Graph-based KNN Text Classification

Zonghu Wang
School of Computer Science and Technology
Xidian University
Xi'an China
zonghuwang@gmail.com

Zhijing Liu
School of Computer Science and Technology
Xidian University
Xi'an China
liuprofessor@vip.163.com

***Abstract*—Vector space model is used in most text categorization methods without considering the important information such as the order and co- occurrence of words within the text. In this paper we describe a novel approach of text classification using graph-based KNN. We reduce the number of features dimensions by a combined feature selection method. Then we present an improved graph–based text representation model and describe a novel graph–based KNN algorithm to predict the category of the texts in the testing set. The result shows that our approach can outperform traditional VSM-based KNN methods in terms of both accuracy and cost time.**

Keywords- text; categorization;graph;feature;KNN

I. INTRODUCTION

The amount of online text data has grown greatly with rapid development of the Internet in recent years. As a result, there is a need to provide auto effective categorization for these huge and unstructured text resource. Text categorization has become an important task in text mining. Many machine learning algorithms have been applied to text categorization, such as Naïve Bayes (NB), k-Nearest Neighbor (KNN) Centroid Classifier Rocchio and Support Vector Machines (SVM).Most of current document categorization methods are based on the vector space model (VSM) [1], which is a widely used data representation. The VSM represents each document as a feature vector of the terms (words or phrases) in the document. Each feature vector contains term weights (usually term-frequencies) of the terms in the document. The similarity between documents is measured by one of several similarity measures that are based on such a feature vector. Examples include the cosine measure and the Jaccard measure. method VSM makes the representation and learning easy and highly efficient, but it ignores the structural information of feature terms within a document which are crucial to natural languages understanding. Therefore, from the view of nature language, these commonly used models are not perfect.

The Graph-based text representation technique was recently developed [2][3].Adam Schenker[4] presented a graph model to perform web documents clustering and classification. However, this model is built on boolean association of the occurrences among feature terms without considering the frequency of such occurrences. Manuel, Aurelio, and Alexander present an approach of information retrieval with conceptual graph matching [5]. Wei jin and Rohini k. also proposed a graph-based model which is capable of capturing term order, term frequency, term co-occurrence and term context in documents and apply it in discovering unapparent associations between two or more concepts from a large text corpus[6].Inspired by these models, we propose a simple but effective graph model to capture the information of a text by introducing centroid feature vector to preserve more information of a document. Then we use a novel graph-based KNN algorithm to perform the text categorization task by considering the weight of both nodes and edges in two graphs .The Result suggested the method used in this paper was indeed providing satisfactory accuracy and efficiency.

The rest of this paper is organized as follows: In section 2, we present the pre-processing and feature selection of text. Section 3 describes the improved graph-based text representation model proposed in this paper and the detail of graph–based KNN algorithm. In section 4 we give the experiment result and the last section is the conclusion.

II. PRE-PROCESSING AND FEATURE SELECTION

One difficulty in text categorization is the high dimension of feature space, Feature selection is necessary in order to remove noise features ,reduce the number of dimensions, simplify the calculation, avoid overfitting and so on. The commonly used text feature selection methods include Information Gain(IG), χ2-statistic(CHI),Document Frequency(DF) and Mutual Information (MI).

Mutual Information (MI)

MI is a concept in information theory, it is used to measure the mutual dependence of two signals in a message. During the field of feature selection it is usually used to calculate the dependence between the feature t and the class c. The formulas which measure mutual information between t and class i are as follows (the two formulas are equivalent):

$$I(t,c) = \log \frac{P_r(t,c)}{P_r(t \mid c) \times P_r(t)} \quad (1)$$

$$I(t,c) = \log P_r(t \mid c) - \log P_r(t) \quad (2)$$

χ2- statistic(CHI)

It measures the lack of independence between term w and category c. If w and c are independent, χ2- statistic has the lowest value of zero.It is defined as:

$$\chi^2(w,c) = \frac{N \times (P(w,c) \times P(\bar{w},\bar{c}) - P(w,\bar{c}) \times P(\bar{w},c))}{P(w) \times P(\bar{w}) \times P(c) \times P(\bar{c})} \quad (3)$$

Where N is the total number of documents in the training set.P(w,c) is the probability when term w and category c appear simultaneously,P(w) is the probability of w in document d,and P(c) is the probability when the text belongs to category

This research was supported by National Science & Technology Pillar Program under Grand No. 2007BAH08B02.

c.Similarly the means of P(w, $\bar{c}$),P($\bar{w}$,c),P($\bar{w}$,$\bar{c}$),P($\bar{w}$),P($\bar{c}$) can be known.

Combined Feature Selection Method(CFM)

Though χ2- statistic is one of the most efficient methods at present, it also has some disadvantages. For example, it increases the weights of words which appear in appointed class with low frequency, but high in other classes and it reduces the weights of the low frequency words. According to formula (1), for such words, when $P(t,c_i)\to 0$, $P(t)$ and $P(c_i)$ are not tending to zero, then $P(t,c_i)/(P(t)\bullet P(c_i))\to 0$, so $I(t,c)$ will tend to negative infinitude, and these words will be thrown out. According to formula (2),

We proposed method for the words which have same $\log P_r(t\mid c)$, the weights of low frequency words will be higher. So the problems above are solved, in this paper we combine χ2- statistic with Mutual Information method is proposed in this paper. The method is the following indicator function:

$$E(t,c)=\alpha E_1(t,c)+\beta E_2(t,c)\quad 0<\alpha,\beta<1,\alpha+\beta=1 \tag{4}$$

Where $E_1(t,c)$ is the weight by χ2- statistic, and $E_2(t,c)$ is the weight using Mutual Information method.

In our experiment, chinese word segmentation was done first ,then we remove the stop words and 1-character Chinese word which contribute little to distinguish different categories. Then we calculate the weight of terms in text using the above feature selection method. Finally, we send the selected feature terms with high weight to the classifier to construct graph nodes.

III. KNN TEXT CATEGORIZATION BASED ON GRAPH

This paper proposed an improved graph-based text representation. After the step of feature selection, we convert all texts into graph basing on the selected feature terms. Then we use the improved graph-based KNN algorithm to predict the category of the texts in testing set..

A. Graph-based text representation model

A graph G is a 3-tuple: G=(V, E,FWM), where V is a set of nodes (also called vertices), E is a collection of weighted edges connecting the nodes.FWM(Feature Weight Matrix) is defined as the feature weight matrix of edges.

- Node：

Unique feature terms obtained from the train set using the feature selecting methods described in section 2 .

- Edge：

Constructed based on order and co- occurrence relationship between feature words. If two feature words appear within a step length (e.g., 2), we assign an edge from the former node to the latter.

- Feature Weight Matrix：

It is defined as (6), the weight w_{ij} can be calculated as co-occurrence frequency of feature term f_i and f_j within a step length. With the fact that two feature words always have fixed order in a sentence. Such as “Information technology”, we never use the reverse situation such as “technology information”. So we just need to calculate the weight w_{ij} when i>j.To improve the effect of graph-based text representation, we construct an improved diagonal matrix to store the structural information of text by introducing the define of centroid feature vector W as follows:

$$W=\{w_{11}, w_{22}...w_{ii}...w_{nn}\} \tag{5}$$

we assign the frequency of feature terms f_i which appears in a text to w_{ii} defined as the i-th diagonal weight of the matrix.By this means,the graph preserves more feature information of text such as frequency, order, co- occurrence of terms. The result of our experiment shows this novel graph-based text representation model improve the performance of classification efficiently.

$$\begin{array}{c} \\ f1 \\ \vdots \\ fn \end{array}\begin{array}{c} \begin{array}{ccc} f1 & \cdots & fn \end{array} \\ \begin{pmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & w_{nn} \end{pmatrix} \end{array} \tag{6}$$

After the processing of converting ,we get the Feature Weight Matrix which represents the text .

B. Traditional KNN Categrization

KNN[7] is a classical statistical pattern recognition algorithm. Its idea is very simple: for a text document to be classified, K nearest neighbors should first be found out according to a certain similarity computing strategy, then all the neighbors similarity be added according to their category respectively. Finally, classify the text document into the category with maximum similarity. KNN can be represented by the following formula:

$$y(t,c_j)=\sum_{d_i\in KNN} Sim(t,d_i)y(d_i,c_j)-b_j \tag{7}$$

Where t is the document to be classified, d_i is the i-th sample document, c_j is the j-th category, and $y(d_i, c_j)\in\{0,1\}$ denotes whether document d_i belongs to category c_j (if d_i belongs to c_j then $y(d_i, c_j)=1$ otherwise $y(d_i, c_j)=0$)， b_j is a preset threshold.

$Sim(t,d_i)$ is the similarity between the document to be classified and the sample documents. If the text is represented by Vector Space Model, the similarity can be computed as follows：

$$Sim(d_i,d_j)=\frac{d_i\cdot d_j}{|d_i|\times|d_j|}=\frac{\sum_{k=0}^{n} w_{ik}\times w_{jk}}{\sqrt{\sum_{k=0}^{n} w_{ik}^{\ 2}}\times\sqrt{\sum_{k=0}^{n} w_{jk}^{\ 2}}} \tag{8}$$

The set of category graphs represent the learned classification model. This model can be used to predict the category of test text.. After the process of feature selection, the dimension of feature terms reduce obviously, most noisy feature terms are removed, but the most discriminative feature terms are preserved. The nodes, edges and their weight in a category graph keep most features of texts of a certain

category. Also the computational complexity is reduced greatly.

C. Improved KNN Classification based on Graph

KNN has advantages as simplicity, stability and high classification precision, but its classification speed is slow. The main reason causing this is that KNN is an example-based algorithm, i.e., to find the K nearest neighbors, it has to compute the similarity between the document to be classified and all of the sample documents one by one. Its time complexity is directly proportional to the sample size. We can see that most of the time is cost in computing similarity between the document to be classified and the large amount of sample documents. Therefore, how to reduce the sample size is critical for the improvement of the classification speed of KNN. Many researches [8] show that KNN's precision is greatly dependent on the sample size. Thus we cannot improve the classification speed simply by reducing the sample size. Then how can we reduce KNN's complexity of similarity computation while retaining the size of the sample documents? We adopt a two-step similarity computation strategy between test text and sample documents. Our method is based on the improved graph-based text representation model proposed in this paper. In this way, the amounts of computation in KNN similarity computation is reduced greatly, while information of the original sample documents is still retained.

As a graph consists of nodes, edges and the weight of edges. We can define the similarity measure of two graphs by those elements. We define two improved classification measures in calculating Similarity (g_i, cg_i) and evaluate them in experiment.

- *FW(Feature Weight):* it describes the similarity between two graphs by weight of both nodes and edges appear in both two graphs. It can be calculated as follows：

Input:
Testing graph g_i , training graph cg_i
Output:
Fw
Procedure:

```
For each edge in g_i
    If edge in cg_i
      If(w_ij(g_i)>=w_ij(cg_i) //w_ij is the weight of edge
          If(j>i)
              Fw+=αw_ij(cg_i)
          Else if(j=i)
              Fw+=w_ij(cg_i)
          End if
      Else If(w_ij(g_i)<w_ij(cg_i)
          If(j>i)
              Fw+=αw_ij(g_i)
          Else if(j=i)
              Fw+=w_ij(g_i)
          End if
      End if
    End if
  End for
```

Algorithm 1. Calculate Feature weight

α is a value we use to enhance the effect of edge weight in classification. We assume that the weight of edges contributes more than nodes in classification. In our experiment, we set α=2.The result of experiment shows it is reasonable.

- *Nft(Node fit percent):* it shows how many nodes in train graph with weight>0 also appear in test graph. It is defined as follows:

$$\text{Nfp}=\frac{|\{node \mid node\in g_i \wedge node\in cg_i\}|}{|\,Number\ of\ Feature\ terms\,|} \quad (9)$$

We first calculate the Nft value of the test and train text, by the define of centroid feature vector W see (5) we can calculate it simply. If Nft value is greater than the threshold we predefined then we calculate the Fw value of two graphs, otherwise we can learn that this two graphs are not belong to the same category ,so the Fw calculation needs not to be done. By this way the complexity of similarity computation can be reduced greatly. The detail of the graph-based KNN which we called GKNN algorithm is described as follows:

Input:
Testing set graphs $G=\{g_1,g_2,\ldots g_i, \ldots g_n\}$, value K
Training set graphs $CG=\{cg_1,cg_2,\ldots cg_i, \ldots cg_n\}$
Output：
Result set $R=\{r_1,r_2,\ldots r_i, \ldots r_n\}$
r_i is the categorization result of text d_i
Procedure:

```
1 For each g_i in G
2.  Initial List RL to store Fw and text category（length is K）
3    For each cg_i in CG
4.       If Nfp(g_i,cg_i)>α
5.            Calculate Feature weight Fw(g_i,cg_i)
6             If RL is not full
7                 Add Fw(g_i,cg_i) and category of cg_i to RL
8             Else If RL is full
9                 If Fw(g_i,cg_i)>min(Fw_i in RL)
10                    Replace Fw_i in RL with Fw(g_i,cg_i)
11                End if
12            End if
13       End if
14  . End For
15   the category of g_i is the category appears most in RL
16   add the category of g_i  to the Result Set R.
16  End For
```

Algorithm 2. Calculate graph similarity

IV. Experiment

We choose corpus collected from the open platform of Chinese natural language processing to perform the classification task in our experiment. It contains 2800 documents in 10 categories. We select 250 documents, which belong to 5 categories with 50 documents in each from the corpus as training set, and the remaining other 100 documents as testing set. The statistics information of the training set and testing set we used is shown in Table I.

TABLE I. CORPUS STATISTICS

Category	Training set		Testing set	
	Feature	*Word*	*Feature*	*Word*
computer	11214	51019	3594	17972
education	9892	41851	2917	10341
economic	10282	45835	1435	3765
transport	7136	21350	2464	9389
political	3710	12389	1276	3641

We compared the proposed approach with Traditional VSM based KNN and use the common measures to evaluate the performance of the classification methods, i.e. Precision,Recall and F1-measure:

$$F_1 = \frac{2 \times \text{Precison} \times \text{Recall}}{\text{Precison} + \text{Recall}} \tag{10}$$

When K=5 we obtain the best result when the feature term number is 2500. When K=10, the best result appears when the feature term number is 3000. The categorization result improves with a bigger K value. The detail result is as follows:：

TABLE II. THE CATEGORIZATION RESULT USING GKNN ALGORITHM

Category	*Recall(%)*		*Precision(%)*		*F1(%)*	
	K=5	*K=10*	*K=5*	*K=10*	*K=5*	*K=10*
computer	96.49	94.7	91.67	90	94.02	92.3
education	83.33	86.9	100	100	90.90	93.02
economic	83.07	82.6	90	95	86.40	88.37
transport	100	100	78.3	85	87.85	91.89
political	94.91	100	95	90	94.12	94.73
Average	91.56	89.2	90.67	88.8	90.66	92.07

TABLE III. THE COMPARISON OF GKNN AND VSM-BASED KNN

Algorithm	*F1(%)*	*COST TIME(second)*
	K=5 feature=2500	*K=5 feature=2500*
GKNN	90.06	264
VSM-BASED KNN	88.46	762

From Table III we see the result of GKNN is promising compared with VSM-Based KNN. GKNN outperforms VSM-Based-KNN both in accuracy and spend time. The result shows we consider the information such as the order and co-occurrence of words in text is reasonable.

The comparison of GKNN with different K value and VSM-Based KNN under the condition of different number of feature terms is shown in Fig. 1.From the curves, we can see the GKNN outperforms KNN. And with a greater K value , the.classification result will be better.

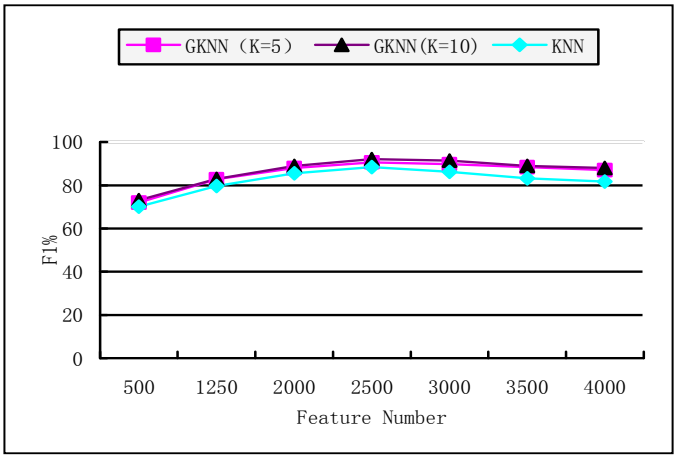

Figure 1. The classification results using KNN and GKNN

V. Conclusions

In this paper we proposed an improved graph-based text representation and applied it in the categorization of Chinese text. According to this model, we presented an graph-based KNN algorithm which reduced the amounts of computation and cost less time than VSM-Based traditional KNN. In the future we will consider more effective feature selection method and similarity calculation algorithm.

ACKNOWLEDGMENT

This research project was incorporated and performed as a part of ”Key Technology Research and Demonstration Project of E-Community” project promoted by Key Projects in the National Science & Technology Pillar Program No. 2007BAH08B02.

REFERENCES

[1] G. Salton, M. McGill, Introduction to Modern Information Retrieval, McGraw Hill, 1983.

[2] Schenker.A., Last.M., Bunke.H.,Kandel,A., “Classification of Web Documents Using a Graph Model International,” Journal of Pattern Recognition and Artificial Intelligence,2004 ,pp. 475-496.

[3] Zhou Zhaotao ,Bu Dongbo ,Cheng Xueqi ,”Towards Graph-based Text Representation”, Journal of Chinese Information Processing,vol.19, pp.36-43 , 2005.

[4] Schenker.A.,Last.M.,Bunke.H. ,”Classification of web documents using a graph model,” In Proc. of 7th International Conf. On Document Analysis and Recognition, Scotland, Computer Society Press, 2003.

[5] Bhoopesh P. “Text clustering using semantics,” The 11th International Word Wide Web Conference, (WWW 2002), Hawai, USA, 2002

[6] Jin Wei,Srihar Rohini K. “Graph-based text representation and knowledge discovery,” Proceedings of the 2007 ACM symposium on Applied computing, 2007, pp.807-811.

[7] Xiaofei Zhang,Heyan Huang,Keliang Zhang.”KNN Text Categorization Algorithm Based on Semantic Centre,”. Proceedings of the 2009 ITCS,2009,pp.249-252.

[8] Dai liuling, Huang heyan,chen zhaoxiong. “A Comparative Study on Feature Selection in Chinese Text Categorization,” Journal of Chinese Information Processing, Vol.18 No.1, 2004: 26-32