Adaptive Committees of Feature-Specific Classifiers for Image Classification

Tiziano Fagni, Fabrizio Falchi, and Fabrizio Sebastiani

Istituto di Scienza e Tecnologia dell'Informazione
Consiglio Nazionale delle Ricerche
Via Giuseppe Moruzzi 1 – 56124 Pisa, Italy
{*firstname.lastname*}@isti.cnr.it

Abstract. We present a system for image classification based on an adaptive committee of five classifiers, each specialized on classifying images based on a single MPEG-7 feature. We test four different ways to set up such a committee, and obtain important accuracy improvements with respect to a baseline in which a single classifier, working an all five features at the same time, is employed.

1 Introduction

We address *single-label* (also known as *multiclass*) *image classification*, i.e., the problem of classifying an image into exactly one from a predefined set of classes (see e.g., [1]). An automated classification system is normally specified by defining two essential components: (i) a scheme for internally representing the data items that are the objects of classification, and (ii) a learning device that takes as input the representations of training data and generates a classifier from them. We describe an image classification system that makes use not of a single representation, but of five different ones for the same data item, based on five different descriptors ("features") from the MPEG-7 standard that each analyze an image under a different point of view. As a learning device we use a "committee" of five classifiers, each based on the representation specific to a single MPEG-7 feature.

As a technique for generating the individual members of the classifier committee we use *distance-weighted k nearest neighbours* (see e.g., [2]). This method does not require a vectorial representation of data items to be defined, since it simply requires that, given two data items, a distance between them is defined. In the discussion that follows this will allow us to abstract away from the details of the representation specified by the MPEG-7 standard, and to simply specify our methods in terms of the distance functions between our data items, which have indeed been specified either by the MPEG-7 group itself or by related literature.

Since distance computation is so fundamental to our methods, we have also studied how to compute distances between data items efficiently, and have implemented an efficient system that uses metric data structures explicitly devised for "nearest neighbour search".

2 Adaptive, Feature-Specific Classifier Committees

Given a set of documents D and a predefined set of *classes* $C = \{c_1, \ldots, c_m\}$, *single-label document classification* (SLC) is the task of automatically building a function $\hat{\Phi}(d_i)$ (the *classifier*) that predicts, for any $d_i \in D$, the correct class $\Phi(c_j)$ to which d_i belongs. The image classifier $\hat{\Phi} : D \to C$ that we generate actually consists of a *classifier committee* (aka *classifier ensemble*), i.e., of a tuple $\hat{\Phi} = (\hat{\Phi}^1, \ldots, \hat{\Phi}^n)$ of classifiers, where each classifier $\hat{\Phi}^s$ is specialized in analyzing the image from the point of view of a single "feature" $f_s \in F$.

As the set F of image features we use a set of five visual "descriptors" as defined in the MPEG-7 standard (see e.g., [3]), each of them characterizing a particular visual aspect of the image: *Colour Layout* (CL – information about the spatial layout of colour images), *Colour Structure* (CS – colour content and its spatial arrangement), *Edge Histogram* (EH – the spatial distribution of five types of edges), *Homogeneous Texture* (HT – texture-related properties of the image), and *Scalable Colour* (SC – a colour histogram in the HSV colour space).

The "aggregate" classifier $\hat{\Phi}$ takes its final classification decision by combining the decisions returned by the feature-specific classifiers $\hat{\Phi}^s$ by means of an *adaptive* combination rule, i.e., a combination rule that pays particular attention to those $\hat{\Phi}^s$'s that are expected to perform more accurately on the particular image that needs to be classified. This is advantageous, since a feature could be more revealing than another for classifying a certain type of images; e.g., for correctly recognizing that an image belongs to class c' the *Homogeneous Texture* feature might be more important than *Colour Layout*, while the contrary might happen for class c''.

For implementing the classifier committee, i.e., for combining appropriately the outputs of the $\hat{\Phi}^s$'s, we experiment with four different techniques. We now describe these techniques, while in Section 2.1 we describe how to generate the individual members of these committees. For reasons of space the description of these techniques is necessarily concise, and only textual in nature; see the full paper [4] for their precise mathematical specifications.

The first technique we test is *dynamic classifier selection* (DCS) [5], which consists in adopting the decision of the feature-specific classifier $\hat{\Phi}^s$ that has performed best on the w training examples closest to the test document d_i. Here, w is a parameter and closeness is interpreted with respect to a (global, i.e., not feature-specific) measure of distance δ to be discussed more in detail in Section 3. In a nutshell, DCS is based on the intuition that similar documents are handled best by similar techniques, and that we should thus trust the classifier $\hat{\Phi}^s$ which performs best on documents similar to the one we need to classify.

The second technique we test is *weighted majority vote* (WMV), which is similar in spirit to the "adaptive classifier combination" technique of [5]. While DCS eventually trusts a single feature-specific classifier, WMV uses a weighted majority vote of the decisions of *all* the $\hat{\Phi}^s \in \hat{\Phi}$, with weights proportional to how well each $\hat{\Phi}^s$ performs on documents similar to the test document.

The third technique we test is *confidence-rated dynamic classifier selection* (CRDCS), a variant of DCS in which each decision of a feature-specific classifier is weighted by the *confidence* with which it has taken this decision. It is indeed true that that, given a test document d_i, our feature-specific classifiers $\hat{\Phi}^s$ return both a class $c_j \in C$ to which they believes d_i to belong *and* a numerical value $\nu(\hat{\Phi}^s, d_i)$ that represents the confidence that $\hat{\Phi}^s$ has in its decision (high values of ν correspond to high confidence); see Section 2.1 for details. The intuition behind the use of these confidence values is that a classifier that has made a correct decision with high confidence should be preferred to one which has made the same correct decision but with a lower degree of confidence.

The fourth technique we test, *confidence-rated weighted majority vote* (CR-WMV), stands to WMV as CRDCS stands to DCS; that is, it consists of a version of WMV in which each decision of a feature-specific classifier is weighted by the confidence with which it has taken this decision.

2.1 Generating the individual classifiers

Each base classifier $\hat{\Phi}^s$ (i.e., each member of the committees described above) is generated by means of the well-known *(single-label, distance-weighted) k nearest neighbours* (k-NN) technique. This technique consists in the following steps; for a test document d_i, (1) identify the set $\chi^k(d_i)$ of the k training examples closest to d_i, where closeness is to be computed according to a feature-specific distance measure $\delta_s(d', d'')$ and k is an integer parameter; (2) for each class $c_j \in C$, gather the evidence $q(d_i, c_j)$ in favour of c_j by summing the complements of the distances between d_i and the documents in $\chi^k(d_i)$ that belong to c_j; (3) pick the class that maximizes this evidence.

Standard forms of distance-weighted k-NN do not usually output a value of confidence in their decision. We naturally make up for this by defining the confidence in the decision taken as the evidence in favour of the chosen class minus the average evidence in favour of all the remaining classes.

3 Efficient implementation of nearest neighbour search by metric data structures

In order to speed up the computations of our classifiers we have focused on implementing efficiently *nearest neighbour search*, i.e., the operation of finding the k objects closest to a given target object, given a suitable notion of distance. The reason we have focused on speeding up this operation is that (i) it accounts for most of the computation involved in classifying objects through the k-NN feature-specific classifiers of Section 2.1, and (ii) it also accounts for most of the computation involved in combining feature-specific classifiers through each of the four methods of Section 2.

Efficient implementation of nearest neighbour search requires data structures that are explicitly devised for this task. To this end we have used an *M-tree* [6], a data structure explicitly devised for speeding up nearest neighbour search in *metric spaces*, i.e., sets in which a distance function is defined between their members that is a metric. We have been able to use an M-tree exactly because (i) as the five feature-specific distance functions δ_s used in the base classifiers, we have chosen the distance measures recommended by the MPEG group (see [3] for details), which are indeed metrics; and because (ii) as the global distance function δ used for assembling the committees we have chosen a linear combination of the previously mentioned five δ_s functions, which is by definition also a metric. As the linear combination weights w_s we have simply adopted the weights derived from the study presented in [7], i.e., $w(CL) = .007$, $w(CS) = .261$, $w(EH) = .348$, $w(HT) = .043$, $w(SC) = .174$.

4 Experiments

The dataset we have used (here called the Stone dataset) is a set of 2,597 photographs of stone slabs, subdivided under 37 classes representing different types of stone[1]. The dataset was randomly split into a training set, containing 780 examples, and a test set, consisting of the remaining 1,817 examples. For each photograph an internal representation in terms of MPEG-7 features was generated and stored into an M-tree. As a measure of effectiveness we have used *error rate* (noted E), i.e., the percentage of misclassified test documents.

As a baseline, we have use a "multi-feature" version of the distance-weighted k-NN technique of Section 2.1, i.e., one in which the distance function δ mentioned at the end of Section 3, and resulting from a linear combination of the five feature-specific δ_s functions, is used in place of δ_s. For completeness we also report five other baselines, obtained in a way similar to the one above but using in each a feature-specific distance function δ_s. In these baselines and in the experiments involving our adaptive classifiers the k parameter has been fixed to 30, since this value has proved the best choice in previous experiments involving the same technique [2]. The w parameter of the four adaptive committees has been set to 5, which is the value that had performed best on previous experiments we had run on a different dataset. In future, larger-scale experiments we plan to optimize these parameters more carefully by cross-validation.

The results of these preliminary experiments are reported in Table 1. We notice that all four committees (2nd row, 2nd to 5th cells) bring about a substantive reduction of error rate with respect to the baseline (2nd row, 1st cell). The best performer proves CRDCS, with a reduction in error rate of 39.7% with respect to the baseline. This is noteworthy, since both this method and the baseline use the same information, only combining it in different ways. The results also show that confidence-rated methods (CRDCS and CRWMV) are not uniformly superior to methods (DCS and WMV) which do not use confidence values.

The results also show that dynamic classifier selection methods (DCS and CRDCS) are definitely superior to weighted majority voting methods (WMV and CRWMV). This result might be explained by the fact that, out of five features, three (CS, CL, SC) are based on colour, and are thus not completely independent from each other; if, for a given test image, colour considerations are not relevant for picking the correct class, in WMV and CRWMV it may be different to ignore them anyway, since they are brought to bear three times in the linear combination. In this case, DCS and CRDCS are more capable of

[1] The dataset can be downloaded from (URL removed to preserve anonymity).

CL	CS	EH	HT	SC
0.479	0.318	0.479	0.410	0.419
Baseline	**DCS**	**CRDCS**	**WMV**	**CRWMV**
0.297	0.183 (-38.4%)	**0.179 (-39.7%)**	0.225 (-24.2%)	0.227 (-23.6%)

Table 1. Error rates of the techniques as tested on the Stone dataset; percentages indicate decrease in error rate with respect to the baseline. The first five results are relative to the five feature-specific baselines. **Boldface** indicates the best performer.

ignoring colour considerations, since they will likely entrust either the EH- or the HT-based classifier with taking the final classification decision.

The same result also seems to suggest that, for any image, there tends to be a single feature that alone is able to determine the correct class of the image, but this feature is not always the same, and sharply differs across categories. For instance, the SC classifier is the best performer, among the single-feature classifiers, on test images belonging to class GialloVeneziano ($E = .11$), where it largely outperforms the EH classifier ($E = .55$), but the contrary happens for class AntiqueBrown, where EH ($E = .01$) largely outperforms SC (.22). That no single feature alone is a solution for all situations is also witnessed by the fact that all single-feature classifiers (1st row) are, across the entire dataset, largely outperformed by both the baseline classifier and all the adaptive committees. This fact confirms that splitting the image representation into independent feature-specific representations on which feature-specific classifiers operate is a good idea.

References

1. Lu, D., Weng, Q.: A survey of image classification methods and techniques for improving classification performance. International Journal of Remote Sensing **28**(5) (2007) 823–870
2. Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of the 22nd ACM International Conference on Research and Development in Information Retrieval (SIGIR'99), Berkeley, US (1999) 42–49
3. Manjunath, B., Salembier, P., Sikora, T., eds.: Introduction to MPEG-7: Multimedia Content Description Interface. John Wiley & Sons, New York, US (2002)
4. Fagni, T., Falchi, F., Sebastiani, F.: Efficient adaptive committees of feature-specific classifiers for image classification. Technical report, Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche (2008)
5. Li, Y.H., Jain, A.K.: Classification of text documents. The Computer Journal **41**(8) (1998) 537–546
6. Ciaccia, P., Patella, M., Zezula, P.: M-tree: An efficient access method for similarity search in metric spaces. In: Proceedings of the 23rd International Conference on Very Large Data Bases (VLDB '97), Athens, GR (1997) 426–435
7. Amato, G., Falchi, F., Gennaro, C., Rabitti, F., Savino, P., Stanchev, P.: Improving image similarity search effectiveness in a multimedia content management system. In: Proceedings of the 10th International Workshop on Multimedia Information System (MIS'04), College Park, US (2004) 139–146