INCREMENTAL SUPPORT VECTOR MACHINES FOR FAST RELIABLE IMAGE RECOGNITION

L. Makili¹, J. Vega², S. Dormido-Canto³

¹Instituto Superior Politécnico da Universidade Katyavala Bwila, Benguela, Angola
²Asociación EURATOM/CIEMAT para Fusión, Madrid, Spain
³Dpto. Informática y Automática - UNED, Madrid, Spain

This paper addresses the reliable classification of images in a 5-class problem. To this end, an automatic recognition system, based on conformal predictors and using Support Vector Machines (SVM) as the underlying algorithm has been developed and applied to the recognition of images in the Thomson Scattering Diagnostic of the TJ – II fusion device. Using such conformal predictor based classifier is a computationally intensive task since it implies to train several SVM models to classify a single example and to perform this training from scratch takes a significant amount of time. In order to improve the classification time efficiency, an approach to the incremental training of SVM has been used as the underlying algorithm. Experimental results show that the overall performance of the new classifier is high, comparable to the one corresponding to the use of standard SVM as the underlying algorithm and there is a significant improvement in time efficiency.

Keywords: conformal prediction, confidence, credibility, incremental support vector machines

1. Introduction

The TJ – II’s Thomson Scattering Diagnostic (TSD) has a high level of automation of its operation, being able to execute automatic analysis processes after its laser shot. To this end, an automated image classification system has been developed and a classifier based on Support Vector Machines (SVM) [1, 2] has been running for years [3, 4].

Despite the success of SVM – based classification, it can only provide bare classifications without an estimation of the corresponding reliability level. Estimates of the reliability of predictions allows strengthening of decisions based on the referred predictions and, based on it warnings can be programmed providing some interactivity and human supervision of automated processes with minimal human intervention.

Conformal prediction [5] is a suitable way to supplement predictions made by machine learning methods with measures of accuracy and reliability and some effort has been done in order to implement this method for TJ – II TS images classification, see for example [6, 7].

SVM is a good candidate for the underlying algorithm of conformal prediction. However, using SVM to this end can be a computationally intensive task since it implies to train several SVM models to classify a single example and training SVM becomes less and less tractable as the number of training examples increases. To overcome this issue, different approaches has been used, by speeding up SVM computation, on the one hand, or by reducing the size of the training set using active learning strategies, on the other hand [7, 8].

Speeding up computations using an incremental approach to the SVM training, instead of training it from scratch for each test example, can result in an important time improvement of the overall classification process [9]. Many incremental approaches to SVM training have been proposed, mainly to address the problems of both learning from very large data sets and active learning. There are several methods to tackle this that provide either approximate or exact solutions. Examples of such different approaches can be found in [10 - 16].

In this paper, we focus on the problem of reliable instance classification, with the aim of improving time efficiency when classifying with confidence the images of the TJ – II’s Thomson Scattering Diagnostic (TSD). We use SVM as the underlying algorithm of a conformal predictor classifier, using an incremental approach for the training.

The layout of this paper is organized as follows. In section 2, we introduce the notion of conformal predictor and in section 3 we outline, on the one hand, the approach to the incremental training of SVM and, on the other hand, how to build
a nonconformity score from them. In section 4, the experiments made with the TJ – II Thomson Scattering’s images are described and results are presented. Finally, the conclusions are shown in section 5.

2. Conformal predictors

Conformal prediction is a method that allows us to make reliable predictions, supplementing the predictions with some measures of its reliability [5].

The basis of the method is to predict that a new object will have a label that makes it similar to the old examples in some specific way [5]. To this end, a numerical score, called nonconformity score (α), is assigned to each example in the sequence formed by the training examples and the new example, measuring how different is the new example from the full sequence of examples.

To do a reliable prediction, a comparison of the α-value assigned to the new example to the other αi’s is needed. A convenient way of making this comparison is to use the p-value. The p-value for each example is computed as the ratio

\[ p_y = \frac{\left| \{ i = 1, 2, \ldots, n : \alpha_i \geq \alpha_n \} \right|}{n} \]  

(1)

3. Incremental Support Vector Machines and nonconformity scores

To classify with a conformal predictor, a key element is the computation of nonconformity scores, and there are different ways to do this. In this paper, we compute nonconformity scores from Support Vector Machines (SVM) [1, 2]. To this end, we can use the Lagrange multipliers, αi, obtained by solving the corresponding dual problem, which are ideal for use as nonconformity scores [5].

Originally, SVM could only handle binary classification problems, but there are standard ways to reduce multi-label classification problems to the binary case [5, 17]. Suppose that we have defined a reasonable way to compute nonconformity scores, A, for binary classification, assuming that the label space in the binary classification problem is \{0, 1\}. In a one-versus-the-rest approach, we compute the nonconformity scores doing a weighted summation among the nonconformity scores computed from each binary problem as

\[ A((x_1, y_1), \ldots, (x_1, y_1), (x, y)) = \lambda \]

\[ x A((x_1, I_{y_i=y}), \ldots, (x_1, I_{y_i=y}), (x, 1)) + \frac{1-\lambda}{|Y|-1} \]

\[ x \sum_{y \neq y} A((x_1, I_{y_{i=y}'}), \ldots, (x_1, I_{y_{i=y}'}), (x, 0)) \]  

(2)

In this case, \((x_i, y_i)\) are the training examples, \(\lambda \in [0, 1]\) is a constant parameter, \(I\) is the indicator function and \(|Y|\) is the cardinality of the label space.

Speeding up SVM training can result in an important improvement in time efficiency. On this way, and reasoning on an off-line learning setting [5], we can do a first training of the SVM and then use the result of this previous training to do a posterior training, at testing time, using an incremental approach.

In this case, if we have a sequence of training examples \((x_1, y_1), \ldots, (x_n, y_n)\), and of unlabeled testing examples \(x_{n+1}, \ldots, x_l\), we can follow the algorithm,

- Do an initial train of the conformal predictor, using the training set and training the SVMs from scratch, and stores it;
- For each example \(x_i, i = n + 1, \ldots, l\), in the testing set do
  - Consider all possible values \(y\) for the label \(y_i\);
  - Compute the p-values \(p_y\) for each possible classification, incrementing the example to the previously stored
solution when training the SVMs;

- Predict the label \( y \) corresponding to the largest p-value calculated;
- Output one minus the second largest p-value as the confidence for the prediction;
- Output the largest p-value calculated as the credibility of the prediction.

End for.

In this paper, to do the incremental training of SVM we follow the approach introduced by Syed, Liu and Sung [10]. Next we outline the referred approach.

3.1. Syed, Liu and Sung’s (SVMi) approach

This approach is based on the property that the SV algorithm preserves the essential information about the decision function in a form of a set of data points, the support vectors, which are preserved for future testing. Usually, only a few portion of the training data points become support vectors, which means that the algorithm can summarize the properties of the data space in a concise manner and that training an SVM only on the support vectors results in the same decision function as training on the whole data set.

On this way, if we train an SVM on several steps, preserving from the previous steps the support vectors that are added to the batch of data to train the next step, we obtain an incremental effect and in the final a similar decision function as if we trained the SVM on the whole data set in one batch.

It has been shown that incrementally trained SVM with this approach compares very well to their non-incremental counterpart [10] and that concept drifts are handled well, provided that the batches used for incremental training are good enough, i.e., their statistical properties don’t differ much from the statistical properties of the whole data set [11, 18], which is accomplished if examples in the data set are independent and identically distributed, as required by the conformal prediction theory.

4. Experiments and results

Now, we’ll discuss the experiments conducted on a set of TJ – II Thomson Scattering’s images.

The algorithm was tested on a set of images from the TJ – II's Thomson Scattering Diagnostic. We obtained a total of 1149 objects, where each object is an image with 576 x 385 pixels, i.e., 221760 possible attributes.

To avoid dimensionality problems, each image was represented by a feature vector that is the vertical detail coefficients of the Haar wavelet transform (WT) at level 4 of decomposition [19] of the original image. This decomposition has shown to have good discrimination capabilities for this kind of signals [20] and permits us to reduce the number of attributes to 36 x 25 (0.41 % of the original).

The incremental SVM approach was implemented using LIBSVM (http://www.csie.ntu.edu.tw/~cjlin/libsvm/) as the base optimizer and as a general framework to the conformal prediction algorithm implementation we used the Spider (http://people.kyb.tuebingen.mpg.de/spider/main.html).

4.1. Predictive accuracy

Algorithm’s performance was measured by the following parameter: the proportion of classifications made correctly over an independent training set and as the error estimator a 10 – fold cross - validation has been used [21].

Table 1 summarize the results of the experiments done. It shows the average success rates (SR) over the 10 folds, and the corresponding standard errors (Std Err), obtained using a radial basis function (RBF) kernel,

\[
K(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right),
\]

in conjunction with the one-versus-the-rest multiclass method. As was expected, performances of SVM-based algorithms depend strongly on parameters selection. Consequently, previously to the classification process a search for the best value of the regularization (C), kernel (\( \sigma \)) and one-versus-the-rest (\( \lambda \)) parameters was made for each classifier.

Other kernel functions, linear and polynomial, have been tested. Due to space considerations we only show the results for the RBF kernel which gives the best success rates.
As a matter of comparison, we present in the table the success rates reached by a conformal predictor using a standard SVM approach to compute nonconformity scores (CP_SVM) and by an SVM classifier without reliability measures (SVM).

4.2. Time efficiency comparison

A set of experiments were carried out in order to do a comparison between time efficiency for the conformal predictors using the incremental SVM approach and using a standard SVM approach to compute nonconformity scores. In this case, as a measure of efficiency we consider the CPU time (in seconds) needed by a classifier to classify a single image.

Then, both classifiers were trained using the same regularization and kernel parameters ($C = \infty$, $\sigma = 2^{11}$) and randomly selected and stratified training sets with different sizes were used; results reported in figure 1 correspond to the average CPU times over 10 runs of each classifier for each size of the training set.

5. Conclusions

In this paper, we focused on the problem of reliable prediction. We described an implementation of the conformal predictor method, using an incrementally trained SVM as the underlying algorithm, applied them to a multiclass classification problem, the prediction of TJ – II's Thomson Scattering images, and the corresponding time efficiency was compared with the one related to a conformal predictor implemented using a standard SVM as the underlying algorithm.

Experiments show that predictions made by the algorithm are accurate and comparable to the performance of the conformal predictor using the standard SVM as the underlying algorithm. The success rates reached are slightly lower than the ones reached by the simple SVM classifier, but the conformal predictor based classifier has the advantage of attaching measures of reliability to the predictions, instead of making bare predictions.

Time efficiency is better when the incremental SVM approach is used as the underlying algorithm. In this case, the CPU time needed to classify a single image remains in the range between 0.4 and 0.6 s, even for large training sets, whilst when the approach using the standard SVM is used it tends to grow up significantly with the size of the training set. For large training sets the difference in processing time is considerable, being the conformal predictor using the incremental SVM approach up to almost 7 times faster than the one using the standard SVM approach.

Acknowledgments

This work was partially funded by the Spanish Ministry of Science and Innovation under the Project No. ENE2008-02894/FTN and by the Grants’ National Institute (INABE) of the Angolan Ministry of Higher Education, Science and Technology.

Authors wish to thank Professor Alex Gammerman for sharing the manuscript of [17].

References
Captions

Fig. 1: Computational time versus number of training examples

Table 1: Average success rates and corresponding standard errors. Results corresponds to a 10-fold cross-validation
Figure 1: Computational Time, kernel RBF (C = ∞, σ = 2^{11})
<table>
<thead>
<tr>
<th></th>
<th>CP_SVMi $(C = \infty, \sigma = 2^{10}, \lambda = 0.4)$</th>
<th>CP_SVM $(C = \infty, \sigma = 2^{11}, \lambda = 0.4)$</th>
<th>SVM $(C = 2^6, \sigma = 2^{11})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SR (%)</td>
<td>97.48</td>
<td>97.74</td>
<td>98.09</td>
</tr>
<tr>
<td>Std Err</td>
<td>0.438</td>
<td>0.434</td>
<td>0.007</td>
</tr>
</tbody>
</table>