Text Mining: Technique, Tools & Mining Process

Shravan Kumar#1, Naazish Rahim*2

# M. Tech Computer Science & Engineering, Lakshmi Narain College of Technology
Jabalpur (M.P), INDIA, shravan51090@gmail.com

*2 Dept. Computer Science & Engineering, Lakshmi Narain College of Technology Jabalpur (M.P), INDIA naazish.rahim786@gmail.com

Abstract— **Text mining is a burgeoning new field that attempts to glean meaningful information from natural language text. It may be loosely characterized as the process of analyzing text to extract information that is useful for particular purposes. Compared with the kind of data stored in databases, text is unstructured, amorphous, and difficult to deal with algorithmically. Nevertheless, in modern culture, text is the most common vehicle for the formal exchange of information. The field of text mining usually deals with texts whose function is the communication of factual information or opinions, and the motivation for trying to extract information from such text automatically is compelling—even if success is only partial. Four years ago, Hearst [Hearst, 1999] wrote that the nascent field of "text data mining" had "a name and a fair amount of hype, but as yet almost no practitioners." It seems that even the name is unclear: the phrase "text mining" appears 17 times as often as "text data mining" on the Web, according to a popular search engine (and "data mining" occurs 500 times as often).**

Keywords— **Cryptography Technique, Network security, Authentication Process.**

I. Introduction

Text mining, also referred to as text data mining, roughly equivalent to text analytics, refers to the process of deriving high-quality information from text. High-quality information is typically derived through the devising of patterns and trends through means such as statistical pattern learning. Text mining usually involves the process of structuring the input text (usually parsing, along with the addition of some derived linguistic features and the removal of others, and subsequent insertion into a database), deriving patterns within the structured data, and finally evaluation and interpretation of the output. 'High quality' in text mining usually refers to some combination of relevance, novelty, and interestingness. Typical text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling (i.e., learning relations between named entities).

Text analysis involves information retrieval, lexical analysis to study word frequency distributions, pattern recognition, tagging/annotation, information extraction, data mining techniques including link and association analysis, visualization, and predictive analytics. The overarching goal is, essentially, to turn text into data for analysis, via application of natural language processing (NLP) and analytical methods.

A typical application is to scan a set of documents written in a natural language and either model the document set for predictive classification purposes or populate a database or search index with the information extracted.

Moreover, the meaning of either phrase is by no means clear: Hearst defines data mining, information access, and corpus-based computational linguistics and discusses the relationship of these to text data mining—but does not define that term. The literature on data mining is far more extensive, and also more focused: there are numerous textbooks and critical reviews that trace its development from roots in machine learning and statistics. Text mining emerged at an unfortunate time in history. Data mining was able to ride the back of the high technology extravaganza throughout the 1990s, and became firmly established as a widely-used practical technology—though the dot com crash may have hit it harder than other areas [Franklin, 2002]. Text mining, in contrast, emerged just before the market crash—the first workshops were held at the International Machine Learning Conference in July 1999 and the International Joint Conference on Artificial Intelligence in August 1999—and missed the opportunity to gain a solid foothold during the boom years.

The phrase ―text mining is generally used to denote any system that analyzes large quantities of natural language text and detects lexical or linguistic usage patterns in an attempt to extract probably useful (although only probably correct) information [Sebastiani, 2002].

The article's major section follows: an introduction to the great variety of tasks that involve mining plain text. We then examine the additional leverage that can be obtained when mining semi-structured text such as pages of the World-Wide Web, which opens up a range 2 of new techniques that do not apply to plain text. Following that we indicate, by example, what automatic text mining techniques may aspire to in the future by briefly describing how human ―text miners‖ who are information researchers rather than subject-matter experts may be able to discover new scientific hypotheses solely by analyzing the literature. Finally we review some basic techniques that underpin text mining systems, and look at software tools that are available to help with the work.

1.1 Applications

The technology is now broadly applied for a wide variety of government, research, and business needs. Applications can be sorted into a number of categories by analysis type or by business function. Using this approach to classifying solutions, application categories include:

- Enterprise Business Intelligence/Data Mining, Competitive Intelligence
- E-Discovery, Records Management
- National Security/Intelligence
- Scientific discovery, especially Life Sciences
- Sentiment Analysis Tools, Listening Platforms
- Natural Language/Semantic Toolkit or Service
- Publishing
- Automated ad placement
- Search/Information Access
- Social media monitoring

II. DIFFERENCE BETWEEN TEXT MINING & DATA MINING

Just as data mining can be loosely described as looking for patterns in data, text mining is about looking for patterns in text. However, the superficial similarity between the two conceals real differences. Data mining can be more fully characterized as the extraction of implicit, previously unknown, and potentially useful information from data [Witten and Frank, 2000]. The information is implicit in the input data: it is hidden, unknown, and could hardly be extracted without recourse to automatic techniques of data mining. With text mining, however, the information to be extracted is clearly and explicitly stated in the text. It's not hidden at all—most authors go to great pains to make sure that they express themselves clearly and unambiguously—and, from a human point of view, the only sense in which it is

―previously unknown is that human resource restrictions make it infeasible for people to read the text themselves. The problem, of course, is that the information is not couched in a manner that is amenable to automatic processing. Text mining strives to bring it out of the text in a form that is suitable for consumption by computers directly, with no need for a human intermediary.

Though there is a clear difference philosophically, from the computer's point of view the problems are quite similar. Text is just as opaque as raw data when it comes to extracting information— probably more so. Another requirement that is common to both data and text mining is that the information extracted should be ―potentially useful. In one sense, this means actionable—capable of providing a basis for actions to be taken automatically. In the case of data mining, this notion can be expressed in a relatively domain-independent way: actionable patterns are ones that allow non-trivial predictions to be made on new data from the same source. Performance can be measured by counting successes and failures, statistical techniques can be applied to compare different data mining methods on the same problem, and so on. However, in many text mining situations it is far harder to characterize what

―actionable means in a way that is independent of the particular domain at hand. This makes it difficult to find fair and objective measures of success. In many data mining applications, ―potentially useful is given a different interpretation: the key for success is that the information extracted must be comprehensible in that it helps to explain the data. This is necessary whenever the result is intended for human consumption rather than (or as well as) a basis for automatic action. This criterion is less applicable to text mining because, unlike data mining, the input itself is comprehensible. Text mining with comprehensible output is tantamount to summarizing salient features from a large body of text, which is a subfield in its own right: text summarization.

III. TEXT MINING AND NATURAL LANGUAGE PROCESSING

Text mining appears to embrace the whole of automatic natural language processing and, arguably, far more besides—for example, analysis of linkage structures such as citations in the academic literature and hyperlinks in the Web literature, both useful sources of information that lie outside the traditional domain of natural language processing. But, in fact, most text mining efforts consciously shun the deeper, cognitive, aspects of classic natural language processing in favor of shallower techniques more akin to those used in practical information retrieval.

The reason is best understood in the context of the historical development of the subject of natural language processing. The field's roots lie in automatic translation projects in the late 1940s and early 1950s, whose aficionados assumed that strategies based on word-for-word translation would provide decent and useful rough translations that could easily be honed into something more accurate using techniques based on elementary syntactic analysis.

IV. TEXT ANALYSIS PROCESS

Information retrieval or identification of a corpus is a preparatory step: collecting or identifying a set of textual materials, on the Web or held in a file system, database, or content corpus manager, for analysis. Although some text analytics systems apply exclusively advanced statistical methods, many others apply more extensive natural language processing, such as part of speech tagging, syntactic parsing, and other types of linguistic analysis.

Named entity recognition is the use of gazetteers or statistical techniques to identify named text features: people, organizations, place names, stock ticker symbols, certain abbreviations, and so on. Disambiguation — the use of contextual clues — may be required to decide where, for instance, "Ford" can refer to a former U.S. president, a vehicle manufacturer, a movie star, a river crossing, or some other entity. Recognition of Pattern Identified Entities: Features such as telephone numbers, e-mail addresses, quantities (with units) can be discerned via regular expression or other pattern matches. Coreference: identification of noun phrases and other terms that refer to the same object. Relationship, fact, and event Extraction: identification of associations among entities and other information in text Sentiment analysis involves discerning subjective (as opposed to factual) material and extracting various forms of attitudinal information: sentiment, opinion, mood, and emotion. Text analytics techniques are helpful in analyzing, sentiment at the entity, concept, or topic level and in distinguishing opinion holder and opinion object.

Quantitative text analysis is a set of techniques stemming from the social sciences where either a human judge or a computer extracts semantic or grammatical relationships between words in order to find out the meaning or stylistic patterns of, usually, a casual personal text for the purpose of psychological profiling etc.

V. LEGAL ISSUE IN CRYPTOGRAPHY

Text mining systems use a broad spectrum of different approaches and techniques, partly because of the great scope of text mining and consequent diversity of systems that perform it, and partly because the field is so young that dominant methodologies have not yet emerged. High-level issues: Training vs. knowledge engineering There is an important distinction between systems that use an automatic training approach to spot patterns in data and ones that are based on a knowledge engineering approach and use rules formulated by human experts. This distinction recurs throughout the field but is particularly star in the areas of entity extraction and information extraction. For example, systems that extract personal names can use hand-crafted rules derived from everyday experience. Simple and obvious rules involve capitalization, punctuation, single-letter initials, and titles; more complex ones take account of baronial prefixes and foreign forms. Alternatively, names could be manually marked up in a set of training documents and machine learning techniques used to infer rules that apply to test documents.

In general, the knowledge engineering approach requires a relatively high level of human expertise—a human expert who knows the domain, and the information extraction system, well enough to formulate high-quality rules. Formulating good rules is a demanding and time consuming task for human experts, and involves many cycles of formulating, testing, and adjusting the rules so that they perform well on new data. Markup for automatic training is clerical work that requires only the ability to recognize the entities in question when they occur. However, it is a demanding task because large volumes are needed for good performance. Some learning systems can leverage unmarked training data to improve the results obtained from a relatively small training set. For example, an experiment in document categorization used a small number of labeled documents to produce an initial model, which was then used to assign probabilistically-weighted class labels to unlabeled documents [Nigam et al., 1998]. Then a new classifier was produced using all the documents as training data. The procedure was iterated until the classifier remained unchanged. Another possibility is t bootstrap learning based on two different and mutually reinforcing perspectives on the data, a idea called ―co-training[Blum and Mitchell, 1998].

5.1 Low-level issues: Token identification

Dealing with natural language involves some rather mundane decisions that nevertheless strongly affect the success of the outcome. Tokenization, or splitting the input into words, is an important first step that seems easy but is fraught with small decisions: how to deal with apostrophes and hyphens, capitalization, punctuation, numbers, alphanumeric strings, whether the amount of white space is significant, whether to impose a maximum length on tokens, what to do with non-printing characters, and so on. It may be beneficial to perform some rudimentary morphological analysis on the tokens—removing suffixes [Porter, 1980] or representing them as words separate from the stem—which can be quite complex and is strongly language-dependent. Tokens may be standardized by using a dictionary to map different, but equivalent, variants of a term into a single canonical form. Some text mining applications (e.g. text summarization) split the input into sentences and even paragraphs, which again involves mundane decisions about delimiters, capitalization, and non-standard characters.

Once the input is tokenized, some level of syntactic processing is usually required. The simples operation is to remove stop words, which are words that perform well-defined syntactic roles but from a non-linguistic point of view do not carry information. Another is to identify common phrases and map them into single features. The resulting representation of the text as a sequence of word features is commonly used in many text mining systems (e.g. for information extraction).

5.2. Basic Techniques

Tokenizing a document and discarding all sequential information yields the ―bag of words representation mentioned above under document retrieval. Great effort has been invested over the years in a quest for document similarity measures based on this representation. One is to count the number of terms in common between the documents: this is called coordinate matching. This representation, in conjunction with standard classification systems from machine learning (e.g. Naïve Bayes and Support Vector Machines; see [Witten and Frank, 2000]), underlies most text categorization systems. It is often more effective to weight words in two ways: first by the number of documents in the entire collection in which they appear (―document frequency) on the basis that frequent word carry less information than rare ones; second by the number of times they appear in the particular documents in question (―term frequency). These effects can be combined by multiplying the term frequency by the inverse document frequency, leading to a standard family of document similarity measures (often called ―tf×idf). These form the basis of standard text categorization and information retrieval systems. A further step is to perform a syntactic analysis and tag each word with its part of speech. This helps to disambiguate different senses of a word and to eliminate incorrect analyses caused by rare word senses. Some part-of-speech taggers are rule based, while others are statistically based.

5.3. Tools

There is a plethora of software tools to help with the basic processes of text mining. A comprehensive and useful resource at nlp.stanford.edu/lionks/statnlp.html lists taggers, parsers, 20 language models and concordances; several different corpora (large collections, particular languages, etc.); dictionaries, lexical, and morphological resources; software modules for handling XML and SGML documents; and other relevant resources such as courses, mailing lists, people, and societies. It classifies software as freely downloadable and commercially available, with several intermediate categories.

One particular framework and development environment for text mining, called General Architecture for Text Engineering or GATE [Cunningham, 2002], aims to help users develop, evaluate and deploy systems for what the authors term ―language engineering. It provides support not just for standard text mining applications such as information extraction, but also for tasks such as building and annotating corpora, and evaluating the applications.

At the lowest level, GATE supports a variety of formats including XML, RTF, HTML, SGML, email and plain text, converting them into a single unified model that also supports annotation. There are three storage mechanisms: a relational database, a serialized Java object, and an XML base internal format; documents can be re-exported into their original format with or without annotations. Text encoding is based on Unicode to provide support for multilingual data processing, so that systems developed with GATE can be ported to new languages with no additional overhead apart from the development of the resources needed for the specific language.

VI. CONCLUSION

Text mining is a burgeoning technology that is still, because of its newness and intrinsic difficulty, in a fluid state—akin, perhaps, to the state of machine learning in the mid-1980s. Generally accepted characterizations of what it covers do not yet exist. When the term is broadly interpreted, many different problems and techniques come under its ambit. In most cases it is difficult to provide general and meaningful evaluations because the task is highly sensitive to the particular text under consideration. Document classification, entity extraction, and filling templates that correspond to given relationships between entities, are all central text mining operations that have been extensively studied. Using structured data such as Web pages rather than plain text as the input opens up new possibilities for extracting information from individual pages and large networks of pages. Automatic text mining techniques have a long way to go before they rival the ability of people, even without any special domain knowledge, to glean information from large document collections.

REFERENCES

[1] Liddell, Henry George; Scott, Robert; Jones, Henry Stuart; McKenzie, Roderick (1984). A Greek-English Lexicon. Oxford University Press.

[2] Rivest, Ronald L. (1990). "Cryptography". In J. Van Leeuwen. Handbook of Theoretical Computer Science 1. Elsevier.

[3] Biggs, Norman (2008). Codes: An introduction to Information Communication and Cryptography. Springer. p. 171.

[4] Bellare, Mihir; Rogaway, Phillip (21 September 2005). "Introduction". Introduction to Modern Cryptography. p. 10.

[5] Archived November 29, 2009 at the Wayback Machine "KDD-2000 Workshop on Text Mining - Call for Papers". Cs.cmu.edu. Retrieved 2015-02-23.

[6] Archived March 3, 2012 at the Wayback Machine Hobbs, Jerry R.; Walker, Donald E.; Amsler, Robert A. (1982). "Proceedings of the 9th conference on Computational linguistics" 1: 127–32. doi:10.3115/991813.991833.

[7] "Unstructured Data and the 80 Percent Rule". Breakthrough Analysis. Retrieved 2015-02-23.

[8] "Content Analysis of Verbatim Explanations". Ppc.sas.upenn.edu. Retrieved 2015-02-23.

[9] "A Brief History of Text Analytics by Seth Grimes". Beyenetwork. 2007-10-30. Retrieved 2015-02-23.

[10] Hearst, Marti A. (1999). "Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics": 3–10. doi:10.3115/1034678.1034679. ISBN 1-55860-609-2.

[11] "Full Circle Sentiment Analysis". Breakthrough Analysis. Retrieved 2015-02-23.

[12] Mehl, Matthias R. (2006). "Handbook of multimethod measurement in psychology": 141. doi:10.1037/11383-011. ISBN 1-59147-318-7.

[13] Zanasi, Alessandro (2009). "Proceedings of the International Workshop on Computational Intelligence in Security for Information Systems CISIS'08". Advances in Soft Computing 53: 53. doi:10.1007/978-3-540-88181-0_7. ISBN 978-3-540-88180-3.

[14] Cohen, K. Bretonnel; Hunter, Lawrence (2008). "Getting Started in Text Mining". PLoS Computational Biology 4 (1): e20. doi:10.1371/journal.pcbi.0040020. PMC 2217579. PMID 18225946. open access publication - free to read

[15] Jenssen, Tor-Kristian; Lægreid, Astrid; Komorowski, Jan; Hovig, Eivind (2001). "A literature network of human genes for high-throughput analysis of gene expression". Nature Genetics 28 (1): 21–8. doi:10.1038/ng0501-21. PMID 11326270.

[16] Masys, Daniel R. (2001). "Linking microarray data to the literature". Nature Genetics 28 (1): 9–10. doi:10.1038/ng0501-9. PMID 11326264.

[17] Joseph, Thomas; Saipradeep, Vangala G; Venkat Raghavan, Ganesh Sekar; Srinivasan, Rajgopal; Rao, Aditya; Kotte, Sujatha; Sivadasan, Naveen (2012). "TPX: Biomedical literature search made easy". Bioinformation 8 (12): 578–80. doi:10.6026/97320630008578. PMC 3398782. PMID 22829734.

[18] Archived October 4, 2013 at the Wayback Machine

[19] "Text Analytics". Medallia. Retrieved 2015-02-23.

[20] Coussement, Kristof; Van Den Poel, Dirk (2008). "Integrating the voice of customers through call center emails into a decision support system for churn prediction". Information & Management 45 (3): 164–74.doi:10.1016/j.im.2008.01.005.

[21] Coussement, Kristof; Van Den Poel, Dirk (2008). "Improving customer complaint management by automatic email classification using linguistic style features as predictors". Decision Support Systems 44(4): 870–82. doi:10.1016/j.dss.2007.10.010.

[22] Pang, Bo; Lee, Lillian; Vaithyanathan, Shivakumar (2002). "Proceedings of the ACL-02 conference on Empirical methods in natural language processing" 10: 79–86. doi:10.3115/1118693.1118704.