International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-2 Issue-4, April 2013

Investigation and Analysis of Efficient Pattern Discovery Method for Text Mining

Asmeeta Mali

***Abstract*— the concept of text mining is nothing but the mechanism of extracting non-trivial and interesting data from the unstructured text dataset. Text mining is consisting of many computer science disciplines with highly oriented towards the artificial intelligence in general such as the applications like information retrieval, pattern recognition, machine learning, natural language processing, and neural networks. The main difference between the search and text mining is that, search needs users attentions means based users requirement search action will perform whereas text mining is the internal process which attempts to find out information in the pattern which is not known before.**

To do the text mining, there are many methods presented still to the date those are having their own advantages and disadvantages. The major problems related to such techniques are efficient use and update of discovered patterns, problems related to the synonymy and polysemy etc. In this paper we are investigating the one such method which is presented to overcome above said problems related to the text mining's. The method presented here is based on innovative as well as effective pattern discovery technique and this consisting of processes like pattern deploying and pattern evolving in order to improve the effectiveness of using and updating discovered patterns for finding relevant and interesting information.

***Index Terms*—Pattern recognition, text mining, knowledge discover, KDD.**

I. INTRODUCTION

The field of knowledge mining is best well-known than that of text mining. A decent example of information mining is that the analyzing of dealing details contained in relative databases, like MasterCard payments or debit card (PIN) transactions. To such transactions numerous further information will be provide: date, location, age of card holder, salary, etc. With the help of this data patterns of interest or behavior will be determined.

But 90 % of all data is unstructured data, and each the share and therefore the absolute quantity of unstructured data will increase daily. Solely a little proportion of data is hold on in an exceedingly structured format in an exceedingly information. The bulk of data that we have a tendency to work with on a daily basis is within the kind of text documents, e-mails or in multimedia system files (speech, video and photos). Looking inside or analysis using information or data mining techniques of this data is not possible, as these techniques work solely on structured data.

Structured data is simpler to go looking, manage, organize, and share and to make reports on, for computers yet as individuals, thence the will to administer structure to unstructured data. These permitting computers and other people to higher manage the knowledge, and permit well-known techniques and strategies to be used.

Text mining, mistreatment manual techniques, was use initial throughout the 1980s. It quickly became apparent that these manual techniques were labor intensive and so expensive. It conjointly prices an excessive amount of time to manually method the already-growing amount of data. Over time there was increasing success in making programs to mechanically method the knowledge, and within the last ten years there has been a lot of progress [1].

Most of the applications like market research as well as business management will profit by the utilization of the data and knowledge extracted from an outsized quantity of information. Knowledge discovery will be viewed because the method of nontrivial extraction of knowledge from massive databases, info that\'s implicitly bestowed within the knowledge, antecedent unknown and doubtless helpful for users [2]. Data mining is thus a necessary step within the process of information discovery in databases. Within the past decade, a big variety of data mining techniques are bestowed so as to perform completely different knowledge tasks. These techniques embody association rule mining, frequent item set mining, serial pattern mining, and most pattern mining and closed pattern mining. Most of them square measure planned for the aim of developing economical mining algorithms to search out specific patterns among an inexpensive and acceptable timeframe [3]. With an outsized variety of patterns generated by victimisation the information mining approaches, a way to effectively exploit these patterns continues to be associate open analysis issue [4]. Text mining is the technique that helps users finds useful information from a large amount of digital text data [5]. It is therefore crucial that a good text mining model should retrieve the information that users require with relevant efficiency. Traditional Information Retrieval (IR) has the same objective of automatically retrieving as many relevant documents as possible whilst filtering out irrelevant documents at the same time. However, IR-based systems do not adequately provide users with what they really need. Many text mining methods have been developed in order to achieve the goal of retrieving for information for users. We focus on the development of a knowledge discovery model to effectively use and update the discovered patterns and apply it to the field of text mining. Most work in knowledge discovery and data mining was concerned with transactional or structured databases. However, a large portion of the available data appears in collections of text articles. The process of knowledge discovery may consist as following:

Manuscript published on 30 April 2013.
* Correspondence Author (s)
Ms. Asmeeta Mali, Information Technology, DYPIET, Pune, India.

i) Data Selection
ii) Data Processing
iii) Data Transaction
iv) Pattern Discovery
iv) Pattern Evaluation.

These steps are used by text mining methods, as we stated in above as well exsting methods still having suffered from many problems like synonymy and polysemy etc, hence in this paper we are investigating the one such method of pattern discovery which effective and overcomes the above said problems. This technique calculates discovered specificities of patterns then evaluates term weights consistent with the distribution of terms within the discovered patterns instead of the distribution in documents for finding the mistaking drawback. It conjointly considers the influence of patterns from the negative training examples to search out ambiguous (noisy) patterns and take a look at to scale back their influence for the low-frequency drawback. The method of change ambiguous patterns will be referred as pattern evolution. The investigated approach will improve the accuracy of evaluating term weights as a result of discovered patterns are a lot of specific than whole documents [6]. In below sections, we will first present the related studied over text mining methods presented so far in section II. In section III we will present the literature review over the text mining concepts in brief. Further in section IV we will present the algorithms those are investigated here for effective pattern discovery.

II. RELATED WORK

There will be a large range of terms extracted from text using data processing strategies. The high spatial property of the feature house results in the machine complexness and over-fitting issues. Solely terms with valuable info area unit elect. the easy thanks to cut back the strategies is that the filtering approach, that filters digressive terms supported the measures derived from the applied mathematics info.

Many varieties of text mining are planned within the past. A standard one is that the bag of words that uses keywords (terms) as elements within the vector of the feature space. In [15], the TFIDF weight theme is employed for text illustration in Rocchio classifiers. Additionally to TFIDF, the worldwide IDF and entropy weight theme is projected in [9] and improves performance by a median of 30 %. Varied weight schemes for the bag of words illustration approach got in [14]. the matter of the bag of words approach is the way to choose a restricted range of options among a vast set of words or terms so as to extend the system expeditiously avoid over lifting[1].Term-based metaphysics mining ways conjointly provided some thoughts for text representations. As an example, stratified agglomeration [17] was wont to confirm synonymy and subordination relations between keywords. Also, the pattern evolution technique was introduced in [25] so as to boost the performance of term-based metaphysics mining. These analysis works have primarily targeted on developing economical mining algorithms for locating patterns from an outsized knowledge assortment. within the presence of those setbacks, sequent patterns employed in data processing community have clothed to be a promising various to phrases [13] as a result of sequent patterns get pleasure from sensible applied mathematics properties like terms. to beat the disadvantages of phrase-based approaches, pattern mining-based approaches or pattern taxonomy models (PTM) [1]) are projected, that adopted the conception of closed sequent patterns, and cropped nonclosed patterns. we tend to conjointly conduct various experiments on the most recent knowledge assortment, Reuters Corpus Volume one (RCV1) and Text Retrieval Conference (TREC) filtering topics, to guage the projected technique. The results show that the projected technique outperforms up-to-date knowledge mining-based ways, concept-based models and also the progressive term primarily based ways. Select RCV1 corpus as our dataset for analysis since RCV1 is that the latest corpus not to mention an outsized quantity of documents and relevancy judgement.

III. LITERATURE REVIEW

3.1 Text Mining

A good example of information mining is that the analyzing of group action details contained in relative databases, like master card payments or charge account credit (PIN) transactions. The sphere of information mining is healthier famous than that of text mining. To such transactions varied further info may be provide: date, location, age of card holder, salary, etc. With the help of this info patterns of interest or behavior may be determined. But at other hand, we found that around 90 % information is unstructured and these percentages of unstructured information are increasing daily. Original unstructured text database contains very less amount of structured information. Most of information on which end user works daily are in the forms of e-mails, text documents, multimedia files like video, speech and photos. Finding inside or analysis based on database or data mining methods of this information does not possible due to reason that such methods only works over structured information [6] [7].

Structured data is simpler to look, manage, organize, and share and to form reports on, for computers further as individuals, therefore the need to provide structure to unstructured info. These permitting computers and other people to raise manage the knowledge, and permit best-known techniques and strategies to be used.

The concept of Text mining initially introduced during the 1980s, based on manual techniques. It quickly became apparent that these manual techniques were labor intensive and so costly. It additionally prices an excessive amount of time to manually method the already-growing amount of data. Over time there was increasing success in making programs to mechanically method the data, and within the last ten years there has been a lot of progress [8].

Currently the study of text mining considerations the event of assorted mathematical, applied mathematics, linguistic and pattern-recognition techniques which permit automatic analysis of unstructured info further because the extraction of prime quality and relevant knowledge, and to form the text as a full higher searchable. High quality refers here, specially, to the mix of the connexion (i.e. finding a needle in an exceedingly haystack) and therefore the effort of recent and fascinating insights.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Initially, this concept of knowledge discovery from the text (KDT) is introduced in Feldman et al. which deals with the machine supported text information analysis. The method is used from the information retrieval, natural language processing and information extraction, further connects them with methods and algorithms of KDD, machine learning, statistics and data mining. And hence, one selects an identical procedure like the KDD method, whereby not knowledge generally, however text documents are focused of the analysis. From this, new queries for the used data processing ways arise. One drawback is that we have a tendency to currently have to be compelled to deal with issues of — from the info modeling perspective— unstructured data sets. If we have a tendency to try and outline text mining, we are able to discuss with connected analysis areas. For every of them, we are able to provide a completely different definition of text mining, which is motivated by the precise perspective of the area:

*Text Mining = Information Extraction***.** Basically this method is based on the assumption like text mining is equal to information extraction.

*Text Mining = Text Data Mining***.** Text mining is nothing but the data mining, as the application of algorithms as well as methods from the field's machine learning and statistics to texts with the goal of finding useful patterns.

*Text Mining = KDD Process***.** Based on the model of knowledge discovery process, we mostly find in literature text mining as a process with a series of partial steps, among other things also information extraction as well as the use of data mining or statistical procedures. In the below sections we will discuss each of them in details [8].

3.2 Knowledge Discovery

Knowledge Discovery in Databases (KDD) is an automatic, wildcat analysis and modeling of enormous information repositories. KDD is that the organized method of characteristic valid, novel, useful, and comprehendible patterns from massive and complicated information sets. Data mining (DM) is that the core of the KDD process, involving the inferring of algorithms that explore the info, develop the model and see antecedently unknown patterns. The model is employed for understanding phenomena from the info, analysis and prediction. The accessibility and abundance information these days makes knowledge discovery and data mining a matter of respectable importance and necessity.

Given the recent growth of the field, it's not stunning that a good sort of strategies is now obtainable to the researchers and practitioners. nobody methodology is superior to others for all cases. The book of facts of knowledge Mining and information Discovery from information aims to prepare all significant strategies developed within the field into a coherent and unified catalog; presents performance analysis approaches and techniques; and explains with cases and software package tools the employment of the various strategies [8].

Data analysis in the KDD is aims of finding the hidden patterns as well as connections in those data. By information we tend to perceive a amount of facts, which may be, for example, information in a database, however additionally information in a straightforward text file. Characteristics that may be accustomed measure the standard of the patterns found within the information are the quality for humans, validity within the context of given datum measures, novelty and utility. Moreover, completely different strategies square measure ready to discover not solely new patterns however to provide at identical time generalized models that represent the found connections. During this context, the expression "potentially useful" means the samples to be found for an application generate a benefit for the user. Therefore the definition couples knowledge discovery with a particular application.

The method of knowledge discovery is consisting of different a processing step which has to be applied to a data set of interest in order to extract useful patterns. Those steps performing based on iteratively as well as most of steps usually need interactive feedback from a user. As per the definition of CRoss Industry Standard Process for Data Mining model having following major steps: (1) business understanding, (2) data understanding, (3) data preparation, (4) modelling, (5) evaluation, (6) deployment.

3.3 Data Mining, Machine Learning and Statistical Learning

There are many research working is going over the fields like knowledge discovery as well as data mining. One indicator for this is the sometimes confusing use of terms. The data mining is also called as KDD which means the data mining is having all the methods knowledge discovery process. Such definition of data mining is commonly used and hence leads to problems of distinguishing proper terms. On the other hand we can also refer data mining as part of KDD processes and presenting modelling phase, i.e. the application of methods and algorithms for the calculation of the searched patterns or models [9].

The author such as Kumar and Joshi assumes data mining in addition as the search for valuable information in *large quantities of data*. The roots of data mining lie in most diverse areas of research, which underlines the interdisciplinary character of this field. Below we are discussing the relations to three of the addressed research areas: Databases, machine learning and statistics.

Databases are necessary so as to investigate massive quantities of information expeditiously. During this association, a information represents not solely the medium for consistent storing and accessing, however moves within the nearer interest of analysis, since the analysis of the info with data mininging algorithms is supported by information's and therefore the utilization of database technology within the data processing process may be helpful.

Machine Learning (ML) is a part of computing involved with the event of techniques which permit computers to "learn" by the analysis of information sets. The focus of most machine learning ways is on symbolic knowledge. ML is additionally involved with the algorithmic quality of procedure implementations. *Statistics* has its grounds in arithmetic and deals with the science and apply for the analysis of empirical knowledge. It's supported applied math theory that may be a branch of maths. at intervals applied math theory, randomness and uncertainty are shapely by applied mathematics. These days several ways of statistics are employed in the sphere of KDD [8].

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

IV. INVESTIGATED ALGORITHMS AND FRAMEWORK

Pattern Taxonomy Model-

As per given in [1] for the investigation purpose we assume that all documents are split into paragraphs. So a given document d yields a set of paragraphs PS (d). Let D be a training set of documents, which consists of a set of positive documents, D+; and a set of negative documents, D-. Let T = {t1; t2; . . . ; tm} be a set of terms (or keywords) which can be extracted from the set of positive documents, D+.

Frequent and Closed Patterns

Frequent Patterns

F – Minimal frequency threshold given by the user

Frequent pattern

A conjunction of literals which covers at least F examples

Algorithms for finding frequent patterns

_ *Propositional data:* the *Apriori* algorithm [Agrawal and Srikant, 1994]

_ *First-order logic:* the WARMR level-wise system [Dehaspe & Toivonen, 1999]

_ *Maximal first-order frequent patterns:* the RAP system [Bla˘t´ak *et al.*, 2002]

Example: Background knowledge.

Closed Sequential Patterns-

A sequential pattern s=<t1; . . . ; tr> (ti elements of T) is an ordered list of terms. A sequence s1= <x1; . . . ; xi> is a subsequence of another sequence s2<=y1; . . . ; yj>, is called s1 is sub-set of s2, iff j_1; . . . ; j_y such that $1<= j_1 < j_2 \ldots < j_y <=j$ and $x_1=y_{j1}$; $x_2=y_{j2}$; . . . ; $x_i=y_{jy}$. Given s1 is sub-set of s2; we usually say s1 is a sub-pattern of s2, and s2 is a super pattern of s1. In the following, we simply say patterns for sequential patterns.

A sequential pattern X is called frequent pattern if its relative support (or absolute support) ≥ min sup, a minimum support. The property of closed patterns can be used to define closed sequential patterns. A frequent sequential pattern X is called closed if not 9 any super pattern X1 of X such that $sup_a(X1) = sup_a(X)$.

Composition Operation-

Let p1 and p2 be sets of term number pairs. P1 P2 is called composition of p1 and p2 which satisfies-

$p_1 \oplus p_2 = \{(t, x_1 + x_2) \mid (t, x_1) \in p_1, (t, x_2) \in p_2\} \cup \{(t, x) \mid (t, x) \in p_1 \cup p_2, \text{not } ((t, \_) \in p_1 \cap p_2)\}$

where _ is the wild card that matches any number.

Example-

$\{(t_1, 3), (t_2, 2), (t_3, 3), (t_4, 3)\} \oplus \{(t_2, 3), (t_5, 4)\}$
$= \{(t_1, 1), (t_2, 5), (t_3, 3), (t_4, 3), (t_5, 4)\}$

Here we add the common elements and which is not common we write as it is.

In the above example t_2 elements are available in both sets so {(t_2 , 2+3)} as composition and another elements of sets we write as it is.

PTM (D+, min_sup)

Input: positive documents D^+; minimum support, min_sup.

Output : d-patterns D P, and supports of terms.

DP = ϕ;
foreach document $d \in D^+$ do
let PS(d) be the set of paragraphs in *d;*
SP =SPMining (*PS(d)*, min _sup);
$\hat{d} = \phi$;
foreach patterns $p_i \in$ SP do
p = { (t,1) |t∈ p_i };
$\hat{d} = \hat{d} \oplus$ p;
end
DP = DP ∪ { $\hat{d}$ };
end
T ={t|(t,f) ∈p,p ∈ DP};
foreach term t ∈ T do
support (t) =0;
end
foreach d-pattern P ∈ DP do
foreach (t, ω) ∈β (p) do
support (t) =support (t) + ω;
end
end

INNER PATTERN EVOLUTION-

In this section, we discuss how to reshuffle supports of terms within normal forms of d-patterns based on negative documents in the training set. The technique will be useful to reduce the side effects of noisy patterns because of the low-frequency problem. This technique is called inner pattern evolution here, because it only changes a pattern's

term supports within the pattern. A threshold is usually used to classify documents into relevant or irrelevant categories. Using the d-patterns, the threshold can be defined naturally as follows:

$$Threshold(DP) = \min_{p \in DP}\left(\sum_{(t,\omega)\in\beta(p)} \text{support}(t)\right)$$

A noise negative document nd in D- is a negative document that the system falsely identified as a positive, that is weight (nd) ≥ Threshold (DP). In order to reduce the noise, we need to track which d-patterns have been used to give rise to such an error. We call these patterns offenders of nd.

IPEvolving (D+, D-, DP, μ).

Input : a training set D = $D^+ \cup D^-$; a set of d-patterns DP; and an experimental coefficient μ

Output : a set of term support pairs np.

np ← ϕ;
threshold = Threshold(DP);
foreach noise negative document nd ∈ D^- do
if weight (nd) ≥ threshold then
$\nabla(nd) = \{p \in DP \mid termset(p) \cap nd \neq \phi\}$;
NDP = $\{\beta(p) \mid p \in DP\}$;
Shuffling(nd , $\nabla(nd)$, NDP , μ);
foreach p ∈ NDP do
np ← np ⊕ p;
end
end

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Stemming Algorithms

A stemming algorithm is a process of linguistic normalization, in which the variant forms of a word are reduced to a common form, for example,

connection
connections
connective ---> connect
connected
connecting

It is important to appreciate that we use stemming with the intention of improving the performance of IR systems. It is not an exercise in etymology or grammar. In fact from an etymological or grammatical viewpoint, a stemming algorithm is liable to make many mistakes. In addition, stemming algorithms - at least the ones presented here - are applicable to the written, not the spoken, form of the language.

For some of the world's languages, Chinese for example, the concept of stemming is not applicable, but it is certainly meaningful for the many languages of the Indo-European group. In these languages words tend to be constant at the front, and to vary at the end:

-ion
-ions
connect-ive
-ed
-ing

The variable part is the `ending', or `suffix'. Taking these endings off is called `suffix stripping' or `stemming', and the residual part is called the stem.

Endings

Another way of looking at endings and suffixes is to think of the suffix as being made up of a number of endings. For example, the French word

Confirmative

can be thought of as `confirm' with a chain of endings,

-atif (adjectival ending - morphological)
plus -e (feminine ending - grammatical)
plus -s (plural ending - grammatical)

-atif can also be thought of as -ate plus -if. Note that the addition of endings can cause respellings, so -e changes preceding `f' to `v'.

Endings fall into two classes, grammatical and morphological. The addition of -s in English to make a plural is an example of a grammatical ending. The word remains of the same type. There is usually only one dictionary entry for a word with all its various grammatical endings. Morphological endings create new types of word. In English -ise or -ize makes verbs from nouns (`demon', `demonise'), -ly makes adverbs from adjectives (`foolish', `foolishly'), and so on. Usually there are separate dictionary endings for these creations.

Language knowledge

It is much easier to write a stemming algorithm for a language when you are familiar with it. If you are not, you will probably need to work with someone who is, and who can also explain details of grammar to you. Best is a professional teacher or translator. You certainly don't need to have a world authority on the grammar of the language. In fact too much expertise can get in the way when it comes to the very practical matter of writing the stemming algorithm.

Vocabularies

Each stemmer is issued with a vocabulary in data/voc.txt, and its stemmed form in data/voc.st. You can use these for testing and evaluation purposes.

Raw materials

A conventional grammar of a language will list all the grammatical endings, and will often summarize most of the morphological endings. A grammar, plus a dictionary, is therefore basic references in the development of a stemming algorithm, although you can dispense with them if you have an excellent knowledge of the language. What you cannot dispense with is a vocabulary to try the algorithm out on as it is being developed. Assemble about 2 megabytes of text. A mix of sources is best, and literary prose (conventional novels) usually gives an ideal mix of tenses, cases, persons, genders etc. Obviously the texts should be in some sense 'contemporary', but it is an error to exclude anything slightly old. The algorithm itself may well get applied to older texts once it has been written. For English, the works of Shakespeare in the customary modern spelling make a good test vocabulary.

From the source text derive a control vocabulary of words in sorted order. Sample vocabularies in this style are part of our Open Source release. If you make a small change to the stemming algorithm you should have a procedure that presents the change as a three column table: column one is the control vocabulary, column 2 the stemmed equivalent, and column 3 the stemmed equivalent after the change has been made to the algorithm. The effects of the change can be evaluated by looking at the differences between columns two and three.

The first job is to come up with a list of endings. This can be done by referring to the grammar, the dictionary, and also by browsing through the control vocabulary.

Rules for removing endings

If a word has an ending, E, when should E be removed? Various criteria come into play here. One is the knowledge we have about the word from other endings that might have been removed. If a word ends with a grammatical verb ending, and that has been removed, then we have a verb form, and the only further endings to consider are morphological endings that create verbs from other word types. At this level the system of endings gives rise to a small state table, which can be followed in devising the algorithm. In Latin derived languages, there is a state table of morphological endings that roughly looks like this:

-IC (adj) -+-> -ATION (noun)
+-> -ITY (noun)
+-> -MENT (adv)
\-> -AT (verb) -+-> -IV (adj) -+-> -ITY (noun)
| \-> -MENT (adv)
\-> -OR (noun)
-ABLE (adj) -+-> -ITY (noun)
\-> -MENT (adv)
-OUS (adj) ---> -MENT (adv)

The ending forms take different values in different languages. In French, -OR becomes `-eur' (m.) or `-rice' (f.), -AT disappears into the infinitive form of a verb. In English, -MENT becomes `-ly', and then one can recognize,

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

-IC-ATION fortification
-IC-ITY electricity
-IC-MENT fantastically
-AT-IV contemplative
-AT-OR conspirator
-IV-ITY relativity
-IV-MENT instinctively
-ABLE-ITY incapability
-ABLE-MENT charitably
-OUS-MENT famously

Trios, -IC-AT-IV etc., also occur, but sequences of length four, -IC-AT-IV-ITY and -IC-AT-IV-MENT, are absent (or occur very rarely).

Using stemming in IR

In earlier implementations of IR systems, the words of a text were usually stemmed as part of the indexing process, and the stemmed forms only held in the main IR index. The words of each incoming query would then be stemmed similarly. When the index terms were seen by the user, for example during query expansion, they would be seen in their stemmed form. It was important therefore that the stemmed form of a word should not be too unfamiliar in appearance. A user will be comfortable with seeing `apprehend', which stands for 'apprehending', `apprehended' as well as `apprehend'. More problematical is `apprehens', standing for `apprehension', `apprehensive' etc., but even so, a trained user would not have a problem with this. In fact all the Xapian stemming algorithms are built on the assumption that it leaves stemmed forms which it would not be embarrassing to show to real users, and we suggest that new stemming algorithms are designed with this criterion in mind.

A superior approach is to keep each word, *W*, and its stemmed form, *s (W)*, as a two-way relation in the IR system. *W* is held in the index with its own posting list. *S (W)* could have its separate posting list, but this would be derivable from the class of words that stem to *s(W)*. The important thing is to have the $W \leftrightarrow s(W)$ relation. From *W* we can derive *s (W)*, the stemmed form. From a stemmed from *s (W)* we can derive *W* plus the other words in the IR system which stem to *s(W)*. Any word can then be searched on either stemmed or unstamped. If the stemmed form of a word needs to be shown to the user, it can be represented by the commonest among the words which stem to that form.

Stopwords

It has been traditional in setting up IR systems to discard the very commonest words of a language - the stopwords - during indexing. A more modern approach is to index everything, which greatly assists searching for phrases for example. Stopwords can then still be eliminated from the query as an optional style of retrieval. In either case, a list of stopwords for a language is useful.

Getting a list of stopwords can be done by sorting a vocabulary of a text corpus for a language by frequency, and going down the list picking off words to be discarded.

The stopword list connects in various ways with the stemming algorithm: The stemming algorithm can itself be used to detect and remove stopwords. One would add into the irregular_forms table something like this,

"", /* null string */
"am/is/are/be/being/been/" /* BE */
"have/has/having/had/" /* HAD */
"do/does/doing/did/" /* DID */
... /* multi-line string */

so that the words `am', `is' etc. map to the null string (or some other easily recognised value).

Alternatively, stopwords could be removed before the stemming algorithm is applied, or after the stemming algorithm is applied. In this latter case, the words to be removed must themselves have gone through the stemmer, and the number of distinct forms will be greatly reduced as a result. In Italian for example, the four forms

questa queste questi questo

(meaning `that') all stem to

Quest

V. CONCLUSION

In this paper had investigated the effective pattern discovery algorithm for the area of text mining. We discussed various methods of text mining along with their drawbacks. We presented the algorithm based PTM and steaming to overcome issues related to existing text mining methods. The investigated framework for text mining here will follow the steps that were previously discussed. The process starts with retrieving the relevant documents from the appropriate databases. Then the data will be extracted and cleaned to remove noises and errors. This cleaned data will then be fed to the analysis process. This thesis will add to this body of knowledge by implementing a new text clustering process.

REFERENCES

[1] K. Aas and L. Eikvil, “Text Categorisation: A Survey,” Technical Report Raport NR 941, Norwegian Computing Center, 1999.
[2] R. Agrawal and R. Srikant, “Fast Algorithms for Mining Association Rules in Large Databases,” Proc. 20th Int’l Conf. Very Large Data Bases (VLDB ’94), pp. 478-499, 1994.
[3] H. Ahonen, O. Heinonen, M. Klemettinen, and A.I. Verkamo, “Applying Data Mining Techniques for Descriptive Phrase Extraction in Digital Document Collections,” Proc. IEEE Int’lForum on Research and Technology Advances in Digital Libraries (ADL ’98), pp. 2-11, 1998.
[4] R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval.Addison Wesley, 1999.
[5] N. Cancedda, N. Cesa-Bianchi, A. Conconi, and C. Gentile,“Kernel Methods for Document Filtering,” TREC, trec.nist.gov/ pubs/trec11/papers/kermit.ps.gz, 2002.
[6] N. Cancedda, E. Gaussier, C. Goutte, and J.-M. Renders, “Word-Sequence Kernels,” J. Machine Learning Research, vol. 3, pp. 1059- 1082, 2003.
[7] M.F. Caropreso, S. Matwin, and F. Sebastiani, “Statistical Phrasesin Automated Text Categorization,” Technical Report IEI-B4-07- 2000, Instituto di Elaborazione dell’Informazione, 2000.
[8] C. Cortes and V. Vapnik, “Support-Vector Networks,” MachineLearning, vol. 20, no. 3, pp. 273-297, 1995.
[9] S.T. Dumais, “Improving the Retrieval of Information fromExternal Sources,” Behavior Research Methods, Instruments, and Computers, vol. 23, no. 2, pp. 229-236, 1991.
[10] J. Han and K.C.-C. Chang, “Data Mining for Web Intelligence,”Computer, vol. 35, no. 11, pp. 64-70, Nov. 2002.
[11] J. Han, J. Pei, and Y. Yin, “Mining Frequent Patterns withoutCandidate Generation,” Proc. ACM SIGMOD Int’l Conf. Management of Data (SIGMOD ’00), pp. 1-12, 2000.
[12] Y. Huang and S. Lin, “Mining Sequential Patterns Using GraphSearch Techniques,” Proc. 27th Ann. Int’l Computer Software and Applications Conf., pp. 4-9, 2003.
[13] N. Jindal and B. Liu, “Identifying Comparative Sentences in TextDocuments,” Proc. 29th Ann. Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR ’06), pp. 244-251, 2006.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

[14] T. Joachims, "A Probabilistic Analysis of the Rocchio Algorithm with tfidf for Text Categorization," Proc. 14th Int'l Conf. Machine Learning (ICML '97), pp. 143-151, 1997.
[15] T. Joachims, "Text Categorization with Support Vector Machines:Learning with Many Relevant Features," Proc. European Conf. Machine Learning (ICML '98),, pp. 137-142, 1998.
[16] T. Joachims, "Transductive Inference for Text Classification Using Support Vector Machines," Proc. 16th Int'l Conf. Machine Learning(ICML '99), pp. 200-209, 1999.
[17] W. Lam, M.E. Ruiz, and P. Srinivasan, "Automatic Text Categorization and Its Application to Text Retrieval," IEEE Trans. Knowledge and Data Eng., vol. 11, no. 6, pp.865-879, Nov./Dec. 1999. IEEE Trans. Knowledgeand Data Eng., vol. 11, no. 6, pp. 865-879, Nov./Dec. 1999.
[18] D.D. Lewis, "An Evaluation of Phrasal and Clustered Representations on a Text Categorization Task," Proc. 15th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '92), pp. 37-50, 1992.