Excluding cascading classifier for face detection

Ivan A. Krestinin, Oleg S. Seredin
Department of Automation and Remote Control
Tula State University, Tula, Russia
ivan@crown-s-soft.com, oseredin@yandex.ru

Abstract

In the article, is discussed the methodology allowing to accomplish the task of learning by instruction in conditions of large training sets. The essence of the methodology comes to introducing of a preliminary cascading classifier removing objects which present “simple” cases from a training selection.

Keywords: *machine learning, large scale train set, cascading classifier, object detection.*

1. INTRODUCTION

When accomplishing a task of searching for a face or another object in an image, a procedure generating a set of fragments is usually used, for example, the sliding-window method or the special-point method [1]. After that, each of the fragments is classified by the binary classifier, which refers an object either to the “face” class or to the “background” class (“non-face”). Thus, the quality of the search is basically defined by the quality of the binary classifier used for this.

However, when creating a quality binary classifier, the following problems arise: a large amount of training material; the large dimensionality of the attribute space; a considerable difference between the number of objects of the “face” class and the number of objects of the “non-face” class. Similar problems arise not only when detecting objects in images; thus, some procedures of solving single-class tasks presuppose generating the second set, which evenly fills the entire object space [2]. It’s obvious that in this case, the number of objects in the generated set is likely to be considerably greater than that of the objects in the original set.

2. EXCLUDING CLASSIFIER TRAINING

In practical tasks of training a binary classifier, we can often make an assumption about the compactness of one of the sets. To identify it, let us call its objects “positive”, and objects of the second class, which is less compact, we’ll call “negative”. Quite often, the number of objects of the positive class turns out to be considerably less than that of negative set objects. In those tasks when there’s no opportunity to make a preliminary judgment on objects’ compactness level, a set containing a greater number of objects or an arbitrary set can be considered “negative”.

An example of the described situation is given in Figure 1.

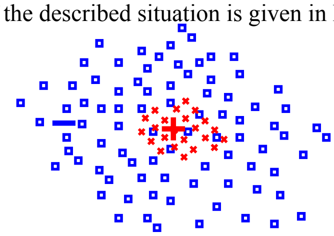

Figure 1: Objects of positive and negative sets.

With such a classifier, let us try to sort out a number of objects, which present “simple” cases. For this, let us train a classifier with the zero error rate for positive objects and, as far as possible, with a low error rate for negative objects (Figure 2).

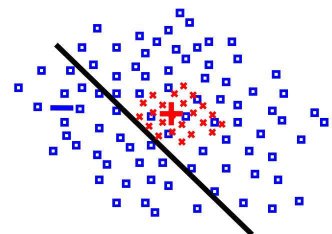

Figure 2: The first “cutting” hyperplane.

After we get the classifier with the zero error rate for positive objects, let us remove the objects, which were unambiguously referred to the negative set by the classifier, from the training set and train another classifier with the zero error rate for positive objects (Figure 3).

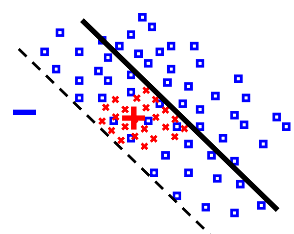

Figure 3: The second “cutting” hyperplane.

We’ll continue following the specified procedure until the number of objects cut off by every new hyperplane is small enough. As a result, we’ll get the situation shown in Figure 4.

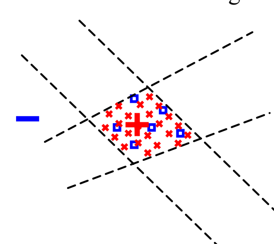

Figure 4: The training set after introducing four cutting hyperplanes.

After the procedure has been completed, we can unite the modeled hyperplanes on the principle of “logic AND”, that is, refer an object to the positive set only if **all** of the classifiers treated the object as positive.

3. ANALYSIS OF TRAINING PROCEDURE RESULTS

It’s quite possible, that the classifier we got in this way is already of a high quality level – in this case, it can be used independently.

The high quality level of such a classifier indirectly indicates that that the hypothesis about the compactness of the positive set, which was originally put forward, is true. In case of negative results, perhaps, it's reasonable to make an assumption about the compactness of the negative set and repeat the procedure interchanging the sets.

Negative results of the procedure can also indicate the high level of noise in data. In this case, is recommended analyzing the positive objects which are most close to the hyperplanes (when using an SVM classifier, frame objects). Such an analysis may help to define the noise source.

4. CLASSIFIER'S CHARACTERISTICS

We can try to improve the results by using the classifier we got as a "preliminary filter" for objects. To do this, the objects of the test set, which were not referred to the negative set by any of the hyperplanes, should be split with a more complex classifier, for example, by moving the objects to a space of larger dimensionality or by using a non-linear nuclear for the support vector method. Here, a procedure similar to that described above can also be used.

Considerably reducing the number of objects in the negative set allows us to use more complex recognition procedures at the next stage of training. Using such procedures for the original training set would be extremely difficult or even impossible due to the large number of objects.

Another advantage of the resultant set of hyperplanes is the ability to check an object incompletely at the recognition stage; thus, if the next hyperplane referred an object to the negative set, checking the object with other cascades doesn't make sense as it can be immediately referred to the negative set.

Among the disadvantages of the algorithm, we should note its extremely high susceptibility to noise in data, which virtually unambiguously defines the sphere of its usage as a preliminary sorting out of "simple" cases before using a complex classifier.

Another disadvantage is that the positive set must be a highly completed set as the decision rule is entirely defined by the hull of the set, which can lead to a high error rate if there are no objects close to a section of the positive set's real hull.

Also, it should be noted that the algorithm is somehow similar to boosting procedures [5]. The algorithm is not a subset of these procedures as the final classifier is trained not as a weighted total of individual classifiers' results, but as uniting them by the "logic AND" operation. However, in some respect, the algorithm can be regarded as a certain ultimate case of a boosting procedure. As compared with this group of algorithms, the described one advantageously differs from them as instead of assigning low importance to the successfully classified objects, it completely excludes them, which positively influences the time performance of the training procedure and of the recognition procedure as well.

5. EXPERIMENTAL RESULTS

To test the described procedure, we used a set of images (objects) of two classes: "faces" and "non-faces". We made an assumption about the compactness of the faces set. The number of positive objects (faces) is 540 thousand, the number of negative objects - 22 mln. The basic problem of this task was that a large number of objects in the negative set did not allow using the procedure of training an SVM classifier in a space with 1200 attributes (due to unreasonable memory and CPU time requirements). The dimensionality of the attribute space for the preliminary cascading classifier was chosen to be 270 attributes. Thus, using 36 hyperplanes at the training stage, we have got a classifier with the zero error rate for the faces set and with the error rate of 2.25% for the non-faces set. Thus, the number of objects in the negative set was reduced from 22 mln. to 510 thousand, which allowed using a classifier with the dimensionality of 1200 attributes at the next stage of training.

6. CONCLUSION

The described methodology of uniting classifiers into a cascade can be highly useful in terms of large amount of the training material, and also, to optimize time performance of complex classifiers by reducing the number of objects being classified at the expense of introducing an additional cascade of simple classifiers passing judgments in simple cases, which are prevailing. This methodology can also be used as an independent classifier, provided that the compactness of objects of one of the classes is high.

7. REFERENCES

[1] Крестинин И. А., Середин О. С. Метод особых точек в задачах поиска лиц на графических изображениях // Известия ТулГУ, серия «Технические науки», Вып. 3 – Тула: ТулГУ, 2008, с. 218-227.

[2] Roberts, S., Tarassenko, L., Pardey, J., and Siegwart, D. A validation index for artificial neural networks. In Proceedings of Int. Conference on Neural Networks and Expert Systems in Medicine and Healthcare, 1994, pages 23–30.

[3] Vapnik, V.: Statistical Learning Theory. Wiley, New York, 1998.

[4] Veropoulos K., Cristianini N. and Campbell C. Controlling the sensitivity of support vector machines. Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI99), Stockholm, Sweden, 1999.

[5] R. Polikar. Ensemble Based Systems in Decision Making. IEEE Circuits and Systems Magazine, vol.6, no.3, pp. 21-45, 2006, http://engineering.rowan.edu/~polikar/RESEARCH/PUBLICATIONS/csm06.pdf

About the authors

Ivan A. Krestinin is a Ph.D. student at Tula State University, Department of Automation and Remote Control. His contact e-mail is ivan@crown-s-soft.com.

Oleg S. Seredin, was born on 18.11.1974. In 1998, obtained the M.S. degree at Tula State University, Department of Automation and Remote Control. In 2001, received the Ph.D. degree at Computing Center of the Russian Academy of Sciences, Moscow (Thesis: Methods and algorithms of featureless pattern recognition). Since 2002, he has been an Associate Professor at Tula State University, Department of Technical Cybernetics. Lecture Courses: Object-oriented Programming, Data Mining. Theoretical Research: featureless pattern recognition, linear methods of data analysis. His contact e-mail is oseredin@yandex.ru.