

# Predicting Box Office Success of Motion Pictures with Text Mining

**Dursun Delen, PhD**

*Department of Management Science and Information Systems, Spears School of Business,  
Oklahoma State University, Tulsa, Oklahoma*

## CONTENTS

Introduction .....	543
Analysis .....	544
Summary .....	556
References .....	556

## INTRODUCTION

Predicting the financial success of a movie prior to its production cycle is arguably one of the most challenging yet essential tasks for decision makers in the motion picture industry. If done accurately, such information could allow decision makers to optimally allocate their resources (financial and otherwise) to maximize their RIO. In-depth knowledge about the factors affecting the financial success of a movie would be of great use in making project selection, investment, and production-related decisions. However, forecasting financial success (box office receipts) of a particular motion picture is considered a very difficult (often impossible) problem. Most domain experts think that “Hollywood is the land of hunches and wild guesses” due largely to the uncertainty associated with predicting the product demand. Jack Valenti, long-time president and CEO of the Motion Picture Association of America, once said, “No one can tell you how a movie is going to do in the marketplace ... not until the film opens in a darkened theater and the sparks fly up between the screen and the audience.” Journals and trade magazines of the motion picture industry have been full of examples, statements, and experiences that support this claim.

The difficulty associated with the perceived unpredictable nature of the problem has intrigued researchers and practitioners to develop “models” for understanding and hopefully forecasting the financial success of motion pictures. Most analysts have used a combination of numerical and nominal variables (e.g., MPAA rating, genre, star power, time of release, special effects, etc.) to

predict the box office receipts of motion pictures after a movie's initial theatrical release. Because they attempt to determine how a movie is going to do based on the early financial figures, the results were quite accurate, but they are not helpful for making investment and production-related decisions, which are to be made during the planning phase. Some studies have attempted to forecast the performance of a movie before it is released but had only limited prediction success. These previous studies, which are either good for predicting the financial success of a movie after its initial theatrical release or are not accurate enough predictors for decision support, leave us with an unsatisfied need for a forecasting system capable of making a prediction prior to a movie's theatrical release. Our ongoing research aims to fill this need by developing and embedding sophisticated forecasting models into a web-based decision support system that can be readily accessed and used by Hollywood managers.

In this tutorial, we used a text mining approach. Using five years of movie data (story lines for over 1,000 movies produced and launched between 2002 and 2006), along with RapidMiner's Text Processing extension, we tried to predict the financial success of movies. Following the format used by [Sharda and Delen \(2006\)](#) and [Delen and colleagues \(2007\)](#), we discretized the dependent variable into nine classes based on the following breakpoints:

**Table O.1** Breakdown of the Dependent Variables into Nine Classes

Class No.	1	2	3	4	5	6	7	8	9
Range (in Millions)	<1 (Flop)	>1 <10	>10 <20	>20 <40	>40 <65	>65 <100	>100 <150	>150 <200	>200 (Blockbuster)

In summary, we used the textual variable of "story line" as the input to the prediction system and used the class numbers (nine-value nominal-ordinal variable as shown above) as the output of our system. Once such a prediction model is trained and deployed, the user can input the story line of a hypothetical movie and find out what success class it belongs to.

## ANALYSIS

Text Processing is an extension to the RapidMiner data mining software tool. In order to use RapidMiner for this text mining project, we must first make sure that our version of RapidMiner includes the Text Processing extension. You can check to see what extensions are already installed on your RapidMiner by clicking Help and then selecting Manage Extensions ([Figure O.1](#)). In the pop-up window you will see all of the installed extensions with their current versions ([Figure O.2](#)). In this interface you can deactivate or uninstall already installed extensions.

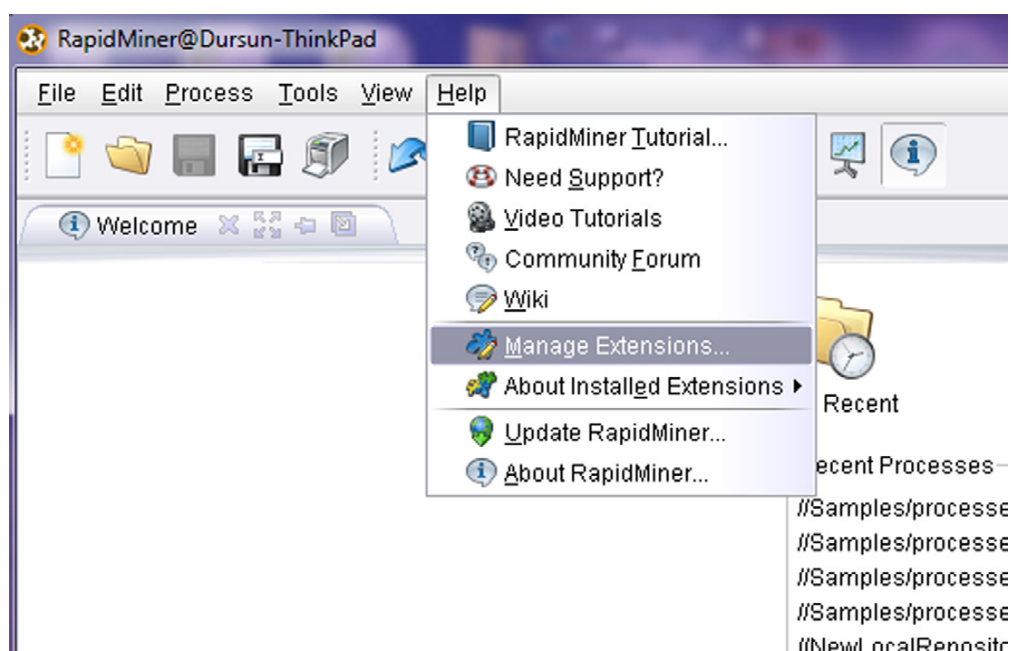


FIGURE O.1

Visualizing the already installed extensions in RapidMiner software tool.

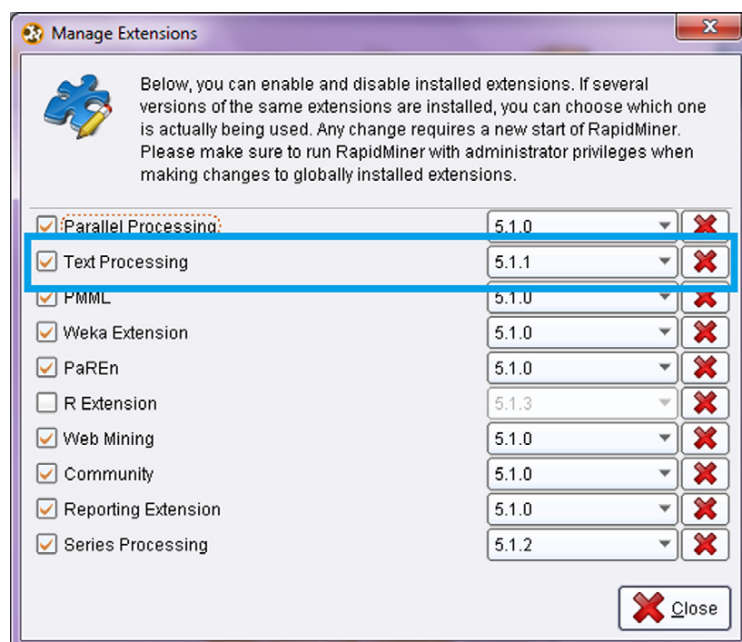
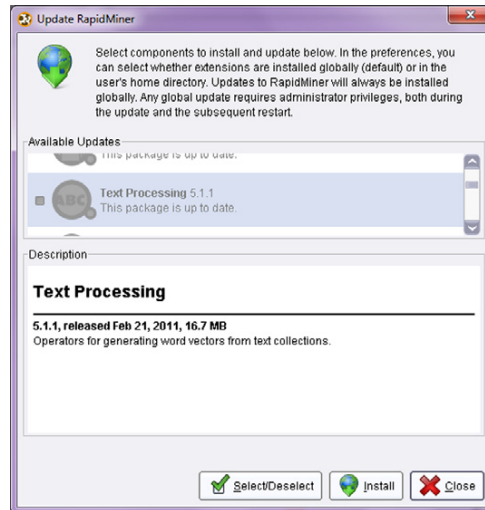


FIGURE O.2

Managing extensions in RapidMiner.

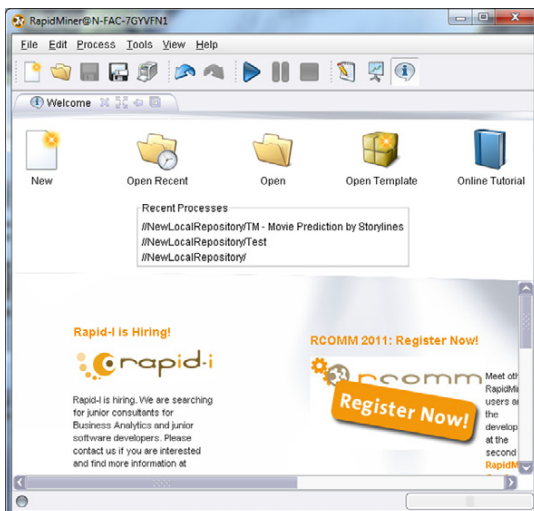
If you don't see Text Processing listed, then it is not installed. All you have to do is go to Help, click on Update RapidMiner, and select and install Text Processing (Figure O.3).



**FIGURE O.3**

Installing the Text Processing extension. Text Processing is grayed out because it is already installed with the latest version.

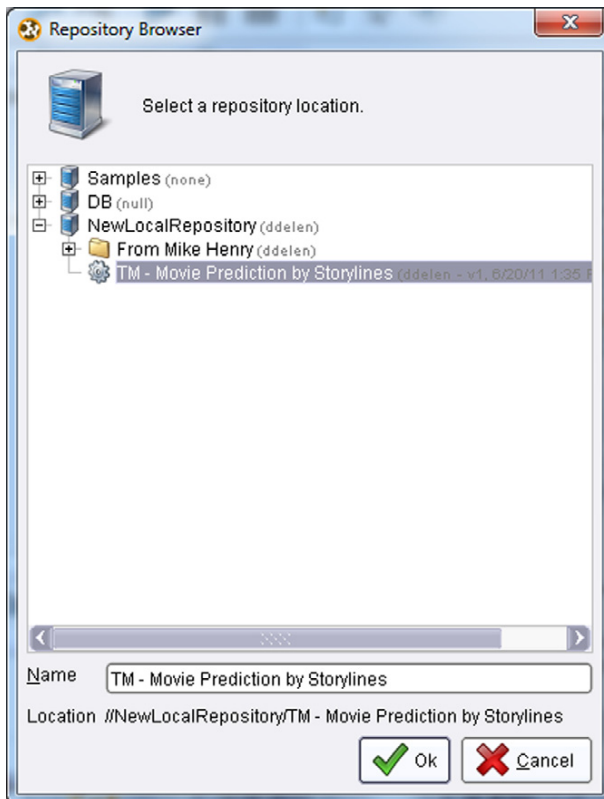
Once we have confirmed the existence of the Text Processing extension in RapidMiner, we can start building our process. For that we need to start a new process (Figure O.4).



**FIGURE O.4**

Starting a new process in RapidMiner.

As soon as you click on a new process, RapidMiner asks you to specify the repository location. To put it simply, a repository in RapidMiner is a centralized location to collect and organize all project related files. In this interface (Figure O.5), you can select an existing repository or to create a new repository.

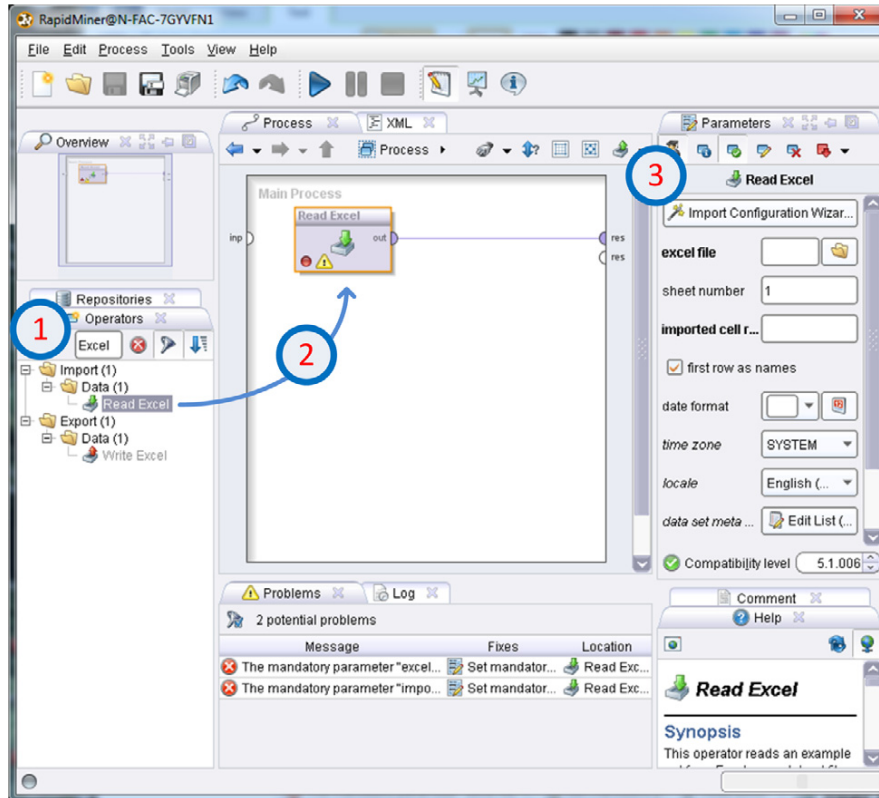


**FIGURE O.5**

Selecting an existing project or creating a new project under a repository.

Once the repository is specified, you will be directed to the process window (Figure O.6). There you can browse, select, or search Operators to drop into the Project workspace. Once located, you can simply drag and drop it into the workspace. Based on your selection, the connections between the input/output ports (and ports between the Operators) can be made automatically or manually (by clicking and dragging the connection from the source port and clicking and dropping it into the destination port).

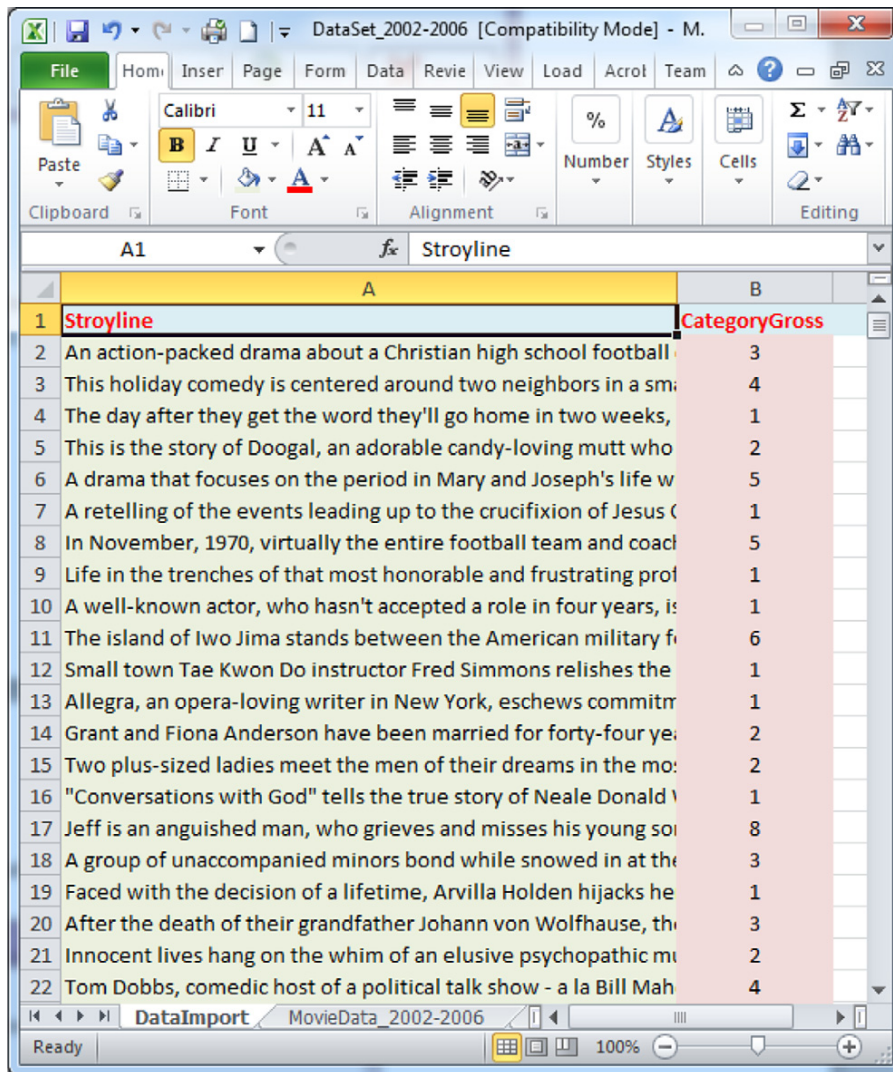
In this project, we will read an Excel file as our input file. Therefore, we search for the Read Excel operator (step “1”). Once located, we can drag and drop it into the Process workspace (step “2”). We can select the Read Excel process in the Process workspace and see its properties in the pane on the right-hand side. There we can click on Import Configuration Wizard to guide us through locating and specifying the properties of our input data file (Figure O.6).



**FIGURE O.6**  
Connecting to an Excel file in the Process workspace.

Once the Import Configuration Wizard has started, we will be asked to go through four simple steps:

**Step 1.** Locate and select your Excel file. A small snapshot of the Excel file is shown in Figure O.7.



The screenshot shows a Microsoft Excel window titled 'DataSet\_2002-2006 [Compatibility Mode] - M.'. The ribbon includes File, Home, Insert, Page, Form, Data, Review, View, Load, Acrol, and Team. The 'Home' ribbon is active, showing options for Clipboard, Font, Alignment, Number, Styles, and Cells. The active cell is A1, containing the text 'Stroyline'. The formula bar shows 'Stroyline'. The spreadsheet has two columns: 'A' and 'B'. Column A is titled 'Stroyline' and contains 22 rows of movie descriptions. Column B is titled 'CategoryGross' and contains 22 rows of numerical values. The data is as follows:

Stroyline	CategoryGross
An action-packed drama about a Christian high school football	3
This holiday comedy is centered around two neighbors in a sm	4
The day after they get the word they'll go home in two weeks,	1
This is the story of Doogal, an adorable candy-loving mutt who	2
A drama that focuses on the period in Mary and Joseph's life w	5
A retelling of the events leading up to the crucifixion of Jesus C	1
In November, 1970, virtually the entire football team and coach	5
Life in the trenches of that most honorable and frustrating prof	1
A well-known actor, who hasn't accepted a role in four years, is	1
The island of Iwo Jima stands between the American military f	6
Small town Tae Kwon Do instructor Fred Simmons relishes the	1
Allegra, an opera-loving writer in New York, eschews commitr	1
Grant and Fiona Anderson have been married for forty-four ye	2
Two plus-sized ladies meet the men of their dreams in the mo	2
"Conversations with God" tells the true story of Neale Donald V	1
Jeff is an anguished man, who grieves and misses his young son	8
A group of unaccompanied minors bond while snowed in at the	3
Faced with the decision of a lifetime, Arvilla Holden hijacks he	1
After the death of their grandfather Johann von Wolfhause, the	3
Innocent lives hang on the whim of an elusive psychopathic m	2
Tom Dobbs, comedic host of a political talk show - a la Bill Mah	4

**FIGURE O.7**

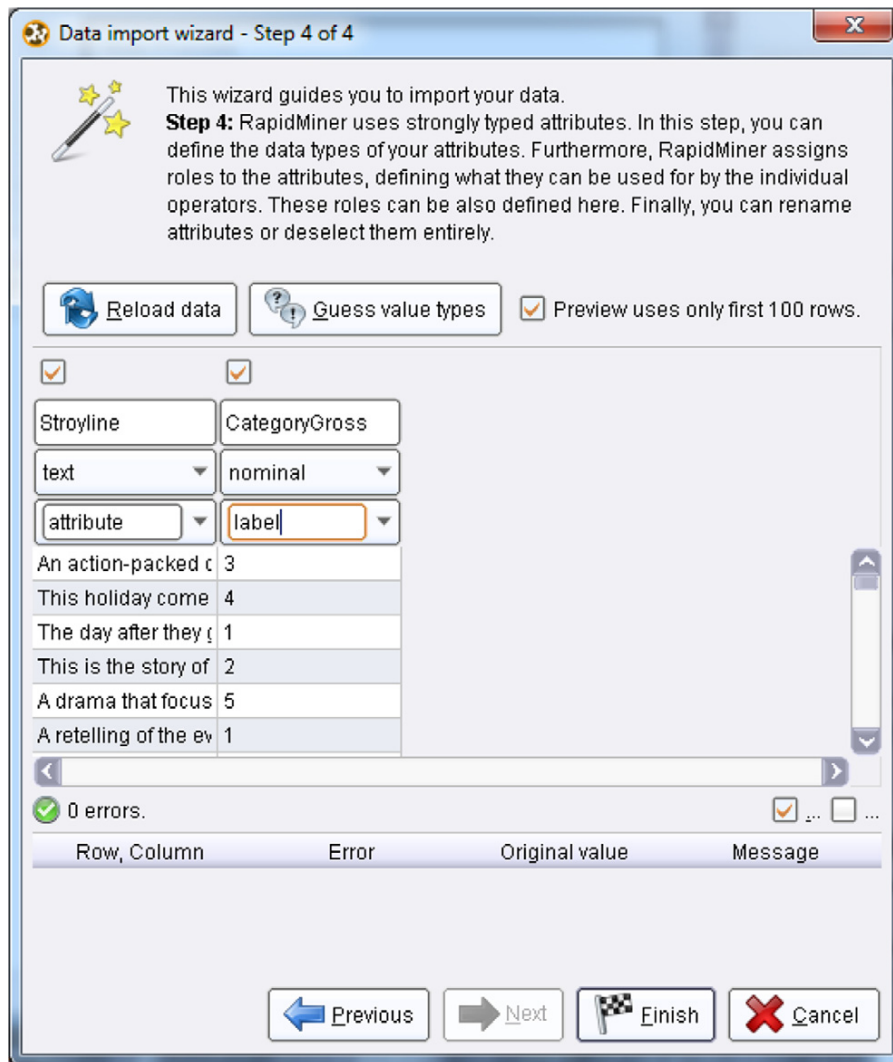
A snapshot for the data set in Microsoft Excel.

**Step 2.** Select your Excel sheet within the selected Excel file.

**Step 3.** Make annotations to your cases (if you so desire).

**Step 4.** Specify the data types for your variables (e.g., numeric, nominal, text, etc.) and the role that they will play (e.g., attribute, label, ID, etc.). The final step in the Import Configuration Wizard is shown in Figure O.8. Then click Finish to save and return to the text mining process window.

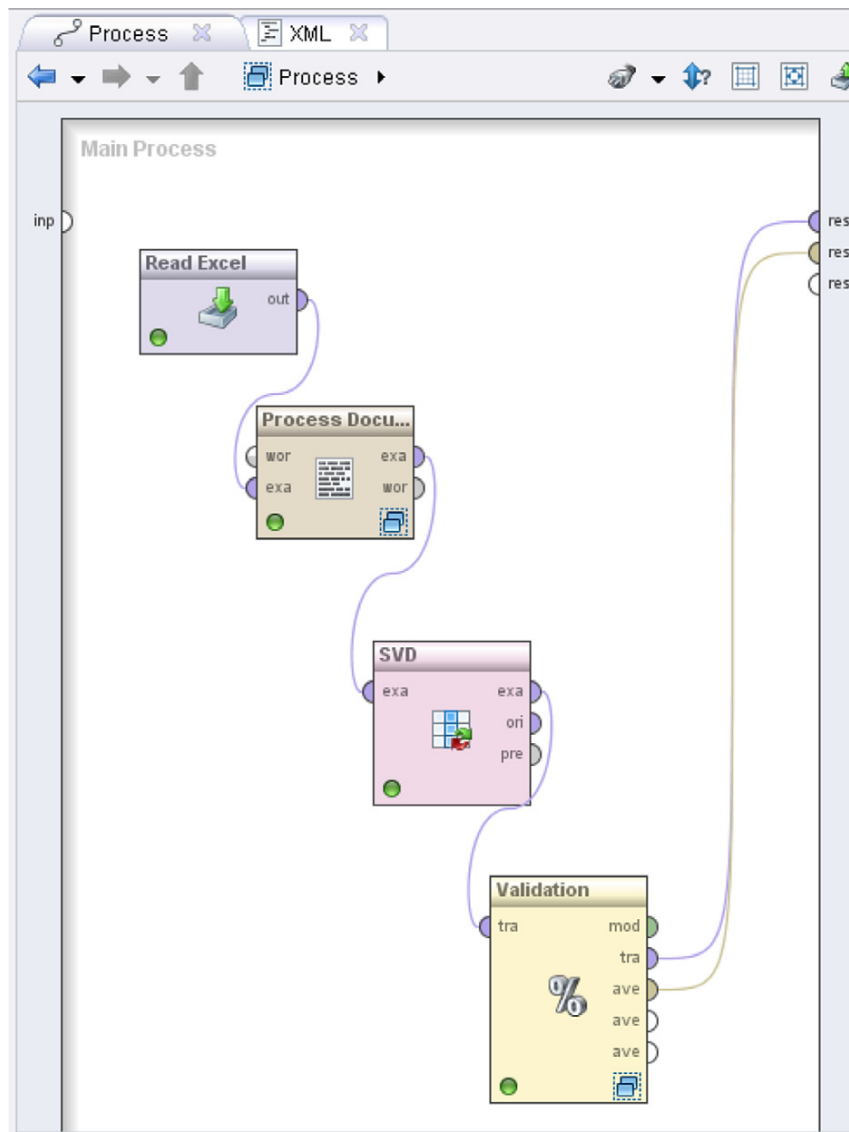


**FIGURE O.8**

Step 4 in the Import Configuration Wizard process.

Once the input data are specified, we can locate and place other operators into the process workspace (Figure O.9). As shown in Figure O.9, immediately following Read Excel operator, we used a Process Documents from Data operator. This operator has a subprocess underneath, where we provide additional operators to convert textual data into a structured form (essentially into a matrix where the rows represent the documents/movies and the columns represent the unique terms/words identified from the collection of story lines). The relationships between the rows and columns are represented by some sort of indices in the intersecting cells.

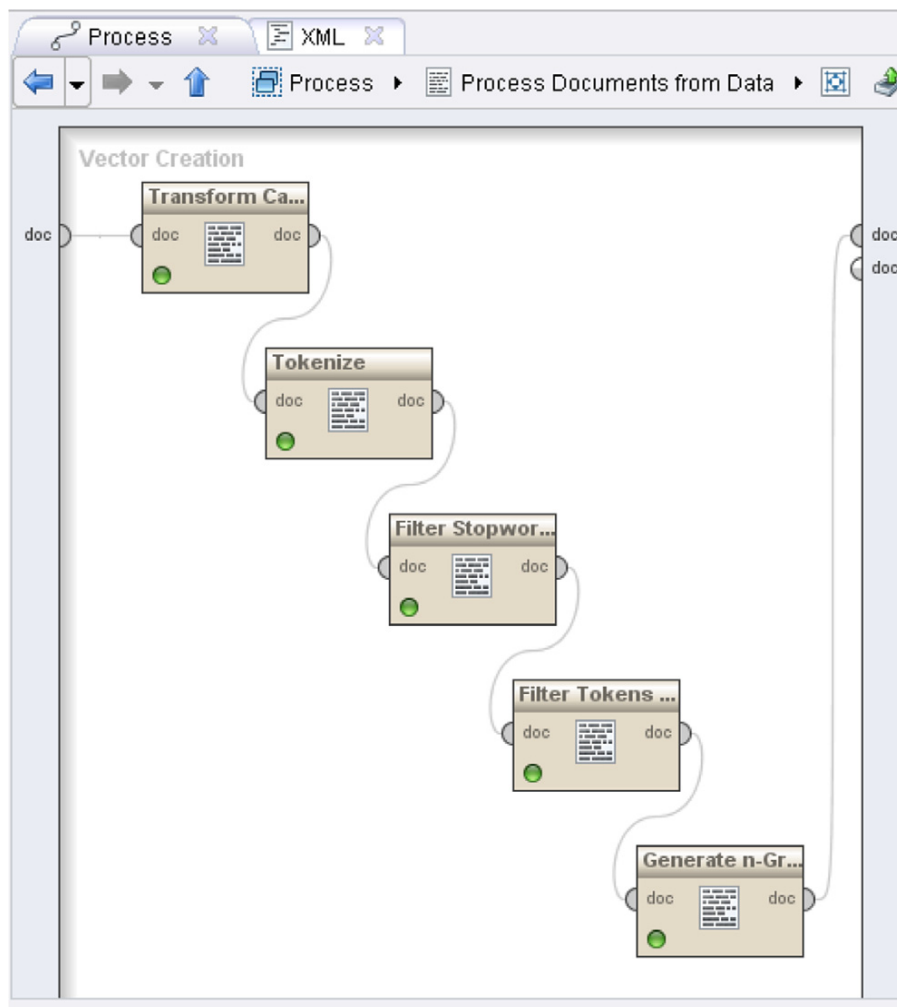


**FIGURE O.9**

The main process for the text mining project.

Figure O.10 shows the subprocess under the Process Documents from Data operator. As can be seen, we used five Operators to convert unstructured text (in the form of story lines) into a matrix. The first Operator (i.e., Transform Cases) converts the text into a single case type (either all lower case or all upper case) so that the case differences between the terms/words would not incorrectly lead to confusion in term/word identification. The next operator (i.e., Tokenize) takes the input of the previous

operator and separates the terms/words from each other. Following the tokenization, comes the Filter Stopwords (English) operator. This operator filters out the most commonly observed words in English from the term/word list. The assumption is that these words (e.g., *a*, *an*, *is*, *am*, *are*, *the*, etc.) repeat many times in text documents and have no information/discrimination value. Next, we used another operator (i.e., Filter Tokens by Length) to remove words/terms that are less than three characters long. Finally, we used an operator (i.e., Generate n-Grams) to identify two- and three-word terms and make them a part of the term-document matrix.



**FIGURE O.10**

Subprocess of the Process Documents from Data operator.

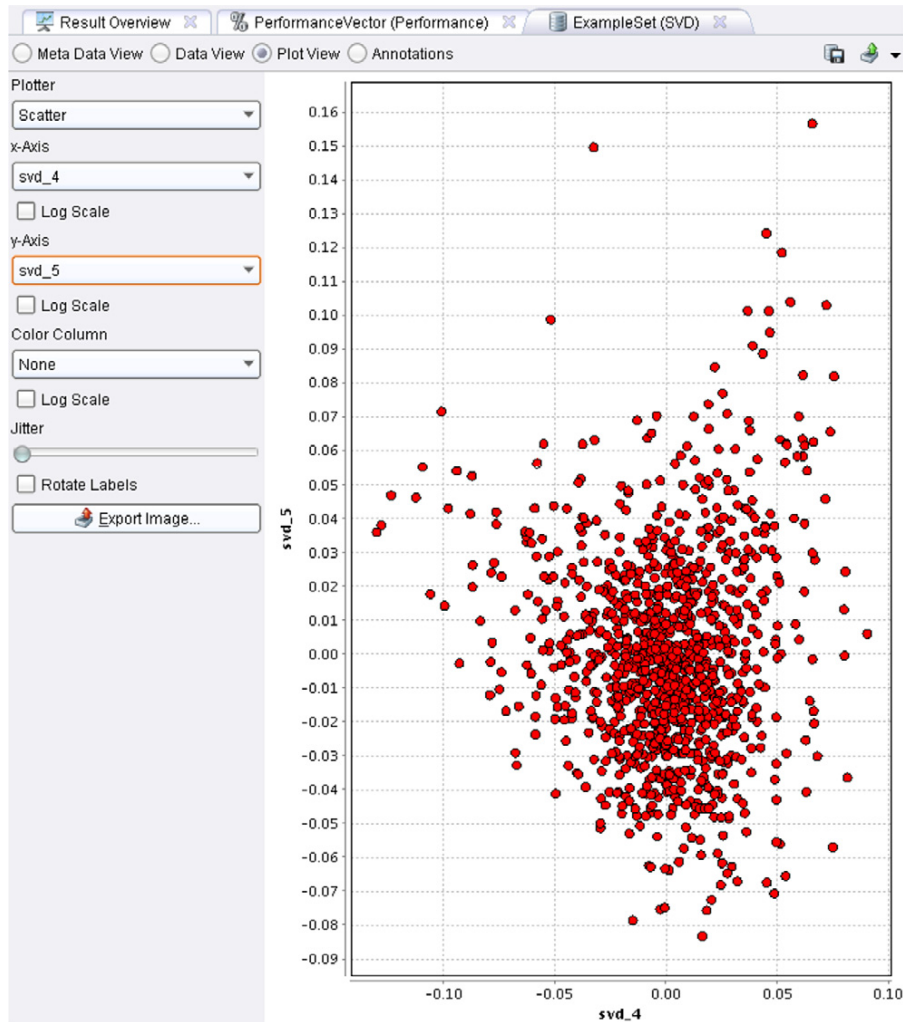
In the main process (see [Figure O.9](#)) following the Process Documents from Data operator, we used a data reduction operator called singular value decomposition (SVD). This operator takes the input data and generates a much smaller number of pseudo-variables (often called the variable reduction/transformation process). The output of the Process Documents from Data operator is a large matrix, more than 1,000 rows and 2,095 columns (variables—words/terms that are created after the five-step text processing). This matrix is shown in [Figure O.11](#). As can be seen, this is a sparse matrix with most of the cells filled with zero values. For classification (which is the case in this study) or clustering type data mining tasks, such large and sparse matrixes are often converted into a smaller size while maintaining underlying patterns. In test mining this reduction process is often performed using SVD. In this example, we reduced 2,095 variables into 5 pseudo-variables (singular values) for further processing.

Row No.	CategoryGross	abandoned	able	abuse	abused	abusive	academy	accept	accepts
30	7	0	0	0	0	0	0	0	0
31	1	0	0	0	0	0	0	0	0
32	9	0	0	0	0	0	0	0	0.100
33	1	0.207	0	0	0	0	0	0	0
34	1	0	0	0	0	0	0	0	0
35	4	0	0	0	0	0	0.211	0	0
36	5	0	0	0	0	0	0	0	0
37	4	0	0	0	0	0	0	0	0
38	1	0	0	0	0	0	0	0	0
39	2	0	0	0	0	0	0	0	0
40	3	0	0	0	0	0	0	0	0
41	2	0	0	0	0	0	0	0	0
42	2	0	0	0	0	0.238	0	0	0
43	1	0	0	0	0	0	0	0	0
44	6	0	0	0	0	0	0	0	0
45	3	0	0	0	0	0	0	0	0
46	2	0	0.11	0	0	0	0	0	0
47	7	0	0	0	0	0	0	0	0.122
48	4	0	0	0	0	0	0	0	0
49	6	0	0	0	0	0	0	0	0
50	5	0	0	0	0	0	0	0	0
51	1	0	0	0	0	0	0	0	0

**FIGURE O.11**

The term-document matrix generated by the Process Documents from Data operator.

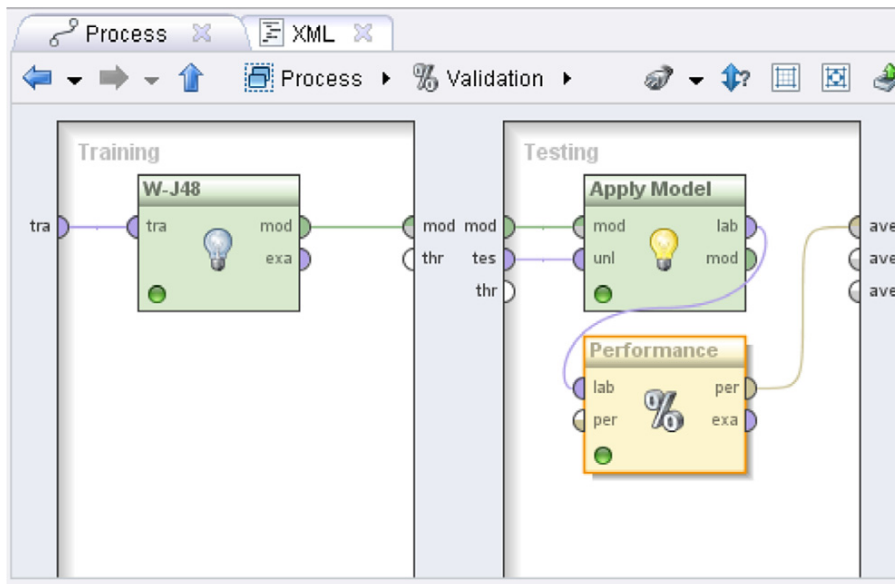
Once the SVD operator completed its processing of the data, we can observe the relationships between the five dimensions in a scatterplot (Figure O.12). By doing so, we can go back and either increase or reduce the dimensions to “optimize” the number of new variables to be used for the classification task that follows.



**FIGURE O.12**

A scatterplot showing the relationship between SVD dimensions 4 and 5.

The last operator in the main process is a subprocess called Validation. This operator is used for classification and regression tasks where the data are automatically split into  $x$  number of folds and the experimentation is repeated for  $x$  times, each time one fold of the data is used as the test sample (holdout sample). Figure O.13 shows the subprocess used under the Validation operator.



**FIGURE O.13**

The subprocess under the Validation operator.

As can be seen in Figure O.13, there are two distinct panes in this subprocess: a model training pane and a model testing pane. In this example, in the model training pane, we used an operator called W-J48. This operator is a part of the Weka library. Weka is another freely downloadable open source data mining tool ([www.cs.waikato.ac.nz/ml/weka/index\\_downloading.html](http://www.cs.waikato.ac.nz/ml/weka/index_downloading.html)). In addition to its own algorithms, RapidMiner also seamlessly incorporates most of Weka's data mining algorithms. J48 is essentially the Java implementation of Quinlan's famous C4.5 decision tree generation algorithm.

On the right-hand side of this subprocess, the trained models are tested and assessed. First, the developed model is applied to the test data (as specified by the current fold) using Apply Model operator, and then its accuracy is assessed using Performance operator.

The classification results of the X-Validation (also known as k-fold cross validation) are shown in Figure O.14 in a confusion matrix format. Herein, all of the folds are combined and aggregated into a single confusion matrix. In this confusion matrix, the columns represent the actuals, while rows represent the predictions. For instance, the very first cell says that 30 of the Class-3 records in the test data sets are accurately predicted as Class-3, accounting for 21.28 percent classification accuracy for Class-3.

As the top of the window shows, the overall accuracy on the test data set is only 15.77. That is, out of all the movies used in this study, this model accurately predicted only 15.77 percent of them correctly. This is not a great prediction accuracy. Considering that the random change of accurately assigning a movie into one of nine classes is roughly 11 percent (i.e.,  $1/9$ ), 15.77 percent accuracy is only marginally better than random chance but obviously not accurate enough for decision making.

☒ Multiclass Classification Performance: ☐ Annotations

☒ Table View ☐ Plot View

accuracy: 15.77% +/- 2.42% (mikro: 15.76%)

	true 3	true 4	true 1	true 2	true 5	true 6	true 8	true 7	true 9	class precis
pred. 3	30	29	27	32	19	16	8	11	4	17.05%
pred. 4	12	21	24	24	13	23	5	8	13	14.69%
pred. 1	30	27	44	37	22	15	6	20	12	20.66%
pred. 2	30	24	46	35	18	21	5	11	12	17.33%
pred. 5	10	13	10	18	11	13	3	6	4	12.50%
pred. 6	11	15	9	18	5	6	2	7	9	7.32%
pred. 8	7	10	2	5	4	1	2	3	2	5.56%
pred. 7	6	4	9	8	4	5	5	9	4	16.67%
pred. 9	5	10	7	4	7	9	5	4	8	13.56%
class recall	21.28%	13.73%	24.72%	19.34%	10.68%	5.50%	4.88%	11.39%	11.76%	

**FIGURE 0.14**

Classification results on holdout samples represented in a confusion matrix.

## SUMMARY

In this tutorial, we used textual data (i.e., story lines) of over 1,000 movies launched between 2002 and 2006 to predict the financial success. As the results indicated, textual information in a story line may not be enough to accurately predict how a movie is going to do at the box office. A short paragraph (a story line) about a movie may not have enough information content to explain potential box office success. At least that is what we deduce from the prediction models developed for this tutorial. It may, however, provide additional variables (words/terms) that can complement other, more traditional data mining-based prediction models.

## References

- Dursun Delen, Ramesh Sharda, and P. Kumar, 2007, "Movie Forecast Guru: A Web-Based DSS for Hollywood Managers," *Decision Support Systems*, 43(4), 1151–1170.
- Ramesh Sharda, and Dursun Delen, 2006, "Predicting Box Office Success of Motion Pictures with Neural Networks," *Expert Systems with Applications*, 30, 243–254.