Research on Classifying Performance of SVM with Modified Kernel Function in HCCR

Limin Sun   Yibin Song
School of Computer Science & Technology
Yantai University
Yantai 264005, China
E-mail: lymansun@126.com
sybw@ytu.edu.cn

Abstract—**Support Vector Machines theoretically show very good performance for two-group classification problem, and the performance largely depends on the kernel function. However, there are no theories concerning how to choose good kernel functions based on practical using problem. In this paper, we tried to modify kernel both in data-dependent and margin-dependent way and applied the method to offline handwritten Chinese character recognition to investigate its classifying performance. Our experiment results show that the performance is improved with the proposed algorithm.**

I. INTRODUCTION

Support vector machines (SVMs) [1] offer a theoretically well-founded approach to automated learning of pattern classifiers for mining labeled data sets. Unlike traditional methods which minimize the empirical training error, SVM aims at minimizing an upper bound of the generalization error through maximizing the margin between the separating hyperplane and the dada. This can be regarded as an approximate implementation of the Structure Risk Minimization principle. Recently, SVMs have received considerable attention in various area because of their superior performance for classification and regression estimate. For the pattern recognition case, SVMs have been used for isolated handwritten digit recognition [2], object recognition [3], speaker identification [4], face detection in image [5] and so on.

Offline handwritten Chinese character recognition (HCCR) is one of means for quick text input and it has a great demand in the area of file recognition, form processing, machine translation and office automation. Currently printed Chinese character recognition technique has achieved the level of commercial use, however it still is a difficult task for offline handwritten Chinese character recognition to put into practical use because of its large stroke distortion, writing anomaly, and no stroke ranking information such as online handwritten Chinese character can get, etc. An efficient classifier occupies very important position for increasing offline HCCR ratio.

In pattern recognition problem, people always expect to enlarge the spatial resolution around the boundary so that the separability of classes is increased. Considering that support vectors always appear near the boundary, it is efficient to enlarge volume elements locally in neighborhoods of support vector. Amari & Wu presented a data-dependent way to implement the idea mentioned above by using a conformal mapping of the input Riemannian space [6]. As an application, in this paper, we proposed a both data-dependent and margin-dependent way to modify kernel function for improving support vector machine classifying performance. Our experiment results in offline handwritten Chinese character recognition show that the classifier of SVM with the modified kernel is efficient.

This paper is organized as follows: In section 2, we briefly review the fundamentals of SVMs for pattern recognition. In section 3, SVM classifier with modified kernel based on data-dependent and margin-dependent is introduced. Section 4 is our experiment data. At last, conclusions are given in section 5.

II. SUPPORT VECTOR MACHINE

The set of labeled training patterns

$$(\mathbf{x}_i, y_i) \quad i = 1,2,\cdots,l. \qquad y_i \in \{-1,\ 1\} \tag{1}$$

is said to be linearly separable if there exists a vector **w** and a scalar *b* such that the inequalities

$$\begin{aligned} \mathbf{w}\cdot\mathbf{x}_i + b \geq 1 \quad &\text{if} \quad y_i = 1, \\ \mathbf{w}\cdot\mathbf{x}_i + b \leq \text{-}1 \quad &\text{if} \quad y_i = -1, \end{aligned} \tag{2}$$

are valid for all elements of training set (1). The inequalities (2) can be written as

$$y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 \qquad i = 1,2,\cdots,l \tag{3}$$

The optional hyperplane

$$\mathbf{w}\cdot\mathbf{x} + b = 0 \tag{4}$$

is the unique one which separates the training data with a maximal margin: it determines the direction $\mathbf{w}/|\mathbf{w}|$ where the distance between the projections of the training vectors

0-7803-9422-4/05/$20.00 ©2005 IEEE

of two different classes is maximal. Therefore we get the decision function

$$y(\mathbf{x}) = \operatorname{sign}(\mathbf{w}\cdot\mathbf{x}+b) \qquad (5)$$

where $\operatorname{sign}(\cdot)$ stands for bipolar sign function.

To optimize hyperplane (4) is the arguments that maximize the distance $2/|\mathbf{w}|$. This amounts to the same thing of minimizing $\frac{1}{2}|\mathbf{w}| = \frac{1}{2}\sqrt{\mathbf{w}\cdot\mathbf{w}}$. Then the minimizing problem is transformed to the following quadratic program problem

$$\min_{\mathbf{w},b} \tfrac{1}{2}|\mathbf{w}|$$
$$\text{subject to } \; y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \ge 1, \quad i = 1,2,\cdots,l \qquad (6)$$

Consider the case where the training data can not be separated without error, to separate the training set with a minimal number of error is expected. Introducing some non-negative variables $\xi_i \ge 0$, the optimal problem of the hyperplane will be:

$$\min_{\mathbf{w},b} \tfrac{1}{2}|\mathbf{w}| + C\sum_{i=1}^{l}\xi_i$$
$$\text{subject to } \; y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \ge 1-\xi_i, \quad \xi_i \ge 0, \quad i = 1,2,\cdots,l \qquad (7)$$

where C is a positive constant, a larger C means a higher penalty is assigned to empirical errors. The optimization problem of (7) is a convex quadratic program which can be solved by using the well-know Lagrange multiplier method. By introducing Lagrange multipliers $\alpha_i \ge 0$ and $\beta_i \ge 0$, the Lagrangian function can be constructed as follows

$$L_{\mathrm{P}} = \tfrac{1}{2}|\mathbf{w}| + C\sum_{i=1}^{l}\xi_i - \sum_{i=1}^{l}\alpha_i[y_i(\mathbf{w}\cdot\mathbf{x}_i + b) - 1 + \xi_i] - \sum_{i=1}^{l}\beta_i\xi_i \qquad (8)$$

For getting minimal L_{P}, one can take the gradient of L_{P} with respect to **w**, b and ξ_i vanish to give the conditions:

$$\mathbf{w} = \sum_{i=1}^{l}\alpha_i y_i \mathbf{x}_i, \quad \sum_{i=1}^{l}\alpha_i y_i = 0 , \quad C - \alpha_i - \beta_i = 0 \qquad (9)$$

Since these are equality constraints in the dual formulation, we can substitute them into Eq. (8) to give

$$\max_{\alpha} L_{\mathrm{D}} = \sum_{i=1}^{l}\alpha_i - \tfrac{1}{2}\sum_{i=1}^{l}\sum\alpha_i\alpha_j y_i y_j \mathbf{x}_i\cdot\mathbf{x}_j \qquad (10)$$
$$\text{subject to } \; 0 \le \alpha_i \le C, \quad \sum_{i=1}^{l}\alpha_i y_i = 0$$

The coefficients α_i can be found by solving the above quadratic programming problem. Training samples ($\mathbf{x}_i$, y_i) with nonzero Lagrangian coefficients are called support vector. Then the decision function (5) should be rewritten as

$$y(\mathbf{x}) = \operatorname{sign}(\sum_{i=1}^{l}\alpha_i y_i \mathbf{x}_i\cdot\mathbf{x} + b) \qquad (11)$$

The threshold b can be gotten with any one of the support vector (SV). According to Karush-Kuhn-Tucker condition in a dual optimization problem, for the SVs $\{\mathbf{x}_i,\ i\in \mathrm{SV}\}$, the corresponding $\{\xi_i,\ i\in \mathrm{SV}\}$ are all zeros. As a result, we have

$$\sum_{i\in \mathrm{NSV}}\alpha_i y_i \mathbf{x}_i\cdot\mathbf{x}_j + b = y_j , \quad j\in \mathrm{SV} \qquad (12)$$

where NSV is normal support vector for those support vectors correspond to $0 < \alpha_i < C$.

For the case of the training samples are not separable with a liner hyperplane in the input space, one can map $\mathbf{x}_i$ to some high dimension (possibly infinite dimensional) Hilbert space **H** via a vector function $\boldsymbol{\varphi}(\mathbf{x})$, $\boldsymbol{\varphi}(\mathbf{x}) : \mathbf{x}\in \mathbf{R}^n \mapsto \boldsymbol{\varphi}(\mathbf{x}) \in \mathbf{H}$. In this high dimensional space, an optimal separating hyperplane, which maximizes the margin between the two closest vectors to the hyperplanes, is constructed. Notice that the only way in which the data appears in the training problem, Eqs. (10)-(12), is in the form of dot products, $\mathbf{x}_i\cdot\mathbf{x}_j$. This means that we needn't explicitly know what $\boldsymbol{\varphi}(\mathbf{x})$ is, only one thing to do is to compute the dot product $\boldsymbol{\varphi}(\mathbf{x}_i)\cdot\boldsymbol{\varphi}(\mathbf{x}_j)$ in training process. Now our work is to find a kernel function $K(\mathbf{x}_i,\mathbf{x}_j)$, such that $K(\mathbf{x}_i,\mathbf{x}_j) = \boldsymbol{\varphi}(\mathbf{x}_i)\cdot\boldsymbol{\varphi}(\mathbf{x}_j)$ satisfy Mercer's condition, replace $\mathbf{x}_i\cdot\mathbf{x}_j$ with $K(\mathbf{x}_i,\mathbf{x}_j)$ in the training algorithm, the algorithm will happily produce a support vector machine which lives in a high dimensional space. All the previous consideration hold, since we are still doing a linear separation, but in a different space.

III. SVM WITH MODIFIED KERNEL

Support Vector Machines theoretically show very good performance for two-group classification problem, and the performance largely depends on the kernel function. However, there are no theories concerning how to choose good kernel functions in a data-dependent way. In pattern recognition problem, people always expect to enlarge the spatial resolution around the boundary so that the separability of classes is increased. Considering that support vectors always appear near the boundary, it is efficient to enlarge volume elements locally in neighborhoods of support vector. Based on the principle mentioned above, by using a conformal mapping of the input Riemannian space, Amari & Wu proposed a data-dependent way to modify kernel function for improving support vector machine classifier [6]. The algorithm can be briefly stated as follows:

Supposing that $K(\mathbf{x}_i,\mathbf{x})$ is a kernel function, for a scalar function $c(\mathbf{x})$, *Definition.*

$$\widetilde{K}(\mathbf{x}_i,\mathbf{x}) = c(\mathbf{x}_i)\cdot c(\mathbf{x})K(\mathbf{x}_i,\mathbf{x}) \qquad (13)$$

is called a conformal transformation of a kernel by factor $c(\mathbf{x})$. According to Mercer's condition [1], $\widetilde{K}(\mathbf{x}_i,\mathbf{x})$ is also a kernel function. To ensure that $c(\mathbf{x})$ have large values at the support vector position, and have small one around other points, it is constructed in a data-dependent way as

$$c(\mathbf{x}) = \sum_{i\in \mathrm{SV}} \alpha_i \exp(-\|\mathbf{x}-\mathbf{x}_i\|^2 / 2\tau^2) \qquad (14)$$

where τ is a free parameter and summation runs over all the support vector. The optimal value of the parameter τ is around $\sigma/\sqrt{n}$ in the case of a Gaussian radial basis function kernel. The spatial resolution is enlarged around support vector with such a $c(\mathbf{x})$ as (14). The training process of the method consists of the two steps:

1) Train SVM with a primary kernel $K(\mathbf{x}_i,\mathbf{x})$, which is then modified according to Eq. (13) and (14),
2) Train SVM with the modified kernel $\widetilde{K}(\mathbf{x}_i,\mathbf{x})$.

In Amari & Wu's method, it doesn't directly take into account the margin between separating hyperplanes to modify kernel function. In practice, especially in the case of the margin is small, we always hope the spatial resolution around the boundary can be increased as large as possible. Therefore, in this paper, we proposed a both margin-dependent and data-dependent way to modify the kernel function such that the Eq. (14) will be changed to be

$$c(\mathbf{x}) = \sum_{i\in \mathrm{SV}} \tfrac{1}{2}|\mathbf{w}| \exp(-\|\mathbf{x}-\mathbf{x}_i\|^2 / 2\tau^2) \qquad (15)$$

From Eq. (15) we can see, the small margin, the large $c(\mathbf{x})$ will be, and all other considerations Amari & Wu mentioned still hold.

IV. CLASSIFYING PERFORMANCE INVESTIGATION

A. Character Base

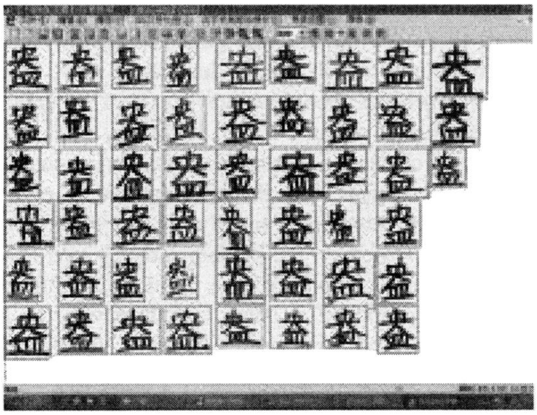

Fig. 1, handwritten Chinese character sample

In our experiments, the handwritten Chinese character base we adopted is SCUT-IRAC, which includes first class Chinese characters of GB2312-80 of all of 3755 characters and 94 symbols, for each character or symbol, there are 50 samples written by different person [9]. Fig. 1 gives an example of handwritten character sample in the region 16 of the database.

B. Feature Extraction

Handwritten Chinese character written by different person has large changes in stroke, but its topological structure is relative stable. So, one common technique adopted in handwritten Chinese character is to extract the structure information. In this paper, block-based relative fuzzy feature extraction method, we proposed in [8], is employed. The method doesn't only make same handwritten Chinese character with different written style having a close stroke distribution probability in the corresponding block, but also takes into account the relativities between strokes and its distribution probability. For extracting each character feature, we firstly divide each character image into some small blocks, i.e. mesh, with horizontal and vertical line. Elastic mesh is constructed such that not only the sum of the number of black pixels in each row in the mesh be equal and but also be equal for column direction dividing. And then, second, to extract sub-block feature and its related fuzzy feature to get feature vector. Interesting reader can see [8] for details.

C. Experiment Data

Our experimental character sets derive from the region 16, 17 of GB2312-80, total 94*50*2 handwritten character samples. In each character set, 40 samples are used to train the network, and other 10 samples are used to test the classifying performance. The character thinning processing for each character image is done before feature extraction. The mesh size varies from 4*4 to 8*8.

Gaussian RBF kernel function is chosen in first step, and set the value of σ to be equal to 0.5. Then to modify kernel function according to Eq. (13) and (15), where the value of τ is set to be around theoretic optimal value $\sigma/\sqrt{n}$ based on different feature dimensions.

Handwritten Chinese character recognition is a multi-class classification problem, and SVM is a classifier for two-class. So, for sharing the superior performance of support vector machine in pattern classification, we should construct a multi-class classifier based on it. In this paper, we adopted one-against-one classifier [7], which considers multi-class problem as a collection of binary classification problem by constructing classifier for arbitrary two classes, and gives total $k(k-1)/2$ classifiers. Similarly some voting scheme applied to determine a new point.

LIBSVM version 2.8 [10] is employed for the SVMs training and test in terms of the quadratic programming problem such as (10) and decision function (11). The experiment data are shown in table I and II.

As a contrast, we also did recognition experiments with classifier of supervised competitive learning neural network, where the competitive strategy of Euclidean distance and Equilibrium Error distance [9] is adopted respectively. The recognition data are given in table III and IV.

TABLE I

RECOGNITION RATIO (%) WITH RBF KERNEL AND MODIFIED KERNEL UNDER DIFFERENT FEATURE DIMENSIONS AND CORRESPONDENT VALUE OF τ IN REGION 16 OF GB2312-80

Mesh configuration		4*4	4*6	6*6	8*8
Feature dimension		64	96	144	256
RBF kernel		88.61	92.66	96.23	97.20
Modified kernel	τ	0.06	0.05	0.04	0.03
	Test set	89.28	93.63	97.20	97.29

TABLE II.

RECOGNITION RATIO (%) WITH RBF KERNEL AND MODIFIED KERNEL UNDER DIFFERENT FEATURE DIMENSIONS AND CORRESPONDENT VALUE OF τ IN REGION 17 OF GB2312-80

Mesh configuration		4*4	4*6	6*6	8*8
Feature dimension		64	96	144	256
RBF kernel		86.53	92.36	94.27	96.87
Modified kernel	τ	0.06	0.05	0.04	0.03
	Test set	87.55	92.41	96.10	97.06

TABLE III

RECOGNITION RATIO (%) WITH CLASSIFIER OF SUPERVISED COMPETITIVE LEARNING NEURAL NETWORK IN REGION 16 OF GB2312-80

Mesh configuration		4*4	4*6	6*6	8*8
Feature dimension		64	96	144	256
Euclidean distance	Training set	47.77	55.40	60.00	69.87
	Test set	47.23	57.23	61.70	68.19
Equilibrium-error distance	Training set	82.26	89.34	93.32	97.13
	Test set	85.00	92.02	94.57	97.34

TABLE IV

RECOGNITION RATIO (%) WITH CLASSIFIER OF SUPERVISED COMPETITIVE LEARNING NEURAL NETWORK IN REGION 17 OF GB2312-80

Mesh configuration		4*4	4*6	6*6	8*8
Feature dimension		64	96	144	256
Euclidean distance	Training set	45.48	55.12	58.90	65.48
	Test set	42.87	56.49	58.51	65.63
Equilibrium-error distance	Training set	81.70	89.34	93.51	96.73
	Test set	83.72	91.70	93.94	97.77

From table I and II, we can see that the classifying performance of the SVM with kernel modified by Eq. (13) and (15) is better than RBF kernel for HCCR. Compared to table III and IV, we found that the classifer of SVM is much better than that of supervised competitive learning neural network based on Euclidean distance, and it also better than that based on Equilibrium-error distance when the mesh configuration is under 6*6. While the mesh size is 8*8, the SVM classifier doesn't show advantage compared to neural network classifier based on Equilibrium-error distance

V. CONCLUSIONS

To enlarge the spatial resolution around the boundary surface is a good idea for increasing the separability of classes by modifying kernel function both in data-dependent and margin-dependent way. Our experiment results show that the performance of the SVM with proposed kernel modifying algorithm is efficient for HCCR, it is better than both classifier of SVM with RBF kernel and that of neural network except for 8*8 mesh size. The lack of our investigation work is that the experiment characters may be not enough, since the experiments were only carried out in region 16 and 17 of GB2312-80. To enlarge the experimental scope of character sets may give some more convinced results.

ACKNOWLEDGMENT

The authors would like to thank reviewer for his/her constructive comments and suggestions about this paper.

REFERENCES

[1] Burges, C.J.C.: "A Tutorial on Support Vector Machines for Pattern Recognition." *Data Mining and Knowledge Discovery*. 2 (1998) pp121-167.

[2] Cortes, C. and Vapnik, V. Support Vector Networks. Machine Learning, 20 (1995)273–297.

[3] Blanz, V., et al.: "Comparison of View–Based Object Recognition Algorithms Using Realistic 3D Models." *Artificial Neural Networks*—ICANN'96, pages 251 – 256, Berlin, 1996. Springer Lecture Notes in Computer Science, Vol. 1112.

[4] Schmidt, M. "Identifying Speaker With Support Vector Networks." *In Interface '96 Proceedings*, Sydney, 1996.

[5] Osuna, E., Freund, R. and Girosi, F. "An Improved Training Algorithm for Support Vector Machines". *Proc. of the 1997 IEEE Workshop on Neural Networks for Signal Processing*, (1997) pp276 – 285, Amelia Island, FL.

[6] Amari S., and Wu, S.: "Improving support vector machine classifier by modifying kernel function." *J. Neural Networks* , 12 (1999)783 - 789.

[7] J. Weston and C. Watkins. Multi-Class Support Vector Machines. Technical Report CSD-TR-9804, Royal Holloway, University of London, Egham, 1998. http://citeseer.ist.psu.edu/context/203915/8884.

[8] Sun, L.M. and Wu, S.H. "Handwritten Chinese Character Recognition Based on Supervised Competitive Learning Neural Network And Block-Based Relative Fuzzy Feature Extraction." *Proc. of IS&T/SPIE Electronics Image*, Vol. 5673, pp65-70, Jan. 2005, San Jose, CA

[9] Dan Liu. "Pre-processing & Recognition of Handwritten Chinese Character." *Ms. Degree Dissertation of South China University of Technology*. Jan. 1996, (in Chinese)

[10] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM--A Library for Support Vector Machines." http://www.csie.ntu.edu.tw/~cjlin/libsvm/