CHAPTER 5

Text Mining Methodology

CONTENTS

Preamble 73
Text Mining Applications 73
Cross-Industry Standard Process for Data Mining (CRISP-DM) 74
Example 1: An Exploratory Literature Survey Using Text Mining 86
Postscript 89
References 89

PREAMBLE

The process that you follow in an endeavor is at least as important as what methodology you follow to accomplish it. There are several data mining process models used commonly by data miners, but there is no accepted process model for text mining. This chapter presents a proposed process for text mining, which will guide you in the performance of any of the five text mining application areas described in the previous chapter.

TEXT MINING APPLICATIONS

Text mining applications are so broad in their scope and so varied in their goals that it is difficult to express the accomplishment of it in general terms. Compared to other well-established statistical methods, text mining is a relatively new and unstandardized analytical technique for knowledge discovery. Therefore, it is challenging to create a road map of operations to perform its methodology. A *methodology* is a documented and somewhat standardized process for executing and managing complex projects that include many interrelated tasks (i.e., extracting knowledge from textual data sources) by the use of a variety of methods, tools, and techniques. A well-designed and properly followed/implemented methodology can help to ensure consistent and successful results. In essence, a methodology is the manifestation of many experiences (both good and not so good) in a given discipline.

Applications of text mining are driven primarily by trial-and-error experiments based on personal experiences and preferences. While data mining methodologies are relatively mature (e.g., CRISP-DM,

Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications. DOI: 10.1016/B978-0-12-386979-1.00005-0

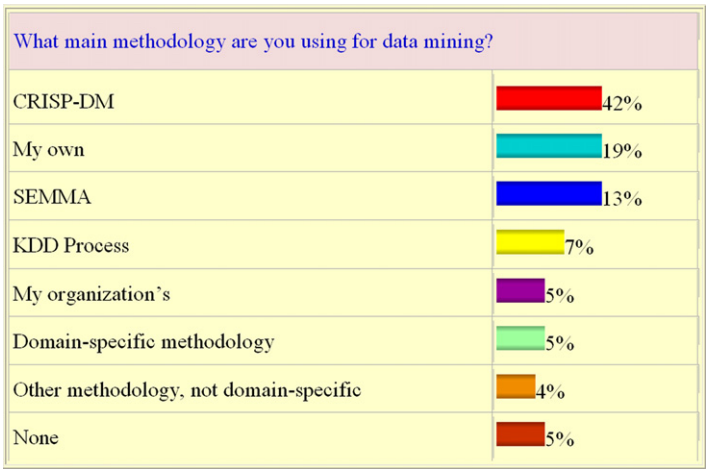

FIGURE 5.1
A user poll on the popularity of data mining methodologies. *Source: KDNuggets.com*

SEMMA, KDD), no commonly accepted methodologies have reflected the essence of best practices in text mining in any domain. The most significant reasons for this void include the following:

- Text mining means different things to different people; even the definition of it and what it encompasses are very unsettled and debatable subjects.
- The unstructured nature of the data opens up a wide range of exploratory avenues.
- There are many different types of unstructured data, some of which can be classified as semistructured (e.g., HTML pages, XML documents, etc.).
- The sheer size of the available data encourages premature sampling and simplification activities.

As the older brother of text mining, data mining went through a similar process of self-definition during the early 1990s that developed several well-known methodologies, including CRISP-DM, SEMMA, and KDD. Even though these three popular methodologies overlap significantly in the way they express and relate the data mining process, they differ in the way they define and scope data mining activities. Among the three, CRISP-DM is the most popular (Figure 5.1).

CROSS-INDUSTRY STANDARD PROCESS FOR DATA MINING (CRISP-DM)

In CRISP-DM, the complete life cycle of a data mining project is represented with six phases: business understanding (determining the purpose of the study), data understanding (data exploration and understanding), data preparation, modeling, evaluation, and deployment. Figure 5.2a shows a slightly modified rendition of the six phases of the CRISP-DM process flow (Chapman et al., 2000), indicating the iterative nature of the underlying methodology.

Within the six phases, CRISP-DM methodology provides a comprehensive coverage of all of the activities involved in carrying out data mining projects. Because the primary distinction between data mining and text mining is simply the type of data involved in the knowledge discovery process, we adopt CRISP-DM

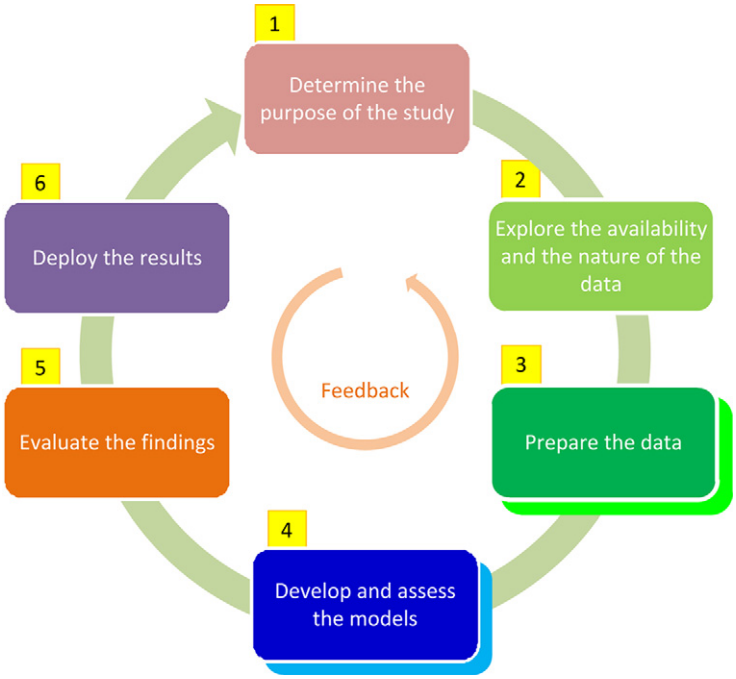

FIGURE 5.2a

A cyclic form of a proposed process flow for text mining, based on CRISP-DM. The feedback loop indicates that the findings and lessons learned at any phase in the process can trigger a backward movement for corrections and refinements, and the completion of a process may lead to new and more focused discovery processes.

as a foundation upon which to derive the text mining methodology followed in this book. Figure 5.2b shows the linear arrangement of phases of a proposed text mining process flow, based on CRISP-DM.

Phase 1: Determine the Purpose of the Study

Like any other project activity, text mining study starts with the determination of the purpose of the study. This requires a thorough understanding of the business case and what the study aims to accomplish. In order to achieve this understanding and define the aims precisely, we must assess the nature of the problem (or opportunity) that initiated the study. Often, we must interact closely with the domain experts in order to develop an in-depth appreciation of the underlying system, its structure, its system constraints and the available resources. Only then can we develop a set of realistic goals and objectives to govern the direction of the study.

Phase 2: Explore the Availability and the Nature of the Data

Once the purpose of the study is determined, we are ready to assess the availability, obtainability, and applicability of the necessary data in the context of the specific study. Some of the tasks in this phase include the following:

- Identification of the textual data sources (digitized or paper-based; internal or external to the organization)

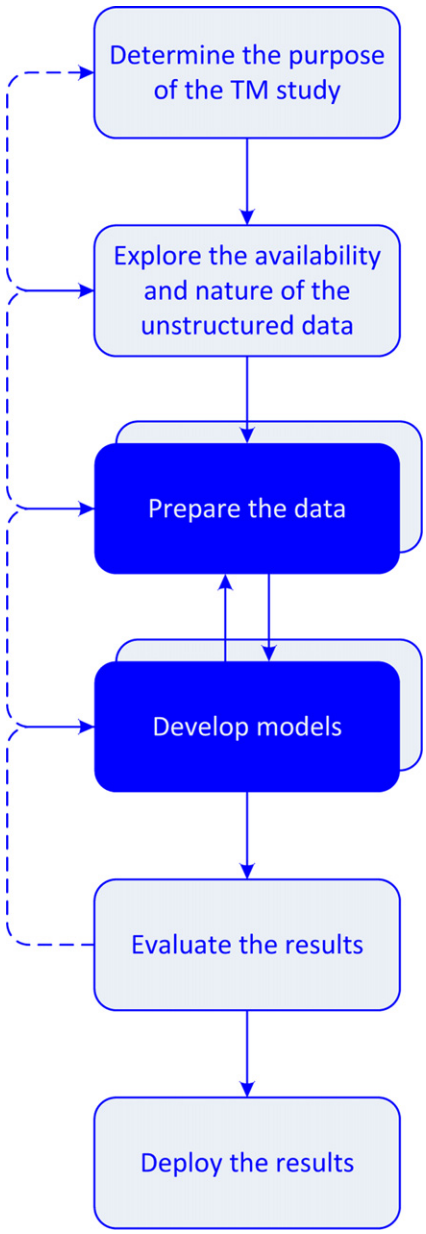

FIGURE 5.2b

A linear expression of the proposed text mining process flow, based on CRISP-DM. The two shaded boxes represent the most significant differences (and require more detailed explanations) for the CRISP-DM methodology as it applies to text mining.

- Assessment of the accessibility and usability of the data
- Collection of an initial set of data
- Exploration of the richness of the data (e.g., does it have the information content needed for the text mining study?)
- Assessment of the quantity and quality of the data. Once the exploration is concluded with positive outcomes, the next phase is to collect and integrate large quantities of data from various sources, which will be used in the study.

Phase 3: Prepare the Data and Phase 4: Develop and Assess the Models

Phases 3 and 4 present the most significant differences between data mining and text mining. In fact, many believe that text mining is nothing but data mining with a more laborious data collection and processing phase. In Figure 5.2b, Phases 2 and 3 are illustrated with shadowed boxes, indicating a more granular, in-depth, text mining–specific delineation. This is discussed after Phases 5 and 6.

Phase 5: Evaluate the Results

Once the models are developed and assessed for accuracy and quality from a data analysis perspective, we must verify and validate the proper execution of all of the activities. For example, we must verify that sampling was done properly and then repeat the steps to validate. Then (and only then) can we move forward to deployment. Taking on such a comprehensive assessment of the process helps to mitigate the possibility of error propagating into the decision-making process, potentially causing irreversible damage to the business. Often, as the analyst goes through these phases, he or she may forget the main business problem that started the study in the first place. This assessment step is meant to make that connection one more time to ensure that the models developed and verified are actually addressing the business problem and satisfying the objectives they were built to satisfy. If this assessment leads to the conclusion that one or more of the business objectives are not satisfied, or there still is some important business issue that has not been sufficiently considered, we should go back and correct these issues before moving into the deployment phase.

Phase 6: Deploy the Results

Once the models and the modeling process successfully pass the assessment process, they can be deployed (i.e., put into use). Deployment of these models can be as simple as writing a report that explains the findings of the study in a way that appeals to the decision makers, or it can be as complex as building a new business intelligence system around these models (or integrating them into an existing business intelligence system) so they can be used repetitively for better decision making. Some of the models will lose their accuracy and relevancy over time. They should be updated (or refined) periodically with new data. This can be accomplished by executing a new analysis process every so often to re-create the models, or, more preferably, the business intelligence system itself can be designed in a way that it refines its models automatically as new and relevant data become available. Even though developing such a sophisticated system that is capable of self-assessing and self-adjusting is a challenging undertaking, once accomplished, the results would be very satisfying.

Phases 3 and 4: Text Mining Process Specifications—A Functional Perspective

Figures 5.2a and 5.2b present the proposed text mining process in terms of process flow through Phases 3 and 4. Figure 5.3 represents a high-level context diagram of the text mining methodology from a functional architecture perspective. This context diagram is meant to present the scope of the process, specifically emphasizing its interfaces with its environment. In essence, it draws the boundaries around the process to explicitly show what is to be included (and/or excluded) from the representation of the text mining process.

Within the context of knowledge discovery, the primary purpose of text mining is to process unstructured (textual) data and structured and semistructured data (if relevant to the problem being addressed) to extract novel, meaningful, and actionable knowledge/information for better decision making. The inputs arrow in Figure 5.3 (on the left edge of the box) to the text-based knowledge discovery process box are the unstructured, semistructured, or structured data that are collected, stored, and made available to the process. The outputs arrow (from the right edge of the box) represents the context-specific knowledge products that can be used for decision making. The constraints (or *controls*) arrow entering at the top edge of the box represents software and hardware limitations, privacy issues, and the difficulties related to processing of the text that is presented in the form of natural language. The enablers entering the bottom of the box represents software tools, fast computers, domain expertise, and natural language processing (NLP) methods.

Figure 5.4 shows that Figure 5.3 can be decomposed into three linked subprocesses that we call "activities." Each has inputs, accomplishes some transformative process, and generates various outputs.

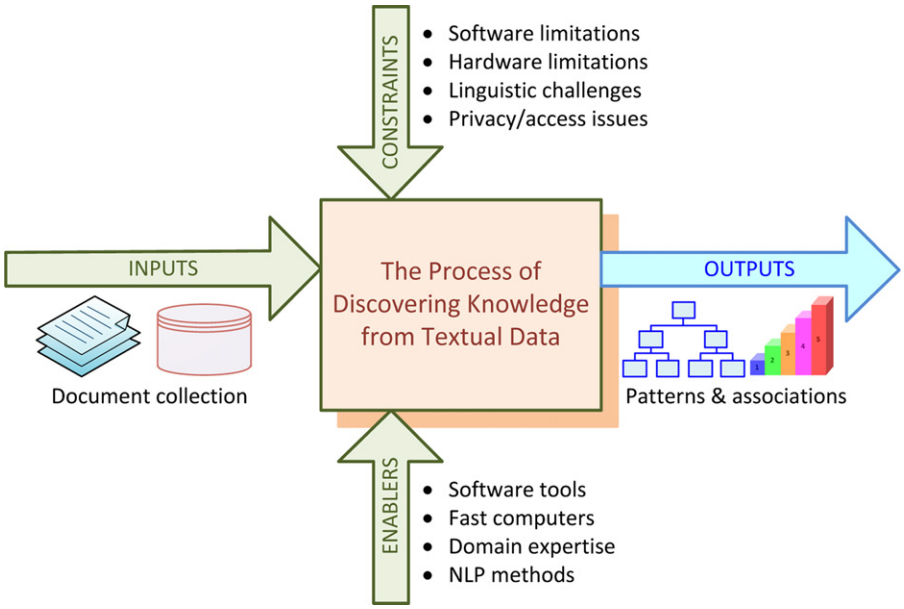

FIGURE 5.3
A high-level context diagram for the text mining process.

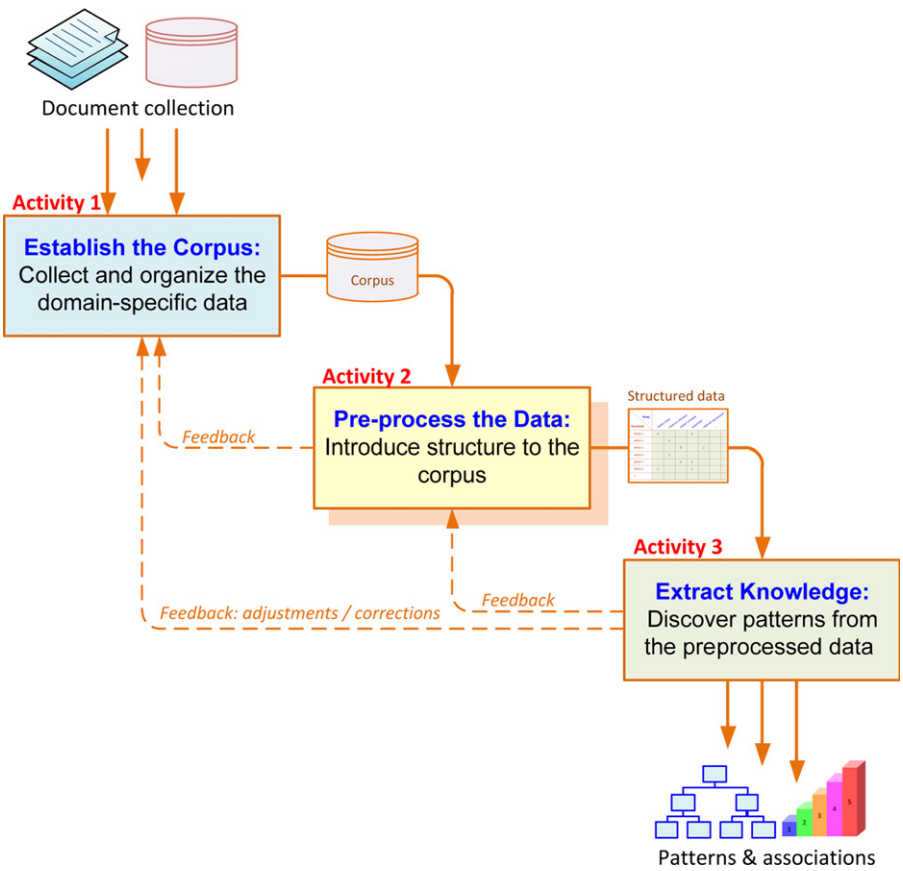

FIGURE 5.4
A detailed view of the context diagram for text mining.

If, for some reason, the output of a subprocess is not what was expected or emerges at an unsatisfactory level, feedback loops redirect information flow to a previous task to permit adjustments and corrections.

Phase 3, Activity 1: Establish the Corpus

The purpose of Activity 1 in Phase 3 is to collect all of the documents that are relevant to the problem being addressed (see Figure 5.4). The quality and quantity of the data are the most important elements of both data mining or text mining projects. Sometimes in a text mining project, the document collection is readily available and is accompanied by the project description (e.g., conducting sentiment analysis on customer reviews of a specific product or service). But usually the text miner is required to identify and collect the problem-specific document collection using either manual or automated techniques (e.g., a web crawler that periodically collects relevant news excerpts from several websites). Data collection may include textual documents, HTML files, emails, web posts, and short notes. In

addition to normal textual data, voice recordings may be included by transcribing using speech-recognition algorithms.

Once collected, the text documents are transformed and organized in a manner such that they are all represented in the same form (e.g., ASCII text files) for computer processing. The organization of these documents can be as simple as a collection of digitized text excerpts stored in a file folder, or it can be a list of links to a collection of web pages in a specific domain. Many commercially available text mining software tools could accept these web pages as input and convert them into a flat file for processing. Alternatively, flat files can be prepared outside the text mining software and then presented as the input to the text mining application.

Phase 3, Activity 2: Preprocess the Data

In this activity, the digitized and organized documents (the corpus) are used to create a structured representation of the data, often referred to as the *term–document matrix* (TDM). Commonly, the TDM consists of rows represented by documents and columns representing terms. The relationships between the terms and the documents are characterized by indices, which are relational measures, such as how frequently a given term occurs in a document. Figure 5.5 illustrates a simplified example of a TDM.

The goal of Activity 2 is to convert the list of organized documents (the corpus) into a TDM where the cells are filled with the most appropriate indices. The assumption we make here is that the "meaning" of a document can be represented with a list and frequency of the terms used in that document. But are all terms equally important when characterizing documents? Obviously, the answer is "no." Some terms, such as articles, auxiliary verbs, and terms used in almost all of the documents in the corpus, have no distinguishing power and therefore should be excluded

Documents \ Terms	market share	resource utilization	project schedule	acquisition	material requirement	...	
Article 1	1			1			
Article 2		1					
Article 3			3		1		
Article 4		1					
Article 5			2	1			
Article 6	1			1			
...							

FIGURE 5.5
A term-by-document matrix

from the indexing process. This list of terms, commonly called *stopterms*, is often specific to the domain of study and should be identified by the domain experts. On the other hand, one might choose a set of predetermined terms under which the documents are to be indexed; this list of terms is conveniently called *include terms* or *dictionary*. Additionally, synonyms (pairs of terms that are to be treated the same) and specific phrases (e.g., "Supreme Court") can also be provided so the index entries are more accurate. Figure 5.6 shows a more detailed view of the TDM with its four tasks.

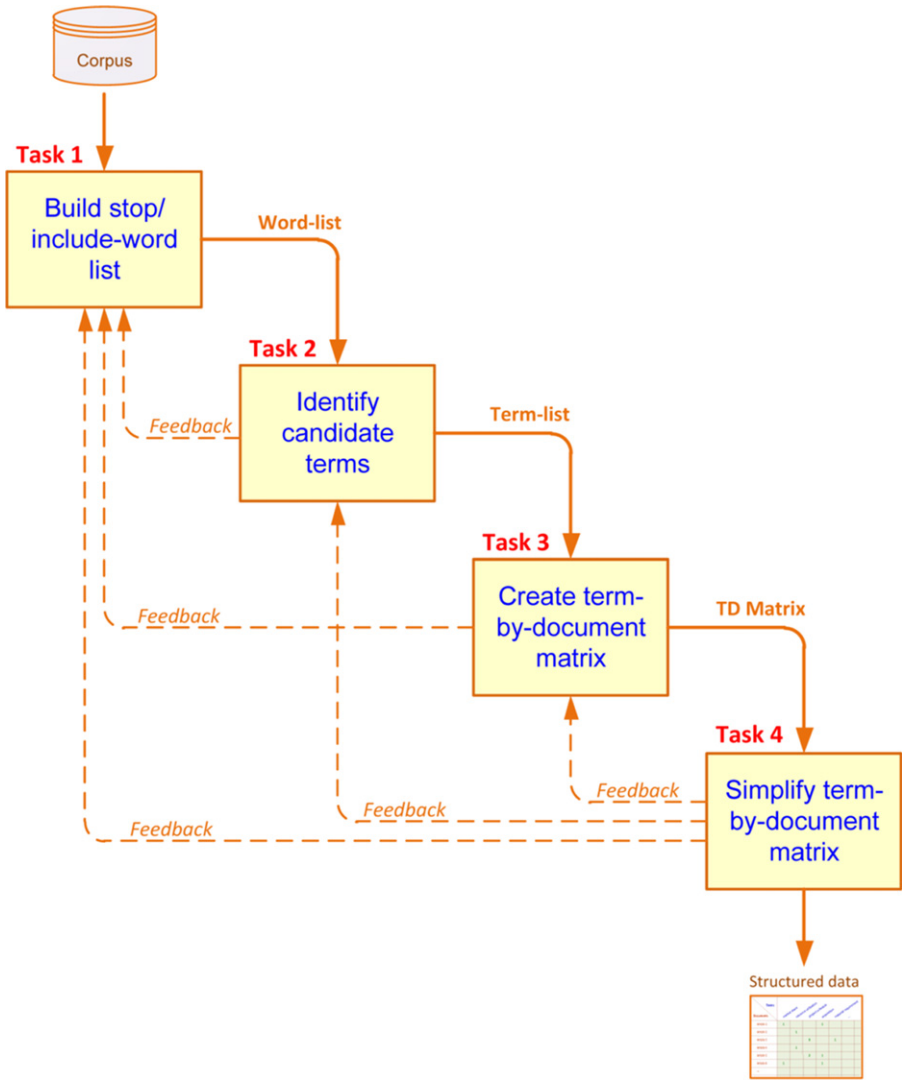

FIGURE 5.6
Decomposition of Activity 2 (preprocess the data) to the task level.

Task 1

The first task generates stopterms (or include terms) along with synonyms and specific phrases.

Task 2

The term list is created by *stemming* or *lemmatization*, which refers to the reduction of terms to their simplest forms (i.e., roots). An example of stemming is to identify and index different grammatical forms or declinations of a verb as the same term. For example, stemming will ensure that *model*, *modeling*, and *modeled* will be recognized as the term *model*. In this way, stemming will reduce the number of distinct terms and increase the frequency of some terms. Stemming has two common types:

1. *Inflectional stemming:* This aims to regularize grammatical variants such as present/past and singular/plural (this is called *morphological analysis* in computational linguistic). The degree of difficulty varies significantly from language to language.
2. *Stemming to the root:* This aims to reach a root form with no inflectional or derivational prefixes and suffixes, which may lead to the least number of terms.

Task 3

Create the TDM. In task 3, a numeric two-dimensional matrix representation of the corpus is created. Generation of the first form of the TDM includes three steps:

1. Specifying all the documents as rows in the matrix
2. Identifying all of the unique terms in the corpus (as its columns), excluding the ones in the stop term list
3. Calculating the occurrence count of each term for each document (as its cell values)

If the corpus includes a rather large number of documents (as is commonly the case), then it is common for the TDM to have a very large number of terms. Processing such a large matrix might be time consuming, and, more importantly, it might lead to extraction of inaccurate patterns. These dangers of large matrices and time-consuming operations pose the following two questions:

- What is the best representation of the indices for optimal processing by text mining programs?
- How can the dimensionality of this matrix be reduced to a more manageable size to facilitate more efficient and effective processing?

To answer question #1, we must evaluate various forms of representation of the indices. One approach is to transform the term frequencies. Once the input documents are indexed and the initial term frequencies (by document) have been computed, a number of additional transformations can be performed to summarize and aggregate the extracted information. Raw term frequencies reflect the relative prominence of a term in each document. Specifically, terms that occur with greater frequency in a document may be the best descriptors of the contents of that document. However, it is not reasonable to assume that the term counts themselves are proportional to their importance as descriptors of the documents. For example, even though a term occurs three times more often in document A than in document B, it is not necessarily reasonable to conclude that this term is three times as important a descriptor of document *B* as it is for document *A*.

In order to have a more consistent TDM for further analysis, these raw indices should be *normalized*. In statistical analysis, normalization consists of dividing multiple sets of data by a common value in order to eliminate different effects of different scales among data elements to be compared. Raw frequency values can be normalized using a number of alternative methods. The following are few of the most commonly used normalization methods (StatSoft, 2010):

- **Log frequencies.** The raw frequencies can be transformed using the log function. This transformation would "dampen" the raw frequencies and how they affect the results of subsequent analysis.
- $$f(wf) = 1 + \log(wf) \quad \text{for } wf > 0$$

  In the formula, wf is the raw term frequency and $f(wf)$ is the result of the log transformation. This transformation is applied to all of the raw frequencies in the TDM where the frequency is greater than zero.
- **Binary frequencies.** Likewise, an even simpler transformation can be used to enumerate whether a term is used in a document.

$$f(wf) = 1 \quad \text{for } wf > 0$$

  The resulting TDM matrix will contain only 1s and 0s to indicate the presence or absence of the respective terms. Again, this transformation will dampen the effect of the raw frequency counts on subsequent computations and analyses.
- **Inverse document frequencies.** In addition to normalized frequency of terms, the importance of a given term in each document (relative document frequency or *df*) is also an important aspect to include in the analysis. For example, a term such as *guess* may occur frequently in all documents, whereas another term, such as *software*, may appear only a few times. The reason is that one might make *guesses* in various contexts, regardless of the specific topic, whereas *software* is a more semantically focused term that is likely to occur only in documents that deal with computer software. A common and very useful transformation that reflects both the specificity of terms (relative document frequencies) as well as the overall frequencies of their occurrences (transformed term frequencies) is the so-called *inverse document frequency* (Manning and Schutze, 1999). This transformation for the ith term and jth document can be written as:

$$idf(i,j) = \begin{cases} 0 & \text{if } wf_{ij} = 0 \\ (1 + \log(wf_{ij}))\log\dfrac{N}{df_i} & \text{if } wf_{ij} \geq 1 \end{cases}$$

where wf_{ij} is the normalized frequency of the ith term in the jth document, df_i is the document frequency for the ith term (the number of documents that include this term), and N is the total number of documents. You can see that this formula includes both the dampening of the simple-term frequencies via the log function (described previously) and a weighting factor that evaluates to 0 if the term occurs in all of the documents [i.e., $\log(N/N = 1) = 0$], and to the maximum value when a term only occurs in a single document [i.e., $\log(N/1) = \log(N)$]. It can be seen easily how this transformation will create indices that reflect both the relative frequencies of occurrences of terms, as well as their document

frequencies representing semantic specificities for a given document. This is the most commonly used transformation in the field.

This brings us to how to reduce the dimensionality of the TDM (question #2). Because, the TDM is often very large and rather sparse (most of the cells filled with zeros), this answer is more tractable to handle. Several options are available for reducing such matrices to a manageable size:

- A domain expert goes through the list of terms and eliminates those that do not make much sense for the context of the study (this is a manual, labor-intensive process).
- Eliminate terms with very few occurrences in very few documents.
- Transform the matrix using singular value decomposition.

Singular Value Decomposition

Singular value decomposition (SVD) is a method of representing a matrix as a series of linear approximations that expose the underlying meaning-structure of the matrix. The goal of SVD is to find the optimal set of factors that best predict the outcome. During data preprocessing prior to text mining operations, SVD is used in latent semantic analysis (LSA) to find the underlying meaning of terms in various documents.

In more technical terms, SVD is closely related to principal components analysis in that it reduces the overall dimensionality of the input matrix (number of input documents by number of extracted terms) to a lower dimensional space (a matrix of much smaller size with fewer variables), where each consecutive dimension represents the largest degree of variability (between terms and documents) possible (Manning and Schutze, 1999). Ideally, the analyst might identify the two or three most salient dimensions that account for most of the variability (differences) between the terms and documents, thus identifying the latent semantic space (is this term the same as *lower dimensional space*?) that organizes the terms and documents in the analysis. When these dimensions are identified, they represent the underlying "meaning" of what is contained (discussed or described) in the documents. For example, assume that matrix A represents an $m \times n$ term occurrence matrix, where m is the number of input documents and n is the number of terms selected for analysis. The SVD computes the $m \times r$ orthogonal matrix U, $n \times r$ orthogonal matrix V, and $r \times r$ matrix D, so $A = UDV'$ and r is the number of eigenvalues of $A'A$.

Phase 3, Activity 3: Extract the Knowledge

Novel patterns are extracted in the context of the specific problem being addressed, using the well-structured TDM, and possibly augmented with other structured data elements (such as numerical and/or nominal variables, potentially including the time and place specifications of the documents). These are the main categories of knowledge extraction methods in text mining studies:

- Prediction (e.g., classification, regression, and time-series analysis)
- Clustering (e.g., segmentation and outlier analysis)
- Association (e.g., affinity analysis, link analysis, and sequence analysis)
- Trend analysis

Classification

Arguably the most common knowledge discovery topic in analyzing complex data sources is the *classification* of certain objects or events into predetermined classes (or categories). The goal of classification

is to assign the data instance into a predetermined set of classes (or categories). As it applies to the domain of text mining, the task is known as *text categorization*, where for a given set of categories and a collection of text documents, the challenge is to find the correct topic (subject or concept) for each document. This challenge is met with building models developed with a training data set that include both the documents and actual document categories. Today, automated text categorization is applied in a variety of contexts, including iterative (automatic or semiautomatic) indexing of text, spam filtering, web page categorization under hierarchical catalogs, automatic generation of metadata, detection of genre, and many others.

The two main approaches to text classification are expert systems (via the use of knowledge engineering techniques) and classification modeling (via the use of statistical and/or machine-learning techniques). With the expert system approach, an expert's knowledge about the categories is encoded into the classification system using a declarative representation in the form of production rules. With the machine-learning approach, a generalized inductive process is employed to build a classifier by "learning" from a set of preclassified examples. As the number of documents increases at an exponential rate and as the availability of knowledge experts becomes scarcer, the popularity trend is shifting toward the machine-learning–based automated classification techniques.

Clustering

Clustering is an unsupervised process whereby objects or events are placed into "natural" groupings called *clusters*. An unsupervised process is one that uses no pattern or prior knowledge to guide the clustering process. Text categorization is a supervised process, where a collection of preclassified training examples is used to develop a model based on the descriptive features of the classes in order to classify a new unlabeled example. In the unsupervised clustering process, the problem is to group an unlabeled collection of objects (e.g., documents, customer comments, web pages) into meaningful clusters without any prior knowledge.

Clustering is useful in a wide range of applications, from document retrieval to enabling better web content searches. In fact, one of the prominent applications of clustering is the analysis and navigation of very large text collections, such as web pages. The basic underlying assumption is that relevant documents tend to be more similar to one another than to irrelevant ones. If this assumption holds, the clustering of documents based on the similarity of their content improves search effectiveness (Feldman and Sanger, 2007).

The two most popular clustering methods are scatter/gather clustering and query-specific clustering. *Scatter/gather* method uses clustering to enhance the efficiency of human browsing of documents when a specific search query cannot be formulated. In a sense, the method dynamically generates a table of contents for the collection and adapts and modifies it in response to the user selection. On the other hand, *query-specific clustering* method employs a hierarchical clustering approach where the most relevant documents to the posed query appear in small tight clusters that are nested larger clusters containing less similar documents, creating a spectrum of relevance levels among the documents. This method performs consistently well for document collections of relatively large sizes.

Association

Association is the process of finding affinities/correlations among different data elements (objects or events). In the retail industry, association analysis is often called market basket analysis. The primary idea in generating association rules is to identify the frequent sets of "things" that go together in

a specific context. A famous example in retail is the association of beer and diapers in the same shopping cart (related to Monday night football games on TV).

In text mining, associations refer specifically to the direct relationships between concepts (or terms) or sets of concepts. An association rule, $X \Rightarrow Y$, relating two frequent concept sets X and Y, can be quantified (or substantiated) by the two basic measures, support and confidence. *Confidence* is the percent of documents that include all the concepts in Y within the same subset of those documents that include all the concepts in X. *Support* is the percentage (or number) of documents that include all the concepts in X and Y. For instance, in a document collection the concept "project failure" may appear most often in association with "enterprise resource planning" and "customer relationship management" with support of 4% and confidence of 55%, meaning that 4% of the documents in the corpus had all three concepts presented together in the same document, and 55% of the documents that included "project failure" also included "enterprise resource planning" and "customer relationship management."

In an interesting text mining study, association analysis was used to study published literature (news articles, academic publication and web postings) to map out the outbreak and progress of bird flu (Mahgoub et al., 2008). The principal goal of this study was to automatically identify the associations among the geographical areas, spreading across species, and countermeasures (i.e., treatments).

A special case of association analysis is where the concepts are associated with one another in an orderly means (e.g., a sequence in which the concepts tend to appear) or over a specific time period. This type of association analysis is called *trend analysis*, which is briefly explained in the following section.

Trend Analysis

The main goal in trend analysis is to find the time-dependent changes for an object or event. Often, trend analysis in text mining is based on the notion that various types of concept distributions over time are functions of the specific document collections; that is, different collections of the same topic representing different time intervals may lead to different concept distributions. It is therefore possible to compare the time-varying changes in two concept distributions that are otherwise identical except that they are from different subcollections. One notable direction of this type of analysis is having two collections from the same source (such as from the same set of academic journals) but from different points in time. Delen and Crossland (2008) applied trend analysis to a large number of academic articles (published in three highly rated academic journals) to identify the evolution of key concepts in the field of information systems.

EXAMPLE 1: AN EXPLORATORY LITERATURE SURVEY USING TEXT MINING

The explosion of text in various literature domains has rendered literature search and review as a very complex and voluminous operation. When creating new text in the process of extending the body of knowledge, it has always been crucial to gather, organize, analyze, and assimilate existing information from the literature in a particular discipline. The thoroughness of a literature search is increasingly difficult to attain in the face of the increasing abundance of potentially significant research reported in related fields—and even in previously unrelated fields. What was unrelated previously might indeed be relevant now.

In new streams of research, the researcher's task may be even more tedious and complex. Trying to ferret out relevant work that others have reported may be difficult, at best, and perhaps nearly

impossible when traditional, largely manual reviews of published literature are required. Even with a legion of dedicated graduate students or helpful colleagues, adequate coverage of all potentially relevant published work is problematic.

Example 2: Semiautomated Analysis

In a study, Delen and Crossland (2008) proposed a method to greatly assist and enhance the efforts of the researchers by enabling a semiautomated analysis of large volumes of published literature with text mining. Using standard digital libraries and online publication search engines, the authors 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 *Journal of Management Information Systems* (JMIS). In order to work with the same time span for all three journals (to enable future comparative studies), the journal with the most recent starting date for the availability of a digital version was used as the start time for this study. For each article, extracted data included the title, abstract, author list, published keywords, volume, issue number, and year of publication. The extracted data, including all of the articles and their above mentioned features, was loaded into a simple database file. Also included in the combined data set was a field that designated the journal type of each article to serve future segmentation operations and discriminatory analyses. Editorial notes, research notes, and executive overviews were omitted from the collection. At the end, over 900 articles were included in the corpus of their study. Table 5.1 shows a snapshot of how their data were organized in a tabular format.

Table 5.1 A Snapshot of the Data Represented in a Tabular Format

Journal	Year	Author(s)	Title	Vol/No	Pages	Keywords	Abstract
MISQ	2005	A. Malhotra, S. Gosain and O. A. El Sawy	Absorptive capacity configurations in supply chains: Gearing for partner-enabled market knowledge creation	29/1	145-187	knowledge management supply chain absorptive capacity interorganizational information systems configuration approaches	The need for continual value innovation is driving supply chains to evolve from a pure transactional focus to leveraging interorganizational partner ships for sharing
ISR	1999	D. Robey and M. C. Boudreau	Accounting for the contradictory organizational consequences of information technology: Theoretical directions and methodological implications	2-Oct	167-185	organizational transformation impacts of technology organization theory research methodology intraorganizational power electronic communication mis implementation culture systems	Although much contemporary thought considers advanced information technologies as either determinants or enablers of radical organizational change, empirical studies have revealed inconsistent findings to support the deterministic logic implicit in such arguments. This paper reviews the contradictory
JMIS	2001	R. Aron and E. K. Clemons	Achieving the optimal balance between investment in quality and investment in self-promotion for information products	18/2	65-88	information products internet advertising product positioning signaling signaling games	When producers of goods (or services) are confronted by a situation in which their offerings no longer perfectly match consumer preferences, they must determine the extent to which the advertised features of
…	…	…	…	…	…	…	…

Using these data, the authors conducted two studies. The first exploratory study was to look at the time-varying perspective of the three journals (i.e., evolution of research topics over time). In order to conduct a time-based associative study, they divided the 12-year period (from 1994 to 2005) into four 3-year periods for each of the three journals. This framework led to 12 text mining experiments with 12 mutually exclusive data sets consisting of abstracts of articles. For each of the 12 data sets, text mining was used to extract the most descriptive terms from these collections abstracts. Results were tabulated and examined for time-varying changes in the terms published in these three journals.

A second exploration used the complete data set (including all three journals and all four periods) and employed clustering to extract knowledge from the abstracts. Clustering is arguably the most commonly used text mining technique employed today. It was used in this study to identify the natural groupings of the articles and then list the most descriptive terms that characterized those clusters. Singular value decomposition was used to reduce the dimensionality of the TDM, and then an expectation-maximization algorithm was used to create the clusters. Several experiments were performed to identify the *optimal* number of clusters, which proved to be nine. After the construction of the nine clusters, the content of those clusters was analyzed from two perspectives: representation of the journal type (Figure 5.7) and representation of time (Figure 5.8). The idea was to explore the potential similarities and differences among the three journals and potential changes in the importance associated with those clusters. The importance was evaluated according to two questions: 1) are there clusters that represent different research themes specific to a single journal? and 2) is there a time-varying characterization of those clusters? Several interesting patterns discovered in this study are illustrated in Figures 5.7 and 5.8. Figure 5.7 shows the representation of the three journals (in terms of

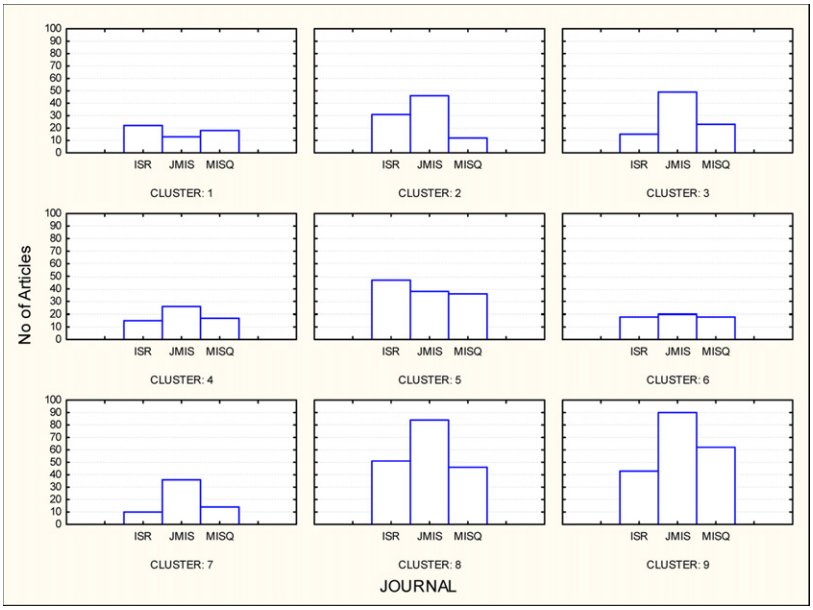

FIGURE 5.7
Distribution articles for the three journals over the nine clusters.

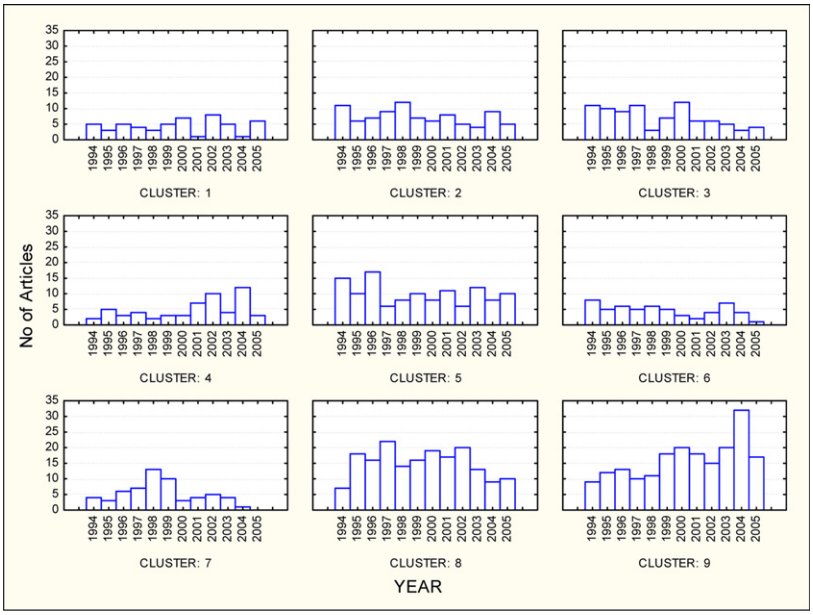

FIGURE 5.8
Distribution of articles over time for each of the nine clusters. *Source: Delen and Crossland, 2008.*

number of articles included) for each of the nine clusters, indicating the overlap (or lack thereof) among the three journals in terms of publishing topics that are represented in nine clusters. Figure 5.8 illustrates the distribution of the articles during the 12-year period for each of the nine clusters.

POSTSCRIPT

Chapters 1–5 present the history, theory, application areas, and a proposed process model to accomplish text mining projects. Chapter 6 introduces you to three common text mining tools that you might use for text mining projects. These text mining software packages are used also in one or more tutorials, although other tools are used also.

References

Chapman, P., J. Clinton, R. Kerber, T. Khabanza, T. Reinartz, C. Shearer, and R. Wirth. (2000). "CRISP-DM—Step-by-step data mining guide." SPSS, Chicago, IL.

Delen, D., and M. Crossland, "Seeding the Survey and Analysis of Research Literature with Text Mining," *Expert Systems with Applications* Vol. 34, No. 3, 2008, pp. 1707–1720.

Feldman, R., and J. Sanger. (2007). *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Boston: ABS Ventures.

Mahgoub, H., D. Rösner, N. Ismail, and F. Torkey. (2008). "A Text Mining Technique Using Association Rules Extraction." *International Journal of Computational Intelligence*, Vol. 4, No. 1, pp. 21–28.

Manning, C. D., and H. Schutze. (1999). *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press.

StatSoft. (2010). *Statistica Data and Text Miner User Manual*. Tulsa, OK: StatSoft, Inc.