Improving support vector machine level-based for person domain categorization

Lijuan Diao
East China Normal University
Shanghai, China
Lijuan_diao@126.com

Xijie Wang
Health Management College of Binzhou Medical University
Yantai, China
bzmcwangxijie@126.com

Lei Cui
YanTai Vocational College
Information Engineering Department
Yantai,China
Ytcuilei66@126.com

Abstract— Classification technology refers to assigning of one or more suitable categories from multiple categories data sets. While previous work in classification focused on single classifier, we propose classification method of improving support vector machine level-based that can classify multiple categories. Actually, we use the weight calculation method of TFIDF and combine DAG-SVM and KNN algorithm to improve precise of classification. An experiment has been carried out to measure the performance of our proposed classification method. The results show that our method performs better for person domain data set comparing with single DAG-SVM method.

Keyword— KNN DAG-SVM KNN-DAG-SVM TFIDF

I. INTRODUCTION

With the rapid proliferation of internet technologies, a large number of the data online are increasing day in day. In order to effectively manage and use the distribution massive data, information retrieval based on document content and data mining have become a hot field. Classification techniques widely apply in the domain of information retrieval and data mining.

The main task of classification is that massive data is divided into different categories, according to the contents of the data and given the label set. There have been many classification algorithms in the recent years. For example, Rocchio algorithm [1], KNN (K Nearest Neighbor) algorithm [2], Bayes networks algorithm [3], Decision tree algorithm [4] and SVM (Support Vector Machine) algorithm [5] and so on.

Most of the traditional classification algorithms, including Bays, Linear classification, Decision tree algorithm, are simple and handle capabilities are weak. These classification algorithms are only apply to two classes and cannot be directly applied to multiple classes.

KNN algorithm has been extensively applied to multiple classes. However, it is a lazy learning method and needs large amount of processing time, because this method does not process the training data before classify to test data. And data sparse influences its precision.

Classification algorithm among multiple classes is important potential solution to achieve classification in multi-category data sets.

In [5], SVM algorithm is proposed by Cortes and Vapnik and settles nonlinear inseparable problem with the promotion of good generalization ability. In order to solving the multiple classes, SVM uses three methods which are one to one, one to many and directed acyclic graphic [6]. The method of one to many has the problem of category overlap and inseparable. However, if there is data skew, it will seriously affect the result of classification. Also, the method of one to one has solved the problem of category overlap and data skew. But there is a serious problem that the number of the classifier is increased with the increasing categories and the number of calling classifier is increased. If the number of category is k, the number of classifier is $k(k-1)/2$. For instance, assume that the number of category is 1000, the number of classifier is 500000. In this situation, time complexity of this method is large. The method of directed acyclic graphic needs to construct $k(k-1)/2$ classifiers, but the times of call classifier are not $k(k-1)/2$. Hence, time complexity of directed acyclic algorithm is decreased comparing to the method of one to one. Moreover, there is a problem that if there are false categories in the initial process of classification, that is no chance to correct this mistake in directed acyclic graphic algorithm. So classification is very important in the beginning. In order to solve the problem of directed acyclic graphic, we combine KNN and directed acyclic graphic support vector machine. Thus, we utilize the advantages of KNN algorithm to solve false categories in the initial stage of classification and the times of calling classifier.

The rest of this paper is organized as follows. Section II focuses on formal definition of categorization, providing a running example and the procedure of text categorization. Section III provides the method of weight calculation and selection feature vector. Section IV presents the method of improving directed acyclic graphic support vector machine which is combining KNN and DAG-SVM. Section V provides performance measures for DAG-SVM and KNN-

DAG-SVM. Section VI states the conclusion of the paper and the future work.

II. THE DEFINITION AND APPLICATION OF CLASSIFICATION

A. A formal definition of classification

Given data unit $d \in X$ and $C = \{c_1, c_2, ..., c_n\}$ is a set of predefined categories, where X is a domain data units set, and given training data unit set $D = < d_i, c_j >$, where $< d_i, c_j > \in X \times C$, categorization is the task of assigning a Boolean value to each pair $< d_i, c_j > \in X \times C$. A value of T assigned to $< d_i, c_j >$ indicates a decision to file d_i under c_j, while a value of F indicates a decision not to d_i file under c_j. More formally, the task is to approximate the unknown target function γ that called the classifier.

$$\gamma : X \to C . \qquad (1)$$

So, each data unit can be classified to a particular categorization, and this classify process is a supervised learning process.

B. A running example

Person domain contains numerous feature data and relationship data. These data of different formats distribute on kinds of webpage. In this paper, we take some person data as running examples. That is, we classify person domain data, according to different occupation.

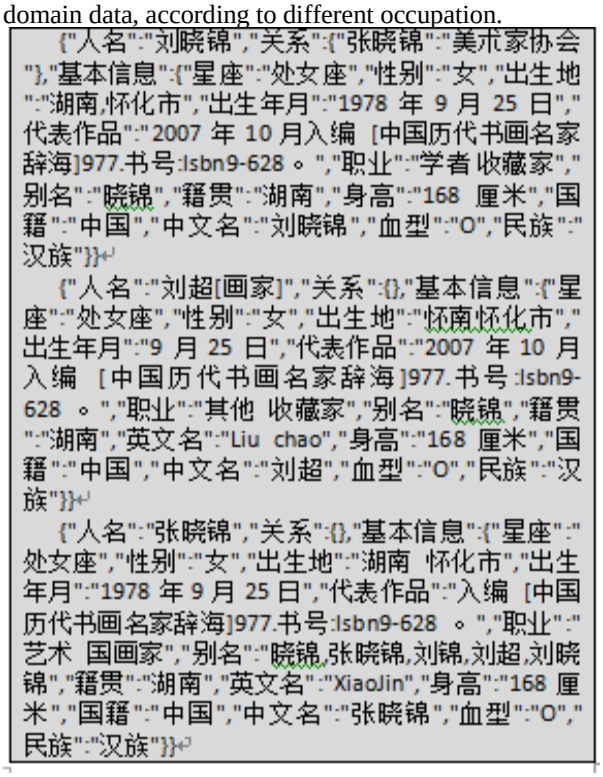

Figure 1. Parts of person domain data

Figure1shows parts of person domain data that is obtained from webpage and is formatted. Our goal is that this format of person data is divided into different classes. More specifically, we plan to give some different labels of categorizations and utilize classifier to classify for the person data set.

C. The procedure of text categorization

- The preprocess of text: The preprocess of text is also called data cleaning. First of all, word segmentation is a fundamental task in text data processing, because word is the most basic unit of semantic representation. And then, filtered to stop word, because the stop word does not contain any actual meaning, is the language of the grammar part.
- The formal text: The classifier does not directly process text form and it need to change text into the form of digit, matrix or vector in the process of categorization. It is divided into two steps in the formalization process. One step is feature selection, which need to remove the noise data. Another step is the weight calculation of the feature and each text corresponds to a feature vector space in the vector space model. The former defines the dimension of feature vectors, and the latter defines the degree of correlation between these feature and text.
- The use of different classification algorithms for text categorization: According to different classification algorithm, we can get the result of text categorization.
- Performance evaluation: The goal of text categorization is the difference between the objective function and the approximation function is minimized. And the value of empirical risk and confidence risk reach the minimum. Evaluation parameters are the correct rate and recall rate. In the case of categorization C, gives the definition of P and R.

$$P = \frac{T_C}{T_C + F_C} \qquad R = \frac{T_C}{T_C + R_C} \qquad (2)$$

Where, T_C denotes the number of that test samples are classified correctly into categorization C. R_C denotes the number of that test samples which should be classified into categorization C are not classified into categorization C. If the value of P and R are closer to 1, the result of text classification is better.

III. THE METHOD OF WEIGHT CALCULATION AND SELECTION FEATURE VECTOR

A. The method of weight calculation

The weight of word denotes the distribution of words in the documents. According to value of word weight, we get the word ability of distinguish categorization and it carries the amount of information content in the process of categorization. We use usually three calculation methods

are information gain, mutual information and $tf-idf$ respectively.

Information gain (IG) [7] is the method of word weight calculation from information theory perspective. It specifically refers to the amount of information content that is carried by feature words in the feature vector. Words which are in the training sample contain the information content which is as the word weight. We can use this for text categorization in the test sample. The specific calculation formula is as follows.

$$IG(\omega) = -\sum_{j=1}^{k} p(c_j)\log p(c_j) + p(\omega)\sum_{j=1}^{k} p(c_j \mid \omega)\log(c_j \mid \omega) + p(\varpi)\sum_{j=1}^{k} p(c_j \mid \varpi)\log(c_j \mid \varpi) \quad (3)$$

Where, $\{c_1, c_2, ..., c_k\}$ denotes a set of categorizations of training sample. $p(c_j)$ is the probability that text belong to c_j in the training sample. $p(\omega)$ is the probability that word ω appear in the training sample. $p(\varpi)$ is the probability that word ω does not appear in the training sample. $p(c_j \mid \omega)$ denotes the probability that word at least once appear in the training sample and belongs to the categorization c_j, while $p(c_j \mid \varpi)$ denotes the probability that word does not appear in the training sample and belongs to the categorization c_j. The advantage of information gain is that the terms do not appear in the categorization impact on classification of text, but it also is the drawback of information gain, because the cost of statistical calculation is too large.

Mutual Information (MI) [7] considers the information content of the low frequency words. Sometimes, the information content of the low frequency is more than the high frequency words. But it focus on low frequency words too much, the effect of this classification is not good in actual application.

The two methods are only applicable to single level classification that is also known as the calculation method of local weights. If texts may be classified into two categorizations, we can use the four methods. tf-idf is the calculation method of global weights and its value is computed in the whole corpus, rather than in the a specific document or categorization. So the most common method is tf-idf.

The main idea of model tf-idf is the number of emergence of word is large in the specific document, but the frequency of appearance of this word is very low in the other documents. So it can be concluded that the degree of distinction of this word is very high in the whole corpus, in other words, this word can distinguish a class of document and the other documents. It mainly contains two aspects of factors. One is the frequency of appearance of this word in the specific document that reflects the importance of this word in this document. Another is the frequency of the inverse document of this word in the corpus that reflects the importance of the word in the whole corpus. The specific calculation formula is as follows.

$$tf-idf_{t,d} = tf_{t,d} \times idf_t = tf_{t,d} \times \log\frac{N}{df_t} \quad (4)$$

Where, $tf_{t,d}$ is the appearance frequency of word t in the document d. df_t is the number of documents where the word appearance in. N is the total number of documents in the whole corpus[8].

B. The feature vector selection based on tf-idf

According to the above calculation weight method analysis, we use the method of tf-idf. Now consider classification algorithm for the example person data given in section II. We need to compute the value of tf-idf for chosen person training data. Parts of the results are given in table I.

TABLE I. THE PARTS OF VALUE OF TF-IDF

Words / TFIDF / Categories	Broker	Team mate	Enemy	Tutor	Comrade-in-arms	...
Military	0	0	0.78	0	12.41	
Education	0.31	0	0	0.60	0	
Acting	2.41	0	0	0.30	0	
Sport	0.90	5.447	0	0	0	
...						

We sort the feature term by the value tf-idf and choose the top of the sorted list feature words in each categorization. We use the chosen feature words to construct feature vector. Take sport categorization vector for example as shown.

[Coach(7.826),Teammate(5.447),Partner(3.112),Player(1.556),Antagonist(1.558),Assistant(0.954),Broker(0.903),Idol(0.791).....]

IV. IMPROVING SUPPORT VECTOR MACHINES

A. KNN and SVM

KNN (K-Nearest Neighbor) algorithm use local information to determine the categorization boundary. Before classify for the test sample, this algorithm does not do anything. That is to say, this method is "lazy learning". When an unknown text vector arrives, KNN algorithm computes the similarity between text vector and text vectors respectively in the training sample, according to the Euclidean distance or cosine value of angle of two vectors. The result of this process is to get K maximum similarity value of documents in training sample. And then, by comparing the number of K documents respectively in the training sample, the algorithm decides to assign test sample into the largest number of documents corresponding to categorization.

Through the above analysis, we can derive this algorithm needs much time to compute and data sparse affects its accuracy. But the importance is that KNN algorithm is suitable for multiple categorizations.

B. SVM (Support Vector Machine)algorithm for induction

Support Vector Machine has strong theoretical foundations and excellent empirical successes. Firstly, we consider SVM in the binary classification setting. We are given training data that are vectors in some space and their labels

$$(x_i, y_i), i = 1, 2, ..., n, x \in R^d, y_i \in \{+1, -1\}, i = 1, 2, ..., l$$

where, x_i is the vector and y_i is its label. In the simplest form, SVM is the hyperplane that separate the training data by a maximal margin. All vectors lying on one side of the hyperplane are labeled as -1, and all vectors lying on the other side are labeled as 1. The training instances that lie closest to the hyperplane are called support vectors. As given in Figure 2. Generally speaking, SVM allow one to project the original training data in space X to a higher dimensional feature space.

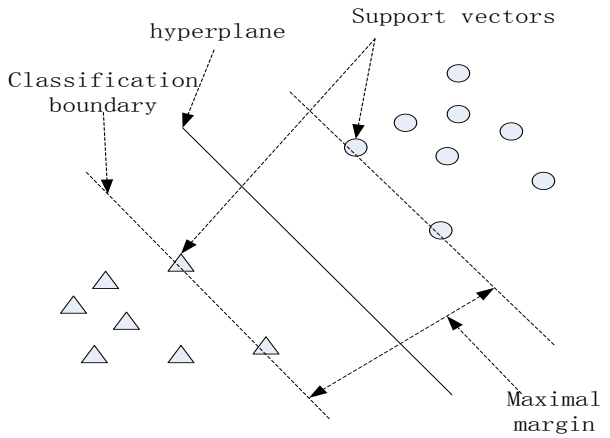

Figure 2. A simple linear support vecotor machine.

In fact, the question of maximal margin is Wolf's dual problem that we can use of Lagrange's function to resolve. And the decision function is as follows.

$$f(x) = \mathrm{sgn}(\sum_{i=1}^{n} y_i \alpha_i^* (x \cdot x_i) + b^*) \quad (5)$$

Here, x_i denotes support vector; α_i^* represents Lagrange coefficient of corresponding with x_i. Therefore, according to decision function $f(x)$, we can get which categories should test sample x belong to.

In the situation of dealing with linear inseparable, kernel function concept is proposed by Vapnik. Actually, input vector is projected to higher dimensional feature space by nonlinear mapping approach. Assume that nonlinear mapping represents $\varphi : R^n \to H$ and kernel function $K(x_i, x_j)$ that meets

$$K(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j) \quad (6)$$

There are some classic kernel functions, for example, polynomial kernel function, the linear kernel function, sigmoid kernel function and radial basis kernel function.

Linear kernel function $K(x, x_i) = (x, x_i)$ (7)

Polynomial kernel function

$K(x, x_i) = ((x, x_i) + 1)^n$, n is parameter (8)

Radial basis kernel function

$$K(x, x_i) = \exp(-\gamma \| x - x_i \|^2) \quad (9)$$

Sigmoid kernel function $K(x, x_i) = S(\upsilon(x, x_i) + c)$ (10)

In the above presented SVM classification algorithm is used to solve the problem of two classifications, and it cannot be directly applied to multiple classes. Therefore, SVM algorithm will be used to solve the multiple classifications problem is an important research goal. We use directed acyclic graph SVM (DAG-SVM) method to solve multiple classifications. In the period of training data, we can construct classifier of two categorizations. If there are k classes, the number of classifier is $k(k-1)/2$. But it is important that it does not call $k(k-1)/2$ classifiers. Before data being classified, we need to organize all classifiers according to a direct acyclic way. For example, if there are five classes, the number of called classifiers is $(k-1)$. The organization structure is given in figure 3.

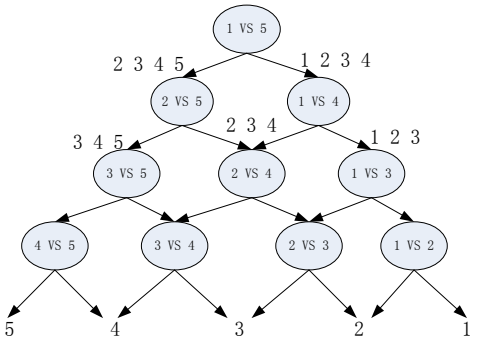

Figure 3 The organization structure of five categories

According to the above classification method, the speed of classifying can be improved and there is no overlap and indivisible phenomenon. But it is more important that if there is error categories in the initial process of classification, that is no chance to correct this mistake. So classification is very important in the beginning.

C. Construction of DAG-SVM organization structure

Based on the above issue, we combine DAG-SVM with KNN algorithm to solve it. The experiments show that this method has better performance. The specific contents are as follows.

First, we use TFIDF to calculate weight of each term and extract feature vector for every categorization.

Second, KNN algorithm views each feature vector of categorizations as a centroid, and calculates distance and constructs classifier between centroids. To ensure the upper class is easy to distinguish, we choose classifier the maximum distance of two classes as a top-level classifier of DAG-SVM.

Finally, according to this method, we can recursively construct the organization structure of DAG-SVM.

On the basis of this method, the process of constructing the direct acyclic graph structure of person domain categorization can be described as follows in Figure 4.

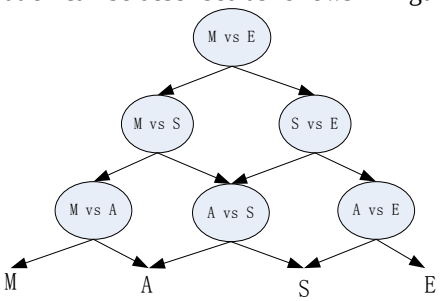

Figure 4 The organization structure of directed acyclic graph.

Here, M is military categorization. And E is education categorization. S and A are sport and acting, respectively.

V. EXPERIMENTS WITH DAG-SVM AND KNN-DAG-SVM

DAG-SVM classifiers have been shown to be fast and effective in text classification [9, 10]. The purpose of this experiment is to explore the use of combining DAG-SVM and KNN classifiers in classifying data. In the experiments, we compared the classification performance indicated by both the standard precision and recall. In our experiment, person domain data collection is used. To conduct our experiment, training data set needs to be manually derived from the person domain data.

The results of our experiment are shown in Table II for the four categories. We compute the standard precision Pr and recall Re of DAG-SVM and KNN-DAG-SVM.

TABLE II. TESTING RESULTS FOR DAG-SVM AND KNN-DAG-SVM

Category	Pr-DS	Re-DS	Pr-DSK	Re-DSK
Military	0.920	0.689	0.971	0.409
Education	0.963	0.597	0.982	0.780
Acting	0.945	0.831	0.964	0.833
Sport	0.938	0.778	0.953	0.760

In summary, our method performed reasonably well for the person domain data set when given enough training data.

VI. CONCLUSIONS AND FUTURE

In this paper, we introduce a method of automate classification among multiple classes. The approach in this paper focuses on using calculation the value of tfidf to choose feature words and construct feature vector. Finally, we present the novel method of classification by combining DAG-SVM and KNN to improve the accuracy and processing speed. We plan to investigate the method of taking advantage of contributions of wrongly classified data sets so that obtain accurate classifier.

REFERENCES

[1] Hull David. Improving text retrieval for the routing problem using latent semantic indexing. The 17th ACM International Conference on Research and Development in Information Retrieval, 1994, pp.282-289.

[2] Lam W, Lai KY. Automatic textual document categorization based on generalized instance sets and a metamodel. IEEE Trans. On Pattern Analysis and Machine Intelligence, 2003, 25(5), pp.628−633.

[3] Tan S, Cheng X, Ghanem MM, Wang B, Xu H. A novel refinement approach for text categorization. In: Otthein H, Hans JS, Norbert F, Abdur C, Wilfried T, eds. Proc. of the 14th ACM Conf. on Information and Knowledge Management (CIKM-05). Bremen: ACM Press, 2005, pp469−476.

[4] Li F, Yang Y. A loss function analysis for classification methods in text categorization. In: Fawcett T, Mishra N, eds. Proc. of the ICML 2003. Washington: AAAI Press, 2003, pp.472−479.

[5] Cortes C, Vapnik V. Support vector networks. International Journal of Machine Learning, 1995, 20, pp.273-297.

[6] Dietterieh T, Bakiri G. Solving Multiclass learning Problem via Error-Correcting Output Codes. Journal of Artificial Intelligence Research, 1995, 2, pp.263-286.

[7] Yang Sheng, Shen Peng-fei. Bidirectional Automated Branch and Bound Algorithm for Feature Selection .Journal of Shanghai University (English Edition), 2005, 9 (3), pp.244-248.

[8] S. Robertson. Understanding inverse document frequency: on theoretical arguments for IDF. Journal of Documentation, 2004, pp.60: 503 – 520.

[9] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization. In Proc. of the 7th Int. Conf. on Information and Knowledge Management, 1998, pp. 148-155.

[10] T. Joachims. Text categorization with support vector machines: learning with many relevant features. In Proc. of the 10th European Conf. on Machine Learning, 1998, pp.137-142.