Efficient Preprocessing and Patterns Identification Approach for Text Mining

Pattan Kalesha[1], M. Babu Rao[2],Ch. Kavitha[3]

[1](M Tech, GEC, Gudlavalleru,)
[2](Professor of CSE, GEC, Gudlavalleru.)
[3](Professor of IT , GEC, Gudlavalleru.)

Abstract – *Due to the rapid expansion of digital data , knowledge discovery and data mining have attracted significant amount of attention for turning such data into helpful information and knowledge. Text categorization is continuing to become the most researched NLP problems on account of the ever-increasing levels of electronic documents and digital libraries. we present a novel text categorization method that puts together the decision on multiple attributes. Since the most of existing text mining methods adopted term-based approaches, all of these are affected by the difficulties of polysemy and synonymy. Existing pattern discovery technique includes the processes of pattern deploying and pattern evolving, to strengthen the impact of using and updating discovered patterns for looking for relevant and interesting information. But the current association Rules methods exist shortage in two aspects once it is used on patterns classification. a person is the strategy ignored the data about word's frequency in a text . The opposite happens to be the method need pruning rules whenever the mass rules are generated. Within this proposed work specific documents are preprocessed before placing patterns discovery. Preprocessing the document dataset using tokenization, stemming, and probability filtering approaches. Proposed approach gives better decision rules compare to existing approach.*

Keywords: *Patterns, Rules, Stemming, Probability.*

I.INTRODUCTION

Many data mining techniques have been proposed for mining useful patterns in text documents. It is a challenging issue to find accurate knowledge (or features) in text documents to really help users to find what they want. In existing, Data Retrieval (IR) provided many term-based methods to solve this challenge. The term-based methods suffer the pain of the problems of polysemy and synonymy. Polysemy signifies a nice word having different meanings, and synonymy means different words having the same meaning. The proposed paper we work with pattern (or phrase)-based approaches which perform better in contrast studies in comparison with other term-based methods. This process improves the accuracy of evaluating support, term weights because discovered patterns are usually more specific than whole documents [2].

Due to the rapid increase of digital data made available recently, knowledge discovery and data mining [1] have attracted a large amount of attention which includes an imminent need for turning such data into useful suggestions and knowledge. Many applications, for instance market analysis and business, may benefit by way of the information and knowledge extracted from a considerable amount of data. Knowledge discovery can be viewed as the method of nontrivial extraction of real info from large databases, information that's implicitly presented among the data, previously unknown and potentially ideal for users. Data mining is therefore a vital help the method of knowledge discovery in databases. Previously decade, a major wide range of data mining techniques have been presented in an effort to perform different knowledge tasks. These techniques include association rule mining, frequent itemset mining, sequential pattern mining, maximum pattern mining and closed pattern mining.

A lot of them are proposed when considering developing efficient mining algorithms to locate particular patterns within one reasonable and acceptable time period. By using a good deal of patterns produced by analyzing statistics mining approaches, the best way to effectively use and update these patterns is still an open research issue.

I. LITERATURE SURVEY

Many types of text representations most certainly been proposed during the past. A proper known is the bag of words that utilizes keywords (terms) as elements within the vector of the attribute space. The issue of one's bag of words approach is how to actually select a limited number of features among an enormous place of words or terms in an effort to raise the system’s efficiency and avoid over fitting. [1], combine unigram and bigrams was chosen for document indexing with regard to text categorization (TC) and evaluated on a variety of feature valuation functions (FEF). A phrase-based textual content representation for Web document management was also proposed in [2].In [3]; data mining techniques have already been utilized for text analysis by extracting cooccurring terms as descriptive phrases from document collections. However, the overall impact of the text mining systems using phrases as text representation showed no significant improvement. The likely reason was that a phrase-based method had “lower consistency of assignment and lower document band or frequency range for terms” as in [4]. In, hierarchical clustering [5], [6] was used to determine synonymy and hyponymy ones between keywords.

Nevertheless, the challenging issue is how to effectively effectively contend with the big quantity of discovered patterns. Regarding the challenging issue, closed sequential designs could have been utilized for text mining in [1], which proposed that the idea of closed patterns in text mining was, useful in addition to had the possible for improving the appearance of the performance of message mining. Pattern taxonomy model was also developed in [4] and [3] to further improve the effectiveness by effectively using closed patterns in text mining. Additionally, a two-stage model that used both term-based methods and pattern based methods made its entrance in [2] to significantly improve the performance of real info filtering. Natural language processing (NLP) serves as a modern computational technology that in fact can assist individuals to understand the meaning of message documents. For a very long time, NLP was struggling for grappling with uncertainties in human languages. Recently, a new concept-based model [3], [4] was presented to bridge the gap between NLP and text mining, which analyzed terms on the sentence and document levels. Pattern based techniques was introduced in [3] to significantly improve the performance of information filtering.

Introduces PTM [2] consider positive documents and negative documents and we adjust the term weights based on term weight of positive document and negative document. Using this technique we can increase maximum likelihood event one documents having more overlapping terms and less content of the document to get accurate results as shown below process. In this phase desired pattern are evolved from the clusters obtained from above phases. This is important phase of the model which actually evolves patterns which will match to the keywords of user who want relevant information from large database which are generally in electronic forms [7].

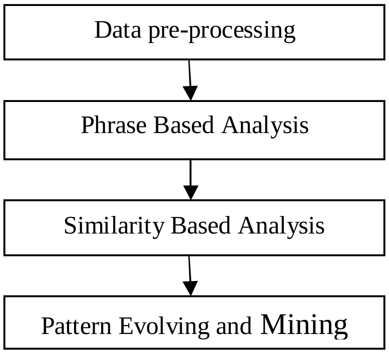


Fig.1:Document Text processing Analysis

III PROPOSED SYSTEM

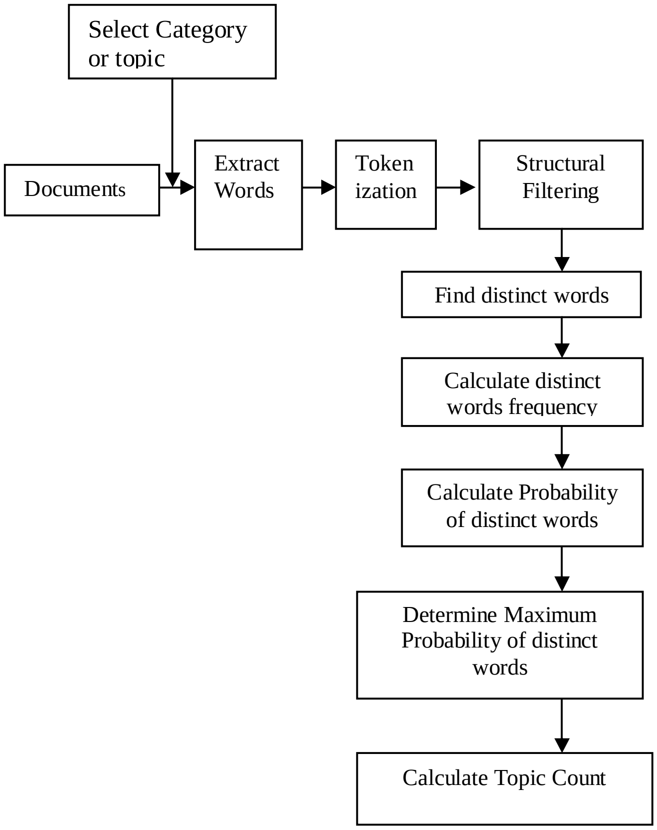

Select Category or topic
Documents
Extract Words
Token ization
Structural Filtering
Find distinct words
Calculate distinct words frequency
Calculate Probability of distinct words
Determine Maximum Probability of distinct words
Calculate Topic Count

Fig.2: Proposed Document Text processing Methodology

Document clustering can loosely be defined as clustering of documents. Clustering is a process of Process of understanding the similarity and/or dissimil arity between the given objects and thus, dividing them into meaningful subgroups sharing common characteristic. Good clustersarethose in which the membe rs inside the cluster have quite a deal of similar characteristics.

Tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters, such as punctuation.

Input: Friends, Romans, Countrymen, lend me your ears;

Output: Friends Romans Countrymen Lend
me Your Ears

A. Data preprocessing:

In the data preprocessing stage there are many steps used to prepare the documents to the next stage. Preprocessing consists of steps that take its input as a plain text document and its output as a set of tokens (which can be single-terms or n-grams) to be included in the vector model. These steps typically can be stated as following:

1. Punctuations, special characters and the numbers are filtered from the document.
2. Partition each document into phrases and then tokenize each phrase into its constituting words.
3. Remove the stopwords which were detected in the stopword list provided by us.
4. Get POS (Part Of Speech) tagger for each remaining word and eliminate each word which is not verb or noun.
5. Remove each word with low frequency or too much occurring words.

B. Document Representation

In this stage each document will be represented in the form as given in Fig.2; that by detecting each new phrase and assigning an id for each cleaned unrepeated phrase (neither when it contains the same words or carries the same meaning). The main challenge is to detect the phrases which do convey the same meaning. This is done by constructing a feature vector for each new phrase then a similarity measure is recursively calculated between each new phrase and the phrases that already have been added to the feature vector and when the similarity exceeds a threshold (assumed value); then one of them will be discarded.

After obtaining the term weights of all topic phrases, it is easy to apply the cosine similarity to compute the similarity of any two documents. Let vectors dx= {x I ,x2, ... , xM}, and dy={yl,y2, ... , yM} denote two documents dx and dy, where xi and yi are the weights of corresponding topic phrase term. Then the similarity of two documents in (1) is calculated by the following formula [6-8].

$$sim_{x,y} = \frac{\vec{d_x} \bullet \vec{d_y}}{|\vec{d_x}| \times |\vec{d_y}|} = \frac{\sum_{i=1}^{M} x_i y_i}{\sqrt{\sum_{i=1}^{M} x_i^2 \sum_{i=1}^{M} y_i^2}} \quad (1)$$

C. Probability Calculation:

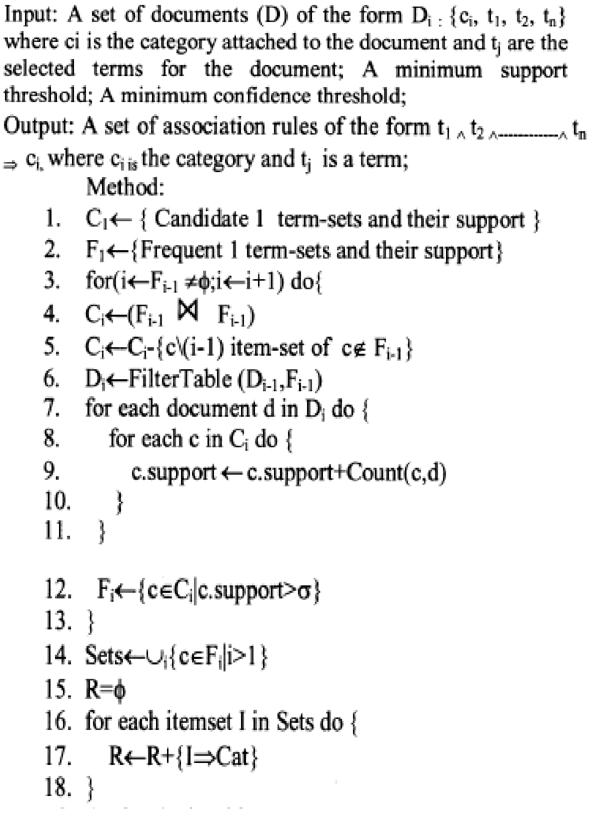
Input : Set S of N document and a set T of K topics
Output: Array of distinct words and array of distinct words count based on K
Method:

```
Read S,N;//Read the training documents one by
one to split all words
Read T,K;//Read the topics
Preprocess D to get Wi //Preprocess to identify
distinct words
For each Wi in D
    For each K of T
        Pi=∑Wi/Wj//Probability calculation of
            distinct words
        count=count+1//Based on the topic
            count will be calculated
    end
end
return Wi;
return count;
```

D. Sample pseudo Code:

Data preprocessing represents the first step. At this stage cleaning techniques can be applied such as stop words removal, streaming or term pruning according to the TF/IDF Values (term frequency/inverse document frequency). The next step in building the associative classifier is the generation of association rules using an apriori - based algorithm. Once the entire set of rules has been generated an important step is to apply some pruning techniques for reducing the set of association rules found in the text corpora. The last stage in this process is represented by the use of association rules set in the prediction of classes for new documents. The first three steps belong to the training process while the last one represents the testing phase. If a document D is assigned to a set of categories C={Ct,C2, c and afterwards pruning the set of term T = { t},t2, t is retained , the following transaction is used to model the document D :{CI,C2 Cm,t},t2, t and the association rules are discovered from such transactions representing all documents in the collection.

The association rules discovered in this stage of the process is further processed to build the associative classifier. Using the apriori algorithm on our transactions representing the documents would generate a very large number of association rules, most of them irrelevant for classification. There are two approaches that we have considered in building an associative text classifier. The first one ARPAC (Association Rule-based Patterns with All Categorized) is to extract association rules from the entire training set following the constraints discussed above. As a result we propose a second solution ARP-BC (Associative Rule-based Pattern by Category) that solves existing problems. In this approach we consider each set of documents belonging to one category as a separate text collection to generate association rules from. If a document belongs to more than one category this document will be present in each set associated with the categories that the document falls into. Algorithm ARP-BC Find association rules on the training set of the text collection when the text corpora are divided in subsets by category.



Input: A set of documents (D) of the form D_i : $\{c_i, t_1, t_2, t_n\}$ where ci is the category attached to the document and t_j are the selected terms for the document; A minimum support threshold; A minimum confidence threshold;

Output: A set of association rules of the form $t_1 \wedge t_2 \wedge$------------$\wedge t_n \Rightarrow c_i$, where c_i is the category and t_j is a term;

```
Method:
1.  $C_1 \leftarrow$ { Candidate 1 term-sets and their support }
2.  $F_1 \leftarrow$ {Frequent 1 term-sets and their support}
3.  for($i \leftarrow F_{i-1} \neq \phi; i \leftarrow i+1$) do{
4.  $C_i \leftarrow (F_{i-1} \bowtie F_{i-1})$
5.  $C_i \leftarrow C_i - \{c \backslash (i-1)$ item-set of $c \notin F_{i-1}\}$
6.  $D_i \leftarrow$ FilterTable $(D_{i-1}, F_{i-1})$
7.  for each document d in $D_i$ do {
8.      for each c in $C_i$ do {
9.          c.support $\leftarrow$ c.support+Count(c,d)
10.     }
11. }

12.  $F_i \leftarrow \{c \in C_i | c.support > \sigma\}$
13. }
14. Sets $\leftarrow \cup_i \{c \in F_i | i > 1\}$
15. $R = \phi$
16. for each itemset I in Sets do {
17.     $R \leftarrow R + \{I \Rightarrow Cat\}$
18. }
```

In ARP-BC algorithm step (2) generates the frequent itemset. In steps (3-13) all the k-frequent item sets are generated and merged with the category in Ci. Steps (16-18) generate the association rules. The document space is reduced in each iteration by eliminating the transactions that do not contain any of the frequent item sets. This step is done by Filter Table (Di-I, Fi-1) function. This problem leads us to the next subsection where pruning methods are presented.

Although the rules are similar to those produced using a rule based induced system, the approach is different. In addition, the number of words belonging to the antecedent, while in some studies with rule-based induced systems, the rules generated has only one or a pair of words as antecedent.

Def 2: Being given two rules Rl and R2 .Rl is higher ranked than R2 if:
(1) Rl has higher confidence than R2
(2) If the confidences are equal, supp (Rl) must exceed supp
(R2)
(3) Both confidences and support are equal, but Rl has less attributes in left hand side than R2

With the sit of association rules sorted, the goal is to select a subset that will build an efficient and effective classifier. In our approach we attempt to select a high quality subset of rules by selecting those rules that are general and have high confidence.

Algorithm Pruning the set of association rules

Input: The set of association rules that were found in the association rule mining phase(s) and the training text collection (D).

Output: A set of rules used in the classification process

Method:

1. Sort the rules according to Definition 1
2. for each rule in the set S
3. Find all those rules that are more specific according to (Definition 2)
4. prune those that have lower confidence
5. a new set of rules S is generated
6. for each rule R in the set S
7. go over D and find those transactions that are covered by the rule R
8. if R classifies correctly at least one transaction
9. select R
10. Remove those cases that are covered by R.

IV RESULTS

A. Existing approach:

In this section existing results are totally based on single attribute based decision system.

BEST RULES USING PTM(IPE):

1. Text=FED SETS 1.5 BILLION DLR CUSTOMER REPURCHASE, FED SAYS
2 ==> class-att=0 2::::: Weight(min sup):(1)
2. Text=OPEC WITHIN OUTPUT CEILING, SUBROTO SAYS Opec remains within its agreed output Ceiling of 15.8 mln barrels a day, and had expected current fluctuations in the spot market of one or two dlrs, Indonesian Energy Minister Subroto said.

He told reporters after meeting with President Suharto that present weakness in the spot oil market was the result of warmer weather in the U.S. And Europe which reduced demand for oil. Prices had also been forced down because refineries were using up old stock, he said. But Opec would get through this period if members
stuck together. REUTER
&#3; 2 ==> class-att=0 2::::: Weight(min sup):(1)

REUTER
&#3; 1 ==> class-att=0 1::::: Weight(min sup):(1)

=== Evaluation ===

Elapsed time: 4.757s

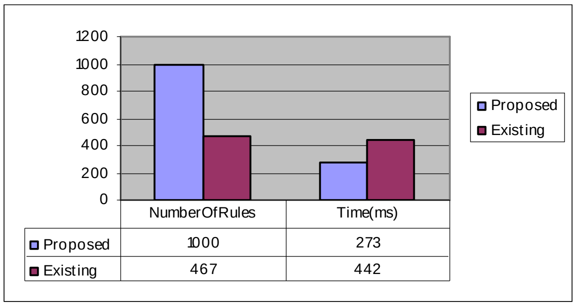
B. Proposed Approach:

Proposed Approach found 1514 rules (displaying top 25 Rules)

1. [reuter=1]: 1441 ----(IMPLIES)------>[&#3=1]: 1441 <RULE CONFIDENCE:(1)> Weighted Accuracy:(99.22)
2. [&lt=1]: 904 ----(IMPLIES)------>[&#3=1]: 904

<RULE CONFIDENCE:(1)> Weighted Accuracy:(62.24)
3. [reuter=1, of=1]: 1077 ----(IMPLIES)------>[&#3=1]: 1077 <RULE CONFIDENCE:(1)> Weighted Accuracy:(74.16)
4. [reuter=1, th=1]: 960 ----(IMPLIES)------>[&#3=1]: 960 <RULE CONFIDENCE:(1)> Weighted Accuracy:(66.1)
5. [reuter=1, to=1]: 931 ----(IMPLIES)------>[&#3=1]: 931 <RULE CONFIDENCE:(1)> Weighted Accuracy:(64.1)
6. [reuter=1, and=1]: 944 ----(IMPLIES)------>[&#3=1]: 944 <RULE CONFIDENCE:(1)> Weighted Accuracy:(65)
7. [reuter=1, in=1]: 915 ----(IMPLIES)------>[&#3=1]: 915 <RULE CONFIDENCE:(1)> Weighted Accuracy:(63)
8. [reuter=1, said=1]: 934 ----(IMPLIES)------>[&#3=1]: 934 <RULE CONFIDENCE:(1)> Weighted Accuracy:(64.31)
9. [reuter=1, &lt=1]: 900 ----(IMPLIES)------>[&#3=1]: 900 <RULE CONFIDENCE:(1)> Weighted Accuracy:(61.97)
10. [reuter=1, a=1]: 866 ----(IMPLIES)------>[&#3=1]: 866 <RULE CONFIDENCE:(1)> Weighted Accuracy:(59.63)
11. [reuter=1, it=1]: 775 ----(IMPLIES)------>[&#3=1]: 775 <RULE CONFIDENCE:(1)> Weighted Accuracy:(53.36)
12. [reuter=1, for=1]: 744 ----(IMPLIES)------>[&#3=1]: 744 <RULE CONFIDENCE:(1)> Weighted Accuracy:(51.23)
13. [reuter=1, mln=1]: 702 ----(IMPLIES)------>[&#3=1]: 702 <RULE CONFIDENCE:(1)> Weighted Accuracy:(48.34)
14. [of=1, th=1]: 906 ----(IMPLIES)------>[&#3=1]: 906 <RULE CONFIDENCE:(1)> Weighted Accuracy:(62.38)
15. [of=1, and=1]: 873 ----(IMPLIES)------>[&#3=1]: 873 <RULE CONFIDENCE:(1)> Weighted Accuracy:(60.11)
16. [of=1, a=1]: 823 ----(IMPLIES)------>[&#3=1]: 823 <RULE CONFIDENCE:(1)> Weighted Accuracy:(56.67)
17. [th=1, to=1]: 873 ----(IMPLIES)------>[&#3=1]: 873 <RULE CONFIDENCE:(1)> Weighted Accuracy:(60.11)
18. [th=1, and=1]: 820 ----(IMPLIES)------>[&#3=1]: 820 <RULE CONFIDENCE:(1)> Weighted Accuracy:(56.46)
19. [th=1, said=1]: 900 ----(IMPLIES)------>[&#3=1]: 900 <RULE CONFIDENCE:(1)> Weighted Accuracy:(61.97)
20. [th=1, a=1]: 813 ----(IMPLIES)------>[&#3=1]: 813 <RULE CONFIDENCE:(1)> Weighted Accuracy:(55.98)
21. [th=1, it=1]: 747 ----(IMPLIES)------>[&#3=1]: 747 <RULE CONFIDENCE:(1)> Weighted Accuracy:(51.43)
22. [to=1, said=1]: 855 ----(IMPLIES)------>[&#3=1]: 855 <RULE CONFIDENCE:(1)> Weighted Accuracy:(58.87)
23. [and=1, in=1]: 776 ----(IMPLIES)------>[&#3=1]: 776 <RULE CONFIDENCE:(1)> Weighted Accuracy:(53.43)
24. [and=1, said=1]: 789 ----(IMPLIES)------>[&#3=1]: 789 <RULE CONFIDENCE:(1)> Weighted Accuracy:(54.33)
25. [and=1, a=1]: 743 ----(IMPLIES)------>[&#3=1]: 743 <RULE CONFIDENCE:(1)> Weighted Accuracy:(51.16)

C. Performance Analysis:



Graph shows that Number of rules generated in the proposed approach vs. time taken to generate rules.

V. CONCLUSION AND FUTURE SCOPE

Many data mining techniques have been proposed in the last decade. These techniques include sequential pattern mining, maximum pattern mining, association rule mining, frequent itemset mining, , and closed pattern mining[1]. However, using these discovered knowledge (or patterns) in the field of text mining is difficult and ineffective. The reason is that some useful long patterns with high specificity lack in support (i.e., the low-frequency problem). In order to rectify the problems in existing approaches, Proposed gives robust pattern discovery with decision making rules. Proposed preprocessing framework gives better execution time compare to existing approaches. In future this work is extended to ontology based framework to give more accurate results in web intelligence.

REFERENCES

[1] Effective Pattern Discovery for Text Mining Ning Zhong, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 24, NO. 1,

[2] Hybrid Approach to Improve Pattern Discovery in Text mining Charushila Kadu, International Journal of Advanced Research in Computer and Communication Engineering

[3] 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.

[4] 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’l Forum on Research and Technology Advances in Digital Libraries (ADL ’98), pp. 2-11, 1998.

[5] R. Baeza-Yates and B. Ribeiro-Neto,Modern Information Retrieval. Addison Wesley, 1999.

[6] T. Chau and A. K. C.Wong, “Pattern discovery by residual analysis and recursive partitioning,” IEEE Trans. Knowledge Data Eng., vol. 11, pp.833–852, Nov./Dec. 1999.

[7] Nitin Jindal, Bing Liu, Ee-Peng Lim, “Finding Unusual Review Patterns Using Unexpected Rules”.