Neurocomputing 72 (2009) 3370–3375

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Letters

Adaptive binary tree for fast SVM multiclass classification

Jin Chen, Cheng Wang *, Runsheng Wang

ATR Laboratory, School of Electronic Science and Engineering, National University of Defense Technology, Changsha, China

ARTICLE INFO

Article history:
Received 23 September 2008
Received in revised form
17 February 2009
Accepted 25 March 2009
Communicated by J. Tin-Yau Kwok
Available online 10 April 2009

Keywords:
Multiclass classification
Support vector machine
Binary tree
Computational complexity

ABSTRACT

This paper presents an adaptive binary tree (ABT) to reduce the test computational complexity of multiclass support vector machine (SVM). It achieves a fast classification by: (1) reducing the number of binary SVMs for one classification by using separating planes of some binary SVMs to discriminate other binary problems; (2) selecting the binary SVMs with the fewest average number of support vectors (SVs). The average number of SVs is proposed to denote the computational complexity to exclude one class. Compared with five well-known methods, experiments on many benchmark data sets demonstrate our method can speed up the test phase while remain the high accuracy of SVMs.

1. Introduction

Support vector machines (SVMs) have been found to be very efficient to solve the classification problems, such as hand-written character recognition [1,2], image classification [3,4], and hyper-spectral classification [5,6]. The high generalization ability of SVMs is ensured by special properties of the optimal hyperplane that maximizes the distance between the closest training samples of each class and the separating hyperplane.

SVMs were originally designed for binary classification. There are two types of strategies to solve the multiclass SVM problem. One, called single machine approach, is by directly considering all data in one optimization formulation [7,8], while the other is by constructing and combining several binary classifiers. The latter type mainly consists of one-against-all (OAA) [9], one-against-one (OAO) [10,11], all-and-one (A&O) [12], direct acyclic graph SVM (DAGSVM) [13], the hierarchical tree-based methods [14,15] and error correcting output codes (ECOC) methods [16,17].

The single machine approach is not practical to many applications, for it generates a large optimization problem, which leads to time-consuming training [14]. Hsu and Lin [18] suggested that OAO and DAGSVM may be more suitable for practical use after comparing the single machine approaches with OAA, OAO and DAGSVM. Rifkin and Klautau [19] did a lot of carefully controlled experimental work and proposed that a simple scheme such as OAA (or OAO) is preferable to a more complex ECOC methods or single machine scheme.

Among the suggested methods, OAA and OAO are the two most common methods. The discrimination of OAA between an information class and all others often leads to the estimation of complex discriminant functions [5]. OAO decomposes the original problem into a set of small problems of two information classes. However, $N(N-1)/2$ binary SVMs are needed for one classification, which may result in slow classification, especially when N is very large. Recently, A&O was proposed to improve the classification accuracy of OAA and eliminate the wrong votes of OAO [12], but it needs N binary SVMs of OAA and one binary SVM of OAO for one classification, which costs more test time than OAA.

To reduce the test computational complexity, DAGSVM [13] and binary tree of SVM (BTS) [14] were proposed. DAGSVM only needs $N-1$ binary SVMs of OAO, while BTS needs $\log_{4/3}((N+3)/4)$ binary SVMs of OAO on average for one classification. Accordingly, both methods can achieve a much faster classification than OAO. BTS can have fewer binary SVMs for one classification than DAGSVM. However, it cannot always assure a faster classification than DAGSVM since the selected binary SVMs for classification may involve a much larger number of SVs, which will result in a relatively slower classification procedure, since the computational complexity of a binary test is proportional to the number of SVs.

In this paper, we propose a new strategy, called adaptive binary tree (ABT), for fast SVM multiclass classification. It focuses on reducing the number of SVs for one classification rather than reducing the number of binary SVMs. It can be faster than OAO, OAA, A&O, DAGSVM, and BTS in terms of test time, while the differences in the accuracy of all methods are very small. The tree selects the binary SVMs with the fewest average number of SVs

* Corresponding author. Tel.: +86 731 4575724; fax: +86 731 4575791.
E-mail addresses: chenjin_wonder@hotmail.com (J. Chen), chwang_nudt@263.net (C. Wang), rswang@nudt.edu.cn (R. Wang).

0925-2312/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2009.03.013

for each internal node, where the average number of SVs is proposed to denote the computational complexity to exclude one class. It also uses the separating planes of some binary SVMs to discriminant other binary problems according to the study of BTS [14], although the binary SVMs are trained for these problems. In the test phase, when an unlabeled sample reaches the leaf node, the final decision will be made (by excluding $N-1$ less similar classes). Experiments on many large multiclass data sets demonstrate that the proposed method can outperform OAO, OAA, A&O, DAGSVM, and BTS in terms of test time, while average classification accuracy of ABT is only 0.05% below the best result of all other methods.

Next section briefly introduces the background of multiclass SVM strategies. Section 3 presents the proposed method. Classification experiments on seven benchmark data sets are performed in Section 4.

2. Multiclass SVM background

This section first introduces five well-known multiclass strategies including OAA, OAO, A&O, DAGSVM, and BTS. Then, test computational complexity is also analyzed. For more details of SVMs, the reader is referred to [7,20].

2.1. Multiclass strategies

Generally speaking, the five methods differ each in the definitions of the binary SVMs and the combining strategy of the binary SVMs. Let $\Omega = \{\omega_i\}_{i=1}^{N}$ be the set of N information classes associated with the data set. The object of multiclass classification is to assign an input sample to one of the classes.

(1) OAA [9] represents the earliest and most common multiclass approach used for SVMs. Each class is trained against the remaining $N-1$ classes that have been collected together. The "winner-takes-all" rule is used for the final decision, where the winning class is the one corresponding to the SVM with the highest output (discriminant function value). For one classification, N binary tests are needed.

(2) OAO [10,11] needs to train $N(N-1)/2$ binary SVMs, where each one is trained on data from two information classes. When testing, for each information class ω_i, score will be computed by a score function:

$$S_i(\mathbf{x}) = \sum_{j=1, j\neq i}^{N} \mathrm{sgn}\{f_{ij}(\mathbf{x})\} \tag{1}$$

and ω_j. Then, the unlabeled sample $\mathbf{x}$ will be associated with the class with the largest score. For one classification, $N(N-1)/2$ binary tests are needed.

(3) A&O [12] combines OAA and OAO to improve the results of both methods. It trains $N(N+1)/2$ binary SVMs, including $N(N-1)/2$ binary SVMs of OAO and N binary SVMs of OAA. In the test phase, unlabeled sample is classified in the OAA framework and two classes whose corresponding SVMs have the two highest values are obtained. At last, the binary SVM trained for the two classes is used to get the final result. For one classification, $N+1$ binary tests are needed.

(4) DAGSVM [13] has the same training phase with OAO. However, in the test phase, it uses a rooted binary directed acyclic graph (consisting of $N(N-1)/2$ internal nodes and N leaves) to combine these binary SVMs. Each internal node is a binary SVM. When an unlabeled sample reaches the leaf node, the final decision will be made. DAGSVM can be seen as a tree-based version of OAO method, which excludes one class at each layer. For one classification, it only needs $N-1$ binary tests.

(5) BTS [14] generates a binary tree to combine the binary SVMs of OAO. It decreases the number of binary classifiers for both training and test. It only needs to train $N-1$ binary SVMs in the best situation and needs $\log_{4/3}((N+3)/4)$ binary tests on average for one classification.

Instead of grouping different classes together to train a global classifier, BTS selects two classes for training in every internal node. After the selection of binary SVM for current internal node, a clustering process is done according to the output of the selected SVM. To get a better result, the probabilistic output is employed to reassign the samples of the other classes which have been assigned to one of the node classes. The reasonability of a sample $\mathbf{x}_i$ in node k belonging to sub-node 0 or 1 is

$$\Delta P_k(\mathbf{x}_i) = P(y = 1|f_k(\mathbf{x}_i)) - 0.5 \tag{2}$$

where $P(y = 1|f(\mathbf{x}))$ is the posterior probability. It can be computed by

$$P(y = 1|f(\mathbf{x})) = 1/(1 + \exp(-f(\mathbf{x}))) \tag{3}$$

If data points of a certain class in node k have been assigned to child node 0 (child node 1) wholly, but these data pints will be assigned to child node 1 (child node 0) if some of their reasonability values

$$|\Delta P_k(\mathbf{x}_i)| < \delta \tag{4}$$

In general, a bigger δ leads to a higher accuracy, while the training time and the test time will increase.

2.2. Computational complexity

Efficiency of the multiclass methods can be verified in terms of generalization capability and computational complexity. A trade-off is often made to obtain an efficient solution for practical problems. Since the differences in accuracy are very small in a lot of experiments [18,19], only the test computational complexity of multiclass SVM method is analyzed in the following.

Both the computational complexity of a binary test and the number of binary SVMs can affect the computational complexity for multiclass problems. Computational complexity of a binary test is $O(n_{SV})$. However, one training data may be a support vector in different binary classifiers. Accordingly, the final test complexity is $O(n_{SV}^{u})$, where n_{SV}^{u} is the number of unique SVs for one classification. It is worth noting that different inputs may have different test complexity, since different sets of binary SVMs may be used for different tests in some methods, such as A&O, DAGSVM, and BTS.

3. Adaptive binary tree

To achieve a fast classification, DAGSVM and BTS reduce the number of binary SVMs for one classification. Interestingly, as analyzed in Section 2.2, the number of SVs also affects the computational complexity. Although BTS can have fewer binary SVMs for one classification than DAGSVM, it cannot always assure a faster classification than DAGSVM. For example, if some binary SVMs have a larger number of SVs, n_{SV}^{l}, and other binary SVMs have a smaller number of SVs, n_{SV}^{s}. Let us assume that binary SVMs with larger number of SVs are selected by BTS and binary SVMs with smaller number of SVs are selected by DAGSVM. When $n_{SV}^{l} = 3n_{SV}^{s}$, even if BTS only needs $(N-1)/2$ binary SVMs, it still