A Survey on Effective Classification for Text Mining using one-class SVM

Safdar Sardar Khan
Barkatullah University Institute of Technology
Barkatullah University, Bhopal M.P, India

Divakar Singh
Head CSE Deppt. Barkatullah University Institute of Technology
Barkatullah University, Bhopal M.P, India

ABSTRACT

Data mining techniques have been widely used to find new patterns and knowledge from large amount of data. While Bayesian models were widely used in the early days, more advanced machine learning methods, such as artificial neural networks and support vector machines, have been applied in recent years these techniques are used in different areas. The problem of text mining has gained increasing attention in recent years because of the large amounts of text data, which are created in a variety of social network, web, and other information-centric applications. Unstructured data is the easiest form of data which can be created in any application scenario. As a result, there has been a tremendous need to design methods and algorithms which can effectively process a wide variety of text applications. This paper will provide an overview of the different methods and algorithms which are common in the text domain, with a particular focus on mining methods.

General Terms

- To develop filtered text, removing stop words, stemming, html tags etc.
- Applying feature selection method TF-IDF.
- Extracting high quality ontopic text using PNLH algorithm.

Keywords

Data mining, text mining, text categorization, partially supervised learning, labelling unlabelled data, feature selection, information filtering, SVM.

1. INTRODUCTION

Most previous studies of data mining have focused on structured data, such as relational, transactional, and data warehouse data. However, in reality, a substantial portion of the available information is stored in text databases(or document databases), which consist of large collections of documents from various sources, such as news articles, research papers, books, digital libraries, e-mails messages, and web pages. Text databases are rapidly growing due to the increasing amount of information available in electronic form, such as electronic publications, various kings of electronic documents, e-mails, and the World Wide Web (which can also be viewed as a huge, interconnected, dynamic text databases). Nowadays most of the information is government, industry, business, and other institutions are stored electronically, in the form of text databases.

Data stored in most text databases are *semi structured data* in that they are neither completely unstructured nor completely structured. For example, a document may contain a few structured fields, such as *title, authors, publication date, and category,* and so on, but also contain some largely unstructured text components, such as *abstract* and *contents*. There have been a great deal of studies on the modeling and implementation of semi structured data in recent database research. Moreover, information retrieval techniques, such as text indexing methods, have been developed to handle unstructured documents.

Traditional information retrieval techniques become insufficient for the increasingly vast amounts of text data. Typically, only a small fraction of the many available documents will be relevant to a given individual user. Without knowing what could be in the documents, it is difficult to formulate effective queries for analysing and extracting useful information from the data. Users need tools to compare different documents, rank the importance and relevance of the documents, or find patterns and trends across multiple documents. Thus, text mining has become an increasingly popular and essential theme in data mining.

2. DEFINITION

"Text mining is finding interesting regularities in large textual datasets" where interesting means: non-trivial, hidden, previously unknown and potentially useful. "Text mining is finding semantic and abstract information from the surface form of textual data…"

Typical text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modelling (*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 main goal is, essentially, to turn text into data for analysis, via application of natural language processing (NLP) and analytical methods.

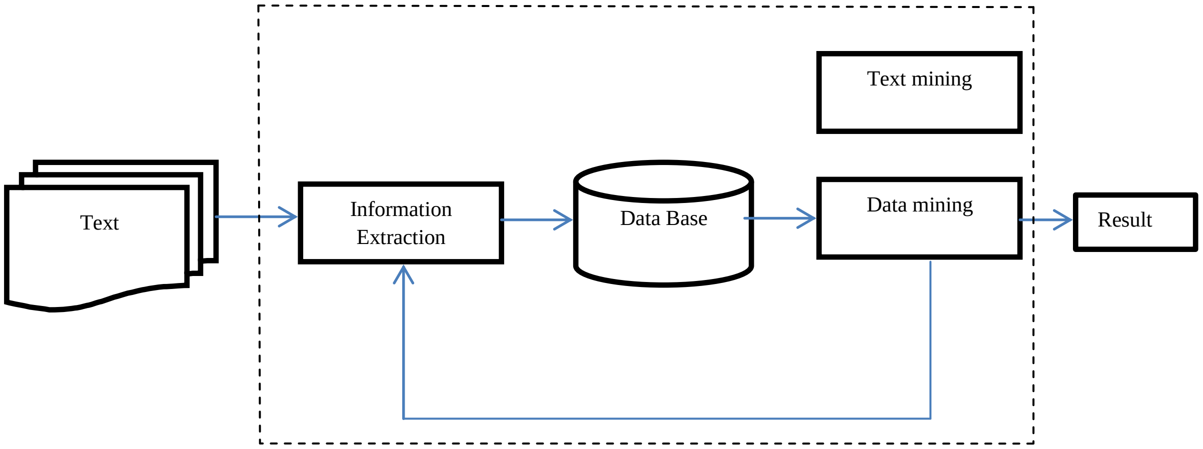

Figure.1 The architectural view of text mining.

2.1 Why text is tough and easy?

Text is Tough due to abstract concepts are difficult to represent, “Countless” combinations of subtle, abstract relationships among concepts, many ways to represent similar concepts (E.g. space ship, flying saucer, UFO), concepts are difficult to visualize, high dimensionality, tens or hundreds of thousands of features.

Text is Easy due tohighly redundant data, most of the methods count on this property. Just about any simple algorithm can get “good” results for simple tasks: Pull out “important” phrases, Find “meaningfully” related words, and create some sort of summary from documents.

2.2 Text Data Analysis:

Text mining is the discovery of interesting knowledge in text documents. It is a challenging issue to find accurate knowledge (or features) in text documents to help users to find what they want. The problem of *text mining*, i.e. discovering useful knowledge from unstructured or semi structured text, is attracting increasing attention [4, 8, 9, 2]. This paper suggests a new framework for text mining based on the integration of *Information Extraction* (IE) and Knowledge Discovery from Databases (KDD), *data mining*. KDD and IE are both topics of significant recent interest. KDD considers the application of statistical and machine-learning methods to discover novel relationships in large relational databases. IE concerns locating specific pieces of data in natural-language documents, thereby extracting structured information from free text.

Traditional data mining assumes that the information to be “mined” is already in the form of a relational database. Unfortunately, for many applications, electronic information is only available in the form of free natural-language documents rather than structured databases. Since IE addresses the problem of transforming a corpus of textual documents into a more structured database, the database constructed by an IE module can be provided to the KDD module for further mining of knowledge as illustrated in figure. Information extraction can play an obvious role in text mining as illustrated.

Due to the rapid growth of digital data made available in recent years, knowledge discovery and data mining have attracted a great deal of attention with an imminent need for turning such data into useful information and knowledge. Many applications, such as market analysis and business management, can benefit by the use of the information and knowledge extracted from a large amount of data. Knowledge discovery can be viewed as the process of nontrivial extraction of information from large databases, information that is implicitly presented in the data, previously unknown and potentially useful for users. Data mining is therefore an essential step in the process of knowledge discovery in databases.

3. INFORMATION RETRIEVAL

Information retrieval (IR) is a field that has been developing in parallel with database system for many years. Unlike the field of data bases systems, which has focused on query and transaction processing of structured data, information retrieval is concerned with the organization and retrieval of information from a large number of text-based documents. Since information retrieval and data base systems each handle different kind of data, some data base systems problem are usually not presenting information retrieval systems, such as concurrency control, recovery, transaction management, and update. Also, such common information retrieval problems are usually not encountered in traditional data base systems, such as unstructured documents, approximate search based on key words, and the notion of relevance.

Due to the abundance of text information, information retrieval has found many applications. There exist many information retrieval systems, such as on-line library catalogue systems, on-line documents management systems, and the more recently developed Web search engines.

3.1 Basic measures for text retrieval:

“Suppose that a text retrieval system has just retrieved a number of documents for me based on my input in the form of a query. How can be assess how accurate or correct the system was? ” Let the set of documents relevant to a query be denoted as {Relevant}, and the set of document retrieved be denoted as {Retrieved}. The set of documents that are both relevant and retrieved is denoted as {Relevant}∩{Retrieved}, as shown in Venn diagram of figure 2. There are two basic measures for assessing the quality of text retrieval[10][11]:

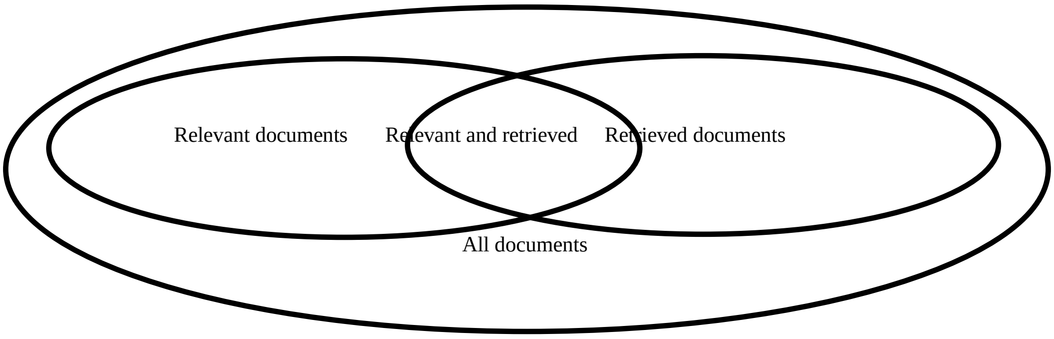

Figure.2 Relationship between all documents.

Precision: This is the percentage of retrieved documents that are in fact relevant to the query (i.e., "correct" response). It is formally defined as

$$\text{Precision}= \frac{|\{\text{Relevant}\}\cap\{\text{Retrieved}\}|}{|\{\text{Retrieved}\}|}$$

Recall: This is the percentage of documents that are relevant to the query and were, in fact, retrieved. It is formally defined as

$$\text{Recall} = \frac{|\{\text{Relevant}\}\cap\{\text{Retrieved}\}|}{|\{\text{Relevant}\}|}$$

An information retrieval system often needs to trade off recall for precision or vice versa. One commonly used trade-off is the F-score, which is defined as the harmonic mean of recall and precision:

$$F - score = \frac{\text{Recall} * \text{precision}}{(\text{Recall} + \text{precision})/2}$$

The harmonic mean discourages a system that sacrifices one measure for another too drastically.

3.2 Feature selection method

The basic idea of the vector space model is the following: We represent a document and a query both as vectors in a high-dimensional space corresponding to all the keywords and use an appropriate similarity measure to compute the similarity between the query vector and the document vector. The similarity values can then be used for ranking documents.

"How do we tokenize text?" The first step in most retrieval systems is to identify keywords for representing documents, a preprocessing step often called tokenization. To avoid indexing useless words, a text retrieval system often associates a stop list with a set of documents. A stop list is a set of words that are deemed "irrelevant." For example, a, the, of, for, with, and so on are stop words, even though they may appear frequently. Stop lists may vary per document set. For example, database systems could be an important keyword in a newspaper. However, it may be considered as a stop word in a set of research papers presented in a database systems conference. A group of different words may share the same word stem. A text retrieval system needs to identify groups of words where the words in a group are small syntactic variants of one another and collect only the common word stem per group. For example, the group of words drug, drugged, and drugs, share a common word stem, drug, and can be viewed as different occurrences of the same word. "How can we model a document to facilitate information retrieval?" Starting with a set of d documents and a set of t terms, we can model each document as a vector v in the t dimensional space R t , which is why this method is called the vector-space model. Let the term frequency be the number of occurrences of term t in the document d, that is, freq(d; t). The (weighted) term-frequency matrix TF(d; t) measures the association of a term t with respect to the given document d: it is generally defined as 0 if the document does not contain the term, and nonzero otherwise. There are many ways to define the term-weighting for the nonzero entries in such a vector. For example, we can simply set TF(d; t) = 1 if the term t occurs in the document d, or use the term frequency freq(d; t), or the relative term frequency, that is, the term frequency versus the total number of occurrences of all the terms in the document. There are also other ways to normalize the term frequency. For example, the Cornell SMART system uses the following formula to compute the (normalized) term frequency:

$$TF(d,t)\begin{cases} = 0 & if\ freq(d,t) = 0 \\ = 1 + \log\big(1 + log\big(freq(d,t)\big)\big) & otherwise. \end{cases}$$

Besides the term frequency measure, there is another important measure, called inverse document frequency (IDF), that represents the scaling factor, or the importance, of a term t. If a term t occurs in many documents, its importance will be scaled down due to its reduced discriminative power. For example, the term database systems may likely be less important if it occurs in many research papers in a database system conference. According to the same Cornell SMART system, IDF (t) is defined by the following formula:

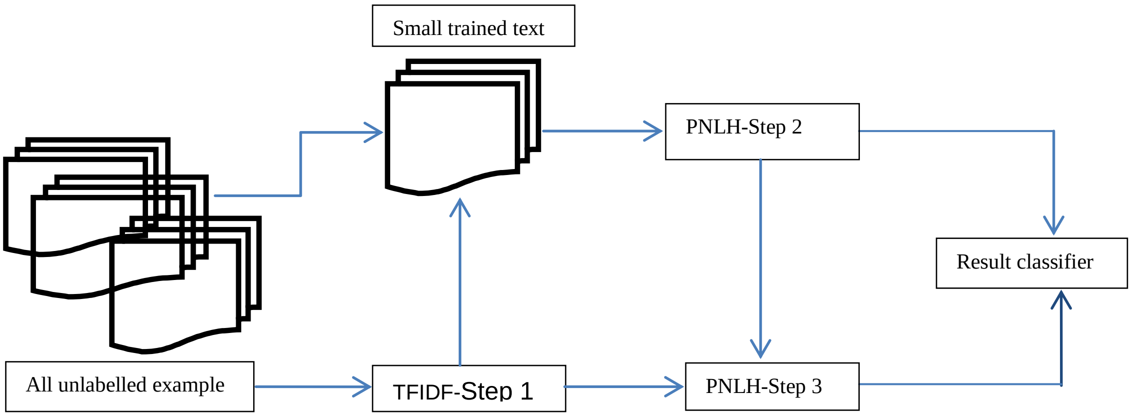

Figure.3 Identifying resultant classifier.

$$TDF = \log\frac{1+|\mathrm{d}|}{|dt|}$$

where*d* is the document collection, and *dt*is the set of documents containing term *t*. If |*dt*|<<|*d*|, the term *t* will have a large IDF scaling factor and vice versa.

In a complete vector-space model, TF and IDF are combined together, which forms the TF-IDF measure:

TF-IDF(d,t) = TF(d,t)×IDF(t).

4. RELATED WORK

Many types of text representations have been proposed in the past. A well-known one is the bag of words that uses keywords (terms) as elements in the vector of the feature space. To the best of our knowledge, there is no work so far on one-class classification under data stream scenario. While a few works found have discussed classification of partially labelled data streams. In [1], Wu *et al.* proposed a semi-supervised classifier, which uses a small number of both positive and negative samples to train an initial classifier, without any further user feedback and training samples. This means that their algorithms cannot cope with concept drift in data streams. Text classifiers using positive and unlabelled example are also discussed in [4][3][5], where the problem of concept drift is not considered as an issue.

Active learning of data streams is proposed in [4][5][7] and [2], which trains initial classifier on partially labelled data stream, and requires the true label of some certain unlabelled samples for enhancing the classifier. The algorithms in [4][5][7] estimate the error of the model on new data without knowing the true class labels. As it is known that concept drift could be caused by changing of user interests, changing of data distribution, or both, the algorithms proposed in [4][5][7] cannot cope with concept drift caused by sudden shift of user interests. Their system cannot detect this kind of concept drift without knowing the true class labels. Moreover, in real-life applications, the system is always fed with overwhelming volume of incoming data, which makes it not applicable to require human investigation for the true class label of some unlabelled samples. Learning concept drift from both label and unlabelled text data is proposed in [9] and [8]. The algorithms proposed by Klinkenberg *et al.* in [9] need more than one scan of the dataset, which makes it not applicable for a data stream scenario. In [2], Dwi *et al.* focused on expanding the labelled training samples using relevant unlabelled data, so as to “extend the capability of existing concept drift learning algorithms”.

Topic tracking [6], a sub-task of topic detection and tracking (TDT), tries to retrieval all *ontopic* news stories from a stream of news stories with a few initial *ontopic* samples, and this is related to our work. For TDT, the concept drift is caused by the evolving of the news story itself. While for our work, the concept drift is caused by changes in the user interests, and/or changes in data distribution. The task of information filtering [6] is to classify documents from a stream as either relevant or non-relevant according to a particular user interest with the objective to reduce information load. In adaptive information filtering [3], a sub-task for information filtering, it is assumed that there is an incoming stream of documents, and that each user interest is represented by a persistent profile. Each incoming document is matched against each profile, and (well-) matched documents are sent to the user. The user is assumed to provide feedback on each document sentto him/her. While for our work, no user feedback is supplied. It is generally believed that ensemble classifier could have better classification accuracy than a single classifier [2]. A range of ensemble classifiers has been proposed by research community [7].

The initial papers [6] on classification data streams by ensemble methods use static majority voting[7] and static weighted voting[8], while the current trends is to use dynamic methods, say, dynamic voting in [1], [4], and [9]; dynamic classifier selection in [5] and [9]. And it is concluded in [8] that dynamic methods perform better than static methods. In [3] and [6], algorithms are proposed to have successfully built text classifiers from positive and unlabelled text documents. We use the same idea discussed in [6] to expand the training data from positive-only to include both

Natural language processing (NLP) is a modern computational technology that can help people to understand the meaning of text documents. For a long time, NLP was struggling for dealing with uncertainties in human languages.

5. CONCLUSION AND FUTURE WORK

Many data mining techniques have been proposed in the last decade. These techniques include association rule mining, frequent item set mining, sequential pattern mining, maximum pattern mining, and closed pattern mining. The main survey of this paper can be summarized as that we firstly tackled the problem of the one-class classification on streaming data. An effective classification of streamed data for text mining by PNLH & one-class classification SVM for text contained audit. The proposition is that fully labelling streaming data is impractical, expensive, and sometimes unnecessary, especially with text streams. By designing a stacking style ensemble-based classifier, and using a series comparative studies, we have dealt with the problems of concept drift, small number of training examples, no negative examples, noisy data, and limited memory space on streaming data classification. The feature space of text stream may evolve constantly. We need to study the dynamic feature space under the one class text stream classification scenario in the future. On the other hand, the further research should also be considered with the one-class classification on streaming data in general.

6. REFERENCES

[1] S. Wu, C. Yang, and J. Zhou. Clustering-training for datastream mining. *Sixth IEEE International Conference onData Mining Workshops*, pages 653–656, 2006.

[2] H. Yu, J. Han, and K. Chang. PEBL: web page classificationwithout negative examples. *IEEE Transactions on Knowledgeand Data Engineering*, 16(1):70–81, 2004.

[3] Y. Zhang and X. Jin. An automatic construction and organizationstrategy for ensemble learning on data streams. *ACMSIGMOD Record*, 35(3):28–33, 2006.

[4] X. Zhu, X. Wu, and Y. Yang. Dynamic classifier selectionfor effective mining from noisy data streams. *Proceedingsof the 4th international conference on Data Mining,(ICDM'04)*, pages 305–312, 2004.

[5] X. Zhu, P. Zhang, X. Lin, and S. Y. Active Learning fromData Streams. *Proceedings of the Sixth International Conferenceon Data Mining, (ICDM'06)*, 2007.

[6] Yang ZhangOne-class Classification of Text Streams with Concept Drift2008 IEEE International Conference on Data Mining Workshops.

[7] Gabriel Pui Cheong Fung, Jeffrey X. Yu, Member, IEEE Computer Society, Hongjun Lu, and Philip S. Yu, Fellow, IEEEText Classification without Negative Examples Revisitieee transactions on knowledge and data engineering, vol. 18, no. 1, January 2006

[8] Z. Jiawei Han and Micheline Kamber Data mining concepts and techniques book referred third edition 2010

[9] Z. Oxford publication Arun kumar pujari book referred second edition 2011

[10] Data Mining: Concepts and Techniques Second Edition Jiawei Han University of Illinois at Urbana-Champaign Micheline Kamber 2006.

[11] X. Li and B. Liu. Learning from Positive and Unlabeled Examples with Different Data Distributions. ProceedingsofEuropean Conference on Machine Learning (ECML-05), pages 218–229, 2005.