TUTORIAL H

Extracting Knowledge from Published Literature Using RapidMiner

Dursun Delen, PhD
Spears School of Business, Oklahoma State University

CONTENTS

INTRODUCTION

The main purpose of this tutorial is to illustrate the text mining capabilities of RapidMiner's text analytics extension using an easily understandable example data set. In fact, the details of the data set, which was used to conduct a similar study with a different set of software tools, can be found in Delen and Crossland (2008).

MOTIVATION

Researchers as well as practitioners conducting reviews of the existing body of knowledge published in literature are facing an increasingly complex and voluminous task. With the increasing wealth of potentially significant research reported in ever increasing numbers of publication outlets (sometimes in related fields and sometimes in what is traditionally deemed as "nonrelated" fields of particular domain of study), the researcher's task is ever more daunting if a thorough job is desired.

In this tutorial, we illustrate a method to assist and enhance the efforts of researchers in this situation by enabling a semiautomated analysis of large volumes of unstructured data (in the form of published journal articles) through the application of text mining. By accessing the extensive number of

Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications. DOI: 10.1016/B978-0-12-386979-1.00016-5

abstracts that are available online, herein we detail how one can use the text mining capabilities of RapidMiner, a free and open source data mining software, to analyze related research.

Using standard digital libraries and online publication search engines, we downloaded and collected all of the available articles for the three major journals in the field of information systems: *MIS Quarterly* (MISQ), *Information Systems Research* (ISR), and the *Journal of Management Information Systems* (JMIS). For each article, we extracted its title, abstract, author list, published keywords, volume, issue number, and year of publication. Also included in the data set was a field that designated the journal type of each article to serve for future pattern analysis. At the end, 901 articles were included in the corpus of their study.

In our text mining part of the analyses, we chose to use only the abstract of an article as the only source of information. We have not included the title or the keywords of the article for two main reasons: Under normal circumstances, the abstract would already include the listed keywords, and therefore inclusion of the listed key words for the analysis would mean repeating the same information and potentially giving them unmerited weight; and the listed keywords may be terms that authors would like their article to be associated with (as opposed to what is really contained in the article), therefore potentially introducing unquantifiable bias to the analysis of the content. We adopted the three-step process (see Figure H.1) of text mining (described in detail in Chapter 5) to execute the text mining project explained in this tutorial.

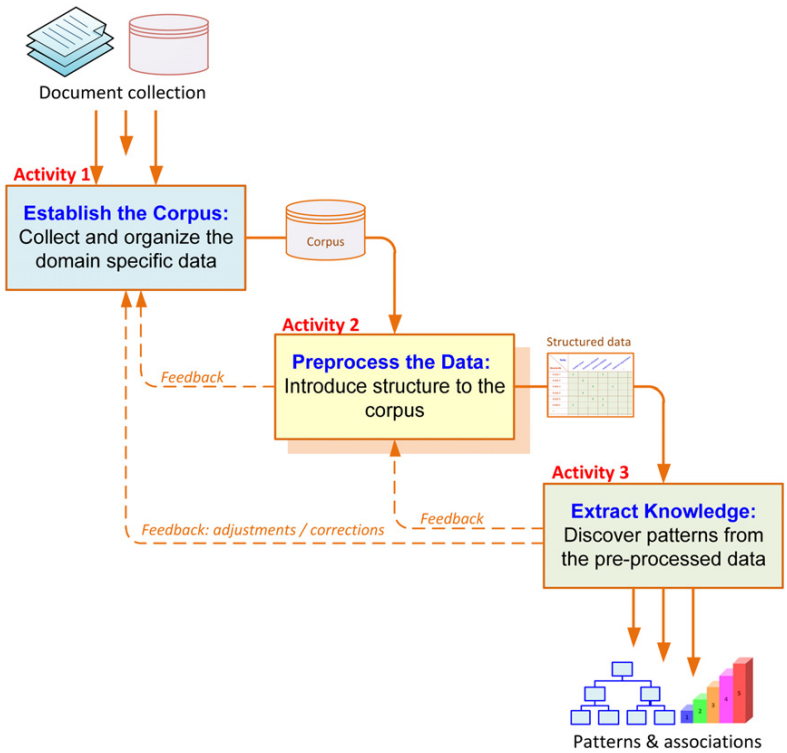


FIGURE H.1
The three-step process for text mining.

A BRIEF INTRODUCTION TO RAPIDMINER

RapidMiner is a free of charge, open source software tool for data and text mining. In addition to Windows operating systems, RapidMiner also supports Macintosh, Linux, and Unix systems. It is available as a stand-alone application for data/text analysis and as a data/text mining engine for the integration into your own products. Thousands of applications of RapidMiner in more than 40 countries are successfully developed to give its users a competitive edge.

The RapidMiner software tool, along with its extensions (including text analytics extension) and documentation, can be found and downloaded from www.rapid-i.com. Once the proper version of the tool is downloaded and installed, it can be used for a variety of data and text mining projects. Its graphical user interface is a little different from the ones we often see in other commercial data mining tools, such as IBM SPSS Modeler, SAS Enterprise Miner, and *STATISTICA* Data Miner. Such differences may lead to a longer learning curve, but once understood it is quite logical and informative.

When the RapidMiner tool is first started, the user is asked to specify a repository; either connect to a remote repository or create a new local repository (see Figures H.2 and H.3). A repository in RapidMiner is a central storage mechanism for all project-related files (processes, models, outputs, etc.).

FIGURE H.2
Specifying a repository for managing a project in RapidMiner.

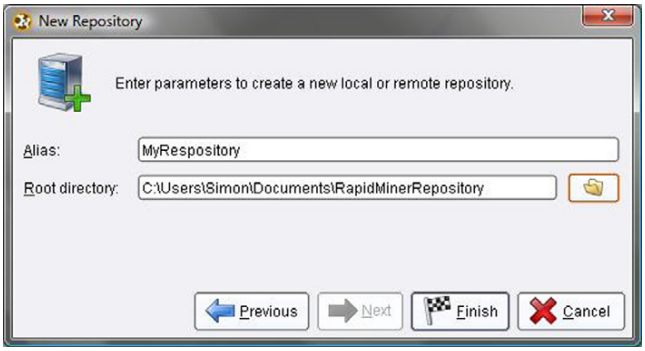


FIGURE H.3
Creating a local repository on your computer the first time you start the program.

Once the local repository and the project name are both specified, you will be forwarded into the so-called Welcome window (Figure H.4). There, the lower section shows current news about RapidMiner if you have an Internet connection. The list in the center of the window shows the analysis processes recently worked on. Users can choose to open one of the recent processes or create a brand new one.

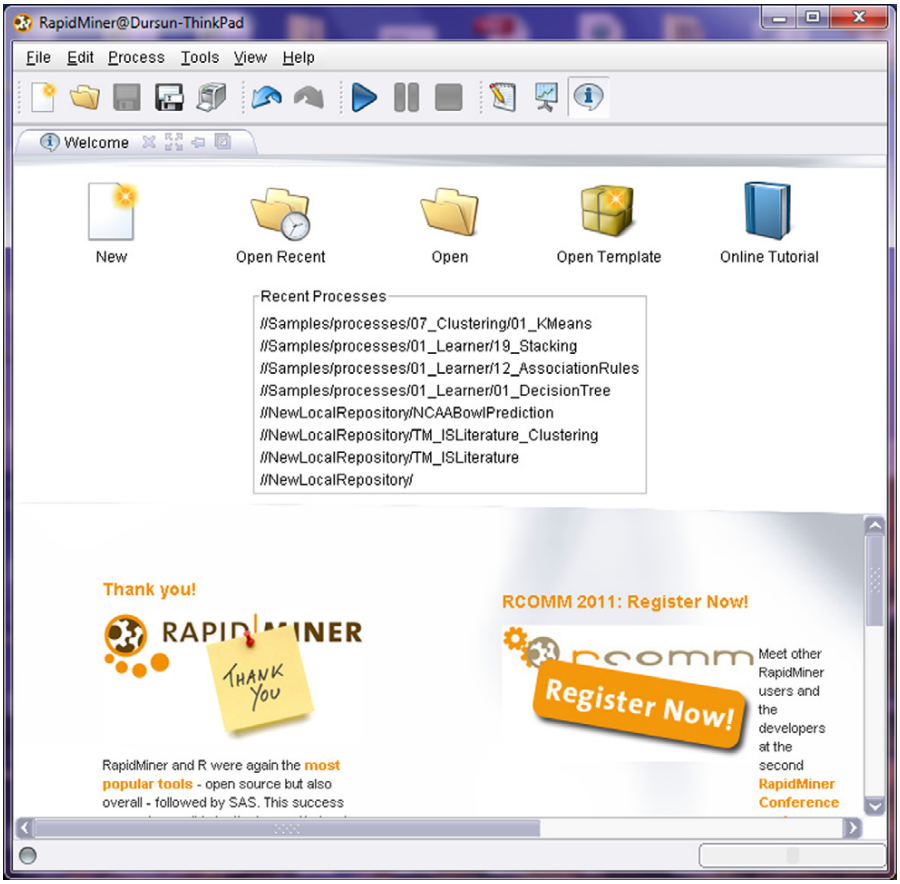


FIGURE H.4
RapidMiner's Welcome window.

In this window the user can open a recently created project, open an existing project from a file, or start a new project either from scratch or by using a prebuilt template.

TEXT ANALYTICS IN RAPIDMINER

In order to use RapidMiner for this text mining project, we first need to make sure that our version of RapidMiner includes the Text Processing add-on extension. You can check to see what extensions are already installed on your RapidMiner by clicking "Help" and then selecting "Manage

Extensions" (Figure H.5). In the pop-up window you will see all of the installed extensions with their current versions (Figure H.6). In this interface you can deactivate or uninstall already installed extensions.

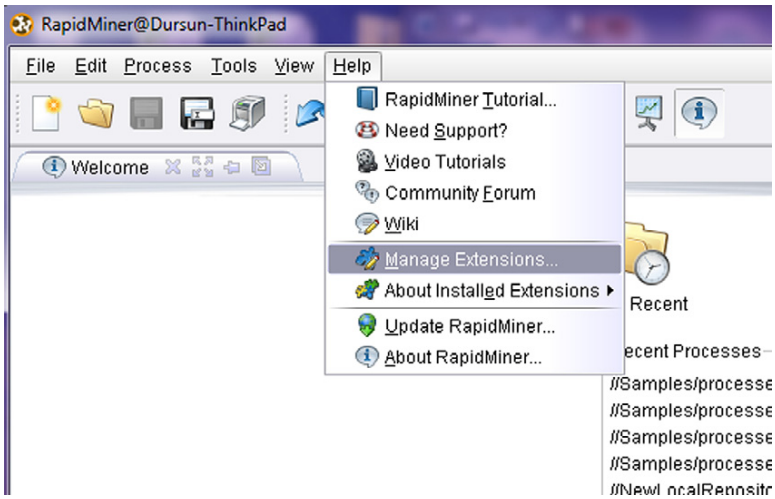


FIGURE H.5
Adding Text Mining extension to the RapidMiner tool.

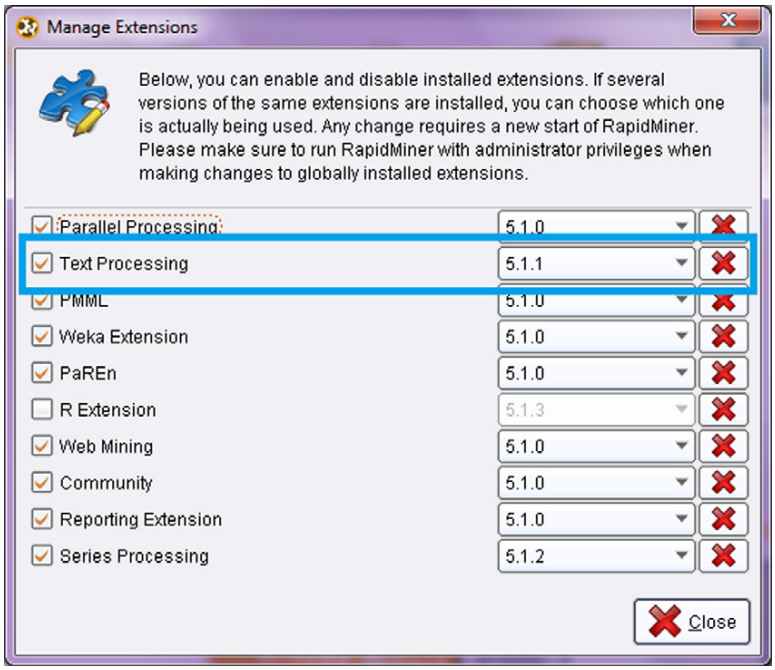


FIGURE H.6
Managing extensions.

If you don't see Text Processing listed, then it is not installed. All you have to do is go to Help, click on Update RapidMiner, and select and install Text Processing (Figure H.7).

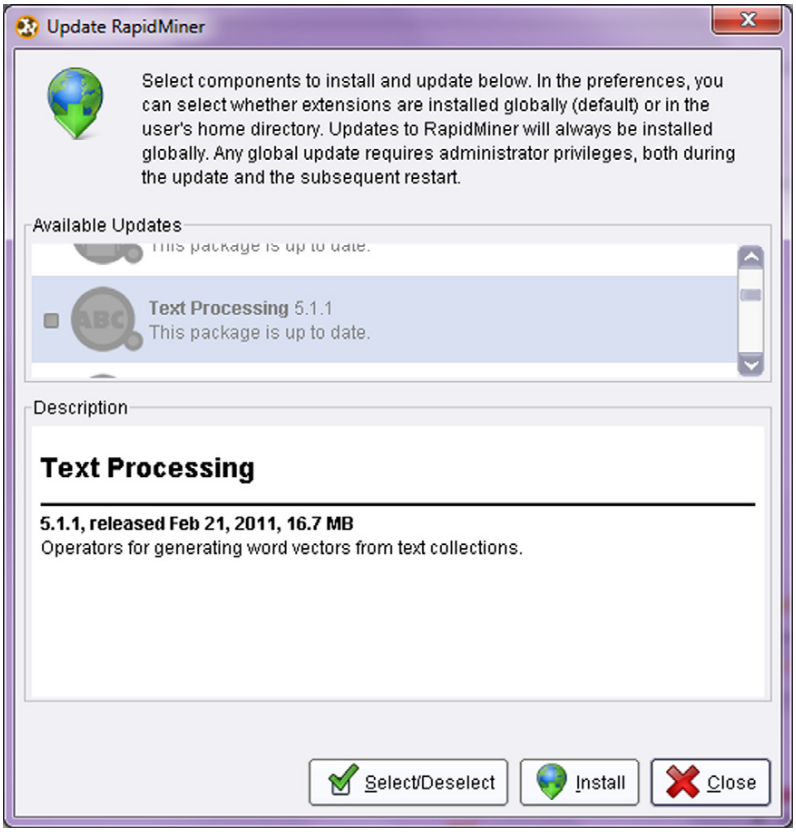


FIGURE H.7
Installing Text Processing extension (in this figure Text Processing is grayed out because it is already installed with the latest version).

STARTING A NEW PROCESS

In the Welcome window, click on New to start a new process. Then select a repository to store the contents of the new project, give it a name, and click OK (Figure H.8).

To load documents into RapidMiner for text mining, you have a number of different options (Figure H.9). You can process documents from data, create documents, read documents, or load the documents from file or mail.

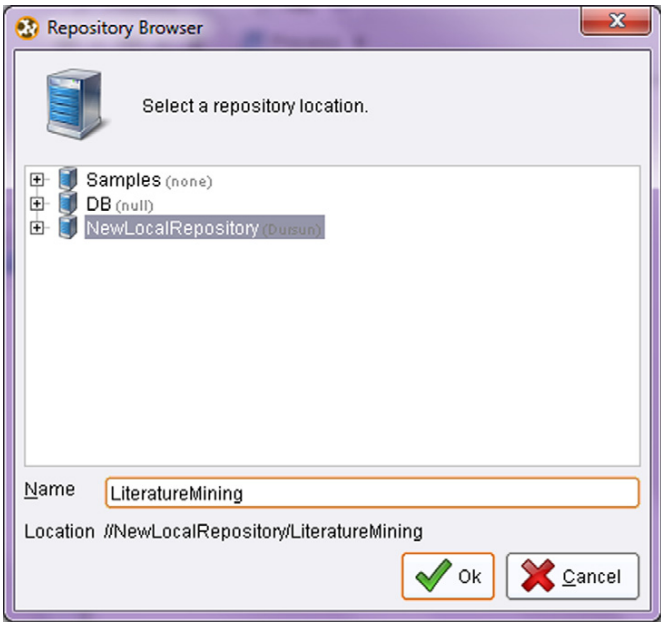


FIGURE H.8
Selecting a repository and naming the new project.

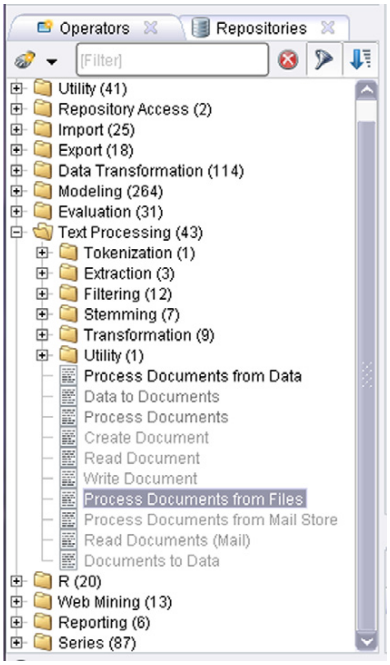


FIGURE H.9
Options to read documents for text processing.

If your documents are saved as individual text files in a folder, then you need to select Process Documents from Files, and in the pop-up dialog box specify the folder (or folders) where the text files are stored (Figure H.10).

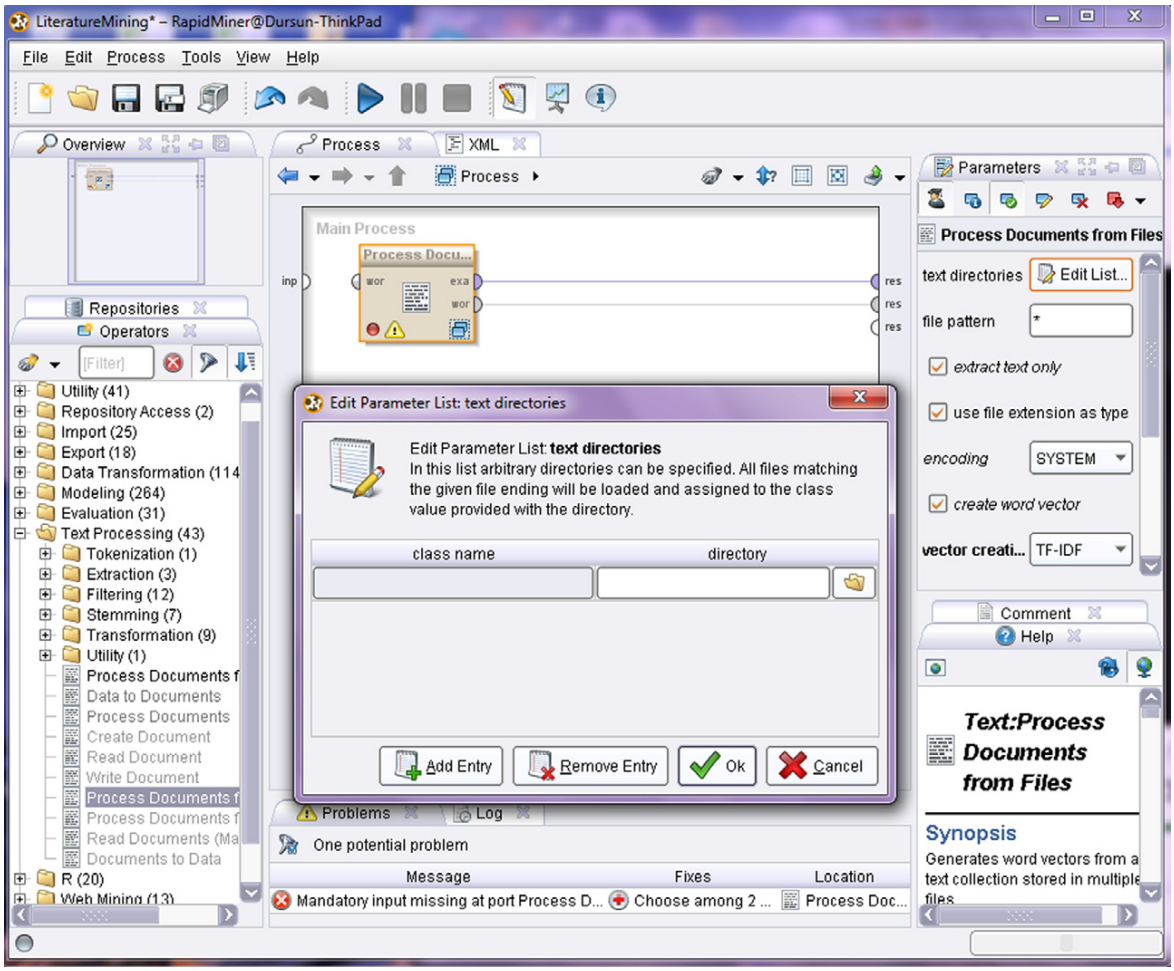
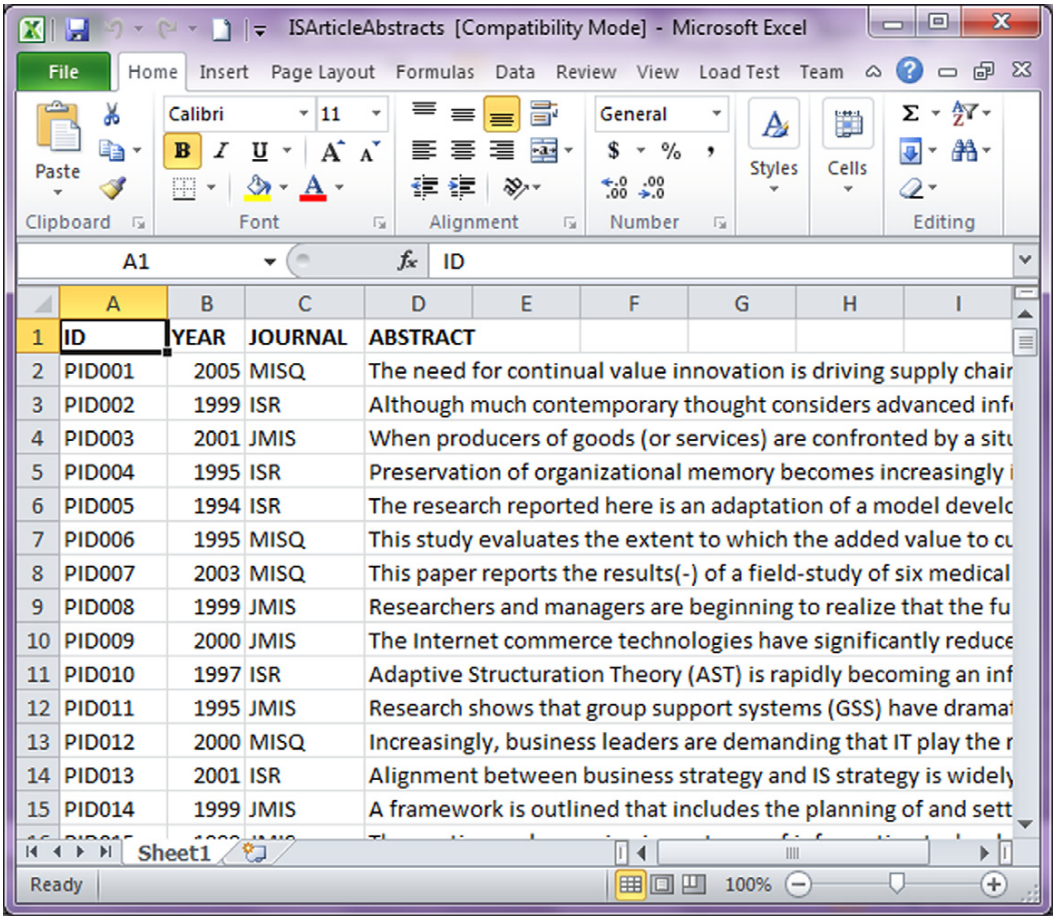


FIGURE H.10
Loading documents from files stored in one or more folders.

In our case, our text data are stored in an Excel file (Figure H.11).

In order to read an Excel file into RapidMiner, you can use the Read Excel operator (i.e., process node). You can find this operator under Import>Data>Read Excel in the Operators tree pane. A quick way to find an operator is to use the Filter (or Search) function, which is located on the top of the Operators pane. There, if you type "Excel," you would get all of the operators that include the word "Excel" in their name (Figure H.12).



	A	B	C	D
1	ID	YEAR	JOURNAL	ABSTRACT
2	PID001	2005	MISQ	The need for continual value innovation is driving supply chair
3	PID002	1999	ISR	Although much contemporary thought considers advanced inf
4	PID003	2001	JMIS	When producers of goods (or services) are confronted by a situ
5	PID004	1995	ISR	Preservation of organizational memory becomes increasingly i
6	PID005	1994	ISR	The research reported here is an adaptation of a model develc
7	PID006	1995	MISQ	This study evaluates the extent to which the added value to cu
8	PID007	2003	MISQ	This paper reports the results(-) of a field-study of six medical
9	PID008	1999	JMIS	Researchers and managers are beginning to realize that the fu
10	PID009	2000	JMIS	The Internet commerce technologies have significantly reduce
11	PID010	1997	ISR	Adaptive Structuration Theory (AST) is rapidly becoming an inf
12	PID011	1995	JMIS	Research shows that group support systems (GSS) have dramat
13	PID012	2000	MISQ	Increasingly, business leaders are demanding that IT play the r
14	PID013	2001	ISR	Alignment between business strategy and IS strategy is widely
15	PID014	1999	JMIS	A framework is outlined that includes the planning of and sett

FIGURE H.11
Textual data (article abstract) stored in an MS Excel file.

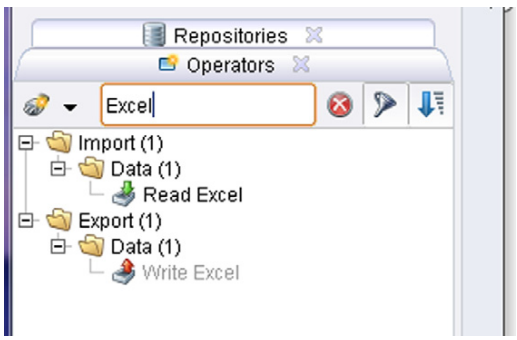


FIGURE H.12
Searching for an operator to manage Excel files.

You can drag and drop the read Excel operator into the process map. Once the operator is selected, its properties are shown on the right side of the screen (Figure H.13).

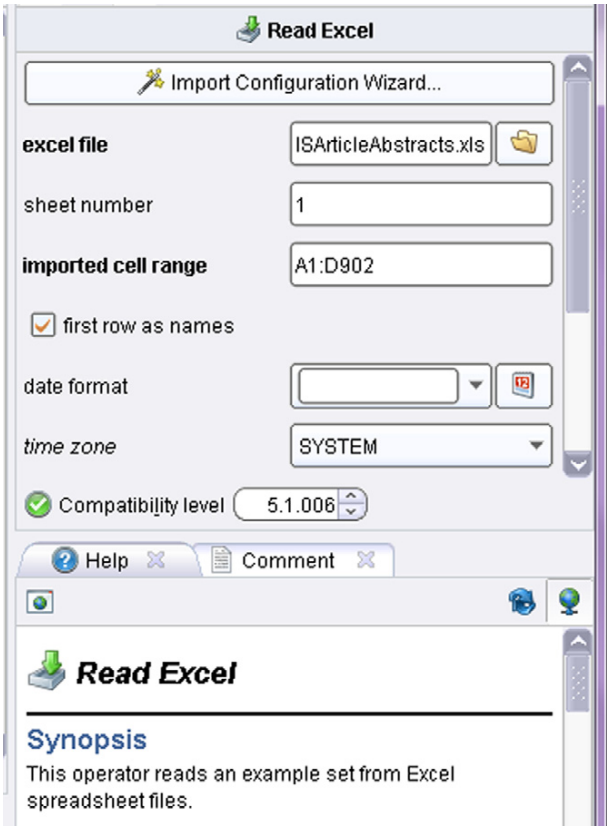


FIGURE H.13
Properties for Read Excel operator.

A quick way to specify the file name and various details about the variables in the data set is to use the Import Configuration Wizard (located on the top of the Read Excel property pane). This wizard takes you through a few steps to set the file and variable specific parameters (see Figures H.14, H.15, H.16, and H.17).

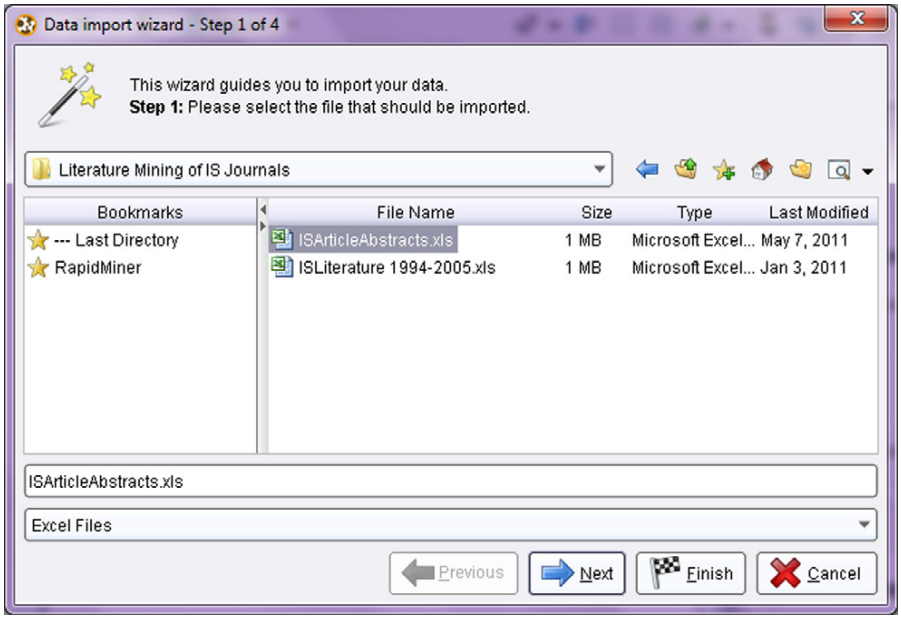


FIGURE H.14
Selecting the Excel file to read.

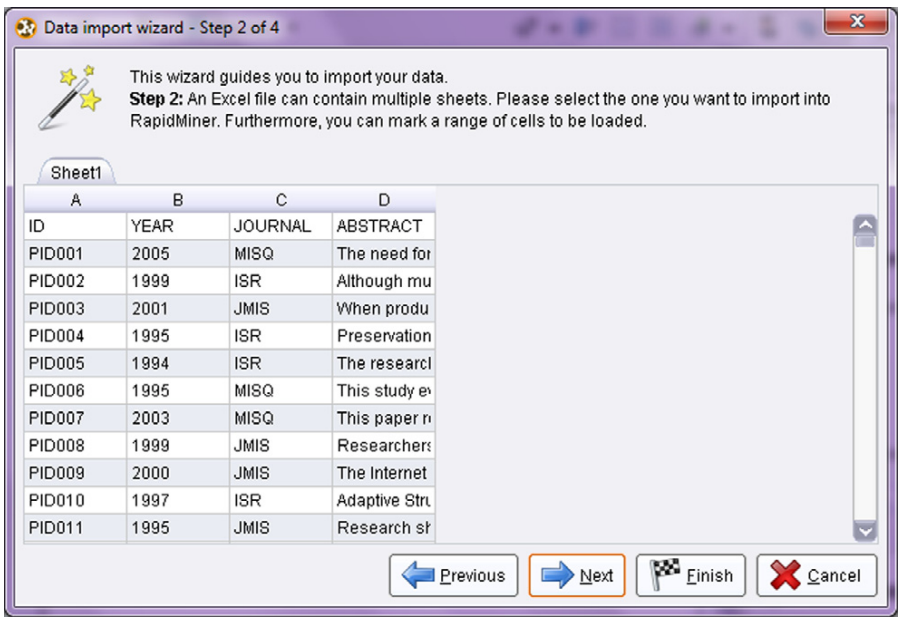


FIGURE H.15
Specifying the sheet within the selected Excel file to process.

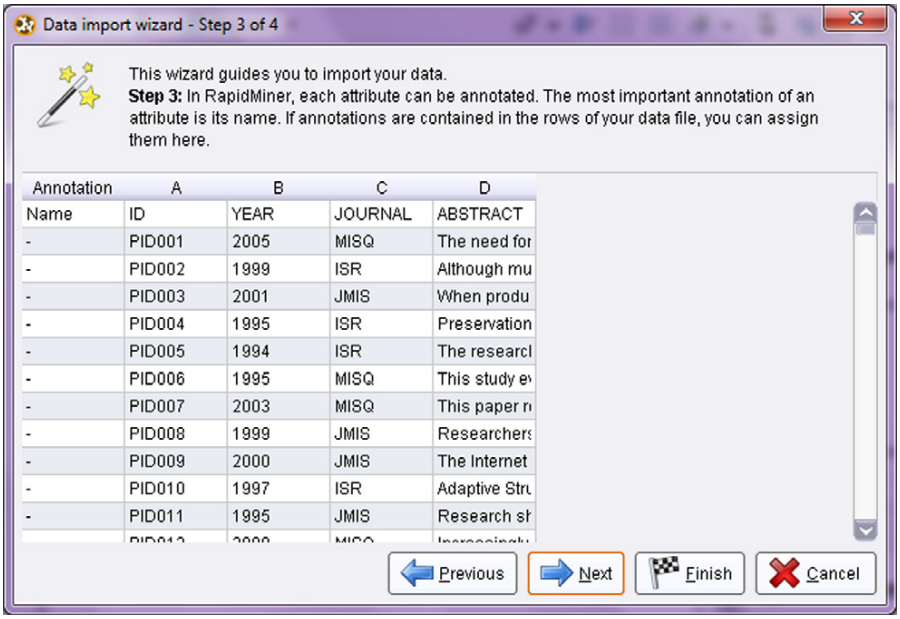


FIGURE H.16
Annotating the attributes (if so desired) interface.

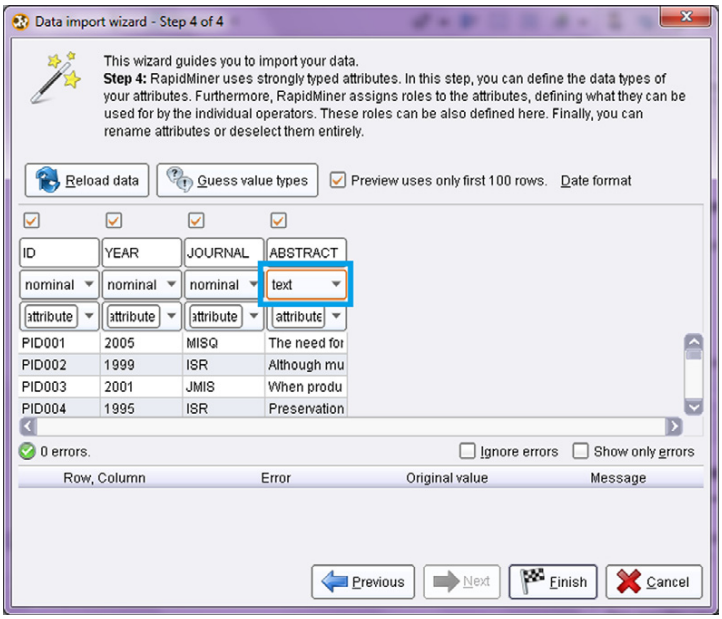


FIGURE H.17
Specifying the types for attributes.

You should make sure the variable types are set properly. The text variable (in this case the ABSTRACT variable) especially needs to have the type specified as *text*. Then you can click Finish to return back to the process window.

Once the data are loaded, you can use the proper operators to process the text. First, you need to drop Process Documents from the Data operator into the process map and connect it to the Read Excel operator (Figure H.18).

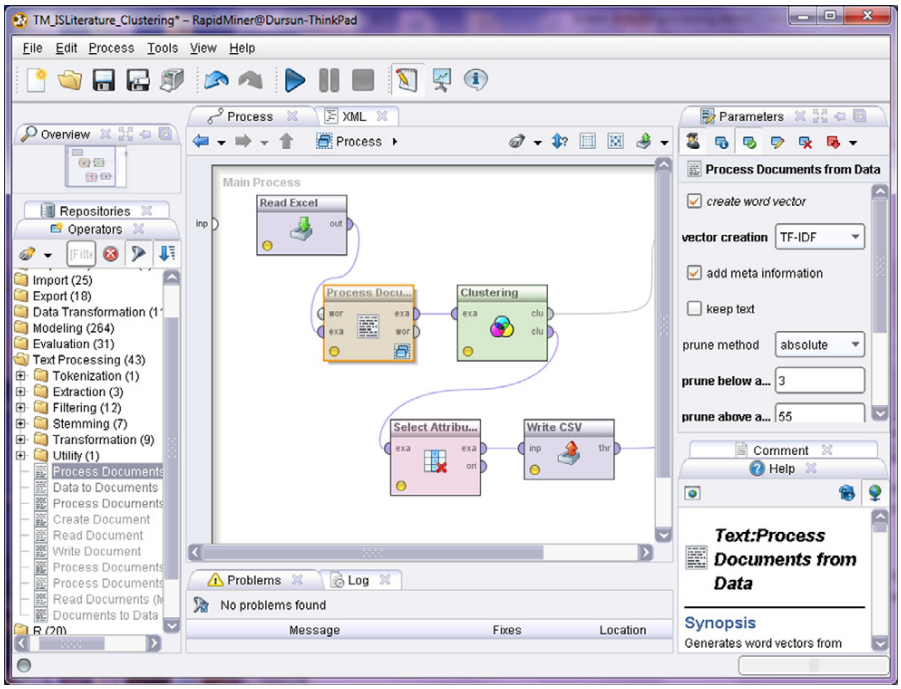


FIGURE H.18
The complete screen shot for the text mining and clustering process.

Figure H.18 shows the complete process where the Process Documents from Data operator is selected. In the RapidMiner window, specific properties of the selected operator are shown on the right side of the window. In Figure H.18, the right side of the window pane shows the properties of the Process Documents from the Data operator. Specific attention is to be paid to the property named "vector creation." Here the user has four different options (Figure H.19), each of which represents the relationship between the words/terms and the documents with different numbers:

1. **Binary Term Occurrence** places 1 in the intersection cell between a document (row) and a word/term (column) if the word/term occurs at least once in that document and places 0 otherwise. The number of occurrences in the document is ignored in this measure.
2. **Term Occurrence** places the exact number of occurrences of a word/term in the intersection cell between the document (row) and the word/term (column). If the word/term does not occur in that document, 0 is placed in the intersection cell.

3. **Term Frequency** places the relative frequency of the word/term in the document in the intersection cell. This measure is calculated by dividing the number of occurrences of a word/term into the number of total words in that document.
4. **TF-IDF** stands for *Term Frequency—Inverse Document Frequency*. It is arguably the most commonly used numerical representation in text mining. It calculates a numerical value that emphasizes both the frequency of the term in a document (more is better) and the rareness of the same term in the collection of all documents (less is better).

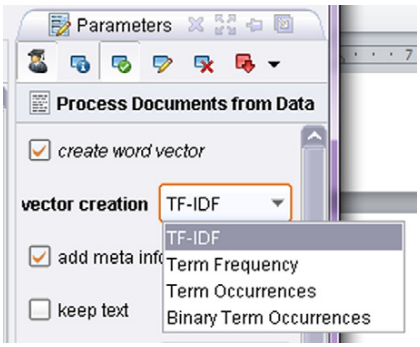


FIGURE H.19
Options to select for vector creation.

The Process Documents from the Data operator has a subprocess where the user is expected to specify the details of how the text should be mined. The operators (like Process Documents from Data) that have an icon that looks like overlapping windows in the lower right corner indicates that there is a subprocess that needs to be specified by the user. User can double click on the parent process to go into the subprocess. In this example the subprocess includes a number of operators (Figure H.20) to convert unstructured text (i.e., article abstracts) into a structured data file (i.e., a term-document matrix).

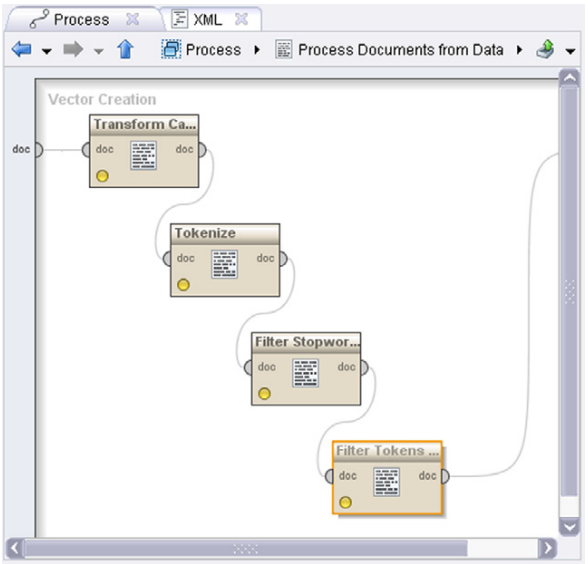
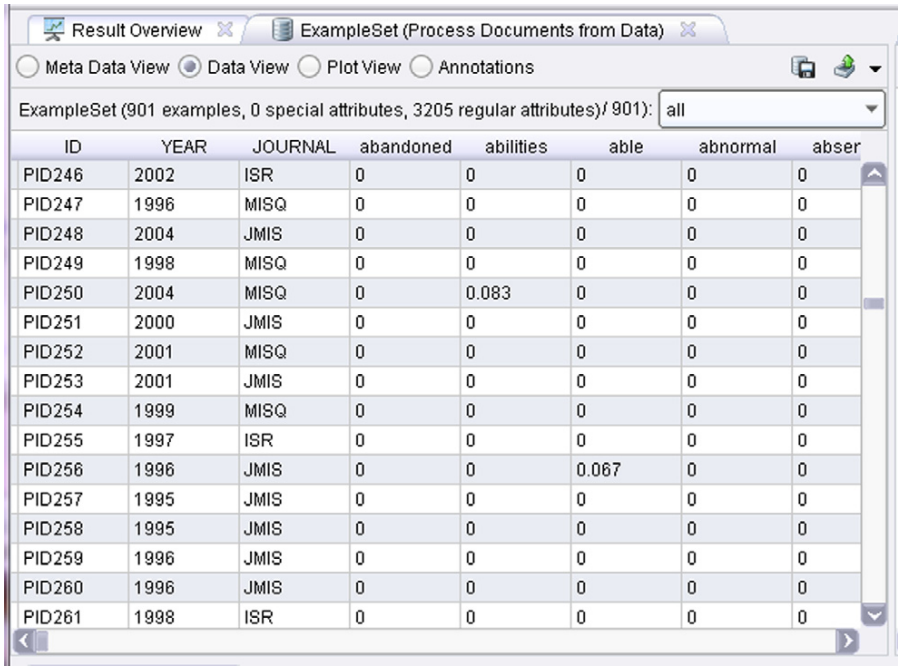


FIGURE H.20
Subprocess under Process Documents from Data.

In this specific subprocess, we used four operators. First, we used a Transform Cases operator to transform all of the characters in the document collection to either lower case (or upper case, if preferred) so the identification of the same words/terms is not biased by the capitalization. Following the transformation, a tokenization operation is applied. Tokenization operator splits the text of documents into a sequence of tokens. There are several options on how to specify the splitting points. You may use all nonletter characters, which is the default setting and works well with English text. This process would result in tokens consisting of one single word/term. Next we applied a Filter Stopwords operator. This operator filters English stopwords from a document collection by removing every token that equals a stopword from the built-in stopword list. Please note that for this operator to work properly, every token should represent a single English word only. The list of stopwords includes words/terms that are commonly found in most text documents (e.g., a, an, is, am, are, the, etc.) that have no contribution to either identification or discrimination of documents from each other. Lastly, we applied the Filter Tokens (by Length) operator to filter out the tokens based on their length (i.e., less than two characters long).

Now we go back to the parent process by clicking on the up arrow at the top of the process window. The output of the Process Documents from Data operator is a term-document matrix where the relationships between the documents and the words/terms are represented with numerical indices—in this case TF-IDF measures (Figure H.21). These structured data are then fed into the Clustering operator.



ID	YEAR	JOURNAL	abandoned	abilities	able	abnormal	abser
PID246	2002	ISR	0	0	0	0	0
PID247	1996	MISQ	0	0	0	0	0
PID248	2004	JMIS	0	0	0	0	0
PID249	1998	MISQ	0	0	0	0	0
PID250	2004	MISQ	0	0.083	0	0	0
PID251	2000	JMIS	0	0	0	0	0
PID252	2001	MISQ	0	0	0	0	0
PID253	2001	JMIS	0	0	0	0	0
PID254	1999	MISQ	0	0	0	0	0
PID255	1997	ISR	0	0	0	0	0
PID256	1996	JMIS	0	0	0.067	0	0
PID257	1995	JMIS	0	0	0	0	0
PID258	1995	JMIS	0	0	0	0	0
PID259	1996	JMIS	0	0	0	0	0
PID260	1996	JMIS	0	0	0	0	0
PID261	1998	ISR	0	0	0	0	0

FIGURE H.21
Output of Process Documents from Data operator (a term-document matrix).

For clustering we selected the k-means clustering algorithm (a popular statistical technique to find natural groupings of records using a simple multidimensional distance measure). In k-means, the user is expected to specify the number of clusters that he or she would like to have. In this tutorial, after some experimentation, we chose to set the number of clusters to 9. We kept the rest of the parameters of Clustering operator as their default values. Figure H.22 shows a screen shot of one of the interesting output reports generated by the Clustering operator. In this two-pane window, all nine clusters are shown in a tree view (left-hand side), and the list of documents that belong to each of the nine clusters is shown in the list view (right-hand side). In this dynamic view, the list of documents changes according to the selection of the cluster number in the tree view.

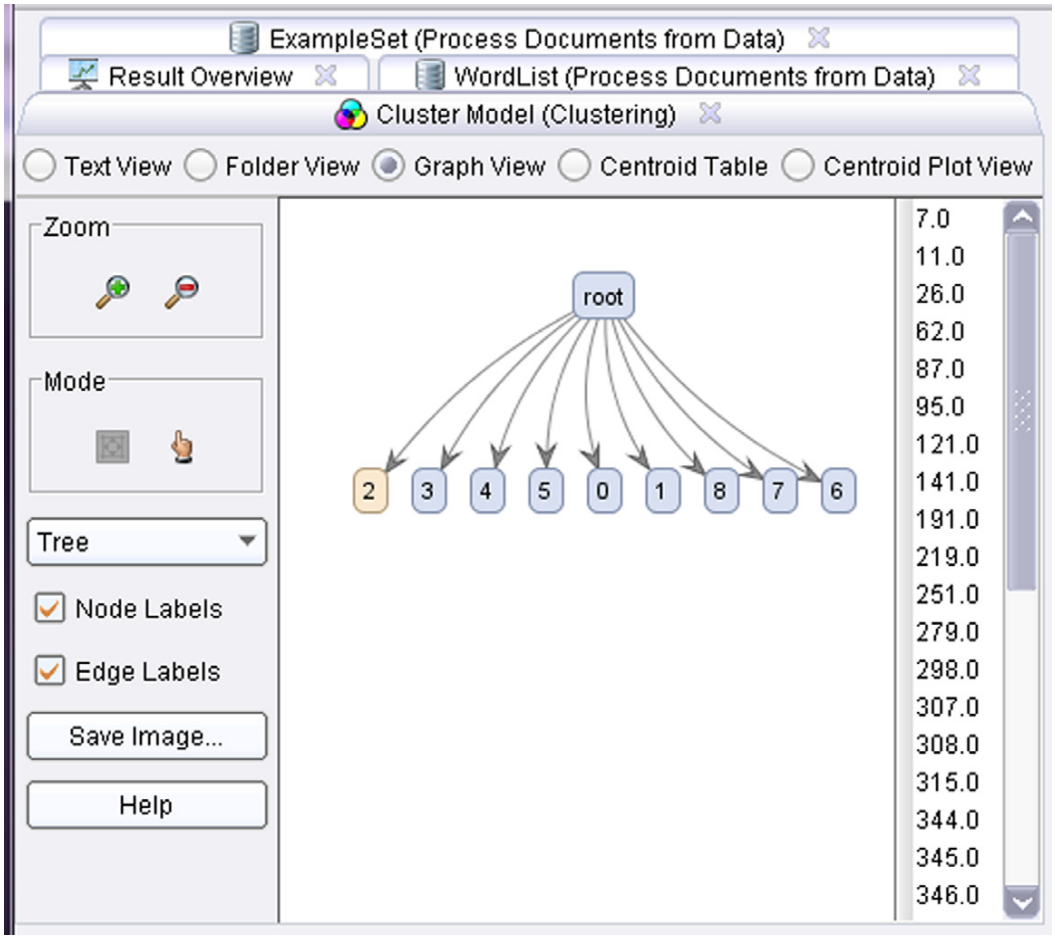
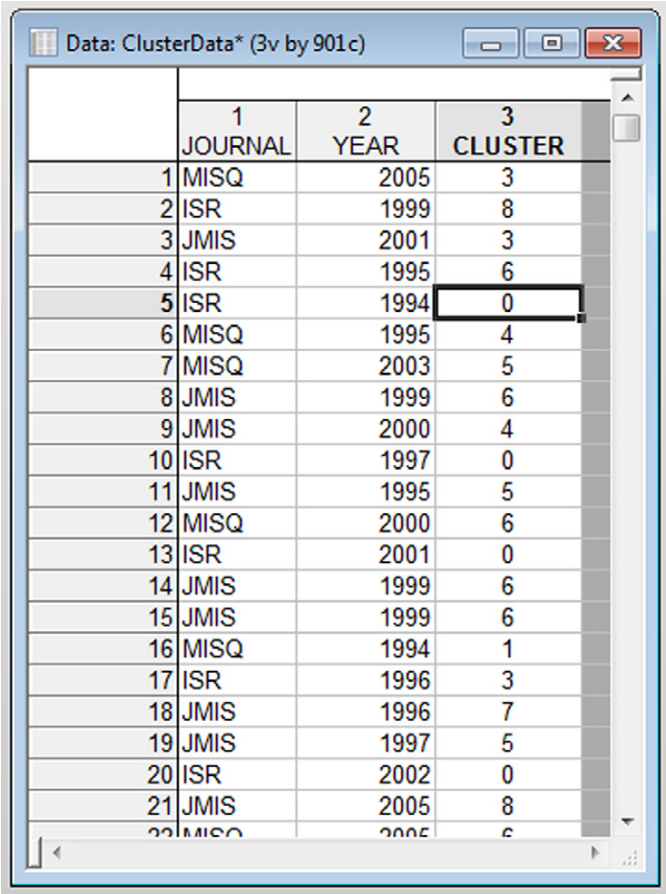


FIGURE H.22

Output of the clustering operator. When one of the nine clusters is selected in the tree view, the document numbers/IDs that fall into that cluster are shown in a list view on the right side.

Next, we wanted to export the clustering results into an easily importable file format; for that we choose the comma separated values (CSV) file format. In this file, in addition to clusters, we also included the original variables (YEAR and JOURNAL). The CSV file is then imported in the *STATISTICA* Data Miner software tool for further graphical reporting (Figure H.23).


Data: ClusterData* (3v by 901c)

	1 JOURNAL	2 YEAR	3 CLUSTER
1	MISQ	2005	3
2	ISR	1999	8
3	JMIS	2001	3
4	ISR	1995	6
5	ISR	1994	0
6	MISQ	1995	4
7	MISQ	2003	5
8	JMIS	1999	6
9	JMIS	2000	4
10	ISR	1997	0
11	JMIS	1995	5
12	MISQ	2000	6
13	ISR	2001	0
14	JMIS	1999	6
15	JMIS	1999	6
16	MISQ	1994	1
17	ISR	1996	3
18	JMIS	1996	7
19	JMIS	1997	5
20	ISR	2002	0
21	JMIS	2005	8

FIGURE H.23
Clustering results in the CSV file are imported into the *STATISTICA* Data Miner tool.

Using these simple data, we created two graphical reports:

1. A histogram (per cluster) that shows time-dependent changes in the number of articles (i.e., documents) for each of the nine clusters (Figure H.24). With this graphical report, one can deduce the increasing/decreasing popularity of topics (represented as clusters) over the 12-year period. Topic specification can be heuristically/annually determined by looking at the dominant terms that define each of the nine clusters.

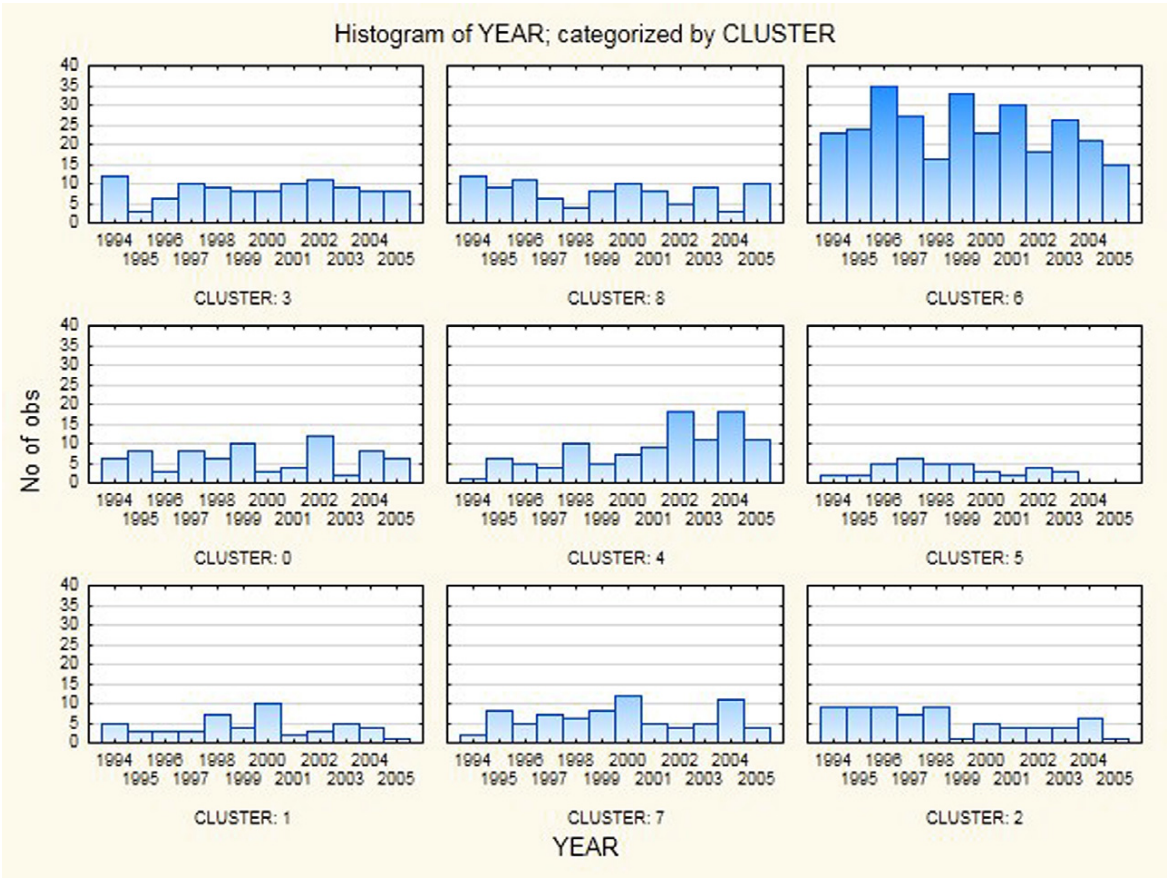


FIGURE H.24
Time-dependent changes in the number of articles published in each of the nine clusters.

2. A histogram (per cluster) that shows the number of articles coming from each of the three journal types (Figure H.25). Such a report could be used to determine if any of the three journals are more inclined to publish certain topics (i.e., clusters).

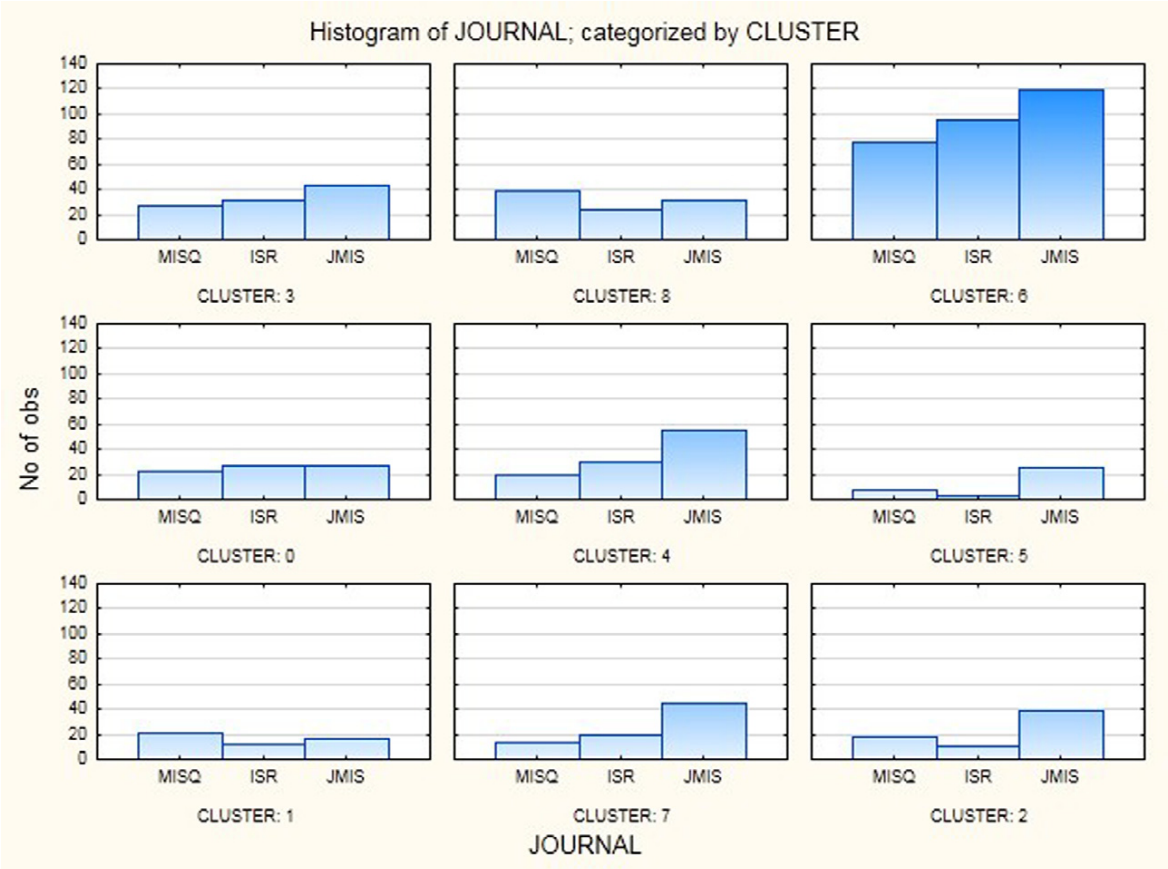


FIGURE H.25
Numerical representation of each of the three journals in each of the nine clusters.

SUMMARY

The amount of unstructured data collected and stored in databases is increasing at a higher rate than any traditional, mostly manual method can keep up with. As the digitized data (either structured or unstructured) become more widely available and accessible, tools that allow us to extract information and knowledge from this mountain of data with ease (e.g., data mining and text mining) are likely to become more valuable. Even though text mining is a relatively new technology, its applications and benefits have already been realized in such fields as medicine, health care, homeland security, law, education, and customer relationship management.

This tutorial showed that it is relatively straightforward to apply text mining techniques to a readily available set of data in the form of downloaded journal article abstracts. A total of 901 articles from three major MIS journals were downloaded and analyzed using text mining, with the objective of

identifying major themes of research and how the major themes may have varied over time, both within individual journals and within the entire set.

The field of text mining is presently in a growth phase in the research literature. Researchers may apply it to a wide spectrum of unstructured, textual data, from genetic sequence analyses to business email. It makes sense to find ways to use this powerful tool to help analyze the status and direction of the research itself.

Reference

Delen, D., and M. Crossland (2008). "Seeding the Survey and Analysis of Research Literature with Text Mining," *Expert Systems with Applications,* 34 (3), 1707–1720.