A Naive Clustering Algorithm for Text Mining

Aishwarya Kappala, Sudhakar Godi
Dept. of Computer Science & Engineering.
Swarnandhra College of Engineering &Technology
JNTUK, East Godavari, India.

ABSTRACT

Predefined categories can be assigned to the natural language text using for text classification. It is a “bag-of-word” representation, previous documents have a word with values, it represents how frequently this word appears in the document or not. But large documents may face many problems because they have irrelevant or abundant information is there. This paper explores the effect of other types of values, which express the distribution of a word in the document. These values are called distributional features. All features are calculated by tfidf style equation and these features are combined with machine learning techniques. Term frequency is one of the major factor for distributional features it holds weighted item set. When the need is to minimize a certain score function, discovering rare data correlations is more interesting than mining frequent ones. This paper tackles the issue of discovering rare and weighted item sets, i.e., the infrequent weighted item set mining problem. The classifier which gives the more accurate result is selected for categorization. Experiments show that the distributional features are useful for text categorization.

Keywords

Text Classification, Text Mining, Machine Learning, Compactness, tfidi, Weighted database.

1. INTRODUCTION

Text Mining is one of an exploratory data mining technique widely used for retrieving valuable correlated data from large documents. The first step to perform text mining was targeted on finding frequent item sets, i.e., patterns whose observation is identifying occurrence of term frequency in the input data. Term Frequency found applications in a number of real-life contexts (e.g., market basket analysis [1], medical image processing [2], biological data analysis [3] and disease data sets [4]). However, several traditional methods ignore the interest of each item within the analysed data sets. So, to allow treating items differently based on their relevance in the text mining process, the notion of score item set has also been introduced. A Score is associated with each data item and distributes its significance within each transaction.

Moreover such type of problems, Text Categorization assigns predefined categories to natural language text according to its relevance item sets. Text categorization has attracted more and more attention because it supports to natural language processing. It is a supervised learning problem, many classifiers widely used in the Machine Learning (ML) community have been applied, such as Naive Bayes, Fuzzy Classifiers, Decision Trees, Artificial Neural Networks, k Nearest Neighbor (kNN), Support Vector Machine (SVM), and AdaBoost. Recently, some excellent results have been obtained by fuzzy logics and SVM.

Every classification technique always described the “bag of words” representation because each document in the large data sets shows how many words are appeared in the document. Mostly it consider a word with value i.e. values should be assigned to every word in the document. Finally it describes how many words in the document and how frequently this word appears in the document [5]. When ever to find out all values in the document then those values will be useful for text categorization. Consider one example, “How are you” and “How you are” are two separate sentences describes to the same vector using the frequency-related values, but their meanings are entirely different. This example clearly describes appearance, frequency of the word and the distribution of word in the document. So, this paper introduces to design and develop some distributional features to measure the appearance of a word’s distribution in a document [6].

Therefore, this paper attempts to design some distributional features to measure the characteristics of a word’s distribution in a document. The first consideration is the compactness of the appearances of a word. Here, the compactness measures whether the appearances of a word concentrate in a specific part of a document or spread over the whole document. In the former situation, the word is considered as compact, while in the latter situation, the word is considered as less compact. This consideration is motivated by the following facts. A document usually contains several parts. If the appearances of a word are less compact, the word is more likely to appear in different parts and more likely to be related to the theme of the document [7].

The contribution of this paper is the following:

1) Designing of distributional features for text categorization. By using these features can help improve the efficiency of performance, while requiring only a little additional cost.

2) How to use the distributional features in large data bases. Adding traditional term frequency with distributional features, that will be increase the performance results.

3) Discussion for identifying which factors are affecting the performance of the distributional features.

4) The advantage of the distributional features is closely related to the length of documents in a corpus data base and supports to natural language documents also.

The paper is organized as follows: In the next section, some related concepts about extraction of distributional features, the section 3 provides utilization of distributional features. In section 4 experiment results of proposed algorithms FDDSS is presented. In section 5 describes maintenance of database and conclusions are given in section 6.

2. EXTRACTION OF DISTRIBUTIONAL FEATURES

As per the features for text categorization are mentioned, the word “feature” generally have two types of related meanings. One is representation of a document or to index of a document, while the other one targets on how to assign an appropriate score to a given feature. Consider “bag of words” as an example. Using the former meaning, the feature is a single word, while term frequency and inverse document frequency score is the feature given another meaning. Finally these features are used for text categorization based on these two meanings. The second meaning of feature is the score assigned to a given feature comes from two sources: intra document and inter document. The intra document-based score uses information within a document, while the inter document-based score uses information in the corpus database. For tfidf, the term frequency part can be regarded as a score from an intra document source, while the inverse document frequency part is a weight from an inter document source.

The inverse document frequency was derived in order to distribute term frequencies evenly on the interval from 0 to 1 [8]. It is used for the importance of each sentence with a score in entire document. Especially, the importance of a sentence was measured by two types. One was to calculate the correlation between the title and a given sentence, while the other one summed the importance of all words appearing in this sentence as the final importance. Given the importance of a sentence, for a word, a score term frequency was used to replace the original term frequency, where each appearance was ranked by the importance of the sentence where this appearance occurred [9].

This section describes how to distributional features are useful for text categorization based on term frequency, inverse document frequency and compactness of the word.

2.1. Method for Word’s Distribution

In word’s distribution, firstly the entire document is divided in to many parts, secondly to find out this word appears how many times in the document. It can be maintained by an array, it is the combination of total number of parts in the document. Every document should be followed by three types of passages below. Here discourse passage deals logic components of the documents such as sentences and paragraphs. The semantic passage deals meaning of the document according to contents. This is more accurate because each paragraph consists corresponding topic or subtopic only [10]. The Window passage is a sequence of all words, it is simple to implement. Here this method may used by discourse passage, Window passage except semantic passage because it purely belongs to meaning of the document. But experiment results show how to explore different sizes in every document [11].

The following is an example for the word of “wheat” in the document. That document has ten sentences, the distribution of the word “wheat” is depicted in Fig. 1, then the distributional array for “wheat” is [3, 1, 0, 0, 2, 0, 0, 1, 0, 2].

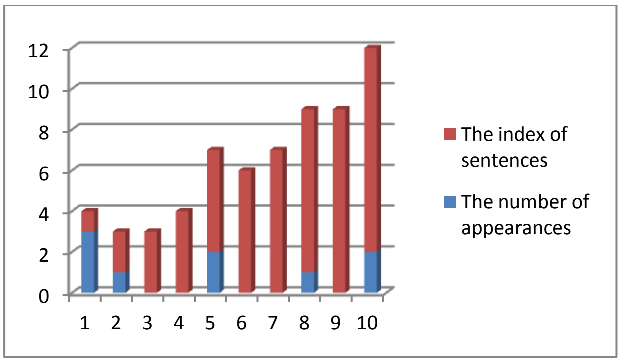

Fig 1. The distribution of ‘Wheat’ in a document.

2.2. Find Position of the Word

To find out compactness of the word in the document, this method followed by three types f partitions as follows. These are useful for position of the word in the document.

ComPactPartNum. It shows a single word appears in how many parts of the document. Suppose a word has low compact it appears in different parts of the document.

ComPactFLDist. This is the difference between first appearance of the word and last appearance of the word. Suppose if it is high compact, the distance between first and last appearance is less.

ComPactPosV ar. It shows various positions of appearances of the word in the document.

Consider a document D with n sentences, the word distribution is array(W,D)= { $C_1, C_2, C_3, \ldots\ldots, C_n$}. The compactness of appearance of the word and position of the word will be defined as follows.

First Appearance (W,D)=Min_i Ci > 0 ? i : n, where i € { 1..n}.

ComPact Part Num (W,D)=$\sum^{n}_{i=1}$ Ci > 0 ? 1 : 0,

Last Appearance (W,D)=Max_i Ci > 0 ? i : -1, where i € { 1..n}

ComPact FL Dist (W,D)= Last Appearance (W,D) - First Appearance (W,D),

Count (W,D)= $\sum^{n}_{i=1}$ Ci ,

The following example shows how to calculate distributional features for “Wheat”.

First Appearance (Wheat, D)

= min{0,7,5,9,3,9,2,5,8,10}= 0,

$\mathrm{ComPact}_{\mathrm{PartNum}}$ (Wheat, D)

= {1+0+0+1+1+0+0+1+1+1} = 6,

Last Appearance (Wheat, D)

= max{0,-1,-2,1,-5,-1,-3,4,8,7}= 8,

$\mathrm{ComPact}_{\mathrm{FLDist}}$ (Wheat, D)= 8-0=8,

Count (Wheat, D) =3+1+2+1+2= 8,

Centroid (Wheat, D)= (3*0+1*1+2*4+1*8+2*7)/8= 3.875

$\mathrm{ComPact}_{\mathrm{PosVar}}$ (Wheat, D)

= (3*3.875+1*2.725+2*0.375+1*2.375+2*3.425)/8= 4.497.

Firstly the process reads text documents and finds out term frequency then it send to corpus data base. This data base identifies length of the documents and load it in to buffer. Then applying of all distributional features on that documents and finally concludes whether this document belongs to which category based on their classification results.

The following figure.2 shows the architecture of extraction of term frequency and distributional features for text categorization.

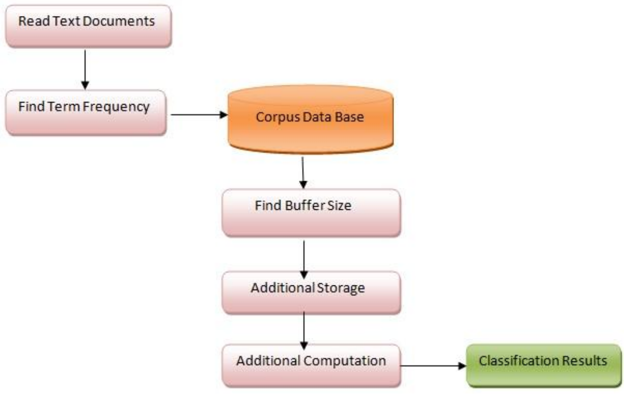

Fig. 2. Extraction of term frequency and distributional features

3. UTILIZATION OF DISTRIBUTIONAL FEATURES

The word relevant score used in the number of documents, those are containing a word to divide the number of documents without this word, instead of the total number of documents in inverse document frequency. Moreover, many researchers and users believed that the inverse document frequency derived directly from text retrieval was not well suited for text categorization where the categories of training documents were available. Before going to extraction process a lot of supervised scores were introduced. The term frequency in tfidf can be treated as a value that measures the priority of a word in a particular document. Whereas the importance of a word can be measured by its term frequency, as well as it measured by the compactness of its appearance of the word and the position of its first appearance of the word [13].

Algorithm:

Input: Take sample trained set and text document.

Output: Classification details with maximum score.

Method:

1. Select all trained set samples. i.e. {C1, C2, C3, C4, C5,....,Cn}

2. Read the input text document 'D'.

3. Pre-process the input document.

4. Applying stop words and stemming process.

5. Find tfidf, first appearance, last appearance, and compactness of the input document.

tfidf (W,D)=importance(W,D)*idf(w)

FA (W,D)=f(First App(W,D), length(d))

CP_{FLD} (W,D)=ComPactFLDist(W,D)+1/ length (D)

6. Compare input document 'D' with all categories of trained samples.

7. If D € {C1, C2, C3, C4, C5,....,Cn} then place it in to particular category otherwise that is act like as an individual cluster.

8. Find score of the input document which belongs to highest category.

9. Stop the process.

4. EXPERIMENT RESULTS

Text Classification is one of the biggest issues in data mining. It uses different types of classifiers, but SVM and kNN are two best classifiers for large data sets. So, this paper also describes and compares these two classifiers for best results.

4.1. Input Data Sets

Data sets are main resources for text classification. Here this paper describes three types of data sets like Reuters-21578 corpus, 20 Newsgroup corpus and WebKB corpus. The Reuters-21578 corpus contains 21,578 articles taken from Reuter's newswire. Some of the categories that have at least one document in both the training set and the test set are extracted. After completion of removal documents, suppose they do not belong to any category they will act like as own category. Here some of the classifications were done in Reuters, those are 7,770 documents in the training set and 3,019 documents in test set. After pre- processing, stemming and stop-word removal, the vocabulary contains 12,158 distinct words that occur in at least two documents of the Reuters corpus [12].

Category C_i		Expert Judgement	
		Yes	No
Classifier Judgement	Yes	TP_i	FP_i
	No	FN_i	TN_i

Table. 1. The Contingency for Category Ci

The 20 Newsgroup corpus contains 19,997 articles taken from the Usenet newsgroup collections. Here also duplicate documents are removed and the documents with multiple labels are detected.

The WebKB corpus is a collection of 8,282 web pages obtained from four academic domains. All HTML tags are removed here after stop words removal and stemming then the vocabulary contains 14,467 distinct words that occur in at least two documents.

4.2. Performance Measurements

Performance measurements are depending upon their ranks of the classifier data. Here SVM and kNN classifiers are shows best results and maximum correlations. This process followed by three data sets and two classification measures. The following figure shows average rank of each candidates in large data sets. Suppose the average rank is less that can be treated as best performance measure of candidates. Candidates may represented by either combination of $\text{TF+CP}_{\text{PN}}\text{+FA}_{\text{GI}}$ nor $\text{TF+CP}_{\text{PN}}\text{+FA}_{\text{GLI.}}$ The following figure shows $\text{TF+CP}_{\text{PN}}\text{+FA}_{\text{GI}}$ performance is best when compared to all candidates [19].

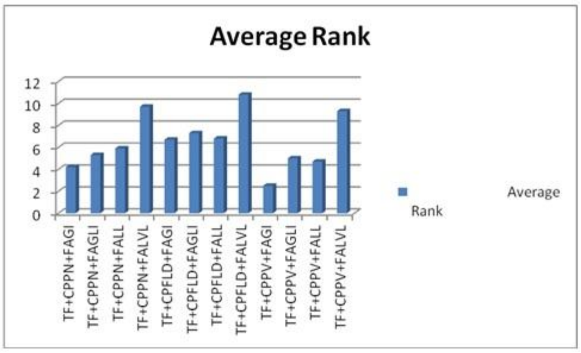

Fig.3 Average Rank of Different Candidates in a data base.

The average rank of TF+CP_{FLD}+FA_{LVL} shows best performance than others. The next best performances are TF+CP_{PN}+FL_{VL}, TF+CP_{PV}+$FA_{LVL.}$

4.3. Analysis for First Appearance Features

The analysis of FA describes about we weighting function 'f' and the strategy of considering the first appearance of a word .It always checked out which parts of the documents affected by FA features. More over a group of weighted term frequency (WET) features are generated by using the weighting function 'f' to weight each appearance of a word in the document. Let us consider a document 'D' with 'n' sentences the distributional array of the word t is array(W,D)=[C_1,C_2,C_3,....,C_n].

The weighted term frequency is calculated as follows:

$$WET\ (W,D)= \sum^{n}_{i=1} \ C_i * f\ (i,\ length\ (D))\ /\ Size(D)$$

The following figure 4 shows comparison between three datasets Reuters, Newsgroup and WebKB. Figure describes applying of kNN classifier in to three data sets. Whenever it applied on particular data sets the Reuters data set consists of negative values. More well as the combination of TF+CPPV+XGI shows out standard performance than CP_{PV}+X_{GI}, TF+X_{GI}, $X_{GI.}$ In this performance measurement of WebKB is efficient than Reuters, Newsgroup [14].

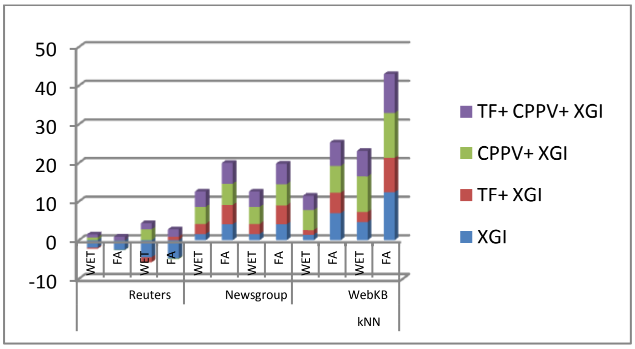

Fig.4 The Comparison between the FA Feature and the WET Feature by using kNN Classifier

The following figure 5 also shows comparison between three datasets Reuters, Newsgroup and WebKB. It was used by the SVM classifier, this classifier performs best result like kNN with the combination of Word Frequency, Compactness and Position Variance. In this case Reuters data sets had worst performance than others [20]. Compared to remaining data sets WebKB provide best performance. Therefore the combination of TF+CP_{PV}+X_{GI} shows out standard performance than CP_{PV}+X_{GI}, TF+X_{GI}, $X_{GI.}$

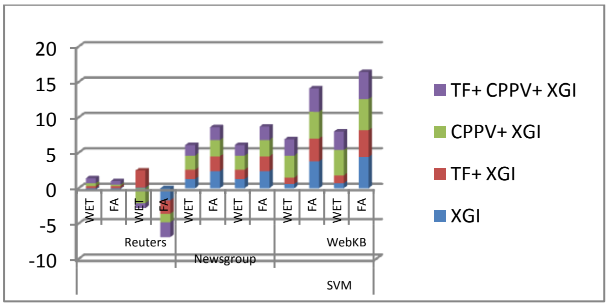

Fig.5 The Comparison between the FA Feature and the WET Feature by using SVM Classifier

Figure 6 shows that FA performs better than WET, especially on 20 Newsgroup and WebKB. The cases where FA performs worse than WET performance. Still WET still improves the base line, it is believed that the effect of FA on 20 Newsgroup and WebKB is brought by both the weighting function and the aggressive strategy that throws all appearances of a word except the first one [15] [16]. For Reuters, the effect of this aggressive strategy is not obvious. It shows the comparison between kNN and SVM classifiers in three data sets. When compared to two classifiers, kNN shows best performance than SVM. Here also in both cases TF+CPPV+WETGI perform best result than individual WET and TF. As per results of two classifiers kNN is best classifier than SVM in these three documents sets [18].

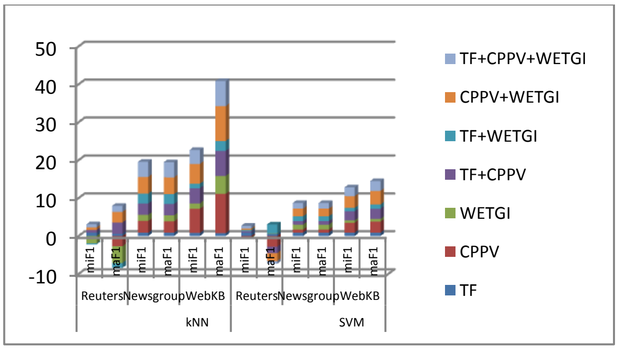

Fig.6 The Comparison between kNN and SVM Classifiers.

5. MAINTAINANCE OF DATABASE

Choose an input file and that file compared to all trained classification files. That is each word of input file should be compared to each word of existed files of classification. Finally, the input file may have different weight or score. We should consider maximum score of the input file, automatically that file was stored based on maximum score of the classification. At the same time the database may note joining and leaving time of the files [15] [17].

Trained Classifications	Score
Acq	2342
Corn	6
Crude	757
Earn	1578
Grain	410
Interest	495
Money-tx	1174
Ship	216
Trade	735
Wheat	0

Table. 2. Classification result for input document..

The classification results show Acq holds highest score than other trained sets. So the input dataset should go to acq data set because it has highest score. Earn, Money-tx data sets also represent the next highest score but the complete input data set goes to acq only. Here wheat data set did not get any related information to input data set that's why it represents zero value.

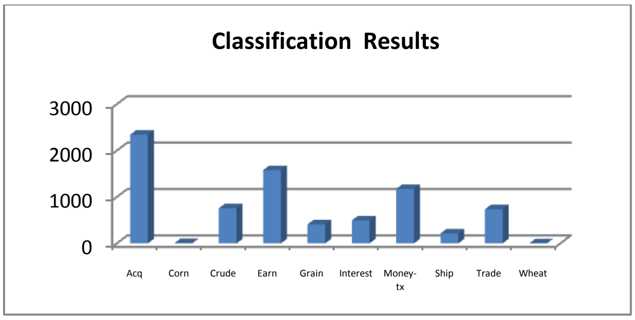

Fig.7. The Comparison of classification results.

6. CONCLUSION

Text Categorization generally uses frequency of the word and appearance of the word. Such type of features is not enough for fully expressing the information contained in a particular document. That's why this paper introduces distributional features of a word in text categorization. These distributional features are very beneficial to classification of the data. Distributional features consists compactness of appearances of a word, the position of the first, last appearance and difference between FL appearances of the word in the document. Three types of compactness-based features and four types of position of appearance of the words are implemented to reflect the different aspects. The term frequency and inverse document frequency style equations are used for word frequency, and the machine learning technique is used to utilize the distributional features. These features are very beneficial to all documents in data sets whenever the documents are long and writing style is casual. Finally these features supports to natural language documents also. This paper maintains the data base for classification details.

7. REFERENCES

[1] R. Bekkerman, R. Elaine, N.Tishby, and Y.Winter, "Distributional Word Clusters versus Words for Text Categorization,"J. Machine Learning Research, vol. 3, pp. 1182-1208, 2003

[2] G. Narasimha Rao, R. Ramesh, D. Rajesh, D. Chandra sekhar."An Automated Advanced Clustering Algorithm For Text Classification". In International Journal of Computer Science and Technology, vol 3,issue 2-4, June, 2012, eISSN : 0976 - 8491,pISSN : 2229 – 4333.

[3] D.CAI, S.P. Yu, J.R. Wen, and WY. Ma, "VIPS: A Vision-Based Page Segmentation Algorithm" Technical Report MSR-TR-2003-79, Microsoft Seattle, Washington, 2003.

[4] J.P. Calan, "Passage Retrieval Evidence in Document Retrieval,"Proc. ACM SIGIR '94, pp. 30310, 1994.

[5] Rao, Gudikandhula Narasimha, and P. Jagdeeswar Rao. "A Clustering Analysis for Heart Failure Alert System Using RFID and GPS." ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India-Vol I. Springer International Publishing, 2014.

[6] M. Craven, D. DiPasquo, D. Freitag, A. K. McCallum, T. M. Mitchell, K. Nigam, and S. Slattery, "Learning to extract symbolic knowledge from the world wide web," in Proceedings of the 15th National Conference for Artificial Intelligence, Madison, WI, 1998, pp. 509–516.

[7] F. Debole and F. Sebastiani, "Supervised term weighting for automated text categorization," in Proceedings of the 18th ACM Symposium on Applied Computing, Melbourne, FL, 2003, pp. 784–788.

[8] T. G. Dietterich, "Machine learning research: Four current directions,"

[9] D. Lewis, "Reuters-21578 text categorization test colleciton, dist. 1.0," 1997. AI Magazine, vol. 18, no. 4, pp. 97–136, 1997.

[10] Y. Yang, "An evaluation of statistical approaches to text categorization," in Inf. Retreival, vol. 1, pp. 69–90, 1999.

[11] S. Shankar and G. Karypis, "A Feature Weight Adjustment Algorithm for Document Classification," Proc. SIGKDD '00 Workshop Text Mining, 2000.

[12] K. Sun and F. Bai, "Mining Weighted Association Rules Without Preassigned Weights," IEEE Trans. Knowledge and Data Eng., vol. 20, no. 4, pp. 489-495, Apr. 2008.

[13] S. Zhu, X. Ji, W. Xu, and Y. Gong, "Multi-labelled classification using maximum entropy method," in Proc. Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2005, pp. 1041–1048

[14] J.P. Calan, "Passage Retrieval Evidence in Document Retrieval,"Proc. ACM SIGIR '94, pp. 30310, 1994.

[15] X. Ling, Q. Mei, C. Zhai, and B. Schatz, "Mining multi-faceted overviews of arbitrary topics in a text collection," in Proc. 14th ACM SIGKDD Knowl. Discovery Data Mining, 2008, pp. 497–505.

[16] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," in J. Mach. Learn. Res., vol. 3, no. 1, pp. 1157–1182, 2003.

[17] T. Joachims, "Transductive inference for text classification using support vector machines," in Proc. Annu. Int. Conf. Mach. Learn., 1999, pp. 200–209.

[18] X.-L. Li, B. Liu, and S.-K. Ng, "Learning to classify documents with only a small positive training set," in Proc. 18th Eur. Conf. Mach. Learn., 2007, pp. 201–213.

[19] Y. Li, A. Algarni, S.-T. Wu, and Y. Xue, "Mining negative relevance feedback for information filtering," in Proc. Web Intell. Intell. Agent Technol., 2009, pp. 606–613.

[20] S.-T. Wu, Y. Li, and Y. Xu, "Deploying approaches for pattern refinement in text mining," in Proc. IEEE Conf. Data Mining, 2006, pp. 1157–1161.