A hybrid multi-class text categorization based on SVM-DT

Ying Fang[1, 2*], Heyan Huang[1]

[1]*College of Computer, Beijing Institute of Technology, Beijing, China, 100081*

[2]*School of computer & technology, ShangQiu Normal College, ShangQiu HeNan, China, 476000*

Received 6 October 2014, www.cmnt.lv

Abstract

How to improve the text categorization efficiency as well as keeping high speed is a research problem. Several factors are effected the processing of the decision tree construction, such as, the degree, the balancing degree, the constructing way, the group number and the division degree between groups etc. Considered the various roles between the above factors, a comprehensive algorithm to construct the SVM-DT (Support Vector Machine - Decision Tree) is proposed. In this method, three conditions are considered respectively. The text categorization experiments on massive corpus demonstrate that the algorithm can improve the efficiency in some degree and decrease the training and testing time largely at the same time. The algorithm to construct the SVM-DT is feasible and adaptable.

Keywords: Text Categorization; Support Vector Machine; Decision Tree; Multi-class category; Corpus; Positive Sequence Tree

1 Introduction

By constructing the best classified hyperplane, the Support Vector Machine (SVM, [1]) can rightly classify the samples, which is proved to be one of the most powerful text categorization methods. However, the SVM was originally developed for binary decision problems. If SVM is used to deal with multi-class classifier with the massive data, the calculating overhead is too huge. The popular methods for applying SVMs to multi-class classification problems usually decompose the multi-class problems into several two–class problems that can be addressed directly using several SVMs. The typical methods are the OAO (One-Against-One) method [2], OAA (One-Against-All) method [3], DAG-SVM (Directed Acyclic Graph-Support Vector Machine) method [4] and SVM-DT (Support Vector Machine-Decision Tree) method etc. Among them, DAG-SVM can solve the problem of "Rejecting Recognition" existing in OAO and OAA, but its generalization ability is limited. SVM-DT takes advantage of both the efficient computation of the decision tree architecture and the high classification accuracy of SVM, which is suitable to large categories [5].

There are many researches to constructing SVM-DT. Based on the information gain of the non-leaf node, Ramaswamy [6] proposed a method to construct a tree called Partial Sequence Tree; in this structure each SVM classifier separates one class from the remainders. The advantage of the method is that the classes at the top level of the tree had very high accuracy (more than 95% sometimes), but the ones at the lower levels (especially the last leaf node) had poor accuracy.

Madjarov etc. [7] applied ensemble learning techniques to constructed a tree structure called Positive Sequence Tree, in which each SVM classifier divides the classes into two groups. By contrast, the classifier with positive sequence tree was faster than the one with the partial sequence tree; and the accuracy between the classifiers on the different levels was relatively stable.

Takahashi and Abe [8] proposed four types of decision trees including the above two trees. But the reparability measure of the Euclidean distance or the Mahalanobis distance could not reveal the features of the texts because it only used the number of the words. And it did not explain which type was better under certain conditions. From these papers we can find that the type of the tree is an important effect to the category.

A balanced decision tree which included the tradeoffs between the sample number and the difficulties to divide was employed in [9]. Although it could decease training time in large degree while did not reduce the recognition rate, and it exaggerated the category effect of the "super class".

According to the massive text sets, in this paper we aim to find the effecting factors of SVM-DT construction and how to integrate many strategies while balancing the category precision and the working speed. The remaining parts of this paper are organized as follows. In Section 2, we introduce the strategies to construct the SVM-DT. Section 3 presents our algorithm and text categorization system based on the Section 2. Section 4 describes our experiments respectively done on English and Chinese corpus. Conclusions and further discussions are given in Section5.

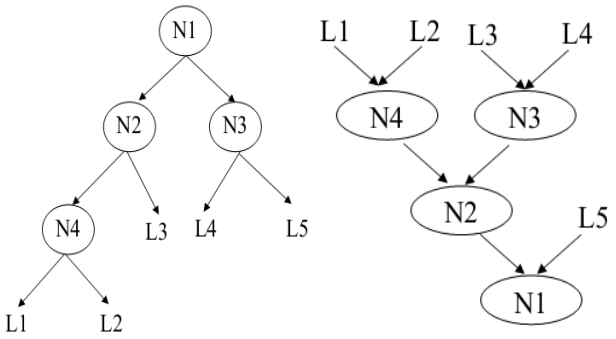
[*]***Corresponding author*** e-mail: yfang@bit.edu.cn

2 Constructing strategies of SVM-DT

In this section, we describe the constructing strategies of the SVM-DT, that is, the depth, the balance degree, the constructing way, the group number and the division degree between groups. These are the most important problems to consider when constructing the SVM-DT.

2.1 SVM-DT CONSTRUCTING MODE

According the constructing direction, the SVM-DT constructing way can be divided into: top-bottom division method and bottom-top accumulation method [10]. The former(Shown as FIGURE 1(a)) begins from the root and grows down to the leaves, which finds two classes which have the minimum distance and then divides the nodes. The similar dividing process will be done until each node points to one class. The latter(Shown as FIGURE 1(b)) first finds two nearest classes from all the samples and combines them into one new group, then recursively continues the finding and combining process until the last two groups are combined into one group.



(a) top-bottom mode (b) bottom-up mode

FIGURE 1 Constructing diagrams of SVM-DT

Both the division method and the accumulating method have $O(n^2)$ (n is the group number of the current notes) time complexity when finding two nearest groups. The calculating consuming is too large for multi-class category when the number of the texts is very big. So, one improving direction is how to reduce the unnecessary calculations.

2.2 EFFECT OF THE SAMPLES SCALE

The scale of the training and the testing sets tends to vary largely. To Reuters corpus, for example, the largest group is composed of 2,877 training documents, at the same time, there are 75 groups of the training documents are less than 10 articles. So when constructing a decision tree it is not appropriate to treat each group equally. Another example is on news sites, the category about the "business and economy" or "entertainment" will contain more pages than that about the "health and medicine" or "education" categories. Therefore, if the larger sets can be broken down earlier, the pressure to the latter work will be greatly reduced.

2.3 CONSTRUCTING STYLE OF DECISION TREE

We have known there are two kinds of constructing way for a decision tree [11]: (1) the Partial Sequence Tree, the two sub nodes of a non-leaf node describe the relation of one-vs.-many (1: n). (2) the Positive Sequence Tree, the relation of the child nodes of a parent node is many-vs.-many (m:n). It has been proved that during the categorization process the partial sequence tree having the bigger depth is slower than the positive sequence tree having smaller depth and higher paralleling degree. So when construct the tree we will try to build a positive sequence tree.

2.4 SIMILARITY BETWEEN THE CLASSES

If the classification performance is not good at the upper node of the decision tree, the overall classification performance becomes worse. So we should put the classes with bigger differences or higher classifying accuracy on the upper nodes of the decision tree, in order that the lower ones may be less affected. To realize that, we can calculate the similarity, then firstly place the groups with less similarity on the upper level.

Based on the above four problems, we present a hybrid strategies to construct SVM-DT: (1) the up-down dividing method is perfect; (2) large set in the samples should be divided out firstly to reduce the calculating overload; (3) the number of sets is tried to be balanced as far as possible to reduce the depth of the decision tree; (4) the dividing ability of the upper nodes should be made stronger as far as possible to reduce the accumulating errors.

3 A hybrid constructing algorithm of SVM-DT and text categorization system

In this section, we will introduce a kind of hybrid constructing algorithm of SVM-DT. The text categorization system based on the corresponding SVM-DT is to be provided then.

The symbols used in the following part are shown in Table 1.

TABLE 1 Notations used in the paper

Sym.	Description
S	the samples
K	the number of the set, $K \in [1,\infty)$
S_i	one sample set, $i \in [1,K]$
$Num(x)$	the number of the texts in set x
N_{dg}	the node on the dg ($dg \in [0,\infty)$) level of a decision tree
∂	the scale effecting factor, $\partial \in (0,1)$
β	the similarity effecting factor, $\beta \in (0,1)$s
SB	the buffer of the total processing set
S_A , S_B	the group centred as A or B
$Sim(A,B)$	the similarity of set A and set B
$IG(N)$	the Information Gain value of node N

3.1 THE HYBRID CONSTRUCTING ALGORITHM OF SVM-DT

In our algorithm, three conditions are considered when building the decision tree:

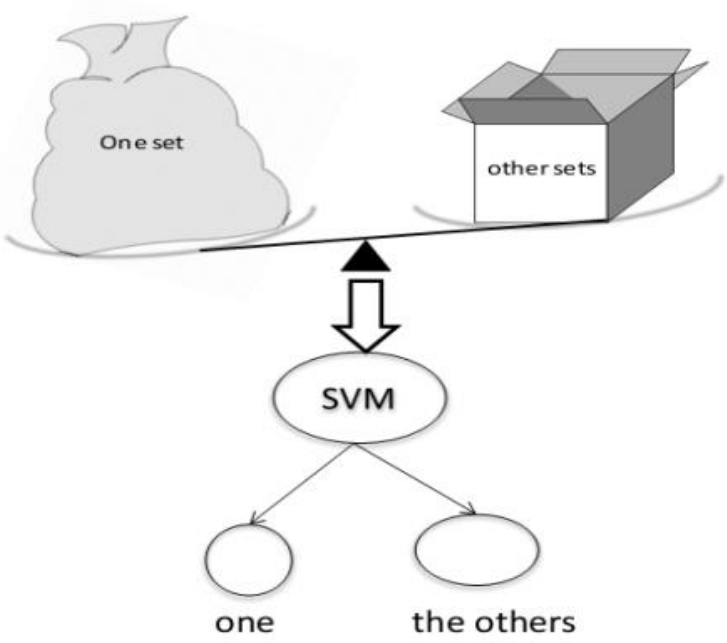


FIGURE 1 The description and the measure of condition (1)

If there is a very large class in the sample sets, we can recognize it only by the number of texts in the set. When the number of texts reaches a certain percentage (∂ here) of the total texts, the set will be put on the upper level first. The parameter ∂ is a value according to the sampling set, usually is set as 1/2 or 1/3 and so on. In our experiment we set it 1/3.

(2) The second condition is that one branch is not too far to divide from the other, which is described as Figure 3.

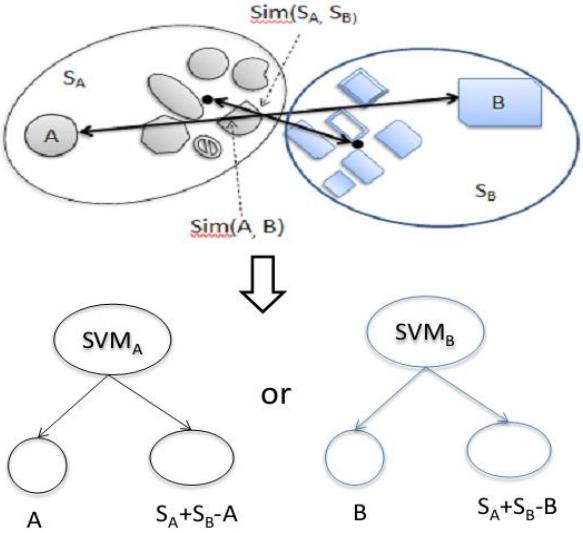


FIGURE 3 The description and the measure of condition (2)

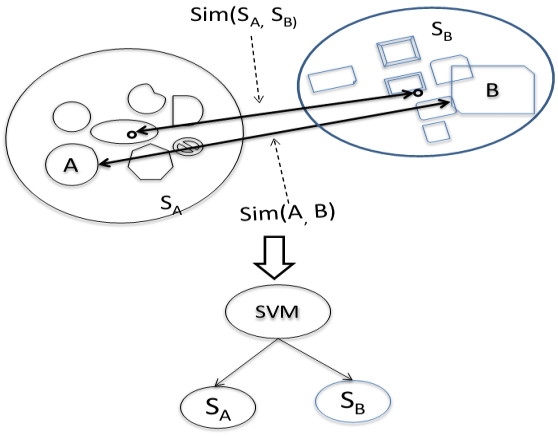


FIGURE 2 The description and the measure of condition (3)

If the gap between the similarity of two groups (S_A, S_B) and the similarity of the two farthest set (A, B) reaches a certain degree, we think A or B is not close enough to the centre of the group. The measure we take is to separate one of them out according to the Information Gain value of the parent node.

(3) The third condition is that two groups are far away enough, shown as FIGURE 4.

This condition should be the most common in the samples. The centre of the group (S_A , S_B) is close to the

dividing node (A , B). So we think two groups can be divided apart. Then a tree structure can be built on S_A and S_B.

The hybrid constructing algorithm for SVM-DT is shown on Algorithm 1.

Algorithm 1 The hybrid constructing algorithm for SVM-DT

Input: training set $S=\{S_1,S_2,...,S_K\}$

Output: a decision tree structure with the corresponding SVM classifiers

// initialization.

$SB=S$; $M=\sum_{i=1}^{K} Num(S_i)$; $dg=0$;

construct a root node N_{dg}.

Step1. let P is the biggest set of SB;

if $Num(P)/M>\partial$ then

{// process the super class P.

construct the sub tree of N_{dg} with the branches of P and $SB-P$;

build a SVM classifier between P and $SB-P$;

$dg++$; $M=M-Num(P)$; $SB=SB-P$;

turn to Step1; // make the sub tree of SB until there is only one set in SB;

}

else {turn to Step2;}.

Step2. //to divide the set into to groups

for each set pair C and D ($C,D\in SB, C\neq D$)

{calculate the similarity $Sim(C,D)$ using formula (3);}

let A and B are the sets which have the lest similarity among those pairs;

for each set P ($P\in SB-A-B$) {

if $Sim(P,A)>Sim(P,B)$

{put P into the set S_A;}

else {put P into the set S_B;} }

if $Sim(S_A,S_B)/Sim(A,B)\leq\beta$

{//to process the class with higher dividing degree

calculate $IG(N_{dg})_A$ with the branches A and $SA+SB-A$;

calculate $IG(N_{dg})_B$ with the branches B and $SA+SB-B$;

if $IG(N_{dg})_A<IG(N_{dg})_B$

{construct the sub tree of N_{dg} with the branches A and $SB-A$;

build a SVM classifier between A and $SB-A$;

$dg++$; $SB=SB-A$; $M=M-L_A$; $N_{dg}=N_{SB-A}$;}

else

{construct the sub tree of N_{dg} with the branches B and $SB-B$;

build a SVM classifier between B and $SB-B$;

$dg++$; $SB=SB-B$; $M=M-L_B$; $N_{dg}=N_{SB-B}$;}

}

else {turn to Step3;}

Step3. //build sub tree of two sets

let S_A and S_B as two branches of the node N_{dg}, construct the tree structure;

build a SVM classifier between S_A and S_B;

$SB=S_A$; turn to Step1;

//make the sub tree of SB until it can not be divided then.

$SB=S_B$; turn to Step1;

//make the sub tree of SB until it can not be divided then;

There are two main factors indicating the algorithm: the huge amount and the distributing character of the texts. So the hybrid algorithm gives an improvement on two sides: (1) the super class can be divided out at the early stage, which aims to speed the construction of the decision tree and reduce much similarity calculation; (2) the balance of two sub trees is taken into consideration, which aims to improve the degree of parallelism and reduce the attraction of the large groups in some degree.

3.2 TEXT CATEGORIZING SYSTEM BASED ON SVM-DT

The text categorizing system (FIGURE 5) is composed of two parts, training part and classifying part, which is connected with the core classifier.

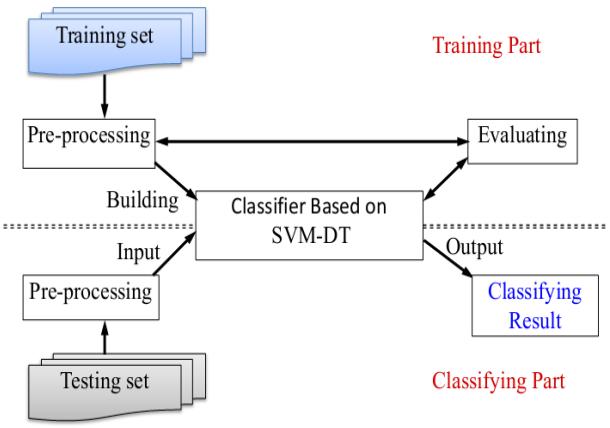


FIGURE 3 Function chart of the text categorizing system based on SVM-DT

Both the training texts and the testing texts should be pre-processed first, which includes the stemming for English texts, word segmentation and tagging for Chinese

texts, stop words removing, feature (words are the features in this paper) selection etc.

The selected feature is weighted by TF-IDF method. $W_{t,d}$ is the weight of word t in text d:

$$W_{t,d} = \frac{TF_{t,d} \times \ln(N / DF_t)}{\sqrt{\sum_{t=1}^{m} [TF_{t,d} \times \ln(N / DF_t)]^2}}. \quad (1)$$

With, $TF_{t,d}$ is the number of t appearing in d. DF_t is the number of the texts including t. N is the total number of the texts.

The text d can be described as a vector V composed of m feature and its weight:

$$V_d = \{(T_1, W_{1,d})(T_2, W_{2,d})...(T_m, W_{m,d})\}. \quad (2)$$

Then, the similarity between vector V_A and vector V_B is calculated as:

$$\mathrm{Sim}(V_A, V_B) = \frac{\sum_{t=1}^{m} (W_{t,A} \times W_{t,B})}{\sqrt{\sum W_{i,A}^2} \sqrt{\sum W_{j,B}^2}}. \quad (1)$$

with, m is the feature both in V_A and V_B.

The aim of the training part is building the classifier based on SVM-DT according to the algorithm 1. Because the classifying result is affected by the feature selection, so we should select the features repeatedly according to the evaluating result.

In training part, each text inputted into the SVM-DT classifier will be divided into one branch of the root node on the basis of the similarities. If the corresponding node is not the leaf node, then we will calculate the similarity and allocate the text to a node recursively until it point to a leaf node. Then the category of the last node is the one that the text to be classified into.

The similarity of two sets can be calculated between the category centres or according to the maximum /minimum distance, the reparability measure matrix. However, the complexity to calculate the reparability measure matrix of two sets is relatively high. Or, much valuable information will be cast away when calculating the similarity with the maximum /minimum distance. So we calculate the reparability between two sets according to the centre vectors similarity of two categories. Let the category L_i contains K texts, C_i is the centre vector: $C_i = \frac{1}{K}\sum_{k=1}^{K} V_k$. Sim_{ij} is the similarity between two categories L_i and L_j, $Sim_{ij} = \left\| C_i - C_j \right\|_2$.

4 Experiment and analysis

In this paper, our experiment system is built on the LIBSVM [12], which is an open source software. The kernel function is RBF. The running environment is 2.66GHz CPU clock speed, 2.0GB memory. We select two types of texts to do our experiments.

English data we used here is from the Reuters21578, which is an open corpus accepted in the field of text category. The total number of the classes is 80. To ensure there is at least one text both in training set and in the testing set, we select 8230 training test and 3200 testing texts. The words in the TITLE and BODY are composed of the texts. After the processes of stemming and stop word removal we get about 20,000 feature words.

To test the application ability of the text categorization system based on the SVM-DT, we collected about 10,000 texts from several famous Chinese news websites such as ifeng, netease, sohu, chnqiang et. al.. The time span of the dataset is a week. There are 36 topics classified by manual works.

4.1 METHOD COMPARISON

To compare with other method, we test three kinds of classifiers.

-- M1: method in Ref. [6] to construct a partial sequence tree.

-- M2: method of Ref. [7] to construct a positive sequence tree.

-- h-M: our method to construct a hybrid type of tree.

Ref.[13] used 8237 texts for testing purpose and 3186 texts to the SVM-DT, which was similar with our texts. The data shown in Ref [13] is our baseline. The precision is used to measure the classified result, calculated as formula (5).

The result of the experiment is shown in Table 2.

TABLE 2 Categorical result of the English texts

Methods	Training Time(s)	Testing Time(s)	Precision(%)
Ref[13]	67	27	94.3
M1	71	36	89.8
M2	62	19	93.7
h-M	56	23	95.9

Comparing with M1 and M2, we could infer that it is quicker to build a positive sequence tree (M2) than a partial one (M1), and the testing time of the former tree is less either. Our hybrid structure (h-M) is the fastest during training period mainly because of the early division of the super class, which avoid a series of the similarity calculation. What's more, testing based on M2 is faster than on h-M, which is reflected by the depth of the tree. Most important, the precision of our method is the best in all the methods. Comprehensively view, our implementation can perform well with relatively quick speed.

4.2 APPLICATIONS

To test the applying ability of the text categorization system based on SVM-DT, we make experiment on the self-built dataset then. The testing texts are randomly held out 1/3 of the total texts and the remaining texts are used to train the system. After 10-fold cross validation, we get the evaluating result. The evaluation index we used here is the standard Precision (P), Recall(R) and F1-measure (F1).

The result based on M1, M2, and h-M is shown on Figure 6.

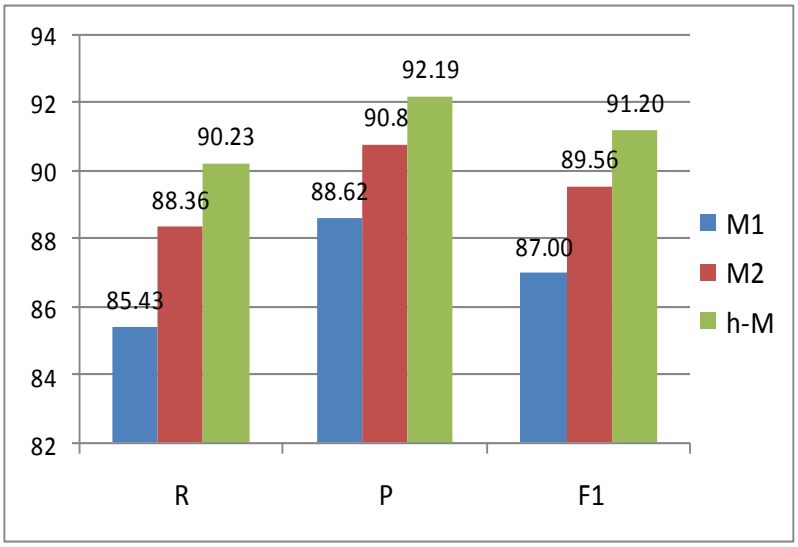


FIGURE 4 Text Categorization result of Chinese texts

On Figure 6, the highest columns are the best results. It is clear that h-M method have got the best values in all the evaluation data. So, we are sure that the text categorization system based on h-M method can distinguish the topics perfectly. Although the time consuming is not the least, the overall performance can meet the demands of the practical application.

5 Conclusion and future work

To SVM, there are two difficulties. One is the lower training speed; the other is the lower precision to deal with the big data. According to the problem, we do improvement on two sides. (1) We notice a hypothesis that the super sampling classes have big differentiation. So, when constructing the tree, we give priority to divide the super class, which can shorten the training time. (2) We make choice to build the branches depending on the disparity of the centre distance between two classes. Our experiment showed that the improvement is effective to reduce the training time, to compress the depth of the tree, to improve the precision comprehensively. Our attempt to balance the classification speed and the efficiency is efficient, which can achieve more feasible method.

Due to the lack of the compared corpus, we did not do the experiment on an open shared Chinese dataset. But our experiment has shown that the hybrid method is not confined to a kind of language. In the future, we can extend our method to the multilingual data. The ability to process the mass data needs to be verified too.

Acknowledgements

This work was supported by National Program on Key Basic Research Project (or 973 Program, No. 2013CB29606), Natural Science Foundation of China (No. 61202244), research funded project of Henan Province and ShangQiu Normal College (No. 2010A520031, No.2013GGJS013).

References

[1] Cortes C, Vapnik V 1995 Support-vector Network *Machine Learning* **20**(3) 273-97
[2] Weston J, Watkinns C 1998 *Multi-class support vector machines* Royal Holloway University of London
[3] KreBerl U PairWise 1999 Classification and support vector machines *Advances in Kernel Methods Support Vector Learning* 255-68
[4] *Deleted by CMNT Editor*
[5] *Deleted by CMNT Editor*
[6] *Deleted by CMNT Editor*
[7] Madjarov G, Gjorgjevikj D, Delev T. 2010 Ensembles of Binary SVM Decision Trees *ICT Innovations Web proceeding* 181-7

[8] Takahashi F, Abe S. 2002 Decision-tree-based multiclass support vector machines ICONIP'02 *Proceedings of the 9th International Conference on IEEE* **3** 1418-22
[9] Diao Z H, Zhao C J, Guo X Y, et al 2011 A new SVM multi-class classification algorithm based on balance decision tree *Control and Decision* **26**(1) 149-52, 156
[10] Qiao Z W, Sun W X. 2009 A multi-class classifier based on SVM decision tree *Computer applications and software* **26**(11) 227-30
[11] Zhu Y P, Dai R W 2005Text classifier based on SVM decision tree *Pattern recognition and artificial intelligence* **18**(4) 412-6
[12] Chang C C, Lin C J 2011 LIBSVB:A Library for Support Vector Machines http://www.csie.ntu.edu.tw/~cjlin/libsvm Mar.13
[13] Zhao T J 2010 Text classifier based on an improved SVM decision tree *Journal of intelligence* **29**(8) 141-3

Authors

Ying Fang, born in January, 1977, Haidian District, BeiJing, P.R. China

Current position, grades: Lecturer of ShangQiu Normal College and Ph.D student of Computer School, Beijing Institute of Technology, China.
University studies: She received her B.Sc in Computer Application from ZhengZhou technology College and M.Sc. in Computer Software and Theory from ShanXi University in China.
Scientific interest: Her research interest fields include Machine Learning, Natural Language Processing
Publications: more than 20 papers
Experience: She has teaching experience of 14 years, has completed three scientific research projects

Heyan Huang, born in October, 1963, Haidian District , BeiJing, P.R. China

Current position, grades: Professor and doctoral tutor of Computer School, Beijing Institute of Technology, China.
University studies: D.Sc from Chinese Academy of Sciences.
Scientific interest: Her research interest fields include Machine Translation, Natural Language Processing
Publications: more than 80 papers published in various journals.
Experience: She has teaching experience of 25 years, has completed more than 40 scientific research projects