Support Vectors selection for supervised learning using an ensemble approach

Li Guo, Samia Boukir and Nesrine Chehata
Institut EGID, GHYMAC laboratory 1 allée F.Daguin 33670 Pessac, France.
li.guo@egid.u-bordeaux3.fr, sboukir@egid.u-bordeaux3.fr, nesrine.chehata@egid.u-bordeaux3.fr

Abstract

Support Vector Machines (SVMs) are popular for pattern classification. However, training a SVM requires large memory and high processing time, especially for large datasets, which limits their applications. To speed up their training, we present a new efficient support vector selection method based on ensemble margin, a key concept in ensemble classifiers. This algorithm exploits a new version of the margin of an ensemble-based classification and selects the smallest margin instances as support vectors. Our experimental results show that our method reduces training set size significantly without degrading the performance of the resulting SVMs classifiers.

Keywords-SVM; ensemble learning; margin;

I. Introduction

Support Vector Machines (SVMs) are based on statistical learning theory as proposed by Vapnik [1], which have been introduced as a technique for solving pattern recognition problems. SVMs have very strong theoretical background and have shown better results than many other classifiers in most applications, which has attracted more and more researchers to devote themselves to it. Perhaps, it has become the most widespread classifier today. However, it has a big weak point: the training is very slow, which limited its use in real-world applications, especially when the data set is huge. To speed up training, several methods have been proposed, which can be classified into two main approaches. One is to extract support vector candidates and then train the support vector machine using these data [2]. The other is to accelerate training by decomposition techniques, in which a small problem is repeatedly solved [3]. This paper proposes a new method to extract support vector candidates based on ensemble learning.

Ensemble learning is a popular learning paradigm, which builds a classification model by integrating multiple component learners. Bagging [4] is one of the most widely used and successful ensemble methods. Bagging is the acronym of bootstrap aggregating. It is made of the ensemble of bootstrap-inspired classifiers produced by sampling with replacement from training instances and uses these classifiers to get an aggregated classifier. The margin theory of ensemble methods was firstly proposed by Schapire et al. [5] to explain the success of boosting. We present here a new ensemble method based on the margin paradigm to refine the training instances of SVMs and thus accelerate its training speed.

II. Support vector machines

The SVM method consists in finding the hyperplane maximizing the distance (called the margin) to the closest training data points known as Support Vectors (SV) in both classes which play an important role of defining the discriminant hyperplane or predicting function. These support vectors are chosen among all the training data by solving a quadratic programming (QP) problem with linear constraints. So, the solution of a QP problem is the crux of the SVMs design, which depends on all the training instances and the selection of a few kernel parameters. However, the memory space for storing the kernel matrix in SVMs QP formulation is \( O(N^2) \), where \( N \) is the number of the training data. In addition, the time complexity of a standard QP solver is \( O(N^3) \) [6]. This indicates that SVMs are unsuitable to problems of large size, which motivates the appearance of methods to speed up the training stage of SVMs.

III. Margin of ensemble methods

A. Theory

Following Schapire’s definition [5], the margin of a sample \( x \) is computed by equation 1, where \( v_y \) is
the number of votes for the true class \( y \) and \( v_c \) is the number of votes for any other class \( c \). The range of the margin is from -1 to +1. A positive margin value of a sample indicates this sample has been correctly classified, a negative value means the sample has been wrongly classified. The larger the margin, the more confidence in the classification.

\[
\text{margin}(x, y) = \frac{v_y - \max_{c \neq y} v_c}{\sum_{c=1}^L v_c} (1)
\]

In addition, the margin of a sample reveals some characteristics of this sample. A large positive value means most of the base classifiers in the ensemble classified this sample correctly, which implies this sample is just in the centre of the distribution of all samples of the related class or nearby the centre. This type of samples represent general informations of the corresponding class. On the contrary, a large negative value shows that only a few base classifiers classified the corresponding sample correctly, which indicates these samples probably represent noise or outliers of the related class. A value close to 0 demonstrates the number of base classifiers which classified this sample correctly and the number of base classifiers which classified this sample as another class are about the same. It indicates this sample is likely on the boundary between these two candidate classes. This type of samples carry specific informations of these two classes.

B. A new definition of the margin

In a classification problem, we are interested in the boundaries of classes because they contain more significant informations about the classes. In this case, the true class labels of these samples are not of significance. To emphasize these particular samples, we propose a new definition of the margin, which can be computed by equation 2, where \( c_1 \) is the most voted class for sample \( x \) and \( v_{c_1} \) is the number of related votes, \( v_{c_2} \) is the second most popular class and \( v_{c_2} \) is the number of corresponding votes. Our margin’s range is from 0 to +1. The smaller the margin, the closer is a priori the related sample to the boundary of the classes, and therefore the more informations they provide. Furthermore, our margin concept does not require the true class label of sample \( x \).

\[
\text{margin}(x) = \frac{v_{c_1} - v_{c_2}}{\sum_{c=1}^L v_c} = \frac{\max_{c=1, \ldots, L} v_c - \max_{c \neq c_1, \ldots, L} v_c}{\sum_{c=1}^L v_c} (2)
\]

IV. Ensemble margin-based support vector candidates selection

According to the architecture of the SVMs, only the training data near the boundaries are necessary to build up the underlying classifier. In addition, because the training time becomes longer as the number of training data increases, the training time is shortened if the data far from the boundary are deleted. Therefore, if we can delete unnecessary data from the training data prior to training, we can speed up training [3].

Based on the above observation, we propose here a new method to extract support vector candidates based on the ensemble margin of each training instance. We use the new margin concept previously introduced. Our method consists in the following steps: (1) Constructing an ensemble classifier with all of the training data. (2) Computing the margin (defined by equation 2) of each training instance. (3) Ordering all the training instances according to their margin. (4) Selecting the first N smallest margin instances as support vector candidates.

To validate our method, we used bagging to create an ensemble, and the base classifier was Classification and Regression Trees (CART) [7]. We also compared our method with the random selection method. Our method is less costly than the SVM approach. Indeed, an ensemble bagging trees time complexity is \( O(tn \log(n)) \), where \( n \) is the number of training data and \( t \) is the number of trees in ensemble, compared to \( O(n^3) \) for a standard implementation of SVMs. In addition, the bagging algorithm is suitable for a parallel implementation.

V. Experimental results

In this section, we report experimental results on a synthetic imbalanced dataset Sin-Square (figure1) and 4 datasets from the UCI Machine Learning repository [8], shown in table I.

For each training set, according to the data selection method used, a portion of the training set (ranging from 20 to 100 percent) was selected as the reduced training set to train the SVM classifier. For each SVM, Gaussian kernels were used. Besides, we applied grid search on a validation set to get the best parameters to each SVM. The whole SVs of each dataset were produced by the SVMs with best parameters trained by the entire training set. All of the results in this paper were the mean value of 10 time calculation.

First of all, to demonstrate the relationship between the margin value of each instance and the features of this instance, we used all instances of data set
Table I. Datasets used in the experiments

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Train.</th>
<th>Test</th>
<th>Valid.</th>
<th>Attrib.</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Magic</td>
<td>2000</td>
<td>2000</td>
<td>1000</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>Pendigits</td>
<td>2000</td>
<td>2000</td>
<td>1000</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>Segment</td>
<td>800</td>
<td>800</td>
<td>710</td>
<td>19</td>
<td>7</td>
</tr>
<tr>
<td>Sin-Square</td>
<td>1000</td>
<td>3000</td>
<td>1000</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Waveform</td>
<td>2000</td>
<td>2000</td>
<td>1000</td>
<td>21</td>
<td>3</td>
</tr>
</tbody>
</table>

Sin-Square to construct our ensemble classifier and a SVM with Gaussian kernels (gamma = 1, cost = 1024). In figure 1, the filled points represent the first 20% smallest margin instances which are found by our ensemble classifier. This result clearly shows that the majority of the depicted small margin instances are at the boundary of the classes, thus behaving like SVs. The SVM method led to 308 SVs (using 5000 training instances) of which our ensemble classifier found 87% using just 25% of smallest margin instances as training data.

Figure 1. A portion of dataset Sin-Square. 20% of smallest margin instances displayed in filled points

Figures 2 and 3 show the curves of the rate of genuine SVs (provided by SVMs) among the selected SV candidates as a function of training data size, achieved by our ensemble-based data selection method and random data selection respectively. For each dataset, our method selected more SVs than the random alternative. In dataset Sin-Square in particular, our approach provides 91% of all SVs using just 30% of training instances.

Figure 2. Rate of genuine SVs among the selected SV candidates in data set Sin-Square.

Figure 3. Rate of genuine SVs among the selected SV candidates in data set Pendigits.

Figure 4 displays the SVMs classification accuracy behaviour with respect to training data size achieved on dataset Sin-Square after a training step using ensemble classifier selected instances and random selected instances respectively. It clearly shows that our method performed better than random selection. Our approach has been validated on four other datasets (see table I). Overall accuracy is also improved compared to a random selection of training data.

Figure 4. Classification accuracy on data set Sin-Square.

Figure 5 shows the accuracy and the training time curves as function of training size percentage of dataset Magic. Training time of SVMs with the whole training dataset was 0.48 seconds and led to a classification accuracy of 86%. With our method, we achieved the same accuracy using just 35% of all training instances with a cost of just 0.11 seconds, thus dividing the SVMs training time by 4. Figure 6 shows the less efficient result obtained among our 5 datasets.

Figure 7 shows the curves of the percentage of instances per class selected by our method on dataset Sin-Square. Class $C_2$ was the smallest class in data set Sin-Square with just 24 instances in the training set. However, our method selected all of instances of this class using just 40% of all training instances.
Figure 8 represents the difference between the SVM accuracy trained using the data provided by our method and the SVM accuracy trained by random selecting the data, for the most difficult class in each dataset. These results clearly show that our SV selection process outperforms a random selection. Indeed, our method is designed to more efficiently handle the classification of difficult and small classes. So, our method is suitable to classify imbalanced datasets.

Figure 5. Accuracy and training time of dataset Magic.

Figure 6. Accuracy and training time of dataset Segment.

Figure 7. Rate of selected training data per class on data set Sin-Square.

VI. Conclusion

In this paper, we have presented a new method to extract support vector candidates based on ensemble margin. Our selection strategy considers the smallest margin instances as support vector candidates. Through our experiments, several observations have been made:

1. Our method selects SV efficiently, especially in datasets with a limited number of SV compared to the whole training data set size.
2. Our method reduces the training set size significantly without degrading the performance of resulting SVM classifiers
3. Our method performs better in terms of classification accuracy, particularly in case of difficult classes.
4. Our method is suitable to imbalanced datasets.

References


