A Bag-of-Pages Approach to Unordered Multi-Page Document Classification

Albert Gordo
Computer Vision Center
Universitat Autònoma de Barcelona, Spain
agordo@cvc.uab.es

Florent Perronnin
Textual and Visual Pattern Analysis
Xerox Research Centre Europe, France
Florent.Perronnin@xerox.com

Abstract

We consider the problem of classifying documents containing multiple unordered pages. For this purpose, we propose a novel bag-of-pages document representation. To represent a document, one assigns every page to a prototype in a codebook of pages. This leads to a histogram representation which can then be fed to any discriminative classifier. We also consider several refinements over this initial approach. We show on two challenging datasets that the proposed approach significantly outperforms a baseline system.

1. Introduction

We consider the problem of classifying documents containing multiple unordered pages. This problem is encountered for instance in the mail room of large corporations where the incoming mail has to be forwarded to the correct workflow (each workflow corresponds to a document class). Each mail typically contains multiple pages (*e.g.* handwritten or typed letters, forms, copies of ID cards or proof of residence) and the order of the pages in the mail is generally non-informative.

There is an abundant literature on *single-page* document classification. This includes methods based on textual features in the case where the documents are OCR-able as well as methods based on the document appearance. In the later case, while most methods require the explicit extraction of the document structure (see *e.g.* [10, 3, 5]) it is possible to circumvent this intermediary step as shown for instance in [9]. Few works also exist on multi-page document classification in the case where the pages are ordered (see *e.g.* [4]). We are not aware of any work on unordered multi-page document classification.

In what follows, we assume that we have a training set of labeled documents. Note that the labels are given *at the document level* but that *we do not have page-level annotations available*. We assume that each page can be described by a feature vector. In this article, we focus on appearance features. However, visual features may be replaced by or combined with other features such as textual features.

The remainder of the article is organized as follows. In Section 2, we describe a baseline system to solve the considered problem and explain its shortcomings. In Section 3, we then propose a novel representation of documents as bags-of-pages. We first learn *page-level clusters* on a training set. To represent a new document, each page is assigned to a cluster and one counts the proportion of pages assigned to each cluster. This histogram representation can then be classified with any discriminative classifier. We also propose several refinements over the original approach: (a) supervised learning of clusters, (b) soft-assignment of pages to clusters and (c) going beyond simple counting. In Section 4, we show on two challenging datasets that the proposed approach significantly outperforms the baseline.

2. Baseline System

Our baseline system is based in the following assumption: *pages belonging to different categories are visually dissimilar*. Under this assumption, we can turn the document classification problem into a page-classification problem.

At training time, we learn page-level classifiers:

1. Extract page-level representations for each page of each training document.
2. Propagate the document-level labels to the individual pages.
3. Learn one page-level classifier per document category using the features of step 1 and the labels of step 2.

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.473

IEEE computer society

In practice, we use Sparse Logistic Regression (SLR) for classification [7].

At runtime, a document is classified as follows:

1. Extract one feature vector per page.
2. Compute one score per page per class.
3. Aggregate the page-level scores into document-level scores for each document class.

The scores we compute at step 2 are the class posteriors. As for step 3, we experimented with different fusion schemes and obtained the best results with a simple summation of the per-page scores.

In most of the scenarios we have encountered, the assumption underlying our baseline system is not verified, *i.e.* two pages may be very similar (from a visual and / or textual perspective) and still belong to documents with different category labels. For instance, the copy of an ID card may be attached to different requests which have to be processed by different workflows.

3. A Bag-of-Pages Approach

Let us assume the existence of page-level categories which are potentially shared across document categories. On the datasets we considered, typical page-level categories could be: “handwritten letter”,“typed letter”, “form X”, “ID copy”, “phone bill”, *etc*. If we had training material with page-level labels, then we could learn page classifiers and then represent an image as a histogram of the number of occurrences of each page category. For instance, a 3 pages documents could be described as: 1 “handwritten letter” + 1 “subscription form” + 1 “phone bill”. This document-level representation might be more amenable to classification than the original representation.

However, this approach is unpractical for two reasons. First, identifying manually page-level categories is a non-trivial task. For instance, should we put a driver’s license and a passport in the same “ID” category or in different categories? Second, even if page-level categories were well-identified, one would need to gather labeled training material for each page category which is a slow and tedious process.

On the other hand, we can try to discover such page categories automatically through an unsupervised process. The next Section describes a plain vanilla implementation of our system. We then describe its limitations as well as various refinements.

3.1 Plain vanilla version

The proposed approach is inspired by the bag-of-patches approach to image classification. An image can be described as an unordered set of local feature vectors whose cardinality might vary from one image to another [2]. Similarly, an unordered multi-page document can be considered as a bag-of-pages: each individual page is described by a feature vector.

The training of the proposed system proceeds as follows:

1. Extract one feature vector for each page of each training document.
2. Perform page clustering on these representations.
3. For each training document, compute a histogram representation by assigning each page-level vector to its nearest cluster and by counting the proportion of pages assigned to each cluster.
4. Learn document classifiers on the document-level histograms.

In our plain vanilla approach, clustering of the page-level feature vectors is performed using K-means and a Euclidean distance. For a fair comparison with the baseline, we used SLR for classification.

At runtime, a document is classified as follows:

1. Extract one feature vector per page of the document.
2. Compute a histogram representation of the document (same as step 3 of training).
3. Classify the document-level representation.

3.2 Limitations and improvements

Ideally, each cluster should correspond to a page category and vice-versa. As there is no ideal clustering algorithm we will typically face the two following issues: pages corresponding to different page categories may be assigned to the same cluster and pages corresponding to the same page category may be assigned to different clusters. There is a third problem which is inherent to quantization: quantization is a lossy process by nature and therefore discriminative information may be lost.

Supervised learning of page clusters. The first issue, pages of different categories being assigned to the same cluster, is problematic since documents which consist of different page categories may then be represented by the same histogram. We note that this is not an issue if the two documents belong to the same document category. It is only prejudicial if the two documents belong to two different document categories. To maximize the chances that two documents which belong to two different document categories have different histogram signatures, we can cluster the pages in a supervised manner: we perform K-means on the pages of each category separately and then put together the clusters of all categories in a single cluster set.

Soft assignment of pages to clusters. The second issue, pages corresponding to the same page category assigned to different clusters, is also problematic. Two documents which consist of the same page categories may then be represented by different histograms. To alleviate this problem, we can assign pages to multiple clusters in a probabilistic manner rather than to a single cluster in a hard manner. The most principled way to perform soft assignment is to assume that the page feature vectors are drawn from a probabilistic model (in our case, a Gaussian mixture model or GMM). The K-means clustering is therefore replaced by the GMM learning using maximum likelihood estimation (MLE) [1]. The computation of the soft-assignments is based on the posterior probabilities of feature vectors to the Gaussian components. Note that the soft-assignment can also enable us to cope with the fuzzy nature of page categories.

Beyond counting. One way to encode more information than the simple counting of cluster occurrences is to use the Fisher kernel framework [6]. Let $X = \{x_t, t = 1 \ldots T\}$ be an unordered set of vectors. In our case, X is a document, T is the number of pages of the document and x_t is the feature vector representing page t. If we assume the existence of a probabilistic generation model of pages with distribution p (a GMM in our case – c.f. the previous paragraph) whose parameters are collectively denoted λ , then X can be described by the following gradient vector:

$$\frac{1}{T} \nabla_\lambda \log p(X|\lambda). \tag{1}$$

It was shown in [8] that in the case of a mixture model, the Fisher representation does not only encode the proportion of features assigned to each component (gradient with respect to the mixture weights) but also the location and spread of features in the soft-regions defined by each component (gradient with respect to the mean and variance parameters respectively). Please, refer to [8] for more details.

4. Experiments

In this Section, we will first describe our experimental setup and then provide results.

4.1 Experimental setup

Since we were not aware of any public dataset of unordered multi-page documents, we experimented on two in-house datasets:

- Our "small" dataset contains 2,060 documents and 10,097 pages divided into 6 document categories.
- Our "large" dataset contains 19,178 documents and 57,530 pages divided into 19 document categories.

In both cases, half of the documents were used for training and half for testing. The classification accuracy was measured as the percentage of documents assigned to the correct category. The experiments were repeated with 5 different train / test partitions, and the reported accuracies are the average of the 5 runs.

Document pages are described using multi-scale run-length histograms, yielding 1,680-dimensional feature vectors. Then, the dimensionality of the feature vectors was reduced to 840 through Principal Component Analysis (PCA).

We evaluated the baseline system as well 6 flavors of our bag-of-pages approach:

- Hard histograms based both on unsupervised and supervised clusters learned with K-means.
- Soft histograms based both on unsupervised and supervised GMMs learned with MLE.
- Fisher vectors based both on unsupervised and supervised GMMs learned with MLE.

As already mentioned, for classification we always used SLR [7].

4.2 Results

Results are presented on Figure 1. We can draw the following conclusions:

- Supervised learning of clusters always improves accuracy over unsupervised learning in the case of histogram representations.
- As expected, soft-assignment of pages to clusters always yields to significant improvements, especially when the clusters are learned in a supervised manner.
- The Fisher kernel framework always improves over histogram representations. This is not surprising given that the Fisher representation goes beyond counting and therefore contains more information than histogram representations. Note that,

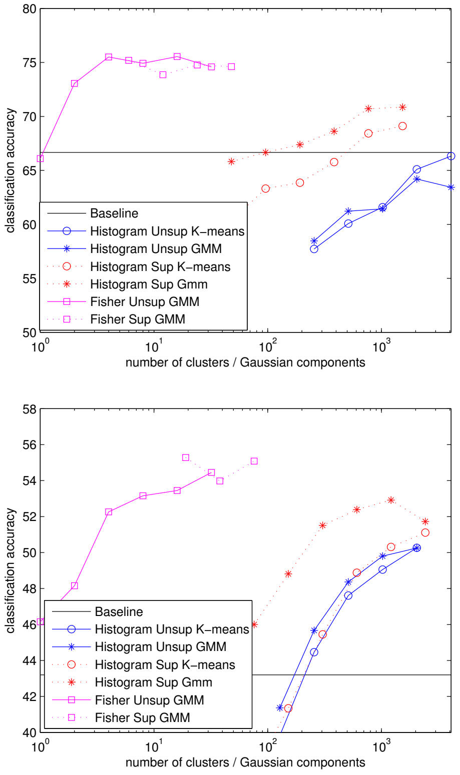

Figure 1. Accuracy of the different systems as a function of the number of clusters / Gaussian components. Top: "small" dataset. Bottom: "large" dataset.

in the case of the Fisher representation, one needs a much smaller number of Gaussians to obtain top accuracy compared to histogram representations.

- Supervised learning does not help in the case of the Fisher representation. This is consistent with the findings of [6, 8]: the accuracy of the Fisher kernel seems to have a weak dependence on the estimation quality of the underlying generative model.

On both datasets, the performance is significantly improved with respect to the baseline system: from 66.7% up to 75.5% on the "small" dataset and from 43.2% up to 54.4% on the "large" dataset (using unsupervised Fisher representations in both cases). It is not surprising to observe that the improvement is more important for a larger number of categories (and therefore a more difficult problem). Indeed, as the number of document categories increases, the assumption underlying the baseline system (different document categories contain visually different pages) becomes more and more incorrect.

5. Conclusion

In this paper we introduced a novel bag-of-pages framework for unordered multi-page document classification. We first proposed a simple version of the system and then tried to overcome some of its shortcomings by using supervised learning, GMM clustering and the Fisher framework. We evaluated this approach and its variations on two datasets and compared it with a baseline method. On both datasets, the proposed approach was shown to improve significantly over the baseline results. The best results were obtained using the Fisher kernel framework (*e.g.* going beyond the simple counting of pages assigned to each cluster).

Future work will focus on adding textual information to the page representations as some document categories are visually very similar, and the main discriminant information is their textual content. Also, while the proposed representation was introduced for classification purposes, we intend to experiment with it on other tasks, *e.g.* document retrieval and clustering.

References

[1] J. A. Bilmes. A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Technical report, 1998.

[2] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *In Workshop on Statistical Learning in Computer Vision, ECCV*, 2004.

[3] M. Diligenti, P. Frasconi, and M. Gori. Hidden tree markov models for document image classification. *IEEE PAMI*, 25(4), 2003.

[4] P. Frasconi, G. Soda, and A. Vullo. Hidden markov models for text categorization in multi-page documents. In *Jour. of Intell. Information Systems*, page 2002.

[5] J. Hu, R. Kashi, and G. Wilfong. Document image layout comparison and classification. In *ICDAR*, 1999.

[6] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, 1998.

[7] B. Krishnapuram, L. Carin, M. Figueiredo, and A. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *IEEE PAMI*, 27(6), 2005.

[8] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007.

[9] P. Sarkar. Image classification: classifying distributions of visual features. In *ICPR*, 2006.

[10] C. Shin, D. Doermann, and A. Rosenfeld. Classification of document pages using structure-based features. *IJDAR*, 3(4), 2001.