Research on Fuzzy Association Classification Algorithm for Large Transaction Database Based on SVM

Wang Wen-qi
Department of Early Warning Intelligence, Air Force Early Warning Academy
Air Force Early Warning Academy
Wuhan, China
wwq505@163.com

Li Qiang
Department of Early Warning Intelligence, Air Force Early Warning Academy
Air Force Early Warning Academy
Wuhan, China
ldxy-cj@163.com

***Abstract*—Aiming at the defects of inefficiency and hard classification boundary in large transaction database classification, A. fuzzy associative classification algorithm based on SVM was proposed, SVM input eigenvector was constructed by weighed index and compatibility measure of fuzzy associative classification role, the effect of quantitative attribute discretization on association classifier was effectively reduced. With reference to decision tree classification algorithm, linear kernel function was used to make the speed of classification because of classification of the test samples are not complete decision tree traversal, and adjustment of parameters when used the nonlinear kernel function was avoided. Experimental results verify the feasibility and effectiveness of the algorithm.**

Keywords- classification; decision tree;SVM; eigenvector;

I. INTRODUCTION

Associative classification algorithm combines association rules mining with classification, uses association rule algorithm to generate frequent item sets, and then uses these frequent item sets to construct the classifier, meanwhile, considers all the properties. Its classification results are better than the traditional methods, such as the classification algorithms of decision tree. Compared with the traditional classification algorithms, associative classification algorithm has higher classification precision and strong adaptability, but many problems still exist.

Execution efficiency of algorithms. For example, the CBA[1] algorithm uses the traditional Apriori algorithm to mine classification association rules, in which I/O will be operated frequently and there will be large numbers of frequent item set combinations generated for large item sets, thus it consumes a large amount of running time; CMAR[2] algorithm uses multiple association rules to classify on the basis of CR-tree data structure; some new association classification algorithms appear in succession, such as Lazy classification association rules pruning method [3], predictive classification algorithm CPAR[4], HARMONY[5], RMR[6] and so on, which improve the algorithm efficiency, but in the process of execution it consumes too large memory, and the compression rate is limited.

To deal with continuous attributes. Most of associative classification algorithms use the attribute discrete methods such as the equal frequency, the equal space and so on while dealing with continuous attributes, which often lead to too hard boundary. In order to solve these problems some scholars use fuzzy association rules for associative classification, for example, the SFPBM algorithm [7] uses fuzzy lattice to obtain fuzzy classification association rules (FCAR) and establish a classifier, which can effectively improve the accuracy of classification. But most of fuzzy associative classification algorithms usually take the fixed membership function to divide attributes, which don't take the data distribution features of quantitative attributes into full consideration and will also reduce the quality of FCAR.

Faced with the above problems, a fuzzy associative classification algorithm based on SVM (support vector machine) is put forward. The weighted index w (Ri) of FCAR as well as the compatibility metric uipx of the training mode in original data set and FCAR is brought into and used to construct feature vectors as the input of SVM. This feature vector can express the compatibility of FCAR and quantitative attribute values, can reduce the effect of quantitative attribute discretization on associative classifiers, and can provide more discriminative knowledge for the training process of SVM. In order to improve the adaptive capacity of SVM algorithm for large-scale data and accelerate its classification speed, a new SVM classification method DTSVM (Decision Tree) is given. The method uses the idea of multiclass SVM algorithm based on decision tree for reference to decompose the original problem into several linear sub-problems, and each sub-problem corresponds to a node on the decision tree, thus a separating hyperplane is got. By modifying the objective functions and constraints of SVM, the hyperplane can classify the hard class completely and divide non hard class in the max classification interval. The DTSVM algorithm uses a linear kernel function in the process of classification, and often don't need to traverse the whole decision tree to classify the test samples; meanwhile, the method avoids the problem of adjusting parameters while using the soft boundary SVM of the nonlinear kernel function, and therefore the classification speed is accelerated greatly. The experiments show that the DTSVM algorithm classification is faster than the classification of traditional SVM algorithm using nonlinear kernel function for the nonlinear classification of the two classes of SVM problems.

II. IMPROVED ALGORITHM BASED ON FUZZY ASSOCIATIVE CLASSIFICATION OF SVM

The optimization problem of quadratic program should be solved to support the training of vector machine, and the traditional method of using standard quadratic model optimization technology to solve the dual problem is the main reason for slowing training algorithm. Firstly, it needs to compute and store the whole kernel function Hessen (Hessian) matrix for many times in the iterative process of training, in which the number of elements is m2 (m is the number of samples); and it is more critical that the Hessen matrix is not sparse, which leads to the algorithm matrix needing to consume a lot of time and taking up considerable memory capacity in computing matrix, especially the nonlinear kernel function matrix. Secondly, large numbers of matrix computations is needed in the process of quadratic model optimization, while the optimization algorithm occupies most time. And because the samples which finally contributes to the classification are only support vectors, and training SVM spends most of the time on the optimization of non support vectors while support vectors are few in the whole sample set, which seriously affects the learning efficiency of SVM. Therefore, supporting vector machine is very effective for small-scale training set, but in practice the training set is often relatively large.

In order to improve the speed of classification, a method DTSVM of linear support vector machine based on decision tree is proposed to replace support vector machine of nonlinear kernel function. The key of this method is to build a decision tree, and each node on the tree corresponds to a hyperplane. Each hyperplane can be obtained by training a linear SVM, and ensure that each hyperplane can divide an area containing only a sample. On the one hand, the method avoids the trouble of adjusting parameters; on the other hand, the hyperplane number of linear estimation classification function are far too fewer than SVs needed in classification. The specific process is described as follows.

Definition 1 (Two classes of SVM problems) Supposing that m1 and m2 are two natural numbers meeting $m=m_1+m_2$, $m_1>0$, $m_2>0$, and supposing $l=\{1,...m\}$, the definitions of positive class and negative class are as follows:

Positive class (class 1): positive class in training samples is made up of the collection $\{x_i\}$, in which $i \in l_1, l_1=\{1,...,m_1\}$, and m1 is the number of the positive class samples, and for all $i \in l_1$, there is $y_i=1$; then C_i is defined as the penalty of a single sample, $i \in l_1$.

Negative class (class 2): negative class in training samples is made up of the collection $\{x_i\}$, in which $i \in l_2, l_2=\{m_1+1,...,m_1+m_2\}$, and m_2 is the number of negative samples, and for all $i \in l_2$, there is $y_i=-1$; then C_i is defined as the penalty of a single sample, $i \in l_2$.

If there is a linear discriminant function (i.e. the optimal separating hyperplane) p, which has a equation, namely $P:\{x \in H \mid \langle w,x\rangle+b=0\}$, $w \in H$, $b \in R$, and can divide the samples in positive class and negative class correctly, then the two classes of sample is linear and separable. Vector w is the orthogonal vector with the hyperplane *p*, and (*w,x*) is the projection length of *x* in the direction of *w*. As shown in Figure 1, the hyperplane is often constructed as a standard hyperplane, and then the geometric margin of classification $\Delta=1/\|w\|$, in which the maximized classification margin is Δ, and the minimized is $\|w\|$.

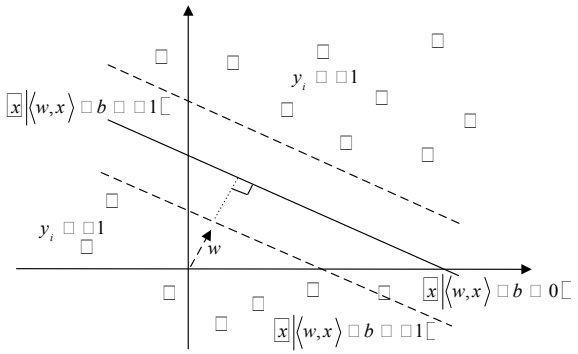

Fig. 1 Hyperplane of the two classes of problems with the maximized boundary

Definition 2 (The initial optimal problem of SVM) Class 1 and class 2 are defined as in definition1, and the initial problem of the hyperplane with the optimal boundary can be defined as follows:

$$\underset{w \in H, b \in R}{\text{minimize}} \quad t(w)=\frac{1}{2}\|w\|^2 \tag{1}$$

$$s.t. \quad y_i(\langle x_i,w\rangle+b) \geq 1, i=1,...,m \tag{2}$$

The corresponding decision function is:

$$f(x)=sign(\langle w,x\rangle+b) \tag{3}$$

Firstly, we can get a new optimal problem of SVM by modifying the constraints and the objective function in definition2.

Definition 3 (The optimal problems of DTSVM hard boundary) positive class and negative class are respectively defined in definition1, and the hyperplane of HartTreeSVM with the optimal boundary is defined as follows:

$$\underset{w \in H, b \in R}{\text{minimize}} \quad t(w)=\frac{1}{2}\|w\|^2-\sum_{i \in l_{\bar{k}}} y_i(\langle x_i,w\rangle+b) \tag{4}$$

$$s.t. \quad y_i(\langle x_i,w\rangle+b) \geq 1, \ i \in l_k \tag{5}$$

where, $k=1, \bar{k}=2$ or $k=2, \bar{k}=1$.

When defining class k as hard class and class $\bar{k}$ as non hard class, from the difference of the initial optimization problems of SVM, it can be seen that if the constraint is modified for $y_i(\langle x_i,w\rangle+b) \geq 1, i \in l_k$ the feasible solution of the optimal problem can divide accurately all samples in class k in the maximized boundary not using slack variables. On

the other hand, by adding a component in the objective function to express that the farther the hyperplane calculated in the distance of samples in class the smaller the objective function value is, it can make sure that the hyperplane which can be found divide samples in class in the minimal error. Similarly, the method of definition 2 turns into dual problem to solve, which is as follows:

Definition 4 (The dual problem of a class of hard boundary) Positive class and negative class are defined in definition 1, and the dual problem of a class of hard boundary is described as followings:

$$\underset{a\in R^m}{\text{maximize}}\ W(a)=\sum_{i=1}^{m} a_i - \frac{1}{2}\sum_{i=1,j=1}^{m} a_i a_j y_i y_j \langle x_i, x_j \rangle \tag{6}$$

$$s.t. \quad 0 \le a_i \le C_i,\ i \in l_k, \tag{7}$$

$$a_j = 1, j \in l_{\bar{k}}, \tag{8}$$

$$\sum_{i=1}^{m} a_i y_i = 0 \tag{9}$$

Where, $k=1,\ \bar{k}=2$ or $k=2,\ \bar{k}=1$. It can been seen that the dual problem is a special case of the original problem, namely, the case where all parameters a_k=1 in a class.

III. The Algorithm Description

The aim of the algorithm is to structure a tree with SVM nodes. Every step of the algorithm has an area which is plotted out of the hyperplane and marked as a class, until the whole space be marked completely.

A. The Decision Tree Made Up of Linear SVM Nodes

As shown in Figure 2, it can achieve quick sort of a date set by structuring a decision tree. The nodes of the decision tree are the hyperplanes obtained by the SVM training of the linear kernel. (In the figure, hc represents a hard class). In the process of structuring the tree, every step is to choose from the predefined hard class and train a SVM until the hyperplane obtained can correctly distinguish all the samples nodes belonging to class l_k , so all of the sample $x_i, i \in l_k$ lie in one side of the hyperplane and samples on the other side of the hyperplane are all belongs to non- hard class $l_{\bar{k}}$. In this way, it can reduce the number of samples needed in next step by deleting the training samples (having been correctly classified) which belongs to the class $l_{\bar{k}}$. This process is repeated until all of the samples belong to the same class.

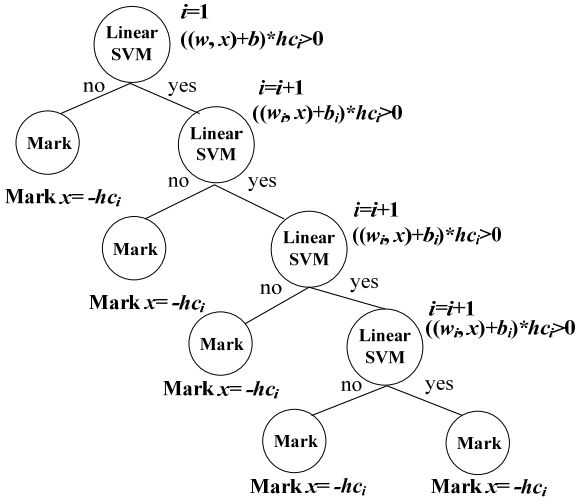

Fig. 2 Linear SVM Decision Tree

B. The algorithm of bias b

After the quadratic programming of definition 3 is solved completely, a node can calculate orthogonal vectors by the below formula:

$$w = \sum_{i=1}^{m} a_i y_i x_i \tag{10}$$

And the following is to calculate the bias b of the hyperplane, in which it must consider that the hyperplane to find should divide the hard class samples without error and try to classify another class of samples with less error. The traditional method of calculating bias can't completely solve the current bias problem and needs to be adjusted as following, when the hard class belongs to positive class.

$$b = \frac{\min_{i \in l_1} \langle w, x_i \rangle + \max_{\{j \in l_2 \cup \langle w, x_j \rangle < 0\}} \langle w, x_j \rangle}{2} \tag{11}$$

When the hard class belongs to the negative class:

$$b = \frac{\max_{j \in l_2} \langle w, x_j \rangle + \min_{\{i \in l_1 \cup \langle w, x_i \rangle > 0\}} \langle w, x_i \rangle}{2} \tag{12}$$

C. The Nonlinear Stretch of SVM Decision Tree

When using the above methods to test the sample classification, it can obtain the conclusion after traversing the whole or part SVM decision tree. Although in the decision tree, only using the linear nodes can get good results, by observing the wrong classification of the instance samples, it can be found that: firstly, most mistakes occur in the last node; secondly, only few samples can reach the last node in the process of classification; to further improve the accuracy of decision tree classification, adding a nonlinear SVM node to the end of the tree is useful, such as using RBF kernel function. As shown in figure 3.

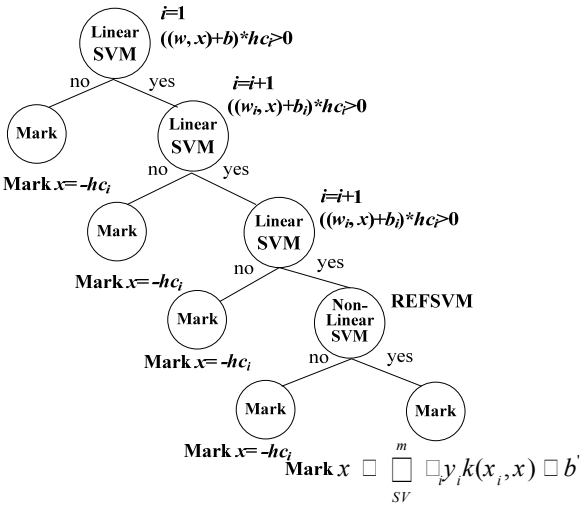

Fig. 3 SVM Decision Tree with Nonlinear Nodes

The process of training SVM decision tree of this stretch is similar to the training of the original decision tree. The first is to build a pure linear SVM decision tree, and then link a nonlinear SVM node to the last node. Compared with other linear nodes, except using different kernel function, the nonlinear node is trained in all the original training set instead of the remaining samples of the last linear node. Although this will generate many SVs in the last nonlinear node and increase the training time, usually only few testing samples will traverse to the last node, so the mean depth and time of classification won't be influenced too much.

IV. Experimental Results and Analysis

The two experiments are respectively designed to verify the effectiveness of fuzzy association classification DTSVM algorithm and SVM classifier construction method which is based on FCAR.

Experiment 1: To show the effectiveness and accuracy of DTSVM algorithm, a series of experiments are conducted on the standard Fourclass data set which is a two dimension data set and put forward by Vienna [5] at the 13th international conference on pattern recognition. In the experiment, the data is divided into two parts: training data and testing data, and randomly select one third data as training data.

The experimental results are shown in table 1 and table 2. Two standardized methods Min-Max and Std-Dev are separately used to train and standardize the samples, then three methods: classic nonlinear SVM, DTSVM methods based on the RBF kernel function and DTSVM method adding enhanced convergence performance are respectively used to classify the training samples. In the table, the last two attributes are ratio between the experimental results of RBF nonlinear SVM method and the results of the last two methods, as shown in the table.

Table. 1 Std-Dev standardized Fourclass sample classification

fourclass (std-dev)	SVM (RBF kernel)	DTSVM	RBF/ DTSVM
The Number of Sample Characteristic	2	2	
The Number of Training Sample	300	300	
The Number of SVs or Hyperplane	127	8	15.88
Training Time (s)	0.4	0.11	3.64
The Number of Testing Sample	605	605	
Classification Accuracy	524	587	0.89
Classification Time	0.22	0.07	3.14
Classification Precision	86.61%	97.02%	0.89

Table. 2 Min-Max standardized Fourclass sample classification

fourclass (min-max)	SVM (RBF kernel)	DTSVM	RBF/ DTSVM
The Number of Sample Characteristic	2	2	
The Number of Training Sample	300	300	
The Number of SVs or Hyperplane	155	17	7.12
Training Time (s)	0.27	0.19	1.42
The Number of Testing Sample	605	605	
Classification Accuracy	512	576	0.89
Classification Time	0.17	0.05	3.4
Classification Precision	84.63%	95.21%	0.89

The experimental results show that although the traditional nonlinear SVM has advantages in small sample classification, under two kinds of data standardization, the number of hyperplane obtained by training the sample using DTSVM algorithm is far smaller than the number of support vector obtained by nonlinear SVM method, the sample training time is three times quicker than traditional method and the actual classification precision is higher than traditional method. By comparing table 1 and table 2, it can be founded that compared with the previous the sample training time and the classification speed, the SVM algorithm has improved much after using the Min-Max standardized method, but the classification precision has reduced. The reason for such differences should be the uncertainty caused by random selecting of samples.

To verify the classification efficiency and accuracy of large-scale data set and multi-featured attribute data, the Isolet data set is selected for training and classification, meanwhile, transverse compare are conducted in the experimental results of three methods. Training samples and testing samples are selected randomly. The experimental results are shown in table 3 and 4.

Table. 3 Std-Dev standardized Isole sample classification

Isolet (std-dev)	SVM (RBF 核)	DTSVM	RBF/ DTSVM
The Number of Sample Characteristic	617	617	
The Number of Training Sample	154647	154647	
The Number of SVs or Hyperplane	33245	275	120.89
Training Time (s)	416.85	853.84	0.49
The Number of Testing Sample	2862	2862	
Classification Accuracy	2698	2665	1.01
Classification Time	198.76	34.53	5.76
Classification Precision	94.27%	93.12%	1.01

Table. 4 Min-Max standardized Isole sample classification

Isolet (min-max)	SVM (RBF 核)	DTSVM	RBF/ DTSVM
The Number of Sample Characteristic	617	617	
The Number of Training Sample	154647	154647	
The Number of SVs or Hyperplane	23952	325	73.70
Training Time (s)	689.45	340.28	2.03
The Number of Testing Sample	2862	2862	
Classification Accuracy	2726	2734	1
Classification Time	199.85	39.48	5.06
Classification Precision	95.25%	95.53%	1

The above experimental results shows that in large-scale multi-featured data set, the samples training time of this algorithm is longer than the training time of traditional nonlinear SVM algorithm. But on the whole, the algorithm has high classification precision under the condition of small sample and big sample, which shows that generalization performance of this method, is better than traditional SVM methods. At the same time, the two methods of data standardization have little influence on this data set.

The Table 5 below shows the comparison of classification results on the USPS data sets [24] among extended DTSVM method adding the nonlinear SVM node, the traditional nonlinear SVM method and the DTSVM method. The data set contains 18063 training data about handwriting recognition research, and 7291 testing samples. About the 256 features of every sample, the experimental results are as follows.

Table. 5 The Classification of Extended DTSVM method on USPS

USPS	SVM (RBF kernel)	DTSVM	DTSVM (extended)
Training Time (s)	27.52	38.06	64.21
The Number of SVs or Hyperplane	4471	105	105
Mean Depth of Classification		9.6	10.9.
The Time of Classification (s)	130.67	29.32	38.74
Classification Precision	96.32%	94.43%	96.17%

The experimental results show that DTSVM method is still much faster than the traditional SVM algorithm on the classification of the test sample, and after joining a nonlinear node in the DTSVM algorithm, its actual test accuracy is close to the traditional method of nonlinear SVM algorithm in classification accuracy, which proves the effectiveness and accuracy of the algorithm. DTSVM method based on compatibility feature vector is faster than the two methods of standardization DTSVM in front both in training time and classification time, meanwhile, the classification accuracy improves a lot. This is because the former has carried on the fuzzy classification association rules mining to the training sample in advance, then according to the produced FCAR to structure compatibility feature vector, which greatly reduces the number of DTSVM features input and the number of training samples, preliminarily deletes the data which has nothing to do with the decision attribute in the original samples, so it has significantly improved both in computational efficiency and accuracy of the meliorated algorithm.

V. CONCLUSION

In view of the defects of traditional classification algorithm, a classification framework combining the fuzzy classification association rules and the SVM is given, which improves the efficiency and accuracy of the classification. And SVM classification method based on decision tree has obtained, which has faster speed and good classification results for large-scale data classification compared with the traditional nonlinear SVM method, at the same time, can solve the problem by decomposing the nonlinear optimization problem into multiple linear SVM, and avoid the process of the kernel function parameter adjustment in the traditional method, simple and easy to implement. How to improve the decision tree model, increase the training speed and classification accuracy are the problems that need further research to make the method more applicable to the mining work of large transaction databases.

REFERENCES

[1] B.Liu, M..Ma, “Integrating Classification and Association Rule Mining,” Proc of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD’98), New York: Menlo Park, 1998,pp.80-86.

[2] M.Li , J.Han ,and J.Pei,“CMAR: accurate and efficient classification based on multiple class association rule,” Proc of the 2001 IEEE International Conference on Data Mining (ICDM’01),USA:IEEE, 2001,pp.369-376..

[3] Baralis,P.Garza, “A Lazy Approach to Pruning Classification Rules,” Proc of the IEEE 2002 International Conference on Data Mining. Japan: IEEE Press, 2002,pp.35-42.

[4] X.Yin, J.Han, “CPAR: Classification based on predictive association rules,” Proc of the Third SIAM International Conference on Data Mining, San Francisco: [s.n.], 2003.

[5] J.Wang, G.Karypis, “Harmony. Efficiently Mining the Best Rules for Classification,” Proc of 2005 SIAM International Conference on Data Mining (SDM’05). California. USA, 2005.

[6] F.Thabtah, P.Cowling, “A Greedy Classification Algorithm Based on Association Rule,” Applied soft computing, vol.3,July.2007, pp.1102-1111.

[7] Y.Hu , R.Chen,and H .Tzeng, “Mining Fuzzy Association Rules for Classification Problems,” Computers & Industrial Engineering, vol.43,Apr.2002