue
Target21	Illness or Injury event
Target22	Security concern / threat

¹ The full reference is available in Appendix B.

The data set contains the following items:

- the safety report named **Text**
- an ID variable named **ID**
- the 22 target variables with the names **Target01** through **Target22** that were defined above
- the variable **Size**, which is only the length of the report in bytes and it is not used

Only the report itself (**Text**) is needed, but **ID** can remain as an ID variable.

Columns		Total rows: 21519 Total columns: 25		
		ID	Size	Text
<input checked="" type="checkbox"/>	Select all			
<input checked="" type="checkbox"/>	ID	1	000001	575 locate _ distance measuring equipment FROM runway ON THE locate
<input checked="" type="checkbox"/>	123 Size	2	000002	421 IN _ I BECAME AWARE THAT MY ULTRASONIC AND EDDY CURRENT ins
<input checked="" type="checkbox"/>	A Text	3	000003	1157 visual flight rules FROM feet PIERCE flight level TO HABERSHAM COUN
<input checked="" type="checkbox"/>	123 Target01	4	000004	567 JUST PRIOR TO rotate A DEER RAN ONTO THE runway. I rotate AND h
<input checked="" type="checkbox"/>	123 Target02	5	000005	393 climb ON _ degree head TO _ feet GOT traffic alert and collision avoida
<input checked="" type="checkbox"/>	123 Target03	6	000006	1661 locate PUERTO PLATA. maintain DELAY result in flight attend go ILLE
<input checked="" type="checkbox"/>	123 Target04	7	000007	580 land runway AND taxi ACROSS airport underground control WITH re
<input checked="" type="checkbox"/>	123 Target05	8	000008	792 JUST PRIOR TO TOUCHDOWN lax airport tower TOLD US TO go around
<input checked="" type="checkbox"/>	123 Target06	9	000009	1238 depart sfo airport ON PORTE _ CZQ transit. AFTER takeoff clearance t
<input checked="" type="checkbox"/>	123 Target07	10	000010	1146 WE WERE clear apparent FROM THE RAMP TO runway BUT hear runw
<input checked="" type="checkbox"/>	123 Target08	11	000011	639 AFTER be vector FOR A VISUAL approach TO runway _ WE advise air
<input checked="" type="checkbox"/>	123 Target09	12	000012	2106 DURING AN instrument flight rule training flight ON AN instrumentfl
<input checked="" type="checkbox"/>	123 Target10	13	000013	2536 WE WERE ON A flight FROM anc airport TO jnu airport reach CRUISE
<input checked="" type="checkbox"/>	123 Target11	14	000014	1252 I FLY FOR A local TV STATION IN NC. DURING THAT TIME I HAVE cover
<input checked="" type="checkbox"/>	123 Target12	15	000015	333 air traffic control facility stl airport approach control. locate stl airport

In this case study, you want to accurately predict the **Target02** and **Target05** events. More generally, you produce a solution that scores new documents with respect to the presence of an operation noncompliance event and an incursion event in the document.

The goal is to develop a system that automatically classifies incidents to 1 of the 22 target labels to avoid the time, cost, and error that is associated with manually labeling the reports. In other words, you build a model on a data set where experts already read the reports and made evaluations. (This is the sort of process that many people use for contextual analysis. That is, they create a data set of labeled cases and then build an automatic classification or prediction system that is based on these known cases.)

Descriptive Statistics for Numeric Variables							
Variable	N	N Miss	Minimum	Mean	Median	Maximum	Std Dev
Target02	21519	0	0	0.5983549	1.0000000	1.0000000	0.4902423
Target05	21519	0	0	0.1422929	0	1.0000000	0.3493584
Target13	21519	0	0	0.1004694	0	1.0000000	0.3006318
Target21	21519	0	0	0.0141271	0	1.0000000	0.1180175
Target22	21519	0	0	0.0266741	0	1.0000000	0.1611329

Approximately 60% of the **ASRS** data exhibit a value of **Target02=1**. Approximately 14% of the ASRS data exhibit a value of **Target05=1**. The balanced nature of the data for **Target02** and **Target05** helps prevent problems that are often associated with a rare target. Of the five labels shown, **Target13**, **Target21**, and **Target22** might be flagged as rare targets. For binary targets with values in (0,1), you must ensure that the software that is used chooses the value 1 as the primary event and 0 as the secondary event.

1. Open Model Studio.

The **ASRS** project should be one of the existing projects. The instructions include complete details, but many of the actions described were already performed. The project name is **ASRS**, the type is **Text Analytics**, the data set is **CASUSER(student).ASRS**, and the language is **English**.

For the **ASRS** data set, the variable **Text** has a role of **Text**, and the variables **Target02** and **Target05** have a role of **Category**.

2. Select the **Concepts** node. Select the **Include predefined concepts** check box.
3. Run the pipeline.
4. Right-click the **Concepts** node and click **Open**.
5. Select **New Concept**. Enter **NONCOMPLIANCE** as the concept name. In the **RulesAndPrompts.txt** file, copy the **NONCOMPLIANCE** concept rules to the **Edit a Concept** editor. Validate the rules.
6. Select **New Concept**. Enter **INCURSION** as the concept name. In the **RulesAndPrompts.txt** file, copy the **INCURSION** concept rules to the **Edit a Concept** editor. Validate the rules.
7. Close the Concepts interactive window. Run the pipeline.
8. Right-click the **Topics** node and click **Open**.

The screenshot shows the SAS Model Studio interface with the 'Topics' pane open. The 'Topics' pane displays a list of 14 topics, each with a checkbox, the topic name, the creator, the number of documents, and a dropdown menu. The topics listed are: tower, clearance, takeoff, hear, runway; flap, knot, brake, gear, +autopilot; approach, visual, runway, visual approach, +locate; +foot, descend, altitude, +autopilot, cross; flap, takeoff, dispatch, flight, depart; engine, emergency, declare, smoke, cabin; airspace, visualflightrules, classb, area, airspace; install, maintain, inspect, minimumequipmentlist, zzz. The 'Documents' pane shows a list of 21519 documents with filters for 'All' and 'Matched'. The 'Text' pane displays a sample document text related to flight operations. The 'Terms' pane on the right lists terms with their roles, document counts, and frequencies, such as aircraft (N, 12789, 38829), runway (N, 8476, 34670), not (ADV, 13719, 31709), airport (N, 11520, 29034), > foot (N, 8102, 25319), approach (N, 6399, 16357), and no (ADV, 9711, 15951).

The screenshot shows the Model Studio interface with the 'Topics' pane selected. The 'Topics (14)' table lists topics such as 'install, maintain, inspect, minimumequipmentlist, zzz', 'degree, head, turn, radial, degree radial', and 'passenger, flightattendant, seat, hi, captain'. The 'Documents' pane shows a sample document with text about a flight from El Monte to Chino. The 'Terms' pane displays a list of terms with their frequencies, such as 'aircraft' (38829), 'runway' (34670), and 'not' (31709).

Topic	Created by	Documents
install, maintain, inspect, minimumequipmentlist, zzz	System	2589
degree, head, turn, radial, degree radial	System	2503
trafficalertandcollisionavoidancesystem, resolutionadvisory, traffic, +foot, climb	System	2192
passenger, flightattendant, seat, hi, captain	System	2063
taxiway, taxiway, taxi, ramp, groundcontrol	System	2043
hold, line, short line, short, short	System	2015
flightlevel, descend, aircraftnumber, center, aircraft	System	1972

Term	Role	Documents	Frequency
aircraft	N	12789	38829
runway	N	8476	34670
not	ADV	13719	31709
airport	N	11520	29034
> foot	N	8102	25319
approach	N	6399	16357
no	ADV	9711	15951

A total of 14 topics are automatically generated.

9. Select the **tower, clearance, takeoff, hear, runway** topic. Because the descriptive terms relate to activities on the ground, the topic is a candidate for an incursion identifier. In the Documents pane, select **Matched**. In the Terms pane, select **Matched**.

The screenshot shows the Model Studio interface with the 'Topics' pane selected. The 'Topics (14)' table highlights the 'tower, clearance, takeoff, hear, runway' topic. The 'Documents' pane shows a sample document with text about a flight from El Monte to Chino. The 'Terms' pane displays a list of terms with their relevancy scores, such as 'tower' (0.253), 'clearance' (0.239), and 'takeoff' (0.211).

Topic	Created by	Documents
<input checked="" type="checkbox"/> tower, clearance, takeoff, hear, runway	System	3393
flap, knot, brake, gear, +autopilot	System	3199
approach, visual, runway, visual approach, +locate	System	3005
+foot, descend, altitude, +autopilot, cross	System	2976
flap, takeoff, dispatch, flight, depart	System	2943
engine, emergency, declare, smoke, cabin	System	2862
airspace, visualflightrules, classb, area, airspace	System	2629
install, maintain, inspect, minimumequipmentlist, zzz	System	2589

Term	Relevancy	Role	Documents	Frequency
tower	0.253	N	4350	10227
clearance	0.239	N	4279	8416
takeoff	0.211	N	3897	7408
hear	0.197	V	2988	4509
runway	0.172	N	8476	34670
control	0.166	N	7019	15301
say	0.159	V	6871	12895

There are 3,393 documents and 826 terms that are matched to this topic.

10. Deselect the first topic and select the **taxiway, taxiway, taxi, ramp, groundcontrol topic.**

The screenshot shows the Model Studio interface with the 'Topics' pane on the left and the 'Terms' pane on the right.

Topics (14) pane:

Topic	Created by	Documents
install, maintain, inspect, minimumequipmentlist, zz	System	2589
degree, head, turn, radial, degree radial	System	2503
trafficalertandcollisionavoidsystem, resolutionadvisory, traffic, +foot, climb	System	2192
passenger, flightattendant, seat, hi, captain	System	2063
<input checked="" type="checkbox"/> taxiway, taxiway, taxi, ramp, groundcontrol	System	2043
hold, line, short line, short, short	System	2015
flightlevel, descend, aircraftnumber, center, aircraft	System	1972

Terms pane:

Term	Relevancy	Role	Documents	Frequency
taxiway	0.472	INCURSION	1888	5048
taxiway	0.425	N	1520	3657
taxis	0.192	INCURSION	3417	6561
ramp	0.157	N	1012	1748
groundcontrol	0.144	INCURSION	875	1379
runway	0.137	N	8476	34670
gate	0.128	N	2444	3829

Documents pane:

All (21519) | Matched (2043 of 21519) | Search

Text:

...WAS TOLD TO TAXI TO runway _ AT taxiway B bywayof taxiway G runway _ left ON taxiway B AT THIS POINT I taxi DOWN taxiway G AND TOOK A right ON runway _ AND taxi ACROSS runway _ AT WHICH POINT I WAS TOLD TO S...

... taxiout OF phi airport generalaviation RAMP WE WERE instruct TO TAXI bywayof taxiway A taxiway D taxiway G AND HOLD SHORT OF runway _ EXPECT runway I taxiway A turn left ON taxiway D cross THE END OF runway ... SHOULD HAVE turn right TO GET TO taxiway G.THERE ARE NO sign show WHICH WAY TO taxiway G.IT IS confuse TO TURN right TO GET TO taxiway G WHEN YOU ARE go TO runway WHICH IS THE OTHER WAY.THE commercialchar ALSO IS confuse BECAUSE IT look LIKE taxiway D doe NOT crossrunway ...

... PUSHBACK FROM GATE AND BOTH engine start I contact groundcontrol FOR taxiclearance.groundcontrol advise TAXI bywayof taxiway B taxiway K AND taxiway east PLAN runway depart FROM taxiway V.WE begin taxi AND start...

Document 1 of 2043

There are 2,043 documents and 701 terms that are matched to this topic.

11. Select both topics that are described above, and then click (Merge topics).

The following display was created after the pipeline was run:

The screenshot shows the Model Studio interface with the 'Topics' pane on the left and the 'Terms' pane on the right.

Topics (15) pane:

Topic	Created by	Documents
<input checked="" type="checkbox"/> taxiway, taxiway, runway, taxi, clearance	User	4170
tower, clearance, takeoff, hear, runway	System	3393
flap, knot, brake, gear, +autopilot	System	3199
approach, visual, runway, visual approach, +locate	System	3005
+foot, descend, altitude, +autopilot, cross	System	2976
flap, takeoff, dispatch, flight, depart	System	2943
engine, emergency, declare, smoke, cabin	System	2862
airspace, visualflightrules, classb, area, airspace	System	2629
install, maintain, inspect, minimumequipmentlist, zz	System	2589

Terms pane:

Term	Relevancy	Role	Documents	Frequency
taxiway	0.216	INCURSION	1888	5048
taxiway	0.193	N	1520	3657
runway	0.155	N	8476	34670
taxis	0.131	INCURSION	3417	6561
clearance	0.130	N	4279	8416
tower	0.091	N	4350	10227
ramp	0.086	N	1012	1748
groundcontrol	0.085	INCURSION	875	1379

Documents pane:

All (21519) | Matched (4170 of 21519) | Search

Text:

... PUSHBACK FROM THE GATE ground WAS call AND clearance TO TAXI TO runway WAS receive.I taxi bywayof taxiway K east A TO runway.HALFWAY TO THE runway I WAS switch TO ANOTHER frequency WHICH WAS THE tower supervise WHO GAVE US A phonenumber TO CALL AFTER arrive IN evg airport I call.supervise WAS concern ABOUT MY choose OF taxiroute TO runway.SINCE HE THOUGHT I WAS instruct TO taxiway B A TO runway.I explain HOW I v TO AVOID RAMP congest SO I TOOK taxiway K east AND NEITHER THE firstofficer OR I hear THE control SAY taxiway B TO runway.FURTHER discuss OF diligent in listen TO radiocall AND follow taxilnstruction occur WITH THE conv...

... PUSHBACK FROM GATE AND BOTH engine start I contact groundcontrol FOR taxiclearance.groundcontrol advise TAXI bywayof taxiway B taxiway K AND taxiway east PLAN runway depart FROM taxiway V.WE begin taxi AND start checklist. clear MY SIDE FOR traffic AS WE cross THE inactive runway BUT INSTEAD of stop AT taxiway B WE continue ACROSS runway TOWARD THE fulllength departureend of runway.THERE WAS NO traffic FOR runway.I WAS think taxiway V WAS ACROSS runway.THE groundcontrol call US immediate AS WE WERE cross runway AND TOLD US WE WERE suppose TO STOP AT taxiway V BUT THERE WAS NO CONFLICT.I DID NOT HAVE MY airportdiagram OUT AN...

switch TO groundcontrol AT AND WE receive instruct TO TAXI TO runway FROM THE GND push ME understand THE instruct TO RE wait ON taxiway B to taxiway T to taxiway 7 to taxiway P runway BECAUSE OF...

Document 1 of 4170

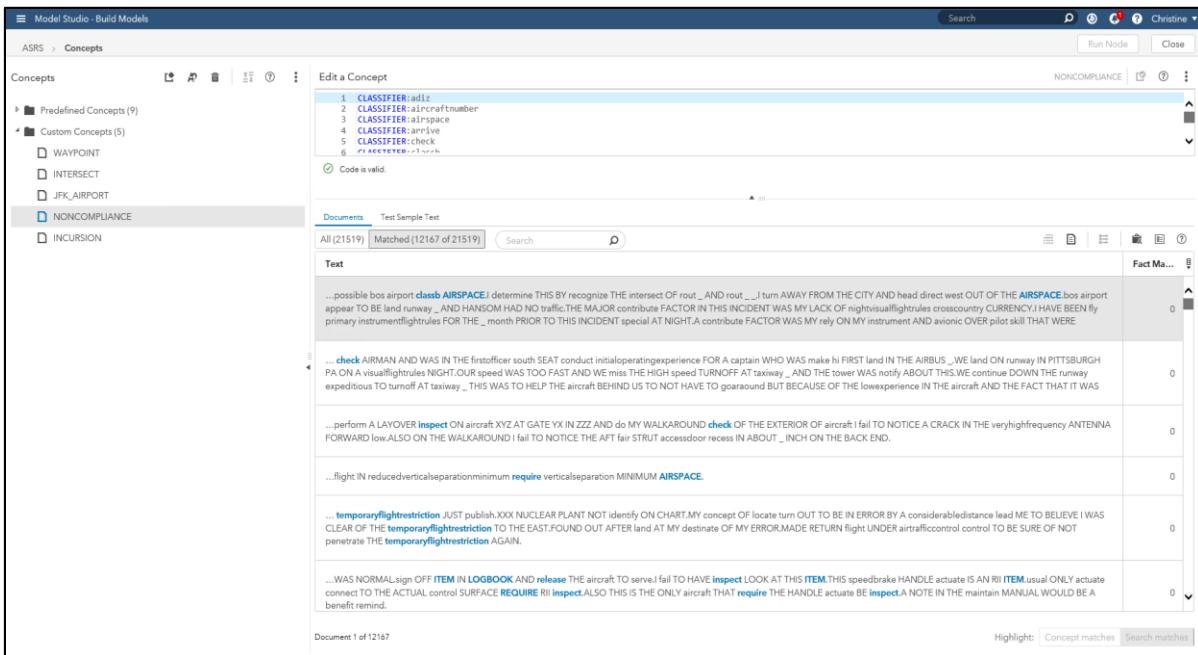
The combined topic identifies 4,170 documents that use 787 terms. The combined topic is added, and the two original topics remain in their original form, which yields 15 total topics.

12. Deselect any other topics and select the merged topic. Then select **Add topics as categories.**

13. Close the **Topics** node to return to the pipeline.
14. Select the **Categories** node and click **Open**.
15. Click  (**New Category**). Enter **NONCOMPLIANCE** to name the category. Enter the following category rule based on the previously defined concept rule in the rule editor:

(OR , "[NONCOMPLIANCE]")
16. Click **New Category**. Enter **INCURSION** to name the category. Enter the following category rule based on the previously defined concept rule in the rule editor:

(OR , "[INCURSION]")
17. Click **New Category**. Enter **NONCOMPLIANCE_2** to name the category. In the RulesAndPrompts.txt file, copy the **NONCOMPLIANCE_2** concept rules to the Edit a Concept editor. Validate the rules.
18. Click **New Category**. Enter **INCURSION_2** to name the category. In the RulesAndPrompts.txt file, copy the **INCURSION _2** concept rules to the Edit a Concept editor. Validate the rules.
19. Close the Categories interactive window.
20. Run the pipeline.
21. Right-click the **Concepts** node and click **Open**.
22. Select the **NONCOMPLIANCE** concept. In the Documents pane, select **Matched**.



The screenshot shows the SAS Model Studio interface. The left sidebar has a tree view under 'ASRS > Concepts' with nodes like 'Predefined Concepts (9)', 'Custom Concepts (5)' (which includes 'NONCOMPLIANCE' and 'INCURSION'), 'WAYPOINT', 'INTERSECT', 'JFK.AIRPORT', and 'NONCOMPLIANCE'. The 'NONCOMPLIANCE' node is selected. The main area is titled 'Edit a Concept' and shows a list of classifier definitions. Below that is a 'Documents' tab with a search bar and results table. The table has columns for 'Text' and 'Fact Ma...'. There are 12,167 rows in the table, with the first few rows visible:

Text	Fact Ma...
...possible bus airport classifies AIRSPACE determine THIS BY recognize THE intersect OF rout_ AND rout_ ... turn AWAY FROM THE CITY AND head direct west OUT OF THE AIRSPACE bus airport appear TO BE land runway ... AND HANSOM HAD NO traffic THE MAJOR contribute FACTOR IN THIS INCIDENT WAS MY LACK OF nightvisualflightrules crosscountry CURRENCY I HAVE BEEN fly primary instrumentflightrules FOR THE ... months PRIOR TO THIS INCIDENT special AT NIGHT A contribute FACTOR WAS MY rely ON MY instrument AND avionic OVER pilot skill THAT WERE	0
... check AIRMAN AND WAS IN THE firstofficer south SEAT conduct initialoperatingexperience FOR A captain WHO WAS make hi FIRST land IN THE AIRBUS ... WE land ON runway IN PITTSBURGH PA ON A visualflightrules NIGHT OUR speed WAS TOO FAST AND WE miss THE HIGH speed TURNOFF AT taxiway ... AND THE tower WAS notify ABOUT THIS WE continue DOWN THE runway expeditious TO turnoff AT taxiway ... THIS WAS TO HELP THE aircraft BEHIND US TO NOT HAVE TO goaround BUT BECAUSE OF THE lowexperience IN THE aircraft AND THE FACT THAT IT WAS	0
... perform A LAYOVER inspect ON aircraft XYZ AT GATE YX IN ZZZ and do MY WALKAROUND check OF THE EXTERIOR OF aircraft I fail TO NOTICE A CRACK IN THE veryhighfrequency ANTENNA FORWARD low ALSO ON THE WALKAROUND I fail TO NOTICE THE AFT fair STRUT accessdoor recess IN ABOUT ... INCH ON THE BACK END.	0
... flight IN reducedverticalseparationminimum require verticalseparation MINIMUM AIRSPACE	0
... temporaryflightrestriction JUST publish XXX NUCLEAR PLANT NOT identify ON CHART MY concept OF locate turn OUT TO BE IN ERROR BY A considerabledistance lead ME TO BELIEVE I WAS CLEAR OF THE temporaryflightrestriction TO THE EAST FOUND OUT AFTER land AT MY destinate OF MY ERROR MADE RETURN flight UNDER airtrafficcontrol control TO BE SURE OF NOT penetrate THE temporaryflightrestriction AGAIN.	0
... WAS NORMAL sign OFF ITEM IN LOGBOOK AND release THE aircraft TO serve I fail TO HAVE inspect LOOK AT THIS ITEM THIS speedbrake HANDLE actuate IS AN RI ITEM usual ONLY actuate connect TO THE ACTUAL control SURFACE REQUIRE RI inspect ALSO THIS IS THE ONLY aircraft THAT require THE HANDLE actuate BE inspect A NOTE IN THE mainten MANUAL WOULD BE A benefit remind.	0

Document 1 of 12167

Highlight: Concept matches Search matches

A total of 12,167 documents match this concept.

23. Select the INCURSION concept. Select Matched in the Documents pane.

The screenshot shows the SAS Model Studio interface with the 'Concepts' node selected in the left navigation pane. The 'INCURSION' category is highlighted. In the 'Documents' pane, the status bar indicates 'Matched (10677 of 21519)'. The results list several documents containing text related to aircraft turnoffs and holds.

Text Sample	Count
... speed TURNOFF AT taxiway ... AND the tower WAS notify ABOUT THIS.WE continue DOWN THE runway expeditious TO turnoff AT taxiway ... THIS WAS TO HELP the aircraft BEHIND US TO NOT HAVE TO goarround BUT BECAUSE OF the lowexperience IN THE aircraft AND THE FACT THAT IT WAS NIGHT WE COULD NOT COMPLETE THE TURNOFF ONTO taxiway ... AND HAD TO CONTINUE TO THIS END OF runway AND turnoff at taxiway ... the tower was notify OF THIS AND WE THEN MADE THE TURNOFF at taxiway ... WE BELIEVE THAT THE aircraft BEHIND US land	0
... clear posit AND HOLD WITH pretakeoffchecklist accomplish WHEN clear FOR takeoff FROM stand STILL advance throttle AND immediate GOT THE takeoffwarninghorn WITH LESS THAN _ knot ON aircraft.throttle immediate reduce AND WE TOLD tower THAT WE need to clear THE runway AND PROCEED BACK TO THE HOLD SHORT .WE exit runway AT taxiway K and return TO THE HOLD .WHEN WE GOT THE HORN immediate notice THE FLAP HANDLE IN the upposition WITH THE leadingedgedevice AND TOTAL TEF south show NORMAL UP.THE takeoffwarninghorn	0
... power UP aircraft.WE cross HOLD SHORT BUT BEFORE WE captain COULD GET OUT word BELOW THE LINE tower call US TO immediate HOLD posit.CESSNA HAD ALREADY COME RIGHT IN FRONT OF US AND DID A TOUCH AND GO WITH _ ON SHORT .FINAL WE WERE ABOUT _ feet OVER THE HOLD SHORT WITH brake NOW SET IT WOULD HAVE BEEN TIGHT FOR takeoff tower TOLD _ HE WAS OK and clear TO LAND.TOLD tower we WERE OVER THE LINE land uneventful THEN WE WENT TO posit AND HOLD AND TOOK OFF.report file BECAUSE WE WERE _ feet	0
... ANYONE WITH A VEHICLE approve TO ENTER THE angleofattack AREA SUCH AS AN AIRLINE VEHICLE OR A VEHICLE operate BY the author SUCH AS AN operate VEHICLE ELECTRICIAN OR ani OF THE PEOPLE WHO WORK FOR the author AND HAVE approve VEHICLE ACCESS.IF YOU ARE ABLE TO GET A RIDE IN THEN THE ONLY check YOU ARE require to DO is TO HAVE THE SECURITY GUARD SWIPE YOUR BADGE ON THE WAY THROUGH THE GATE.NO wand OR ani OTHER physic check TO SEE IF YOU ARE IN possess OF AN ILLEGAL DEVICE SUCH AS A WEAPON	0
... THEY signal A STOP .BETTER train FOR groundpersonnel is ESSENTIAL THE rampersonnel IN THIS CASE signal A STOP TOO LATE.I HAVE notice deficiency IN rampersonnel train IN ALL OF OUR city IE IMPROPER signal NOT FOLLOW STANDARD procedure ETC.THE ONLY OBVIOUS REMEDY I SEE FOR OUR AIRLINE IN THIS AREA IS BETTER COMPENSATION FOR OUR RAMP worker AND BETTER train.IT IS a seriousproblem.RAMPERS routine WALK OUT OF VIEW DURING engine start marshal aircraft AT NIGHT WITHOUT light wand etc.	0

A total of 10,677 documents match this concept.

24. Close the Concepts node to return to the pipeline.

25. Right-click the Categories node and click Open.

26. Select the NONCOMPLIANCE category. In the Documents pane, select Matched.

The screenshot shows the SAS Model Studio interface with the 'Categories' node selected in the left navigation pane. The 'NONCOMPLIANCE' category is highlighted. In the 'Documents' pane, the status bar indicates 'Matched (12167 of 21519)'. The results list several documents containing text related to aircraft inspections and malfunctions.

Text Sample	Sentiment	Relevancy
... consult THE aircraft minimumequipmentlist .IT require US TO RETURN TO THE GATE A no goitem.I TOLD THE mechanic WHO WAS push US BACK ABOUT THE cargodoor engineindicationandcrewalertingsystemmessage AND THAT OUR minimumequipmentlist require A RETURN TO GATE FOR A DOOR inspect .THE mechanic TOLD ME ALL THE inspect require WAS A VISUAL check OF THE EXTERNAL DOOR LOCK MECHANISM AND THAT WE DIDN'T HAVE TO RETURN TO THE GATE.WE	33.000	
... AND perform serve check ON aircraft TAB Inspect WAS perform ON aircraft BY OTHER HANGER PERSONNEL aircraft require MX NOTE TO CARRYOVER function,flight check DUE TO elevate TAB HINGE BOLT replace LOGBOOK WAS DONE proper BUT minimumequipmentlist sticker WERE NOT place IN COCKPIT proper BY MECHANIC do MX NOTE ON aircraft,airworthiness WAS sign AND aircraft WAS dispath without placard be STUCK ON airspeedindicator AS procedurecall	31.000	
... check XXXX BACKUP generate OIL LEVEL and oilfilter check revise DATE november _ I WAS an assign mechanic TO ACCOMPLISH THIS COMPONENT check . check THE center DISPLAY control PANEL AND IT display THAT the right BACKUP generate OIL LEVEL require service.I open THE right FAN COVER ON aircraft XYZ AND proceed TO serve THE right BACKUP generate IN accord WITH COMPONENT check XXXX.I misunderstand THE COMPONENT check AND HAD realize I HAD	31.000	
aircraftnumber _ WAS return FROM A PATROL MISSION TO mfe airport.aircraftnumber _ WAS ON a trainingflight IN THE PATTERN FOR runway _ AT mfe airport.AT _ feetabovegroundline AND _ nauticalmile west OF the airport aircraftnumber _ pass UNDER aircraftnumber _ .BY AN estimate _ feet,SYNOPSIS aircraftnumber _ WAS A lawenforcement helicopter return FROM A PATROL MISSION.aircraftnumber _ WAS A rentalaircraft ON a trainingflight.THE pilot OF aircraftnumber _ HAD fly on	30.000	
... WAS AN INSTRUMENT check flight FOR OUR _ experiment:RESEARCH aircraft.THE aircraft HAD BEEN extensive modify OVER PAST _ month.MISSION WAS TO check RECEPTION OF ADSB timemeinservice trafficinformation DATA link FROM ZZZZ SUBSEQUENT TO COMPASS SWING AT _ preflight BRIEF indicate THAT RECEPTION SHOULD OCCUR AT locate JUST north OF RICHMOND VA.unfortunate WE WERE UNABLE TO RECEIVE SIGNAL SO WE proceed north toward THE DC AREA SINCE I	29.000	
... AND weather WAS plan us duat.AN instrumentflightrules TRIP seem APPROPRIATE AND WAS file FOR _ ON august _ FOR MMV RAWER bottle RDM AT _ feet.THE		

A total of 12,167 documents appear for this category. This is identical to the number of documents that were identified as having the NONCOMPLIANCE concept in the Concepts node.

27. Select the **NONCOMPLIANCE_2** category. In the Documents pane, select **Matched**.

The screenshot shows the Model Studio interface with the 'Categories' pane open. The 'NONCOMPLIANCE_2' category is selected. The 'Documents' pane displays a list of 12167 matched documents from a total of 21519. Each document entry includes a snippet of text, a red circular icon for sentiment, a numerical relevancy score, and a 'Highlight' button.

Text Snippet	Sentiment	Relevancy
...consult THE aircraft minimequipmentlist IT require US TO RETURN TO THE GATE A no goitem.I TOLD THE mechanic WHO WAS push US BACK ABOUT THE cargodoor engineindicationandrewarningsystemMESSAGE AND THAT OUR minimequipmentlist requires A RETURN TO GATE FOR A DOOR inspect .THE mechanic TOLD ME ALL THE inspect require WAS A VISUAL check OF THE EXTERNAL DOOR LOCK MECHANISM AND THAT WE DIDN'T HAVE TO RETURN TO THE GATE.WE	?	33.000
...AND perform servie check ON aircraft TAB inspect WAS perform ON aircraft BY OTHER HANGAR PERSONNEL aircraft require MX NOTE TO CARRYOVER function.flight check DUE TO elevate TAB HINGE BOLT replace LOGBOOK WAS DONE proper BUT minimequipmentlist sticker WERE NOT place IN COCKPIT proper BY MECHANIC DO MX NOTE ON aircraft.airworthy WAS sign AND aircraft WAS dispatch WITHOUT placard be STUCK ON airspeedindicator A5 proceduralcall	?	31.000
... check XXXX BACKUP generate OILLEVEL and oilfilter check revise DATE november _I WAS an assign mechanic TO ACCOMPLISH THIS COMPONENT check .I check THE center DISPLAY control PANEL AND IT display THAT THE right BACKUP generate OIL LEVEL require service.I open THE right FAN COVER ON aircraft XYZ AND proceed TO serve THE right BACKUP generate IN accord WITH COMPONENT check XXXX I misunderstood THE COMPONENT check AND Had realize I HAD	?	31.000
aircraftnumber _ WAS return FROM A PATROL MISSION TO mfe airport. aircraftnumber _ WAS ON a trainingflight IN THE PATTERN FOR runway _ AT mfe airport.AT _ feetabovegroundlevel AND _ nauticalmile west OF the airport. aircraftnumber _ pass UNDER aircraftnumber _ BY AN estimate _ feet.SYNOPSIS aircraftnumber _ WAS A lawenforcement helicopter return FROM A PATROL MISSION. aircraftnumber _ WAS A rentalaircraft ON a trainingflight.THE pilot OF aircraftnumber _ HAD fly on	?	30.000
...WAS AN INSTRUMENT check flight FOR OUR _ experiment RESEARCH aircraft.THE aircraft HAD BEEN extensive modify OVER PAST _ month.MISSION WAS TO check RECEPTION OF ADSB timesservice trafficinformation DATA link FROM ZZZZ SUBSEQUENT TO COMPASS SWING AT _ preflight.BRIEF indicate THAT RECEPTION SHOULD OCCUR AT locate JUST north OF RICHMOND VA.unfortunate WE WERE UNABLE TO RECEIVE SIGNAL SO WE proceed north toward THE DC AREA.SINCE I	?	29.000
...AND weather WAS plan us dust.AN instrumentflightrules TRIP seem APPROPRIATE AND WAS file FOR _ ON august _ FOR MMV RAWER bottle RDM AT _ feet.THE	?	29.000

The same documents are identified. This is somewhat surprising because noun (@N) and verb (@V) modifiers were used in the category rules, but only CLASSIFIER rules were used in the NONCOMPLIANCE concept. Because only 20 terms were used as classifiers, this is unusual but not impossible.

28. Select the **INCURSION** category. In the Documents pane, select **Matched**.

The screenshot shows the Model Studio interface with the 'Categories' pane open. The 'INCURSION' category is selected. The 'Documents' pane displays a list of 10677 matched documents from a total of 21519. Each document entry includes a snippet of text, a red circular icon for sentiment, a numerical relevancy score, and a 'Highlight' button.

Text Snippet	Sentiment	Relevancy
taxi JUST bare PAST THE instrumentlandingsystem HOLD SHORT LINE AT approach END OF runway AT sdf airport._ aircraft WERE SENT AROUND.THIS occur FOR sever REASON _ THE control say TAXI to runway HOLD SHORT OF THE approach.NOT HOLD SHORT OF instrumentlandingsystem LIVE HOLD SHORT OF runway HOLD SHORT OF taxisway D of WHICH WOULD HAVE MADE SENSE.I THOUGHT THIS WAS STRANGE term BUT HE mean HOLD SHORT OF THE approach TO THE	?	83.000
...WAS clear TO TAXI bywayof taxisway taxisway left FOR a runway depart AS I near taxisway south ON taxisway left I was ask IF I WAS FAMILIAR WITH ORANGE transit I state I WAS UNFAMILIAR AND WOULD LIKE A progressivetaxi instruct.A COMMUTER JET WAS taxis OUT FROM TERMINAL ON ORANGE transit AND I WAS TO FOLLOW region JET TO runway.I identify RJ AND follow ON taxisway J TO taxisway XJJ WAS instruct TO HOLD SHORT OF runway <stop> STOP ON THE HOLD SHORT LINE</stop>	?	71.000
...WERE instruct TO TAXI from THE gate TO runway fulllength.instrust WERE TO TAXI bywayof taxisway B FM HOLD SHORT OF runway ON taxisway M.WE WERE JUST HOLD SHORT OF taxisway I head west ON taxisway M WHEN THE groundcontrol ask IF WE COULD MAKE A left _ degree TURN ONTO taxisway T AND clear AN aircraft THAT WAS IN FRONT OF US hold SHORT OF runway.WE decide THAT WE HAD wingtipclearance AND TOLD THE groundcontrol THAT WE COULD MAKE THE TURN.HE	?	66.000
...airport.THIS WOULD REQUIRE cross THE activetrunway .THE pilot OF THE ROCKET WAS instruct BY THE train TO HOLD SHORT OF runway AT taxisway _ THE pilot OF THE ROCKET readback THE HOLD SHORT instruct.I observe THE ROCKET approach THE HOLD bar ON taxisway _ AND determine THAT THE aircraft WAS NOT go TO STOP .I instruct THE pilot OF THE ROCKET to STOP .THE aircraft stop between THE HOLD bar AND the runwayedge.THE pilot state I AM hold SHORT OF THE runway.	?	65.000

A total of 10,677 documents are identified by the rule. This is the same number of documents that were identified by the INCURSION concept.

29. Select the **INCURSION_2** category. In the Documents pane, select **Matched**.

The screenshot shows the Model Studio interface with the 'Categories' pane open. In the 'Categories' tree, 'INCURSION_2' is selected. The 'Textual Elements (22337)' pane shows a table of words and their frequencies. The 'Documents' pane displays a list of 11,658 matched documents, each with a snippet of text and sentiment/relevancy scores.

String	Role	Frequency
aircraft	N	38829
runway	N	34670
not	ADV	31709
airport	N	29034
» foot	N	25319
approach	N	16357
no	ADV	15951
control	N	15301
then	ADV	13255
say	V	17895

A total of 11,658 documents are identified by the rule. The use of noun and verb modifiers generates different results than for the INCURSION concept. As expected, more documents are identified when noun or verb parent terms are expanded to include stemmed versions of the term. However, anything can occur. That is, more or fewer documents are possible.

For example, the NONCOMPLIANCE_2 category rule includes **plan@N**. This eliminates verb matches, because **plan@V** is not included. If there are many documents in which the word *plan* is used as a verb, these can be identified by **CLASSIFIER:plan**, but not by **plan@N**.

30. Select **runway, hold, taxiway, taxiway, taxi**, the combined topic rule. This is named **Combined Topic** in the results table at the end. Select **Matched** in the Documents pane.

String	Role	Frequency
aircraft	N	38829
runway	N	34670
not	ADV	31709
airport	N	29034
» foot	N	25319
approach	N	16357
no	ADV	15951
control	N	15301
then	ADV	13255
sav	V	12895

31. Close the Categories interactive window.
32. Open the results window to get appropriate statistics for the original target categories.
- | Rule Source | Precision | Recall | F1 | Misclassification |
|-------------|-----------|--------|--------|-------------------|
| Target02/1 | 0.6835 | 0.8517 | 0.7584 | 0.3247 |
| Target05/1 | 0.6478 | 0.7988 | 0.7154 | 0.0904 |
33. Close the results window.
34. Right-click the **Categories** node and select **Download score code**. The ZIP file is stored on the client machine. Unzip the file.
35. Open SAS Studio by clicking the action **Develop SAS Code** to open ScoreCode.sas.



36. The header portion of the ScoreCode.sas file must be modified to provide valid values for the various macro variables. One possible solution is shown below, but the instructor can provide up-to-date values for your Virtual Lab environment.

```
%let input_table_name = "ASRS";
%let key_column = "ID";
%let document_column = "Text";
%let output_caslib_name = "casuser";
%let output_categories_table_name = "_out_categories_ASRS";
%let output_matches_table_name = "_out_matches_ASRS";
```

Note: The **key_column** variable must be numeric and must be unique for every document.

37. Save the edited ScoreCode.sas file as ASRS_CatScoreCode.sas in the course sassrc folder.
38. Run the program. Verify that the categories and matches tables were created. You can write programs to use these tables to calculate additional statistics of interest for all the rules created in the Categories node.

End of Demonstration

5.4 Retrieving Mortgage Complaints from the CFPB Customer Complaints Data (Self-Study)

Consumer Complaint Data

Directive: Find documents that complain about mortgages.

Data were obtained with permission from the Consumer Financial Protection Bureau (CFPB).



Flag_Mortgage

- Doclength
- Product
- Complaint_Narrative



17

Copyright © 2018, SAS Institute Inc. All rights reserved.

sas

The **complaints** data are from the US Consumer Financial Protection Bureau (CFPB).

Note: The CFPB logo is trademarked and should not be used in any external material without prior permission from the Bureau.



Exploring and Categorizing Consumer Complaints (Self-Study)

This demonstration illustrates how to use SAS Visual Text Analytics to explore and categorize consumer complaints that are related to the banking and finance industry.

With permission, the **complaints** data set was obtained from the Consumer Financial Protection Bureau (CFPB). The data are augmented for education purposes. The original data are located at <https://www.consumerfinance.gov/data-research/consumer-complaints/>.

The table below provides a brief description of the provided and generated variables within the **complaints** data set.

Variable	Type	Len	Format	Informat	Description
Company	Char	9			The complaint is about this company.
Complaint_ID	Char	8			Unique identification number for a complaint.
Complaint_Narrative	Char	16384			Consumer submitted description of “what happened” from the complaint. *
Consumer_disputed_	Char	3	\$3.	\$3.	Whether the consumer disputed the company’s response.
Date_received	Num	8	MMDDYY10.		Date on which the CFPB received the complaint.
Date_sent_to_company	Num	8	MMDDYY10.		Date on which the CFPB sent the complaint to the company.
Doclength	Num	8			Length of the complaint narrative submitted by the consumer.
Flag_Communication	Num	3			Binary variable indicating that <i>Communication</i> was the issue.
Flag_DebtCollect	Num	3			Binary variable indicating that <i>DebtCollect</i> was the issue.
Flag_IncorrectInfo	Num	3			Binary variable indicating that <i>IncorrectInfo</i> was the issue.

Variable	Type	Len	Format	Informat	Description
Flag_Mortgage	Num	3			Binary variable indicating that <i>Mortgage</i> was the issue.
Issue	Char	80	\$80.	\$80.	Issue that the consumer identified in the complaint.
Product	Char	76	\$76.	\$76.	Type of product that the consumer identified in the complaint.
State	Char	2	\$2.	\$2.	State of the mailing address provided by the consumer.
Sub_Issue	Char	85	\$85.	\$85.	Sub-issue that the consumer identified in the complaint.
Sub_product	Char	42	\$42.	\$42.	Type of sub-product that the consumer identified in the complaint.
Timely_response_	Char	3	\$3.	\$3.	Whether the company gave a timely response.
ZIP_Prefix	Char	3			Leading three values of the complaining consumer's ZIP code.

* Consumers must opt in to share the narrative. Narratives are not published unless the consumer consents, and consumers can opt out at any time.

The **Complaints** project exists on the server in the Virtual Lab. Nonetheless, the complete steps for setting up the project are included so that users can repeat the steps to obtain mastery with respect to using the software.

1. Open Model Studio.
2. Select **New Project**. Name the project **Complaints**. Select **Text Analytics** as the type of project and select the consumer **complaints** data set as the data source. If the data are not already loaded to memory, import the local file from the server.

New Project

Name: *

Type: *

Data source: *

Project language: *

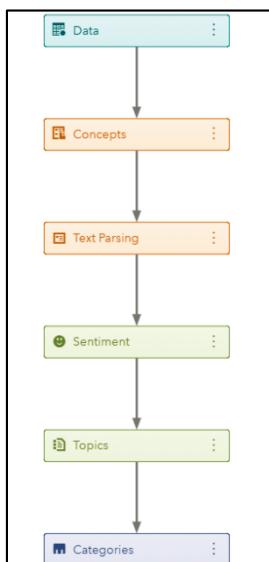
Description:

CFPB Complaints Project

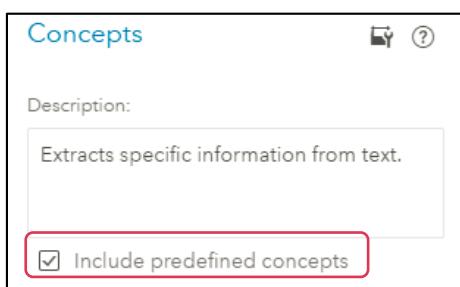
3. To assign the correct roles to the variables, click the corresponding variable. In the menu on the right, select the **Role** drop-down menu and choose the appropriate role.
 - a. Assign **Complaint_Narrative** to the Text role, and assign the Category role to the character variable **Consumer_disputed_**.
 - b. Verify that an ID variable, named **_uniqueid_**, was automatically created and assigned as the key variable. If not, **Complaint_ID** could also be used.

Variable Name	Type	Role	Display Variable
uniqueid	Numeric	Key	
Company	Character		
Complaint_ID	Character		
Complaint_Narrative	Character	Text	✓
Consumer_disputed_	Character	Category	✓
Date_received	Numeric		
Date_sent_to_company	Numeric		
DocLength	Numeric		

4. Click the **Pipelines** tab, and a default pipeline is populated as follows:
Data ⇒ Concepts ⇒ Text Parsing ⇒ Sentiment ⇒ Topics ⇒ Categories



5. Select the **Concepts** node and select the **Include predefined concepts** check box.



6. Right-click the **Concepts** node and select **Run** to apply the settings above.

Exploring Concepts

1. Right-click the **Concepts** node and click **Open**.
2. Under Predefined Concepts in the left pane, investigate **nlpMoney** and **nlpPercent**. Select one at a time and then click the **Matched documents** tab at the bottom to explore the automatic matches of these out-of-the-box entity definitions.

The screenshot shows a list of predefined concepts under a folder icon. The concepts listed are: nlpDate, nlpMeasure, nlpMoney, nlpNounGroup, nlpOrganization, nlpPercent, nlpPerson, nlpPlace, and nlpTime. The 'nlpMoney' and 'nlpPercent' items are highlighted with a red rectangular border.

Documents matching the nlpMoney predefined concept:

The screenshot shows a search interface with the following details:

- Header: Document Test Sample Text
- Search Bar: All (25681) Matched (9864 of 25681) Search
- Table Headers: Complaint_Narrative and Consumer_disputed_
- Table Data:
 - Row 1: ...subdivision is (\$290000.00). The identical Lot that my house sits on is (\$20000.00). This is the same amount as the appraisal that RELs XXXX, through XXXX valued my house for, back in XXXX. I have an additional (\$28000.00) in upgrades, and a (\$31000.00) screened in pool. This (with simple math) puts my house at (\$370000.00). ... | Yes
 - Row 2: ... my from (\$24000.00) to XXXX. Secondly, XXXX needs to disclose the amount my mortgage was bought for because since my address did not change, XXXX sold this note, I am only legally obligated for the amount this mortgage was sold for. I also believe these payments with XXXX to be illegal and the CFPB should investigate this company. | No
 - Row 3: ... has been (\$540.00) and the lender has XXXX raised it to (\$2000.00). I have contacted the lender by phone several times to resolve this matter but they are unable or unwilling to resolve this matter. | N/A
 - Row 4: ...been approx. (\$1100.00) with 20 years left of my loan at 2.35 %. XXXX I am paying over (\$2200.00) on a 40 year loan (I will XXXX years old when I finally pay it off) at over 5 %. This does not include my XXXX mortgage with the same company. ... | No
 - Row 5: ...check for (\$100.00) stating that my car notes had went down and my account was over. I had been dealing with XXXX over the issue of not including my fire policy. | N/A
- Bottom Left: Document 1 of 9864
- Bottom Right: Highlight: Concept matches Search matches

Documents matching the nlpPercent predefined concept:

Document 1 of 2742

Highlight: Concept matches Search matches

3. Expand **Custom Concept** and double-click to open the custom concept named **INTEREST_RATE**. You can disambiguate some of the mentions of percentages that are captured by nlpPercent. You can isolate mentions of APR and interest rates from other contexts such as *%loan to value*, *% of repair costs*, and *% of income*.

Examine the LITI rule associated with the custom CONCEPT_RULE named INTEREST_RATE:

CONCEPT_RULE:(SENT,(DIST_6, "_c{nlpPercent}"),(OR,"rate","apr","interest","introductory"))

The rule type prefix and operators show a blue color when they are properly specified, as shown below. However, you should always validate the rule.

```
1 CONCEPT_RULE: (SENT, (DIST_6, "_c{nlpPercent}"), (OR, "rate", "apr", "interest", "introductory"))
2
```

4. View the Documents window. Click the **Matched** tab to view documents that match the entities identified by this custom concept.

Document 1 of 1245

Highlight: Concept matches Search matches

The idea behind this simple concept is that although it looks for contextual cues that the percentage is related to interest rates, it extracts **only** the cited percentage into a new field. In some cases, you might want to extract only the number and not the symbol so that you derive a numeric value from the text on which you can now do math. This is a great way to enrich a record with new, structured attributes that were previously buried in the text field.

Sometimes the concepts that you build are the main goal for your analysis, and sometimes they are simply building blocks on the path to achieving a larger objective. These are *supporting* or *helper* concepts. For example, you can reference concepts in categorization rules to add extra power and reach to those definitions.

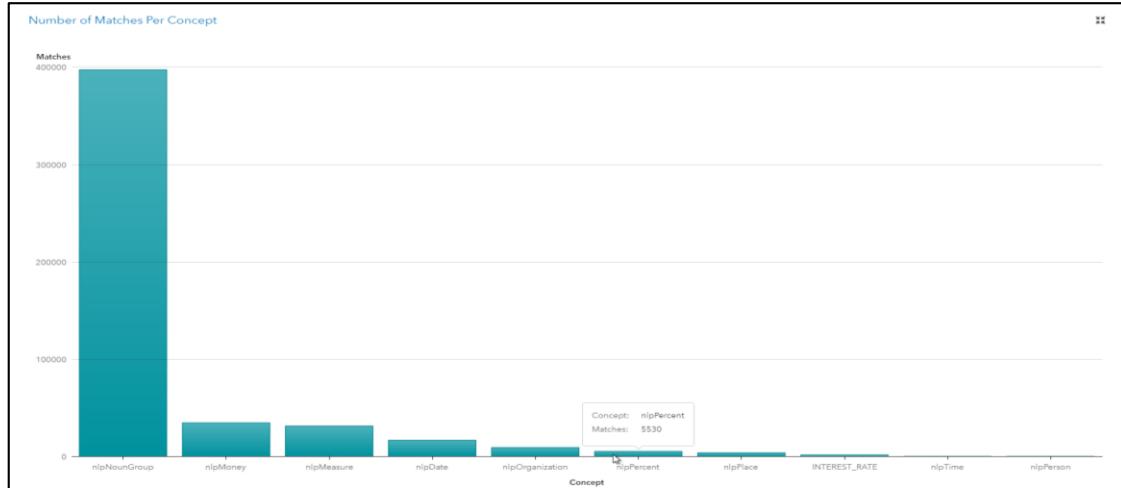
5. Identify documents containing the value 3 % using the Search tool. Enter 3 % within double quotation marks in the search window and click **Search**.

Documents Test Sample Text	
All (115 of 25681) Matched " 3 %"	
Complaint_Narrative	Consumer_disputed_
...to 3 % AND at least have half of the XXXX balance " forgiven. " HA. I paid (\$3000.00) for basically nothing XXXX it was tax refund XXXX so I had some funds XXXX. This XXXX calls themselves XXXX. Their address is : XXXX, XXXX, XXXX. Now I have received a foreclosure notice from XXXX, XXXX, XXXX, XXXX based on the lack of payments to XXXX. I have not ever been able to pay XXXX ...	No
...to 3 %, XXXX this changes to 4.99 % We called and sent a letter stating incorrect amounts on the settlement papers. These were ignored and we were told we had to sign the papers or forfeit getting the new plan for a reduced mortgage rate. In the settlement papers they charged XXXX of unpaid interest this interest had been paid in our prior monthly payments and we should not have been doubled charged. The new principle for this loan was (\$320000.00)....	N/A
...to 3 %, the 7th year my rate would go to 4 %, and the 8th year my rate would go to 4.5 % fixed for the remainder of my loan. I have never made a late payment. I have a credit score of XXXX. I was fully in agreement with the terms of my modification. I went through a divorce and I was awarded the house. XXXX I had to assume the mortgage and remove my x wife's name from the note.	No

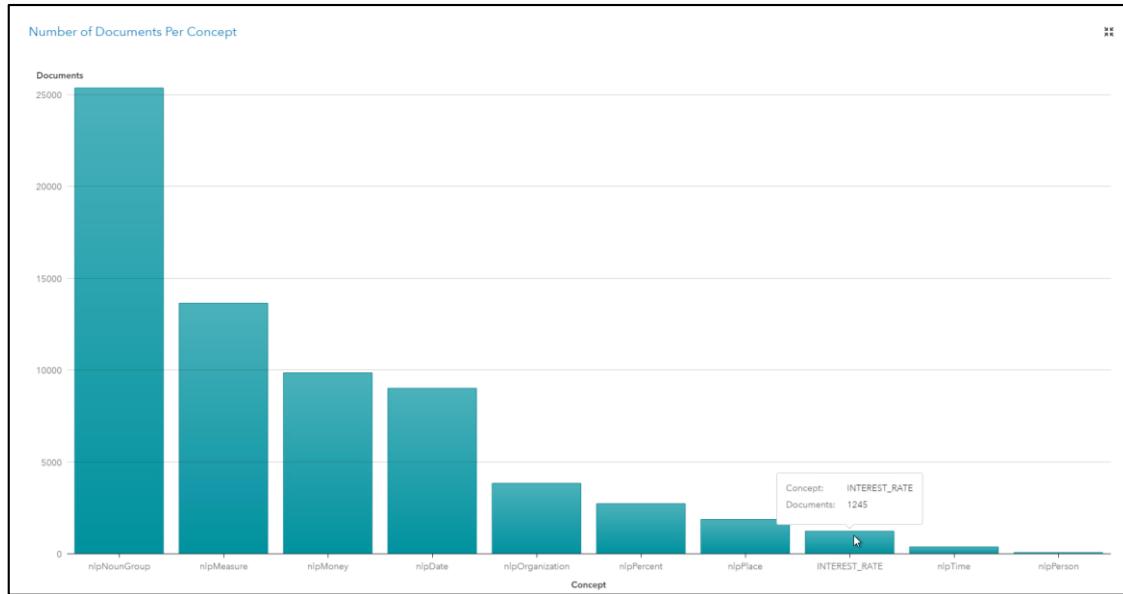
Document 3 of 115

6. Using the button in the top right of the window, close the **Concepts** node.
7. To examine Concept node results, right-click the **Concept** node and open the Results window. The Results window displays three charts:

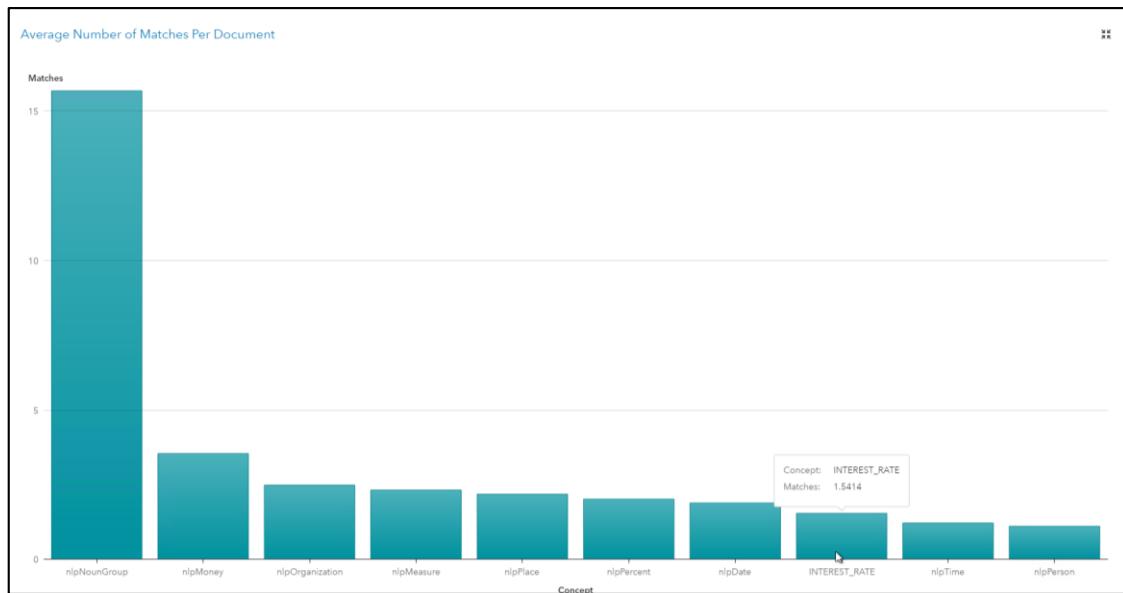
- Number of Matches Per Concept



- Number of Documents Per Concept



- Average Number of Matches Per Document



Exploring Terms

- Right-click the **Text Parsing** node and click **Open**.
- You should explore the data and ensure that the terms are what you would expect in this document collection. On the left, you can easily inspect all the Kept terms that occur in the text collection, and they are sorted descending by frequency. On the right, you can review the terms that are dropped from the downstream topic analysis either because they are on the loaded stop list, or because you interactively dropped a term from this list because it offers little value to your analysis.

View the Kept terms versus the Dropped terms.

The terms xxxx, xx, xxx-xxx, and so on, found in the Dropped list are masked values that represent personal, private, or sensitive terms.

The screenshot shows a table titled 'Kept Terms (22172)' under the 'Text Parsing - Manage Terms' section. The table has columns for 'Term', 'Role', 'Documents', and 'Frequency'. The data includes common verbs like 'not', 'payment', 'loan', 'mortgage', 'pay', 'receive', 'make', 'tell', 'send', 'no', 'call', and 'adv' (ADV).

Term	Role	Documents	Frequency
not	ADV	20102	66522
► payment	N	14406	57201
► loan	N	15186	48174
► mortgage	N	16303	42675
► pay	V	11669	27315
► receive	V	11670	25063
► make	V	11562	24031
► tell	V	10276	23703
► send	V	10364	21707
no	ADV	11448	21122
► call	V	9534	20540

If you expand a term in the Kept list, you can see that the variations are resolved to that parent term, including verb conjugations, plurals, and any synonyms that you imported. For example, to view how the word *charge* is used in context, select it in the list and then view the Matched documents pane.

The screenshot shows two tables side-by-side: 'Kept Terms (39)' and 'Dropped Terms (104679)'. The 'Kept Terms' table includes rows for '► change', '► exchange', '► payment change', '► unchange', 'in exchange for', 'name change', '► rate change', and '► change'. The 'Dropped Terms' table includes rows for 'xxxx', 'the', '► be', 'to', 'i', 'and', 'a', '► have', 'my', '► they', and '► of'. Below the tables is a 'Documents' pane showing a list of matched documents. One document entry is highlighted, showing text from a 'Complaint_Narrative' document: "...never intending to change their opinion on the appraisal, but made me look like a fool in making me jump through these hoops anyway. This seems the very definition of "bad faith"!". The 'Consumer_disputed' field for this document is set to 'Yes'.

Here, *change* is used as a verb, as shown by its detected role. This is made possible by the automatic part-of-speech tagging. The part-of-speech tags are available to you when you write rules.

3. Expand **change** to show its stemmed verb forms and to show how it is used in context in individual documents on the Matched tab.

The screenshot shows a software interface for managing terms in customer complaints. At the top, there's a search bar with the term 'change'. Below it is a table titled 'Kept Terms (39)' with columns for 'Term', 'Role', 'Documents', and 'Frequency'. The table lists various forms of the word 'change' and their frequencies. A cursor is hovering over the row for 'change' (Role: V, Documents: 2770, Frequency: 3710). Below the table, there's a section titled 'Documents' with tabs for 'All (25681)' and 'Matched (2770 of 25681)'. Under the 'Complaint_Narrative' tab, several snippets of text are shown, each containing the word 'change' in blue, indicating it has been expanded or is a key term. One snippet reads: "...never intending to **change** their opinion on the appraisal, but made me look like a fool in making me jump through these hoops anyway." Another snippet discusses a lender's arbitrary **changed** interest rate.

Term	Role	Documents	Frequency
change	V	2770	3710
changed	V	1703	2067
change	V	878	1024
changing	V	429	459
changes	V	155	160
▷ change	N	1292	1740
▷ exchange	N	72	87
▷ exchange	V	47	49
▷ payment change	nlpNounGroup	28	36
▷ unchange	V	28	35

4. Scroll up or down to find *change* listed as a noun. Review how it is used in that context.

5. Select the noun **change** and view the matched documents below.

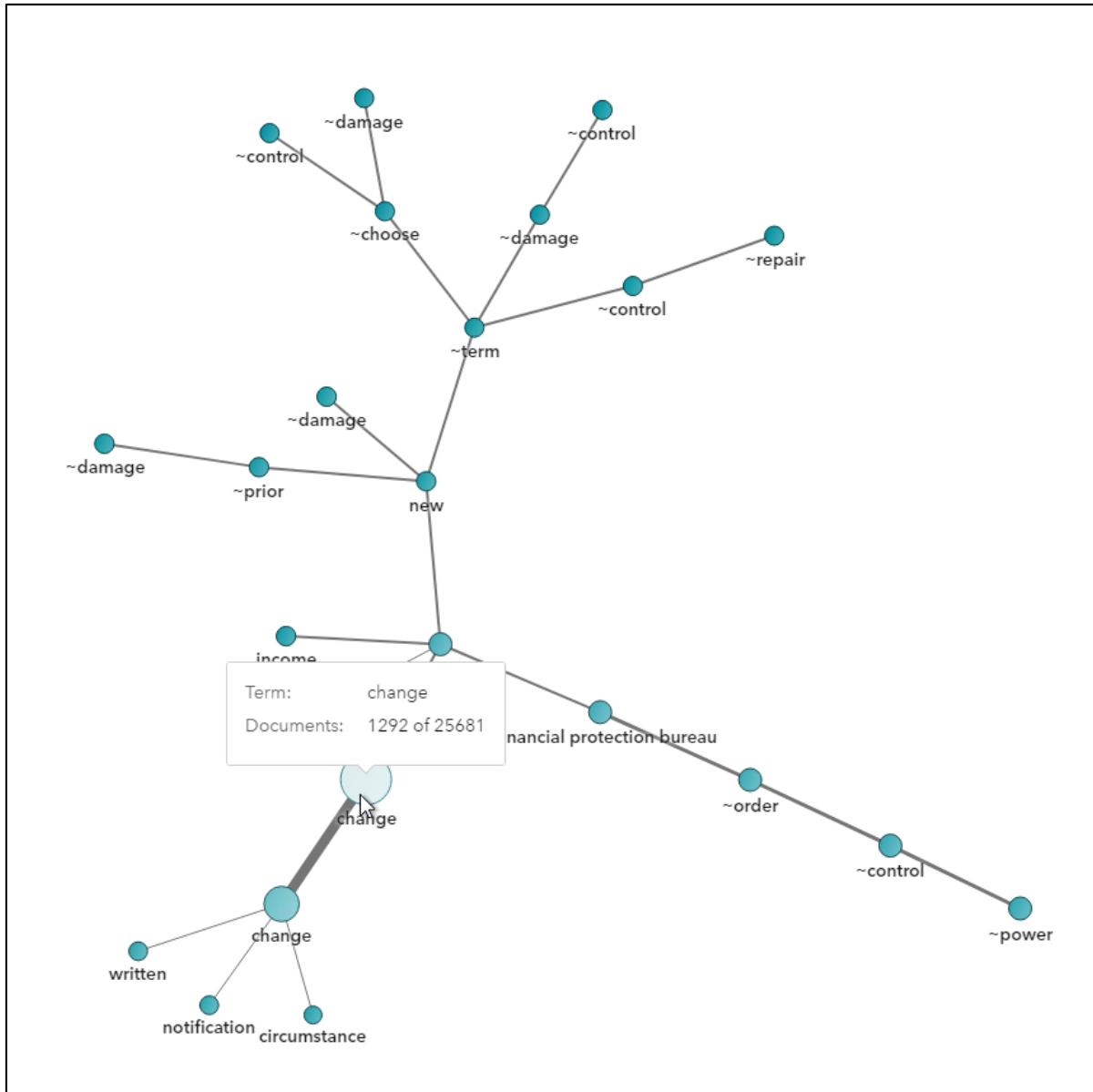
The screenshot shows a table titled "Kept Terms (39)" with a search bar containing "change". The columns are "Term", "Role", "Documents", and "Frequency". The term "change" appears in various roles (V, N) across different documents, with a total frequency of 1292. A specific row for "change" as a noun (N) is highlighted with a blue background, showing 1292 documents and a frequency of 1740. Below the table, there is a section titled "Documents" with tabs for "All (25681)" and "Matched (1292 of 25681)". Under the "Complaint_Narrative" tab, three examples of text are shown where the word "change" has been highlighted in blue. At the bottom left, it says "Document 1 of 1292".

Term	Role	Documents	Frequency
change	V	2770	3710
changed	V	1703	2067
change	V	878	1024
changing	V	429	459
changes	V	155	160
change	N	1292	1740
change	N	977	1263
changes	N	402	477
exchange	N	72	87
exchange	V	47	49
payment			

The term map shows other terms that are most commonly found in documents with your selected term across the entire text collection. It also indicates how reliably those other terms predict that your chosen term might also be present.

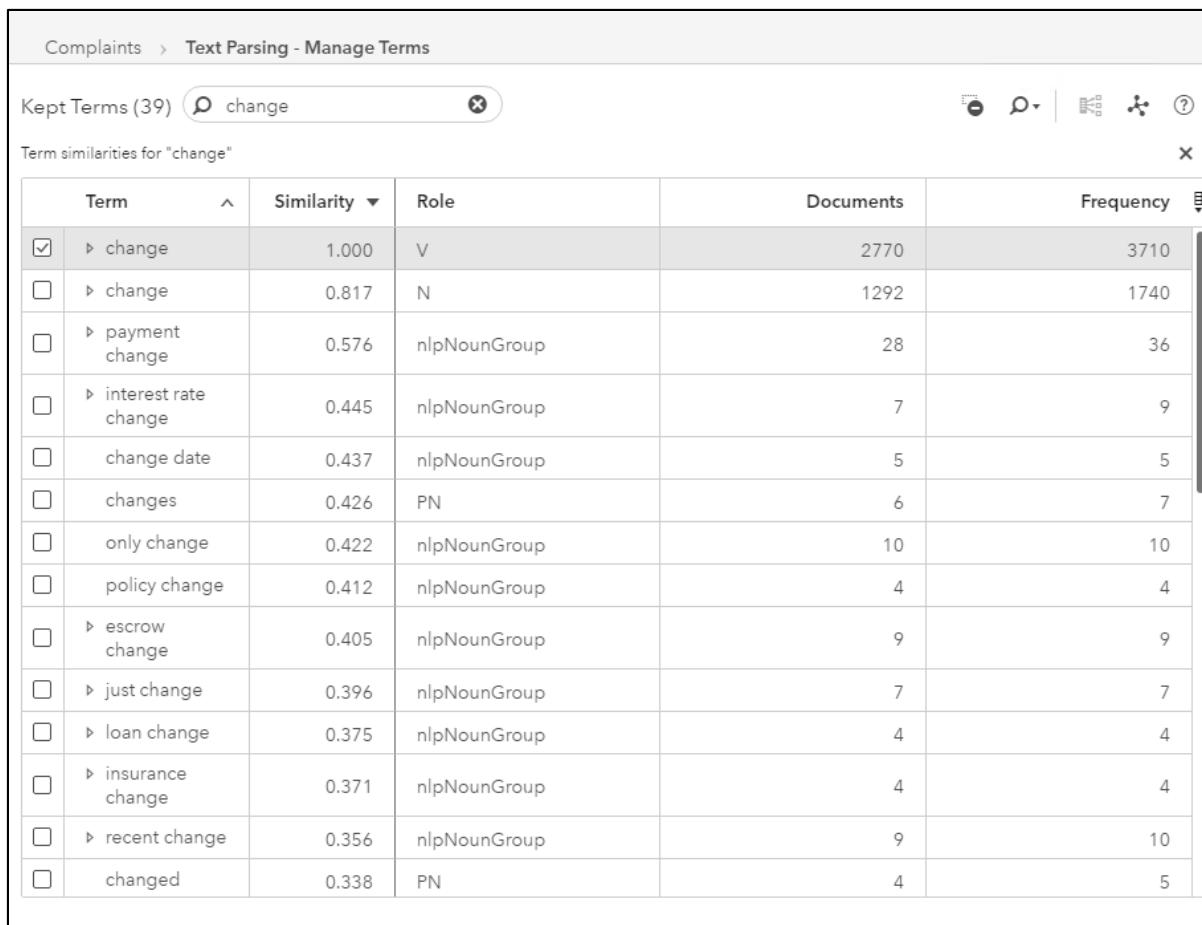
6. Select the noun **change** and right-click or click the **Show Term Map** icon. If the icon is unavailable, you must first click **Run Node** in the top right.

The screenshot shows a table titled "Kept Terms (78962)" with a search bar containing "Filter". The columns are "Term", "Role", "Documents", and "Frequency". The "Show Term Map" icon (a circular icon with a network-like symbol) is located in the top right corner of the interface.



In addition to the term map, you can also view a list of similar terms (that is, terms that are used in a similar or approximately equivalent way as your selected term in the text collection). This can be one way to obtain ideas for additional keywords to build into your category or concept definitions.

7. Select the verb **change** and right-click or click  (Show Similarity Scores).



The screenshot shows a software interface titled "Text Parsing - Manage Terms". In the top left, it says "Complaints > Text Parsing - Manage Terms". Below that, there's a search bar with "Kept Terms (39)" and a dropdown set to "change". To the right of the search bar are several icons: a magnifying glass, a refresh, a dropdown arrow, a grid, a person icon, and a question mark.

The main area displays a table titled "Term similarities for 'change'". The table has columns: Term, Similarity, Role, Documents, and Frequency. The "Term" column lists various forms of the word "change" and related terms, each with a checkbox next to it. The "Similarity" column shows the cosine similarity score for each term. The "Role" column indicates the part of speech or role assigned by the NLP model. The "Documents" column shows how many documents contain each term, and the "Frequency" column shows the raw frequency of each term.

Term	Similarity	Role	Documents	Frequency
▶ change	1.000	V	2770	3710
▶ change	0.817	N	1292	1740
▶ payment change	0.576	nlpNounGroup	28	36
▶ interest rate change	0.445	nlpNounGroup	7	9
change date	0.437	nlpNounGroup	5	5
changes	0.426	PN	6	7
only change	0.422	nlpNounGroup	10	10
policy change	0.412	nlpNounGroup	4	4
▶ escrow change	0.405	nlpNounGroup	9	9
▶ just change	0.396	nlpNounGroup	7	7
▶ loan change	0.375	nlpNounGroup	4	4
▶ insurance change	0.371	nlpNounGroup	4	4
▶ recent change	0.356	nlpNounGroup	9	10
changed	0.338	PN	4	5

8. Click the **Close** button at the top right to return to the pipeline view.

9. Select the **Topics** node and right-click to select **Open**. Review the auto-generated topics.

Complaints > Topics			
Topics (14)			
	Topic	Created by	Documents ▾
<input type="checkbox"/>	+house, +year, +home, +help, +try	System	4219
<input type="checkbox"/>	+check, +payoff, +send, +modification, +company	System	3998
<input type="checkbox"/>	+modification, +loan modification, +application, +deny, +submit	System	3989
<input type="checkbox"/>	+payment, +interest, +principal, monthly, +amount	System	3989
<input type="checkbox"/>	+call, +call, +phone, +speak, +customer	System	3860
<input type="checkbox"/>	+property, +file, +complaint, +debt, +note	System	3282
<input type="checkbox"/>	+fee, +late, +late fee, +charge, +payment	System	3056
<input type="checkbox"/>	+rate, +refinance, +interest, +closing, +close	System	2960
<input type="checkbox"/>	+tax, +escrow, +escrow account, +property tax, +property	System	2924
<input type="checkbox"/>	+credit, +report, +report, +credit report, +bureau	System	2772
<input type="checkbox"/>	+sale, +short, +short sale, +offer, +buyer	System	2304
<input type="checkbox"/>	+insurance, +policy, +flood, +insurance company, +company	System	2188
<input type="checkbox"/>	pmi, +appraisal, +remove, +value, ltv	System	1563
<input type="checkbox"/>	green tree, tree, green, green, +tree	System	507

10. Select a topic. Explore this newly discovered topic by viewing the important terms list at the right and investigating how they are used on the Matched documents tab at the bottom.

Complaints > Topics				Run Node	Close	
Topics (14)				Terms		
	Topic	Created by	Documents ▾	All (22172)	Matched (828 of 22172)	Filter
<input type="checkbox"/>	+house, +year, +home, +help, +try	System	4219			
<input type="checkbox"/>	+check, +payoff, +send, +modification, +company	System	3998			
<input type="checkbox"/>	+modification, +loan modification, +application, +deny, +submit	System	3989			
<input checked="" type="checkbox"/>	+payment, +interest, +principal, monthly, +amount	System	3989			
<input type="checkbox"/>	+call, +call, +phone, +speak, +customer	System	3860			
<input type="checkbox"/>	+property, +file, +complaint, +debt, +note	System	3282			
<input type="checkbox"/>	+fee, +late, +late fee, +charge, +payment	System	3056			
<input type="checkbox"/>	+rate, +refinance, +interest, +closing, +close	System	2960			

Term	Relevancy ▾	Role	Documents	Frequency
> payment	0.278	N	14406	57201
> interest	0.232	N	4277	7559
> principal	0.206	N	1433	2393
monthly	0.201	A	3458	5381
> amount	0.178	N	6286	11948
> rate	0.175	N	3001	6129
> balance	0.174	N	2978	5129

11. Promote one or more selected topics. In this demonstration, the topic **+payment**, **+interest**, **+principal**, **monthly**, **+amount** was promoted to a category by right-clicking and selecting **Add topics as category**.

This action converts statistically discovered topics into Boolean rules that you can alter in the Categories node if you want to fine-tune the context.

12. Close the **Topics** node and run the pipeline.

Exploring Categories

1. Open the **Categories** node.
 2. Select the **Categories** node and right-click to open it.

Categories		Categories	Run Node	Close
<input checked="" type="checkbox"/> All Categories (5)				
<input checked="" type="checkbox"/> Consumer_disputed_				
<input type="checkbox"/> No				
<input checked="" type="checkbox"/> Yes				
<input type="checkbox"/> N/A				
<input type="checkbox"/> +payment,+interest,+principal,monthly,+amount				
		Edit a Category	Consumer_disputed_>Yes	
<pre>[OR,(AND,(OR,"desires","desire","desired"),"mandatory"),(AND,(OR,"assignment","assignments"),(OR,"violate","violating","violated","violates")),(AND,(OR,"identities","identity"),(OR,"attached","attaches","attaching","attach")),(AND,(OR,"fraud","frouds"),"intentionally"),(AND,"other","requests"),(AND,"account"),(AND,(NOT,"approves"),(NOT,"approved"),(NOT,"approve"),(NOT,"approving"),(OR,"rights","right"),(OR,"rescind","rescinded"),(OR,"rescinds","rescinding")),(AND,"loan","pretender"),(AND,(OR,"owner","owners"),(OR,"assignment","assignments"),(AND,(OR,"refer","referring"),(OR,"referred","refers"),(OR,"xxx sale","xxx sales"))),(AND,(NOT,"cares"),(NOT,"care"),(NOT,"updates"),(NOT,"update"),(AND,"transferred"),(AND,"cphb","cphb case"),(AND,(OR,"referencing","references"),(OR,"referenced","reference"),(OR,"act"),(AND,(NOT,"shorten"),(NOT,"short"),(NOT,"shortest"),(OR,"attaching","attaches"),(OR,"attach","attached"))),(AND,"nys","(OR,"record","records"))),(AND,(NOT,"th"),(cphb,"misconduct"),(AND,"latest correspondence"),(AND,"discharged"),(AND,(NOT,"care"),(NOT,"care"),(NOT,"updates"),(NOT,"update"),"all"),(OR,"xxx sends","xxx send"),(AND,"th"),(OR,"grounds","ground"),(also),((AND,(NOT,"nd"),(NOT,"th e"),(OR,"letter","letters"),(OR,"evidences"),(OR,"evidences"),(OR,"evidence","evidencing"))),(AND,(OR,"balance reduction","balance reduction"))),(AND,(NOT,"calls"),(OR,"call"),(NOT,"called"),(NOT,"th"),(OR,"information"),(OR,"informations"),(OR,"investigation"),(OR,"investigations"),(AND,(OR,"explain"),(OR,"explain"),(AND,(OR,"interest penalty"),(OR,"interest penalties"),(AND,"inadequately"),(AND,"hypothetical"),(AND,"never experienced anything"),(AND,"because"),(AND,"sheriff"),(AND,(OR,"theft"),(OR,"thief"),(AND,(NOT,"nd"),(OR,"letter"),(OR,"letters"),(OR,"credit"),(AND,(NOT,"postpone"),(NOT,"postpone"),(NOT,"postingponed"),(OR,"violates"),(OR,"violated"),(OR,"violates"),(OR,"violating"))),(AND,(NOT,"called"),(NOT,"call"),(NOT,"calling"),(NOT,"call"),(false),((AND,(NOT,"nd"),(NOT,"th e"),(NOT,"automate"),(NOT,"automated"),(OR,"letters"),(OR,"letter"),(OR,"deed"),(AND,"u.s."),(AND,"tree"),(AND,(OR,"xxx loan"),(OR,"xxx loans"))),(AND,(NOT,"shorter"),(NOT,"short"),(NOT,"shortest"),(OR,"attaches"),(OR,"attach","attached"),(OR,"attaching"))),(AND,(OR,"settlements"),(OR,"settlement"))),(AND,(OR,"frauds"),(OR,"fraud"))),(AND,(NOT,"automate"),(NOT,"automated"),(OR,"statement"),(OR,"statements")))]</pre>				

The screenshot shows the Textual Elements interface. On the left, a table lists strings with their roles and frequencies. On the right, a list of matched documents is shown, each with a preview, sentiment score (red), relevance score (90.000), and consumer status (N/A). The documents relate to mortgage complaints.

String	Role	Frequency
not	ADV	66522
> payment	N	57201
> loan	N	48174
> mortgage	N	42675
> pay	V	27315
> receive	V	25063
> make	V	24031
> tell	V	23703
> send	V	21707
no	ADV	21122

Document 1 of 11993

Highlight: Category matches Search matches

System-generated Boolean rules associated with the predefined category **consumer_disputed** = 'Yes' and the matched documents identified by this category rule are displayed above.

You can inspect the Boolean logic and, if necessary, edit the rules or extend them with additional operators and linguistic qualifiers.

- Right-click the promoted topic category **+payment, +interest, +principal, monthly, +amount** and rename it as **Payments**. Rerun the **Category** node.

The screenshot shows the Textual Elements interface. On the left, a table lists strings with their roles and frequencies. On the right, a list of matched documents is shown, each with a preview, sentiment score (red), relevance score (varies), and consumer status (Yes or No). The documents relate to mortgage payments and balances.

String	Role	Frequency
not	ADV	66522
> payment	N	57201
> loan	N	48174
> mortgage	N	42675
> pay	V	27315
> receive	V	25063
> make	V	24031
> tell	V	23703
> send	V	21707
no	ADV	21122

Document 1 of 5427

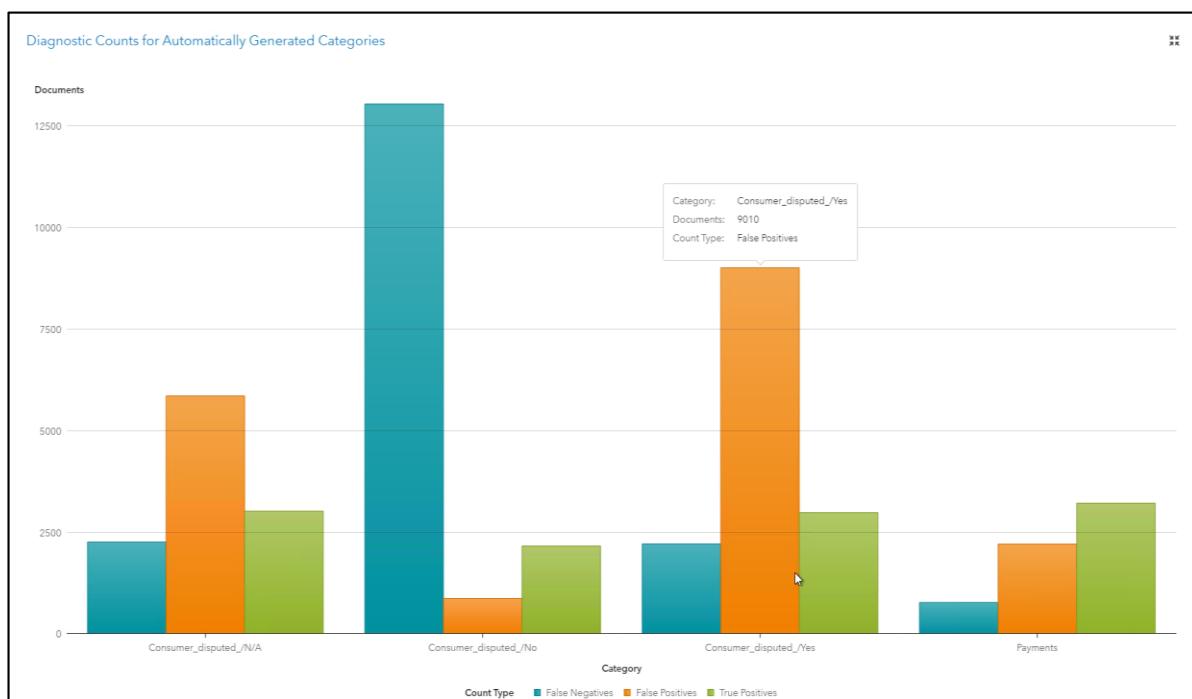
Highlight: Category matches Search matches

System-generated Boolean rules associated with the promoted category **Payments** and the matched documents identified by this category rule are displayed above.

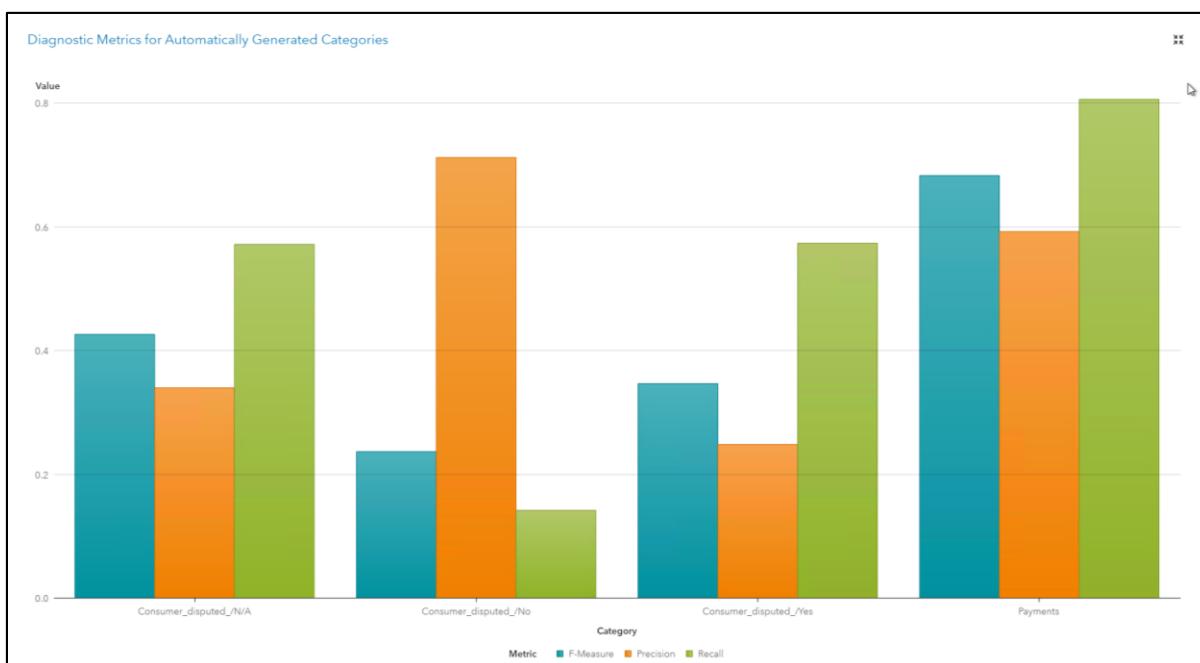
- Using the button on the top right of the window, close the **Category** node.

5. To examine Category node results, right-click the **Category** node and open the Results window. The window displays two charts:

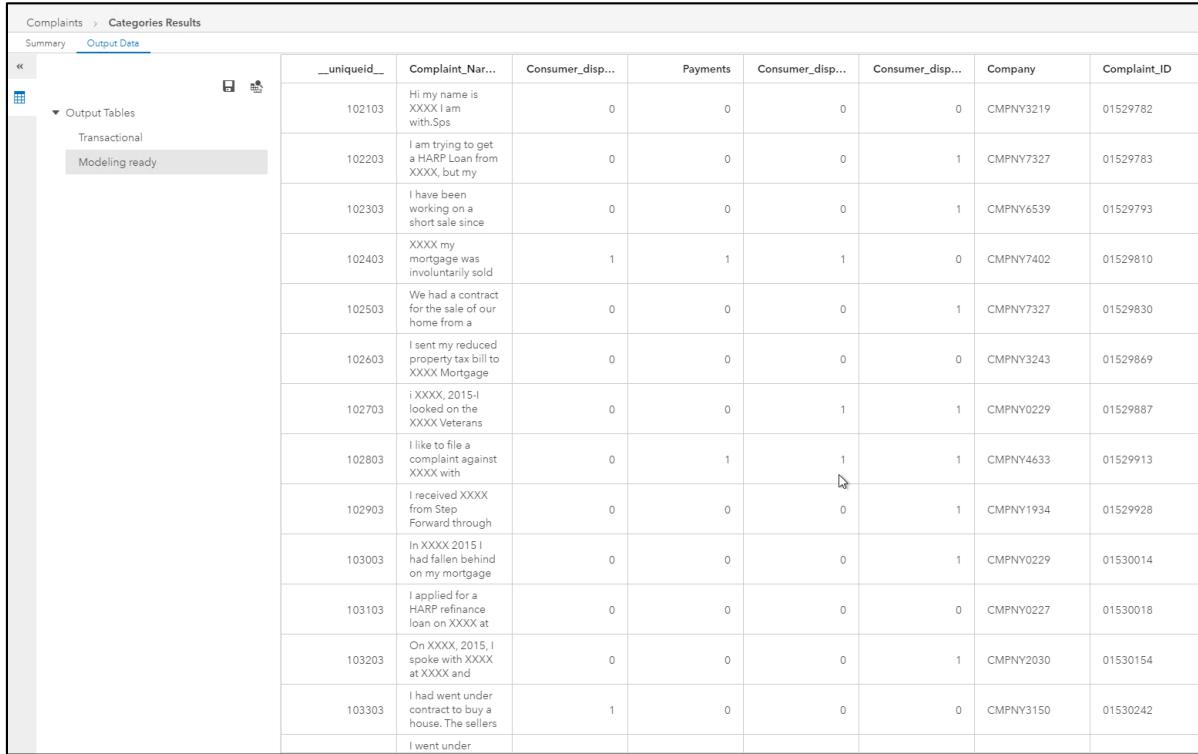
- Diagnostic counts (False negative, False positive, and True positive) for the predefined and topic-promoted category are displayed on the first chart.



- Diagnostic metrics (precision, recall, and F-measures) for the predefined and topic-promoted category are displayed on the second chart.



The Category Results window also displays the Modeling ready output table that contains classification results.



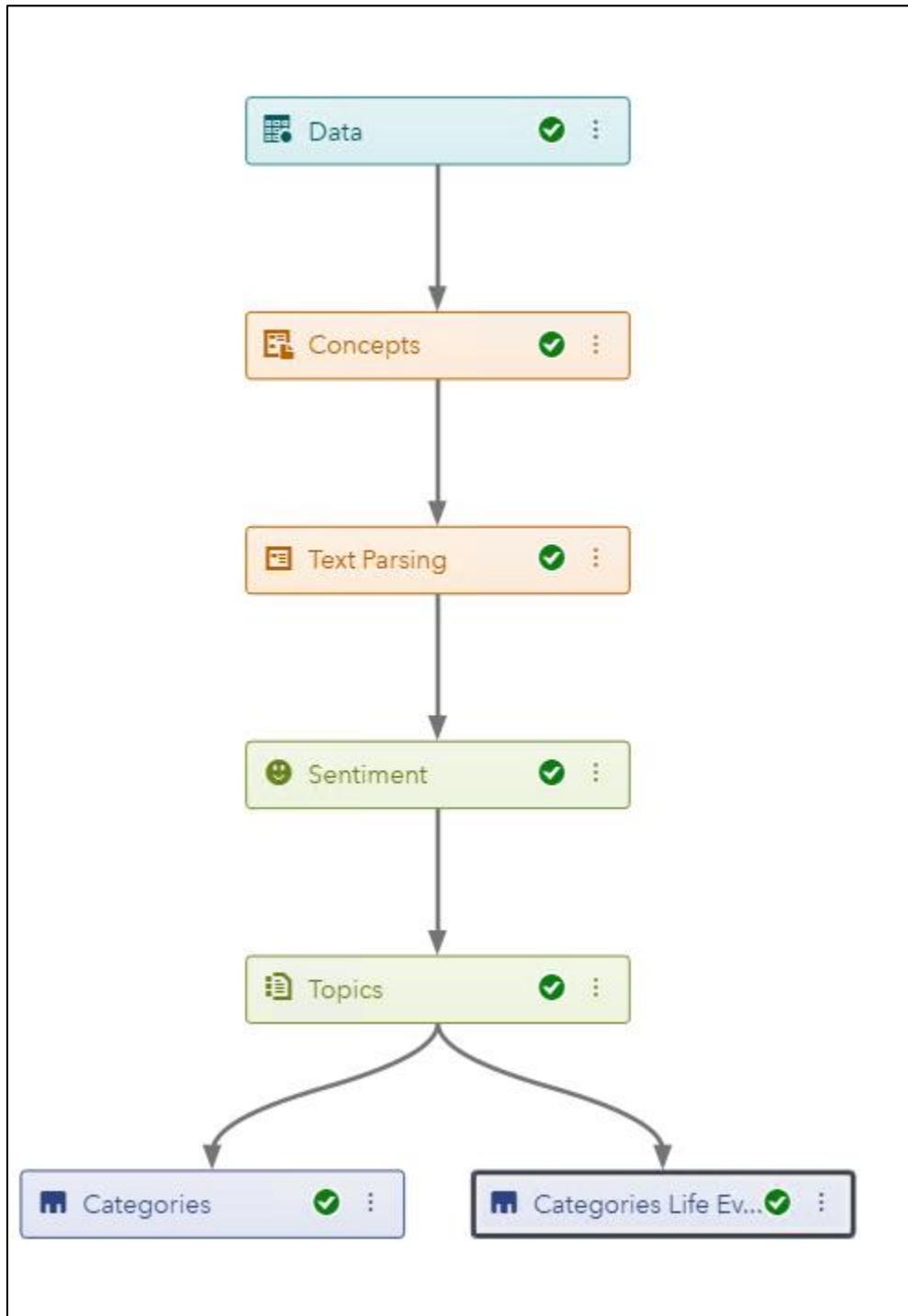
The screenshot shows the 'Categories Results' window in SAS Visual Analytics. At the top, there are tabs for 'Summary' and 'Output Data'. Below the tabs, there's a sidebar with a tree view under 'Output Tables': 'Transactional' is collapsed, and 'Modeling ready' is expanded, showing several rows of data. The main area displays a table with the following columns: __uniqueid__, Complaint_Narrative, Consumer_dissatisfied, Payments, Consumer_dissatisfied, Consumer_dissatisfied, Company, and Complaint_ID. The data consists of 12 rows, each representing a consumer complaint with its unique ID, narrative, and various dissatisfaction and payment counts, along with the company name and ID.

__uniqueid__	Complaint_Narrative	Consumer_dissatisfied	Payments	Consumer_dissatisfied	Consumer_dissatisfied	Company	Complaint_ID
102103	Hi my name is XXXX I am with Sps	0	0	0	0	CMPNY3219	01529782
102203	I am trying to get a HARP Loan from XXXX, but my	0	0	0	1	CMPNY7327	01529783
102303	I have been working on a short sale since	0	0	0	1	CMPNY6539	01529793
102403	XXXX my mortgage was involuntarily sold	1	1	1	0	CMPNY7402	01529810
102503	We had a contract for the sale of our home from a	0	0	0	1	CMPNY7327	01529830
102603	I sent my reduced property tax bill to XXXX Mortgage	0	0	0	0	CMPNY3243	01529869
102703	i XXXX, 2015 I looked on the XXXX Veterans	0	0	1	1	CMPNY0229	01529887
102803	I like to file a complaint against XXXX with	0	1	1	1	CMPNY4633	01529913
102903	I received XXXX from Step Forward through	0	0	0	1	CMPNY1934	01529928
103003	In XXXX, 2015 I had fallen behind on my mortgage	0	0	0	1	CMPNY0229	01530014
103103	I applied for a HARP refinance loan on XXXX at	0	0	0	0	CMPNY0227	01530018
103203	On XXXX, 2015, I spoke with XXXX at XXXX and	0	0	0	1	CMPNY2030	01530154
103303	I had went under contract to buy a house. The sellers	1	0	0	0	CMPNY3150	01530242
	I went under						

- Using the button at the top right of the window, close the **Category** node.

Exploring Documents Associated with Life Events Categories Using Custom Category Boolean Rules

1. Go to the pipeline.
2. Right-click **Categories Life Events** and click **Open**.



Category rules and matched documents associated with the two of the life events , Natural Disaster and Career Trouble, are displayed next.

Categories

- All Categories (14)
 - NaturalDisaster
 - Retirement
 - HealthIssues
 - CareerTrouble
 - Children
 - Education

Textual Elements (22172)

String	Role	Fre...
not	ADV	66522
payment	N	57201
loan	N	48174
mortgag	N	42675

Edit a Category

```
(OR,"hurricane@","tornado@","flood@","water damage@","tree")
```

Code is valid.

Documents

Complaint_Narrative	Sentiment	Relevancy	Consumer_dis...
...serviced by Green Tree Servicing XXXX (Green Tree). I believe that errors have occurred concerning Green Tree 's servicing of my loan, and my belief has been confirmed in various telephone...	:(28.000	No
...with a Standard Flood Hazard Determination Form. Mv Mortgaoe			

Categories

- All Categories (14)
 - NaturalDisaster
 - Retirement
 - HealthIssues
 - CareerTrouble
 - Children
 - Education
 - FinancialTrouble
 - HomeProperty

Textual Elements (22172)

String	Role	Fre...
not	ADV	66522
payment	N	57201
loan	N	48174
mortgag	N	42675

Edit a Category

```
(OR,(DIST_5,(OR,"lose@","laid off","fired","down-sized","down sized","out of","looking for","unable to"),(OR,"job","career","work")),"unemployed","unemploy","unemployment")
```

Code is valid.

Documents

Complaint_Narrative	Sentiment	Releva...	Consumer_disputed...
...aware I was unable to work and asked for an FHA XXXX due to my XXXX. The representative on the phone told me that they would not provide any documents to me to file the forbearance because I was...	:(16.000	N/

All the Life Events category rules used in this project are provided below.

NaturalDisaster: (OR,"hurricane@","tornado@","flood@","water damage@","tree")

Retirement: (OR,"retire@","retirement@")

HealthIssues:

(OR,"medical","illness","sickness","medicine","disease@","hospital@","healthcare","health care","hospice","nursing home@","nurse@","doctor@","radiation treatment","assisted living","medicaid")

CareerTrouble: (OR,(DIST_5,(OR,"lose@","laid off","fired","down-sized","down sized","out of","looking for","unable to"),(OR,"job","career","work")),"unemployed","unemploy","unemployment")

Children: (OR,"baby@","newborn","new born","child@","kid@","son@","daughter@","grandchild@","grandson","granddaughter")

Education:

(OR,"tuition", (DIST_5,(OR,"college","university","school@")), (OR,"loan@","expense@","cost@")))

FinancialTrouble: (OR,(ORDDIST_2,(OR,"need@"), (OR,"money","cash","fund@N")), "bankrupt@", "bankruptcy","insufficient income", "late on payment@","behind on payment@","chapter11","chapter 11","insolvency@","late fee@","debt","loss mitigation")

HomeProperty: (OR,"home","house","property","condo","rental property","apartment")

Vehicle: (SENT,(OR,"buy@","purchase@","get@","new"), (OR,"car@","truck@","automobile@","vehicle@","van@","auto"))

Modify a Previously Created Custom Category Rule

1. Select the **NaturalDisaster** category and update the rule to include a new term such as *earthquake* and refine the usage of the term *tree* as follows:

```
(OR, "hurricane@", "earthquake", "tornado@", "flood@", "water damage@", (SENT, (OR, "fell@", "fall@", "damage*"), "tree"))
```

2. Run the node and examine matched documents. Close the node.

The screenshot shows the SAS Model Studio interface. In the top navigation bar, it says 'Model Studio - Build Models' and 'student'. The main area has a 'Categories' pane on the left listing 'All Categories (14)' with 'NaturalDisaster' selected. To the right is an 'Edit a Category' pane containing the code block shown above. Below these are 'Textual Elements (22172)' and a preview of a document's narrative.

String	R...	...
not	AD V	665 22
payment	N	572 01
loan	N	481 74

Documents Test Sample Text

All (25681) Matched (671 of 25681) Search

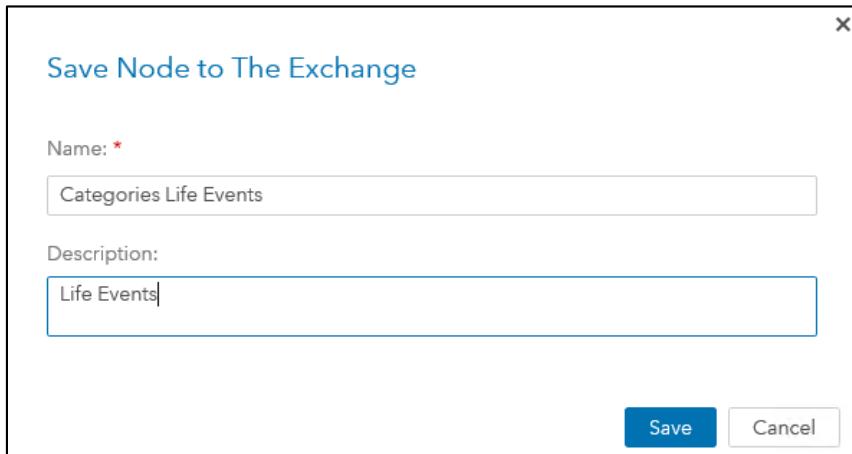
Complaint_Narrative Sentiment Releva... Consumer_disputed_

...with a Standard **Flood** Hazard Determination Form. My Mortgage has been sold several times over the past 2 years, and is XXXX with Loan Care. My home owner's insurance is with XXXX and **Flood**...

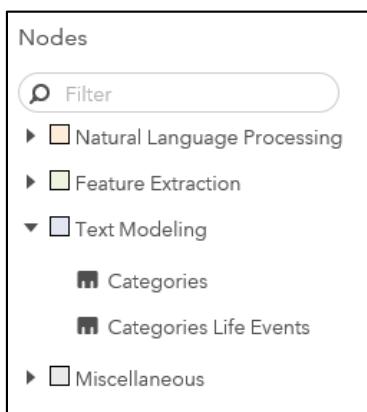
...to increase my **flood** insurance to match the coverage of my

Saving the New Life Events Category Node Taxonomy to the Exchange

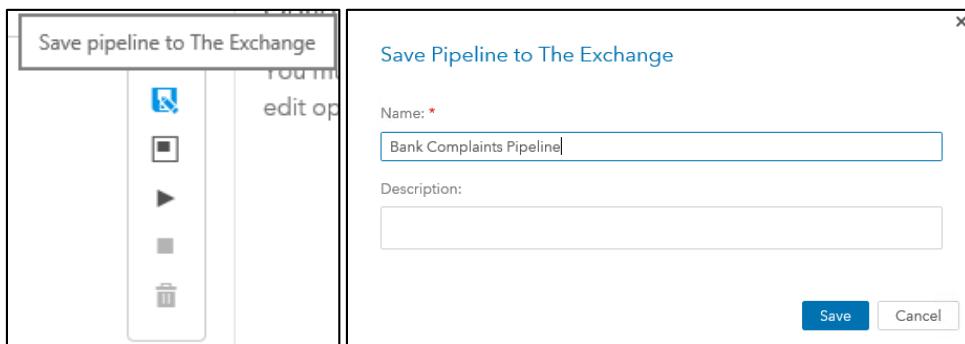
1. Right-click the **Life Events** category node and click **Save** to save the new taxonomy to the Exchange.



This saved node shows up as a selection in the Nodes pane.



2. Save the pipeline in the Exchange by clicking the **Save** icon above the Run pipeline icon.



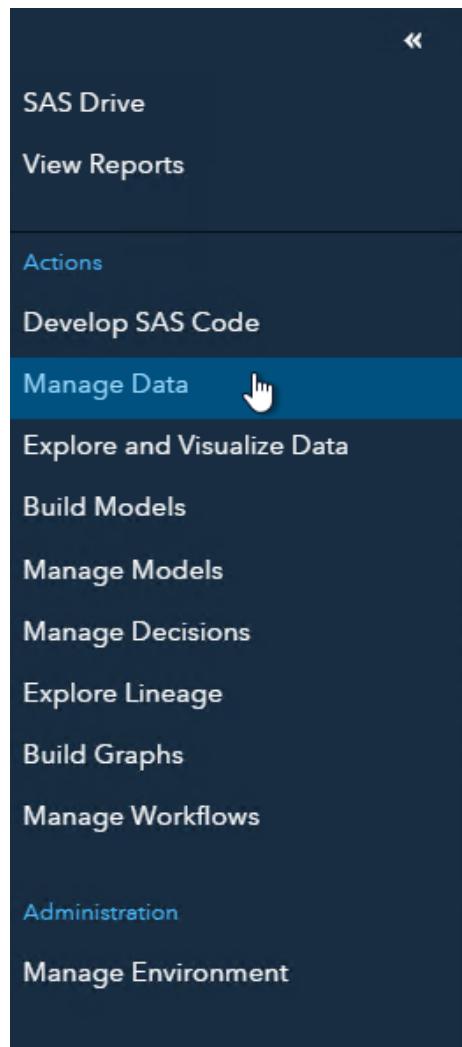
Model Deployment

The next step is the deployment of models. You can score a new data set using the model score code from Visual Text Analytics models.

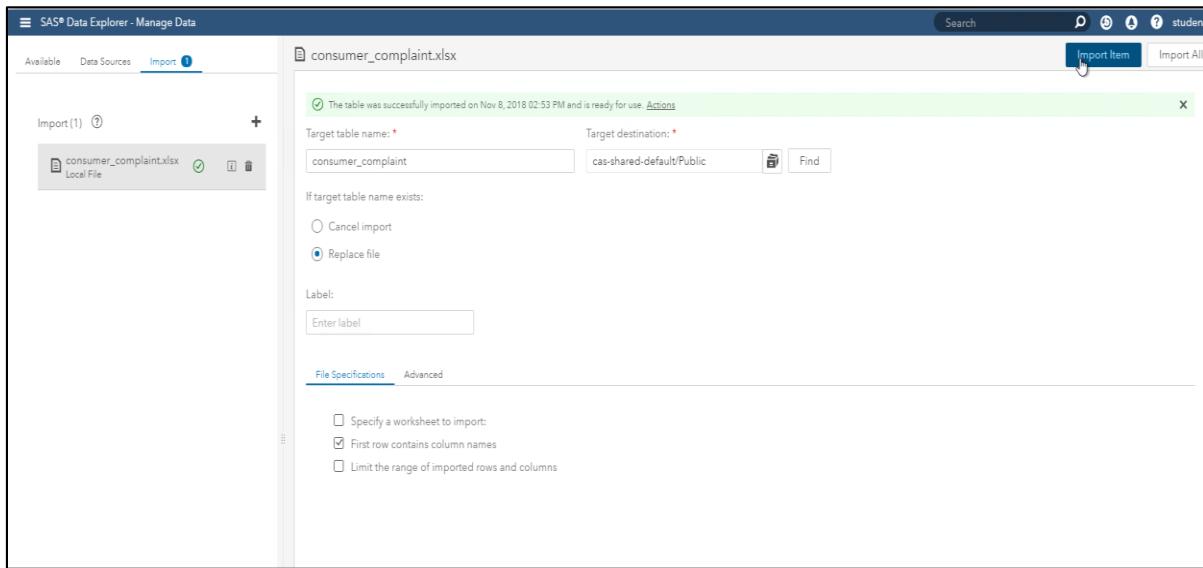
SAS Cloud Analytics Services (or CAS) score code is generated for four types of models:

- Concepts, using the action set textRuleScore
- Sentiment, using the action set sentimentAnalysis
- Topics, using the action set astore
- Categories, also using the action set textRuleScore

The first step in scoring is preparing the scoring data and making it available to CAS.



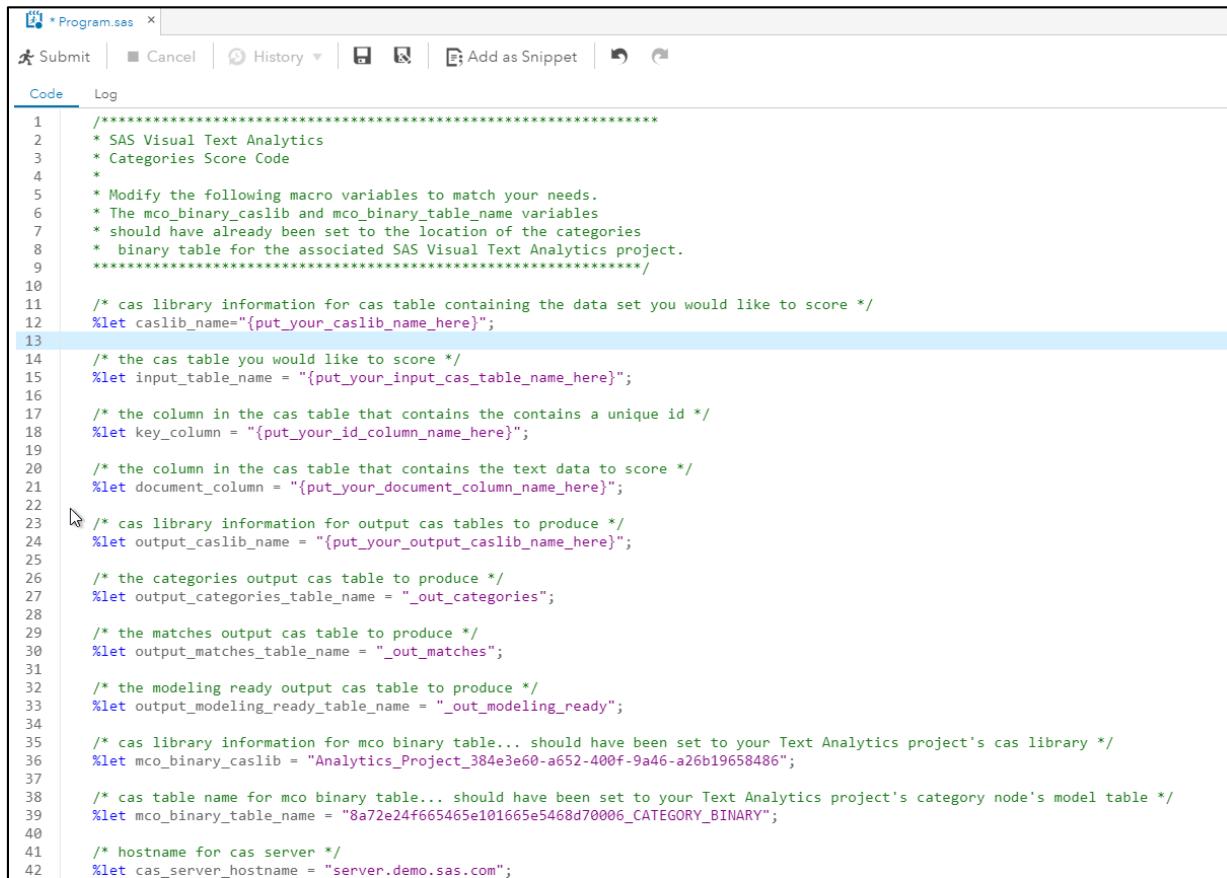
1. Click **Manage Data**. Import the new score data (consumer_complaint.xls) and make it available to CAS.



2. The scoring code for the Category node is included in the results. Right-click the **Life Events** Category node and select **Results**.
3. Copy the score code in the Results pane.
4. Open SAS Studio by clicking **Develop SAS Code** and then open a new Code Editor.



5. Open a new program file and paste the score code from the Visual Text Analytics Category node results window.



```

1  ****SAS Visual Text Analytics
2  * Categories Score Code
3
4
5  * Modify the following macro variables to match your needs.
6  * The mco_binary_caslib and mco_binary_table_name variables
7  * should have already been set to the location of the categories
8  * binary table for the associated SAS Visual Text Analytics project.
9  ****
10
11 /* cas library information for cas table containing the data set you would like to score */
12 %let caslib_name="{put_your_caslib_name_here}";
13
14 /* the cas table you would like to score */
15 %let input_table_name = "{put_your_input_cas_table_name_here}";
16
17 /* the column in the cas table that contains the contains a unique id */
18 %let key_column = "{put_your_id_column_name_here}";
19
20 /* the column in the cas table that contains the text data to score */
21 %let document_column = "{put_your_document_column_name_here}";
22
23 /* cas library information for output cas tables to produce */
24 %let output_caslib_name = "{put_your_output_caslib_name_here}";
25
26 /* the categories output cas table to produce */
27 %let output_categories_table_name = "_out_categories";
28
29 /* the matches output cas table to produce */
30 %let output_matches_table_name = "_out_matches";
31
32 /* the modeling ready output cas table to produce */
33 %let output_modeling_ready_table_name = "_out_modeling_ready";
34
35 /* cas library information for mco binary table... should have been set to your Text Analytics project's cas library */
36 %let mco_binary_caslib = "Analytics_Project_384e3e60-a652-400f-9a46-a26b19658486";
37
38 /* cas table name for mco binary table... should have been set to your Text Analytics project's category node's model table */
39 %let mco_binary_table_name = "8a72e24f665465e101665e5468d70006_CATEGORY_BINARY";
40
41 /* hostname for cas server */
42 %let cas_server_hostname = "server.demo.sas.com";

```

6. Update the necessary fields in the score code to point to the location of the new data set and the output library for the results. Modify the remaining macro variable definitions as described in the comments in the Code Editor.

The modified macro parameters are presented below.

```

1  ****
2  * SAS Visual Text Analytics
3  * Categories Score Code
4  *
5  * Modify the following macro variables to match your needs.
6  * The mco_binary_caslib and mco_binary_table_name variables
7  * should have already been set to the location of the categories
8  * binary table for the associated SAS Visual Text Analytics project.
9  ****
10 /*
11  * cas library information for cas table containing the data set you would like to score */
12 %let caslib_name="Public";
13
14 /* the cas table you would like to score */
15 %let input_table_name = "Consumer_Complaint";
16
17 /* the column in the cas table that contains the contains a unique id */
18 %let key_column = "Complaint_Id";
19
20 /* the column in the cas table that contains the text data to score */
21 %let document_column = "Complaint_Narrative";
22
23 /* cas library information for output cas tables to produce */
24 %let output_caslib_name = "Public";
25
26 /* the categories output cas table to produce */
27 %let output_categories_table_name = "Comp_out_categories";
28
29 /* the matches output cas table to produce */
30 %let output_matches_table_name = "Comp_out_matches";
31
32 /* the modeling ready output cas table to produce */
33 %let output_modeling_ready_table_name = "Comp_out_modeling_ready";
34
35 /* cas library information for mco binary table... should have been set to your Text Analytics project's cas library */
36 %let mco_binary_caslib = "Analytics_Project_384e3e60-a652-400f-9a46-a26b19658486";
37
38 /* cas table name for mco binary table... should have been set to your Text Analytics project's category node's model table */
39 %let mco_binary_table_name = "8a72e24f665465e101665e5468d70006_CATEGORY_BINARY";
40
41 /* hostname for cas server */
42 %let cas_server_hostname = "server.demo.sas.com";

```

- Click the **Submit** button to execute the scoring code.

Three output data sets, **_Out_Categories**, **_Out_matches**, and **_Out_modeling_ready**, are generated during scoring and are stored in CAS.

CAS Library	Name	Label	Number of Rows	Number of Columns
Public	Comp_out_categories		5389	4
Public	Comp_out_matches		25257	5
Public	Comp_out_modeling_ready		2627	14

The resulting three tables are stored in CAS and can be used in operations or analyzed further in SAS Visual Analytics, SAS Visual Statistics, or SAS Visual Data Mining and Machine Learning.

End of Demonstration

Appendix A Cheat Sheet Template: Concept and Category Rules

A.1	Introduction	A-3
A.2	Sample Concept Rules.....	A-4
A.3	Sample Category Rules	A-7

A.1 Introduction

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 concept rule. For example, you can create a custom concept named **SAS Output Delivery System, (ODS)**, and then write a rule that extracts all the documents in your document set that contain the word *ODS*. In other words, all the documents that are displayed for the concept *SAS Output Delivery System* would contain *ODS*. Each document is evaluated separately for concept matches. Matches do **not** span documents.

Facts (also called *predicates*) are related pieces of information in text that are located and matched. Facts can be identified within a custom concept. 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.

For information about editing rules when you use the interface and when you use property settings, see “Concepts Page” in the SAS Visual Text Analytics Help documentation. For a list of rule types, see “Which Rule Type Should I Use?” on page 79 in *SAS® Visual Text Analytics 8.3: User’s Guide*.

A.2 Sample Concept Rules

Here are examples of sample Concept LITI rules. For more information about writing concept rules, see “Writing Concept Rules: Basic LITI Syntax” in the online Help documentation or in SAS® *Visual Text Analytics 8.3: User’s Guide* on page 77.

Test Document	Rule Types	Rule Capabilities	Rule Syntax	Matched Words
web report studio Web Report Studio	CLASSIFIER	To match specific word or strings	CLASSIFIER: web report studio	web report studio Web Report Studio
Visual Analytics Visual Statistics	CONCEPT	To match any following word	CONCEPT: visual _w _w must be in lowercase.	Visual Analytics Visual Statistics
Predictive Analytics Predictive analytics PREDICTIVE MODELING	CONCEPT	To match any following word that begins with an uppercase letter	CONCEPT: Predictive _cap The text _cap must be in lowercase.	Predictive Analytics Predictive analytics PREDICTIVE MODELING
Business Analytics predictive analytics PREDICTIVE ANALYTICS	CONCEPT	To match any previous word	CONCEPT: _w analytics	Business Analytics predictive analytics PREDICTIVE ANALYTICS

Test Document	Rule Types	Rule Capabilities	Rule Syntax	Matched Words
SAS introduced the Output Delivery System (ODS) with Version 7, making output much more flexible. We show some examples using ODS here.	CONCEPT_RULE	To match an abbreviation and expanded term in the document	CONCEPT_RULE: (OR, "_c{ODS}", "_c{Output Delivery System}")	SAS introduced the Output Delivery System (ODS) with Version 7, making output much more flexible. We show some examples using ODS here.
SAS Visual Text Analytics uses concepts, terms, topics to categorize document collection. It is not prediction based. SAS Visual Text Analytics uses concepts, terms, topics to categorize document collection.	CONCEPT_RULE	To match several terms if there is no specific term	CONCEPT_RULE: (AND, (OR, "_c{concepts}", "_c{topics}", "_c{terms}"), (NOT, "Prediction"))	SAS Visual Text Analytics uses concepts, terms, topics to categorize document collection. It is not prediction based. SAS Visual Text Analytics uses concepts, terms, topics to categorize document collection.
SAS Visual Text Analytics is a product from the SAS Text Analytics family. Visual Text Analytics uses machine learning and subject matter expertise to define categorization and extraction models.	C_CONCEPT	To match a specific term (Text Analytics) when it follows a specific term (SAS)	C_CONCEPT: SAS _c{Text Analytics}	SAS Visual Text Analytics is a product from the SAS Text Analytics family. Visual Text Analytics uses machine learning and subject matter expertise to define categorization and extraction models.

Test Document	Rule Types	Rule Capabilities	Rule Syntax	Matched Words
SAS Visual Text Analytics 14.1 uses concepts, terms, topics to categorize document collection. It is not prediction based.	REGEX	To match a number that uses decimal notation , such as 14.1, 9.4, and 13.1	REGEX: [0-9][0-9],\.[0-9]+[0-9]	SAS Visual Text Analytics 14.1 uses concepts, terms, topics to categorize document collection. It is not prediction based.
In the data mining section several Enterprise Miner and text miner papers were also presented.	PREDICATE_RULE	To extract a fact associated with two sets (section and topic) of terms	PREDICATE_RULE: (section, topic):(AND, "_section{data mining}", "_topic{enterprise miner}")	In the data mining section, several Enterprise Miner and text miner papers were also presented.
SAS introduced the Output Delivery System (ODS) with Version 7, making output much more flexible. We show some examples using ODS here.	C_CONCEPT with COREF	<p>Use the coreference operator (_ref) when you want to link pronouns and other words with the canonical form (full form) of the terms that they reference.</p> <p>> (Multiple matches)</p> <p>Locates multiple instances of a match that is specified by the coreference operator (_ref).</p>	C_CONCEPT: _c{Output Delivery System }, _ref{ODS}>	SAS introduced the Output Delivery System (ODS) with Version 7, making output much more flexible. We show some examples using ODS here.

A.3 Sample Category Rules

Category rules (Boolean rules) resolve to *true* or *false*. *True* results in a match. Boolean rules use operators, arguments, and modifiers to define the conditions that are necessary for category matches. Category rules are simpler to write than LTI rules and are recommended when there is no need to extract specific information from the data.

General Rules for Syntax:

- Boolean operators are enclosed in ***parentheses*** and separated by ***commas***. Strings within arguments are included in quotation marks (" "). Example: **(AND, "SCA", "Text Miner")**
- Rules can be nested. Example: **(AND, (OR, "courage", "courageous"), (OR, "brave", "bravery"))**
- Concept names can be referenced in category rules. If you reference a concept name, all concept matches also match in the category. Concept names must be enclosed in braces ([]) and quotation marks. For example, to reference the concept BUSINESS_INTELLIGENCE in a category rule, you could write the rule **(OR, "[BUSINESS_INTELLIGENCE]")**.

Here are examples of sample Boolean category rules. For information about writing Boolean rules, see “Writing Category Rules: Boolean Rules” in the online Help documentation.

- Subcategory: equation:
 - **(AND, (OR, "equation", "equations"))**
- Subcategory: measure & relationship:
 - **(AND, (OR, "measuring", "measures", "measured", "measure"), (OR, "relationship", "relationships"))**
- Subcategory: text & ~create & ~ods:
 - **(AND, "text", (NOT, "odds"), (NOT, "ods"), (NOT, "creating"), (NOT, "create"), (NOT, "creates"), (NOT, "created"))**
- Subcategory: Business Intelligence or Enterprise Miner ~ (Enterprise Guide, DDE, Drug):
 - **(AND,(NOT,(OR,"Enterprise Guide","DDE","Drug")), (OR,"Business Intelligence","Enterprise Miner"))**

To specify the previously defined concept rule Business Intelligence (BUSINESS_INTELLIGENCE), use the following:

(OR, "[BUSINESS_INTELLIGENCE]")

Appendix B References

B.1 References	B-3
----------------------	-----

B.1 References

- Albright, Russell. 2004. SAS Institute white paper. "Taming Text with the SVD." Available <http://ftp.sas.com/techsup/download/EMiner/TamingTextwiththeSVD.pdf>.
- Albright, R., J.A. Cox, and K. Daly. 2001. "Skinning the Cat: Comparing Alternative Text Mining Algorithms for Categorization." *Proceedings of the 2nd Data Mining Conference of DiaMondSUG*. Chicago, IL. DM Paper 113.
- Allan, E., M. Horvath, C. Kopek, B. Lamb, T. Whaples, and M. Berry. 2008. "Anomaly detection using nonnegative matrix factorization." In M. Berry and M. Castellanos, Editors, *Survey of Text Mining II: Clustering, Classification, and Retrieval*. pages 203–218. Springer-Verlag London Limited.
- Berry, Michael W., and Murray Browne. 2005. *Understanding Search Engines: Mathematical Modeling and Text Retrieval*. Philadelphia: SIAM.
- Cassell, David L. 2007. "The Basics of the PRX Functions." *Proceedings of the SAS Global Forum 2007 Conference*. Cary, NC: SAS Institute Inc. Available <http://www2.sas.com/proceedings/forum2007/223-2007.pdf>.
- Chakraborty, Goutam, Murali Pagolu, and Satish Garla. 2013. *Text Mining and Analysis: Practical Methods, Examples, and Case Studies Using SAS®*. Cary, NC: SAS Institute Inc.
- Cherniak, Eugene. 1993. *Statistical Language Learning*. Cambridge, Massachusetts: The MIT Press.
- Evangelopoulos, Nicholas, Xiaoni Zhang, and Victor R. Prybutok. 2010. "Latent Semantic Analysis: Five methodological recommendations." *European Journal of Information Systems*. pages 1–17.
- Jurafsky, Daniel, and James H. Martin. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Upper Saddle River, New Jersey: Prentice Hall.
- Kiefer, Manfred. 2012. *SAS® Encoding: Understanding the Details*. Cary, NC: SAS Institute Inc.
- Langville, Amy N., and Carl D. Meyer. 2006. *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton, New Jersey: Princeton University Press.
- Manning, Christopher D., and Hinrich Schütze. 2002. *Foundations of Statistical Natural Language Processing*. Cambridge, Massachusetts: The MIT Press.
- Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. New York: Cambridge University Press.
- Miner, Gary, Dursun Delen, John Elder, Andrew Fast, Thomas Hill, and Robert A. Nisbet. 2012. *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*. Waltham, Massachusetts: Academic Press.
- Quigley, Ellie. 1998. *PERL by Example*. Upper Saddle River, New Jersey: Prentice Hall PTR.
- SAS Institute Inc. 2018. *SAS® Visual Text Analytics 8.3: User's Guide*. Cary, NC: SAS Institute Inc. Available http://go.documentation.sas.com/?docsetId=ctxtug&docsetTarget=titlepage.htm&docsetVersion=8.3&local_e=en.
- Shannon, C.E. 1948. "A Mathematical Theory of Communication." *Bell System Technical Journal*. Vol. 27. pages 379–423 and 623–656.
- Strang, Gilbert. 1993. "The Fundamental Theorem of Linear Algebra." *The American Mathematical Monthly*. 100:9. pages 848–855.
- Thisted, Ronald A. 1988. *Elements of Statistical Computing*. New York: Chapman and Hall.

Wakefield, Todd. August 6, 2004. "A Perfect Storm is Brewing: Better Answers are Possible by Incorporating Unstructured Data Analysis Techniques." *Information Management*. Available <https://www.information-management.com/news/a-perfect-storm-is-brewing-better-answers-are-possible-by-incorporating-unstructured-data-analysis-techniques>.

Wicklin, Rick. 2010. *Statistical Programming with SAS/IML Software*. Cary, NC: SAS Institute Inc.