Pattern Recognition Letters 29 (2008) 1565–1570

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Growing a multi-class classifier with a reject option

D.M.J. Tax *, R.P.W. Duin

Information and Communication Theory Group, Mekelweg 4, 2628 CD Delft, The Netherlands

ARTICLE INFO

Article history:
Received 11 October 2007
Received in revised form 30 January 2008
Available online 30 March 2008

Communicated by F. Roli

Keywords:
Multi-class classification
Outlier detection
Rejection

ABSTRACT

In many classification problems objects should be rejected when the confidence in their classification is too low. An example is a face recognition problem where the faces of a selected group of people have to be classified, but where all other faces and non-faces should be rejected. These problems are typically solved by estimating the class densities and assigning an object to the class with the highest posterior probability. The total probability density is thresholded to detect the outliers. Unfortunately, this procedure does not easily allow for class-dependent thresholds, or for class models that are not based on probability densities but on distances. In this paper we propose a new heuristic to combine any type of one-class models for solving the multi-class classification problem with outlier rejection. It normalizes the average model output per class, instead of the more common non-linear transformation of the distances. It creates the possibility to adjust the rejection threshold per class, and also to combine class models that are not (all) based on probability densities and to add class models without affecting the boundaries of existing models. Experiments show that for several classification problems using class-specific models significantly improves the performance.

1. Introduction

In standard problems one has to classify an object and assign it to one of a set of known classes. In practice one also has to reject the objects that do not fit to any of the classes (Dubuisson et al., 1985). In these applications some classes may be known, but novel classes can appear and these are unknown. In a face recognition problem a model for each person in the training set has to be trained. The system should recognize a novel person and it should not assign this outlier person to one of the known persons. Furthermore, an extra practical demand is that the system should be easily extendible to include new persons, and it should be simple to remove known persons. Moreover, such an extension should not affect the decision boundaries between existing models. These types of demands are not only typical for face recognition (Kang and Choi, 2006), but also for the classification of crops, industrial products, disease detection in medical imaging etc.

The standard approach to rejection in pattern recognition is to estimate the class conditional probabilities, and to reject the most unreliable objects, that is, the objects that have the lowest class posterior probabilities. This is called the *ambiguity reject* (Chows rule, Chow, 1970). This reject rule is optimal when the posterior probabilities are estimated without error. In the case of estimation errors, it was recognized in Fumera et al., 2000 that a per-class threshold may be required.

Furthermore, using the class posterior probability for rejection ignores the possibility of having objects from unknown classes. These objects do not typically appear in areas with a low posterior probability (i.e. in areas between the known classes), but they are often distributed around the known classes, where the total data probability density is low, but the posteriors are high. In Dubuisson and Masson, 1993 the ambiguity reject was extended to the *distance reject* in which objects are rejected for which the full data density is below a threshold. This can be seen as outlier detection, or novelty detection, and numerous other outlier detection algorithms in a wide range of scientific fields have been proposed (Davies and Gather, 1993; Japkowicz et al., 1995; Tarassenko et al., 1995; Cerioli and Riani, 1999; Baker et al., 1999; Pan et al., 2000; Ramaswamy et al., 2000; Tax and Duin, 2001; Marsland, 2001).

It appears that some of these outlier detection methods do not rely on a probability density estimate. To estimate a probability density requires a large amount of training data, and when the feature space is large in comparison to the training set size, density estimators suffer from the curse of dimensionality (Duda et al., 2001). It is therefore often better to avoid an explicit density estimation and to use an approximate model. Unfortunately, this makes the combination of the models to a multi-class classifier more complex, in particular when one wants to do more than simple voting. Confusion often occurs in situations where objects are accepted by more than one model. Unfortunately, in many cases it is just two models, and in this situation voting is not applicable.

* Corresponding author. Tel.: +31 (0) 15 27 88434; fax: +31 (0) 15 27 81843.
E-mail addresses: D.M.J.Tax@tudelft.nl (D.M.J. Tax), R.Duin@ieee.org (R.P.W. Duin).

0167-8655/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2008.03.010

For these situations the soft outputs of the models have to be compared. But because each model may have a different way to measure the similarity of an object to its class, the output values of the different class models have to be normalized.

In this paper we investigate and compare two rescaling heuristics for one-class models. Both heuristics are constructed such that the decision boundaries between the classes and the outliers are not affected. The first scaling makes use of the assumption that all class models give a fixed output for outlier objects, and for that it requires a non-linear scaling of distances. The second scaling assumes that the average output for a class is constant, and assumes that classes are relatively well sampled. In Section 2 we start with discussing the standard approach of rejecting objects. Next, the combination of models and the required normalization are presented in Section 3. In Section 4 the experimental evaluation is done and the paper finishes with conclusions in Section 5.

2. Multi-class classifiers with reject and class models

Assume we are given objects $\boldsymbol{x}$ from c classes $\omega_1, \ldots, \omega_c$, with prior probabilities $p(\omega_i)$. All objects are represented by p-dimensional feature vectors in a bounded area in the feature space: $\boldsymbol{x} \in \mathscr{D} \subset \mathbb{R}^p$. A training set $\mathscr{X}_i^{tr} = \{\boldsymbol{x}_j, j = 1, .., n_i\}$ is available for each of the classes ω_i. The standard pattern recognition approach to classification is to estimate the class conditional probabilities $p(\boldsymbol{x}|\omega_i)$, $i = 1,...,c$. By applying Bayes rule the posterior probabilities $p(\omega_i|\boldsymbol{x})$ can be computed using the class conditional probabilities and the class priors:

$$p(\omega_i|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\omega_i)p(\omega_i)}{\sum_{j=1}^{c} p(\boldsymbol{x}|\omega_j)p(\omega_j)} = \frac{p(\boldsymbol{x}|\omega_i)p(\omega_i)}{p(\boldsymbol{x})}. \tag{1}$$

In the standard rejection approach, the ambiguity reject (Chow, 1970), the objects $\boldsymbol{x}$ are rejected for which the maximum posterior probability $\max_i p(\omega_i|\boldsymbol{x})$ is below a threshold.

In real applications objects from other, novel classes may appear. This situation can be modeled by an extra reject (or outlier) class ω_0 that has a uniform distribution in the area $\mathscr{D}$. To distinguish this outlier class from the c known classes, one can put a threshold on the total data density of the known classes (Bishop, 2006). The total classifier with reject therefore becomes:

$$\hat{y} = \begin{cases} \omega_0 & p(\boldsymbol{x}) \leqslant \theta, \\ \omega_i & p(\omega_i|\boldsymbol{x}) > p(\omega_j|\boldsymbol{x}), \quad i \neq j \quad \text{and} \quad p(\boldsymbol{x}) > \theta. \end{cases} \tag{2}$$

This approach is suitable when a sufficiently large training sample is available for all of the classes and when the training sample is not contaminated by outliers. In this case $p(\boldsymbol{x}|\omega_i)$ can be estimated reliably by some model $\hat{p}(\boldsymbol{x}|\omega_i)$. A first problem may be that the different classes in the training data may be contaminated by different amounts of outliers. In that case a different rejection threshold θ_i per class (Fumera et al., 2000) has to be used. In this case a *density-based* one-class model for class ω_i is obtained:

$$\hat{y} = \begin{cases} \omega_0 & \hat{p}(\boldsymbol{x}|\omega_i) < \theta_i \\ \omega_i & \text{otherwise}. \end{cases} \tag{3}$$

Even when we know the class priors and we are using proper density models (as in (3)), we cannot just use the standard Bayes rule (Eq. (1)) for finding the most probably class. The Bayes rule does not incorporate the model thresholds θ_i, and these thresholds may vary significantly in value, especially when classes have a large spread. For classes with a large spread, the probability density values tend to be low, because the probability densities are normalized to integrate to one. On the other hand, classes that are very compact will have a much higher probability densities. When a single rejection threshold is chosen in the standard Bayes rule, most of the rejected objects will therefore come from the class with the highest spread, while (almost) none of the objects from a very compact class is rejected.

A second problem is that to apply Bayes rule (1) a good estimate of the class priors $p(\omega_i)$ should be available. When the training set reflects well what can be expected in the practical application, these priors can be simply obtained. For situations that new classes may appear, for instance because new types of diseases or new types of defects may appear in the objects that should be classified, it may be hard to find these priors.

A third significant problem for formulation (2) is that density estimation is a hard problem. For high dimensional feature spaces many training objects are required to avoid the curse of dimensionality (Duda et al., 2001). When a limited training set is available, approximations to the class densities have to be made, like k-means clustering centers, self-organizing maps, subspace models using PCA or hypersphere models inspired by the support vector machines (Tax, 2001). These methods often use a distance to prototypes or subspaces, and are therefore called *distance-based* class models. In contrast to the density models, the distance-based class models give a high output to the outliers:

$$\hat{y} = \begin{cases} \omega_0 & d_i(\boldsymbol{x}) > \theta_i \\ \omega_i & \text{otherwise}, \end{cases} \tag{4}$$

where $d_i(\boldsymbol{x})$ is the distance of object $\boldsymbol{x}$ to the model of class ω_i.

Although these distance-based class models (4) may describe a class better, they lack a common output scaling and it is not clear how they can be compared and combined. A similar problem appears when density-based models and distance-based models are combined, because one cannot directly compare (3) with (4); the first one increases while the second one decreases when one approaches a class.

To generalize formulation (2) to both density and distance-based class models, a normalization has to be defined. We define this normalization with two demands. The first demand is that the normalized output for a class is high for objects that come from that class. The second demand is that the decision boundaries of the different models between the outlier objects and their corresponding class objects are not changed.

Because each model characterizes the same outlier class with their threshold θ_i, these thresholds should coincide. On the other hand, each model characterizes a different 'target' class, and therefore these class outputs have to be compared to find the most probable output class. The exact construction of the normalization is explained in the next section.

3. Combination of class models

We propose to use the following transformation for the normalization the outputs of models (3) and (4). For the density-based models we chose to use a simple linear rescaling, for the distance-based models we have a possibly nonlinear transformation g:

$$\tilde{p}_i(\boldsymbol{x}) = \begin{cases} \frac{1}{Z_p}\hat{p}(\boldsymbol{x}|\omega_i) + p_0 & \text{density-based models}, \\ \frac{1}{Z_d}g(d_i(\boldsymbol{x})) + d_0 & \text{distance-based models}, \end{cases} \tag{5}$$

with the two free parameters Z_p, p_0 for the density models, and g, Z_d, d_0 for the distance models.

To remove the first free parameter, we use the assumption that all one-class models are assumed to model the *same* outlier distribution with their threshold θ_i. The rejection thresholds θ_i in (3) and (4) should therefore coincide for all classes. This removes one of the free parameters.

To fix the second free parameter two alternatives are possible; the first is based on the expected output for the outlier class data, the second is based on the expected output for the target class: