DOCUMENT IMAGE AND ZONE CLASSIFICATION THROUGH INCREMENTAL LEARNING

Mohamed-Rafik Bouguelia, Yolande Belaïd, Abdel Belaïd

DOCUMENT IMAGE AND ZONE CLASSIFICATION THROUGH INCREMENTAL LEARNING

Mohamed-Rafik Bouguelia, Yolande Belaïd, Abdel Belaïd

Université de Lorraine, LORIA
UMR 7503, Vandoeuvre-les-Nancy, F-**54506**, France

ABSTRACT

We present an incremental learning method for document image and zone classification. We consider an industrial context where the system faces a large variability of digitized administrative documents that become available progressively over time. Each new incoming document is segmented into physical regions (zones) which are classified according to a zone-model. We represent the document by means of its classified zones and we classify the document according to a document-model. The classification relies on a reject utility in order to reject ambiguous zones or documents. Models are updated by incrementally learning each new document and its extracted zones. We validate the method on real administrative document images and we achieve a recognition rate of more than 92%.

Index Terms— Document Image Analysis, Incremental Learning, Zone Classification, Document Classification

1. INTRODUCTION

Today, companies deal with many heterogeneous documents that are daily digitized and must be processed quickly and efficiently. One important problem in document image analysis systems is the identification of the content type of different zones that constitute the document. Different zones may be obtained by a physical layout analysis system or by using some segmentation techniques [1, 2, 3, 4]. We focus in this paper on the identification of the content type of the detected zones through the classification of these zones into different classes such as logo, signature, table, handwritten annotation, stamp etc. Zone classification is useful because it allows document image analysis systems to use content-specific algorithms which may improve their results. For instance, if we know that a given zone inside a document represents a table, then we can use some specialised methods for the extraction of informations from tables.

Beside document zone classification, document classification allows automatic identification of documents type, which is important for document routing to topic-specific processing and information extraction mechanisms, or routing document images directly to humans or service departments that are specialised in their management [5]. The document types that we deal with in this context consists of bank checks, medical receipts, invoices, prescriptions, etc. They are very diverse and of a variable quality, which makes them more difficult to be processed efficiently.

More importantly, a major difficulty is that most of state of the art methods for document image and zone classification like [6, 7, 8] operate on two phases: the learning phase where a model is learned and a classification phase where new data is classified according to that model. Consequently, these methods perform in a batch mode where the learning phase need the whole training dataset to be available beforehand. However, this requirement is inconvenient in an industrial and real-world application[1] for two reasons: (1) Companies usually deal with a massively and continuously arriving document flow where the documents become available progressively over time. Therefore, it would be important to consider an incremental learning configuration where each new document can be visited only once and used to update the learned model incrementally as soon as it is available. (2) In state of the art methods we need to manually build a large enough set of annotated documents and zones for the learning to be efficient. However, obtaining sufficiently numerous labelled training documents and zones is costly and time-consuming. Therefore, we let our method chose which data is more convenient for labelling and reject it to get its true class-label from a human annotator. There are many recent related works on incremental learning, like the ones surveyed in [9], however, most of these methods do not consider rejecting ambiguous data during the incremental learning process.

This paper deals mainly with document image topic identification and document zone type identification, by proposing a learning method which can be trained incrementally from a continuously arriving stream of documents so that it do not need the whole training documents and zones to be labelled and available beforehand.

This paper is organized as follows. In section 2, we describe the general scheme of the method. In section 3, we

[1] Based on communication and direct collaboration on real-world industrial problem with the ITESOFT http://www.itesoft.com

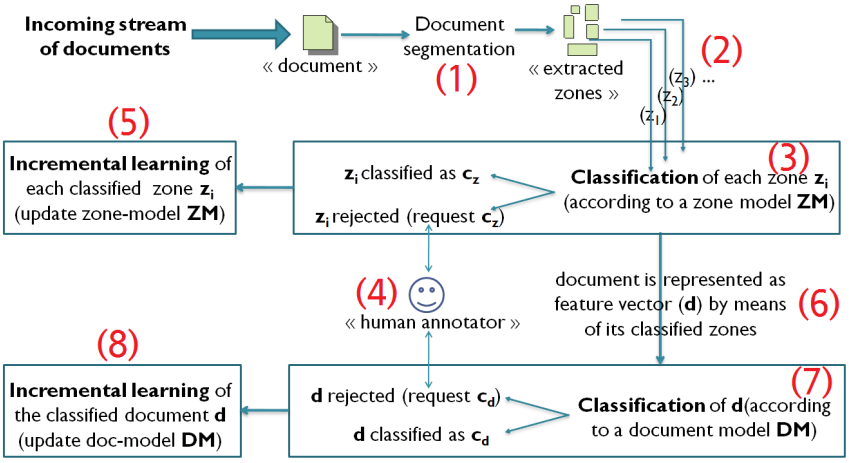

Fig. 1: General scheme

briefly describe the document image segmentation using an existing method. In section 4, we present our proposed incremental learning and classification method. In section 5, we present our experimental evaluation on a real digitized administrative document dataset. Finally, we give the conclusion and we present some perspectives on this work in section 6.

2. GENERAL SCHEME

The general scheme can be expressed according to Fig.1. Each new incoming document from the stream, is segmented into zones (Fig.1(1)). This is done by firstly analysing the document using an OCR and regrouping the extracted words into lines then into paragraphs (i.e. printed-text zones). Printed-text zones are then removed from the original document image and the remaining zones are extracted by regrouping connected components according to their distance and size. The segmentation process is briefly explained in section 3. Each obtained zone is represented as a feature-vector z_i (Fig.1(2)) by applying some simple morphological features like run-lengths according to [10], bilevel co-occurence according to [11] and connected components (size and density). Each zone z_i is then classified into a class c_z (e.g. logo, table, handwritten annotation etc.) according to a zone-model (ZM) using the method in section 4.1 (Fig.1(3)). The classification method relies on a reject utility in order to reject an ambiguous zone (i.e. that is uncertain, or not sure how to classify it) and ask a human annotator for its true class-label c_z (Fig.1(4)). The classified zone (z_i, c_z) is then given to the incremental learning process which is described in section 4.2, in order to update and improve the zone-model (Fig.1(5)). Simultaneously, the original document image is represented by means of its classified zones as a feature-vector containing the number of occurrences of each zone type and its size, combined with the number of times each word occurs in the printed text zones (Fig.1(6)). It is then classified and learned using the same method as for zones, but this time according to a document-model (Fig.1(7, 8)).

A document-model (respectively zone-model) is represented as a set of document-representatives (respectively zone-representatives) which are feature-vectors that are continuously maintained and updated by the incremental learning algorithm (section 4.2).

3. DOCUMENT SEGMENTATION

This step allows to segment the document into physical regions, we use an existing method which was already experimented in our team [4]. In the following, there are some default threshold values for the segmentation parameters $\text{seg\_param} = \{s_1, s_2, \dots\}$ which come from the experimental study in [4] of the dispersion of pixels in the documents, in order to combine connected components as good as possible while reducing noise. However, a manual adjustment of these parameters may lead to a better segmentation for some specific documents.

Printed-text zones: The OCR is able to recognize characters, words and lines of a document and propose a segmentation. However, errors occur when the document is complex and do not contain only printed-text. For this purpose, lines are reconstructed as following: let $\bar{h}$ be the average hight of each word (in pixels). Words on the same horizontal line shifted with no more than $\bar{h} \times s_1$ (e.g. $s_1 = 15\%$) and having a spacing of less than $\bar{h} \times s_2$ (e.g. $s_2 = 2$ pixels), are regrouped. To regroup lines into paragraphs, their vertical spacing should not exceed $\bar{h} \times s_3$ (e.g. $s_3 = 2$ pixels) and their left margin alignment should not exceed $\bar{h} \times s_4$ (e.g. $s_4 = 8$ pixels). Furthermore, text-printed results may be validated using a dictionary which contains words that are mostly present in the documents and some regular expressions that allow to recognize phone numbers, addresses, politeness for-

mulas and dates, to indicate the presence of printed-text which was correctly recognized.

Non-printed-text zones: The segmentation of non-printed-text zones is done by regrouping the remaining connected components (after removing printed-text zones from the original document image). Let m be the hight of the image in pixels. If the distance between two connected components is less than $m \times s_5$ (e.g. $s_5 = 4\%$), then the two components are regrouped.

4. INCREMENTAL LEARNING AND CLASSIFICATION OF ZONES AND DOCUMENTS

The same classification and learning algorithms are used for both zones and documents; thus, for a simplification matter, in the following explanation we use the notation x to refer to a data-point as a feature-vector which represents a zone z_i or a document d_j, we use M to refer to a model (ZM or DM), $y \in M$ refers to a data-representative (zone-representative or document-representative), c_y refers to the class-label of the data-representative y.

We initially get a small number of labelled documents and labelled zones extracted from them. We use these data to initialize the two models.

4.1. Classification with reject utility

For each new document or zone x, we use K-Nearest Neighbours method [12] and we derive a probability of belonging to its two most probable classes.

Let $\mathrm{KNN}(x) = \{(y_1, c_{y_1}), ..., (y_K, c_{y_K})\}$ be the K-nearest data-representatives selected from M, sorted in ascending order according to their Euclidean distance to x. Let $P(c|x)$ the probability that the data-point x belongs to the class c. It is determined as

$$P(c|x) = \frac{\sum_{(y_i, c_{y_i}) \in \mathrm{KNN}(x)} f(y_i, c_{y_i})}{K} \quad (1)$$

where

$$f(y_i, c_{y_i}) = \begin{cases} 1 & \text{if } c_{y_i} = c \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Let $c_1 = \underset{c}{\operatorname{argmax}}\, P(c|x)$ and $c_2 = \underset{c \neq c_1}{\operatorname{argmax}}\, P(c|x)$, i.e. c_1 and c_2 are respectively the first and the second most probable classes given the data-point x, such that $P(c_1|x) \geq P(c_2|x)$. If the probability of a data-point x belonging to its most probable class c_1 is close to the probability of belonging to its second most probable class c_2 (i.e. $P(c_1|x) - P(c_2|x)$ is small), then we say that x is ambiguous according to the current model M and should consequently be rejected.

To decide if a new document or zone x should be rejected, we can then define a small probability threshold value δ and reject x if $P(c_1|x) - P(c_2|x) < \delta$. If it is rejected, then its true class-label is queried from a human annotator, because knowing the true class-label of such data-point would be useful for M (and for the learning algorithm) to better discriminate between these classes. Otherwise, the document x is classified as c_1 (its most probable predicted class).

4.2. Incremental learning

Let x be a new data-point and c its queried or predicted class-label. Let y be the nearest data-representative from x (distance(x, y) is the smallest one).

1. if x is *far enough* from y then:
   $M \leftarrow M \cup \{(y_{new}, c) | y_{new} = x\}$, i.e., a new data-representative y_{new} labelled with c is generated based on x.

2. if x is *close enough* to y then:

   - if $c = c_y$:
     we say that x is assigned to y and we update y:
     $y \leftarrow y + \epsilon \times (x - y)$, i.e., updating the feature vector y to be *less* distant from x (i.e. moving y towards x by a learning rate ϵ, $0 < \epsilon \ll 1$).
   - if $c \neq c_y$:
     $y \leftarrow y - \epsilon \times (x - y)$, i.e., updating the feature vector y to be *more* distant from x (i.e. moving y far away from x).

We consider x to be *far enough* (respectively *close enough*) from a data-representative y, if the distance between x and y is higher (respectively smaller) than a distance threshold T_y. The threshold T_y of a data-representative y, depends on a local Gaussian distribution of the distances to data-points previously assigned to y (for which y was the nearest data-representative). The further away the new data-point x is far from its nearest data-representative y, the more likely that x should not be assigned to y (and should become itself a new data-representative); this is basically why we use a local Gaussian distribution of the distances around each data-representative y.

Let $\bar{d}_y$ be the mean distance from data-representative y to its previously assigned data-points, and σ the corresponding standard deviation. Let d be a random variable distributed according to the Gaussian distribution of $\mathrm{mean} = \bar{d}_y$ and $\mathrm{variance} = \sigma^2$. The threshold T_y is defined according to formula 3 as the distance value T which is determined such that the probability $Pr_y(d > T)$ is low. This is the probability that a random distance d distributed according to $\mathrm{Gaussian}(\bar{d}_y, \sigma^2)$ is higher than T. However, $\bar{d}_y$ and σ can be computed if at least two data-points are assigned to y; in case where y has less than 2 assigned data-points, we

consider the threshold T_y as the distance from y to its nearest data-representative.

$$T_y = \begin{cases} T, \text{where } \Pr_y(d > T) = P_{low} & \text{if } n_y \geq 2 \\ \min\limits_{\tilde{y}} \text{distance}(y, \tilde{y}) & \text{otherwise} \end{cases} \quad (3)$$

where n_y is the number of data-points assigned to y, and P_{low} is a parameter representing a low probability value (e.g. $P_{low} = 0.05$).

Note that we do not need to save data-points (zones or documents) that are already seen, in order to compute this threshold. It is incrementally computed each time a new data-point comes, by updating some information associated to each data-representative (e.g. number of data-points n_y assigned to data-representative y, the sum of their distances to this data-representative, etc.). The threshold T_y depends only on the parameter P_{low} and evolves dynamically according to new data.

5. EXPERIMENTS

We test the proposed method on a dataset provided by ITESOFT company, which consists of 597 heterogeneous administrative documents of different types (16 classes) and resulting in 1117 zones of 5 classes (handwritten annotations, tables, stamps, signatures and logos). The documents and zones are represented by feature-vectors in a 637 and 101 dimensional space respectively. The models are initially initialized using only 20 labelled documents and labelled zones extracted from them. The system is then trained incrementally by considering documents one by one. $\sim 33\%$ of documents and zones are used for recognition (testing).

Method (labels %)	Recognition %	Recall %	Precision %
Zones dataset			
Ours (33.953%)	**92.546**	**79.500**	**81.428**
KNN (33.953%)	87.267	78.772	78.494
KNN (100%)	90.993	78.573	80.286
Documents dataset			
Ours (19.598%)	**95.979**	**94.848**	**88.763**
KNN (19.598%)	74.371	60.425	70.973
KNN (100%)	95.477	94.757	88.531

Table 1: Validation results

The obtained results are shown in Table 1. The number of true class-labels that were queried from a human annotator by our method during learning (as shown in section 4.1) is 33.9535% of the total training set. Since the proposed method uses KNN in section 4.1, we also compare the results to KNN in two cases: (1) using the same number of labels as the one obtained by our method (i.e. by labelling 33.9535% of zones chosen randomly from the training set), and (2) using the whole labels (i.e. by labelling all the zones of the training set). Results in Table 1 show that the proposed method achieved the best performances for both zones and documents, in terms of recognition rate, recall and precision.

δ	Reject %	Error %	Recognition %
0.01	1.863	6.521	91.614
0.05	5.590	5.900	88.509
0.10	12.422	2.484	85.093

Table 2: Error rate optimization for the zones dataset with variable values of the reject parameter δ during test

Table 2 shows how the error rate decreases with variable values of parameter δ (used for rejection) during testing, despite the possible decrease in the recognition rate. Indeed, in an industrial context, we may prefer to reject uncertain data, because we give more importance to lowering the error rate than to increasing the recognition rate, since doing an error is more costly.

6. CONCLUSION AND FUTURE WORK

This paper presents a general scheme and incremental learning method for document image and zone classification, which incrementally processes documents from a continuously arriving stream and can then perform a long-life learning. Experiments on a real digitized administrative documents show that a good classification performance is achieved for both zones and documents, while requiring few data to be manually labelled for learning (by querying the true class-label of a new data only if it is not sure how to classify it according to the current model). This makes the method convenient for an industrial real-world application where documents become available progressively over time.

Nonetheless, further work still needs to be done. For example, parameter P_{low} is involved in computing the threshold T_y, which determines when the incremental learning consider a new document or zone to be close or far enough from its nearest document or zone representative y (respectively zone-representative). The value of this parameter is externally set by a user and is data-dependent. Future work will be devoted to automate the choice of such parameter. Another direction for future work is to reuse a previous segmentation experience (of the previously classified documents) in order to segment more efficiently a new incoming document. That is, reusing good parameter values that were used to segment well a previously classified document in order to segment more efficiently the new similar documents. Indeed, let y be a document-representative from the document-model DM. If we associate to y the "good" segmentation parameters seg_param that were used to segment the documents assigned to y, then we can reuse these parameter values in order to segment more efficiently the new document x if it is *close enough* to y (distance$(x, y) < T_y$). In the end, we plan to integrate the proposed method as a first step in a case-based reasoning system for document image analysis [13].

7. REFERENCES

[1] A. Antonacopoulos, S. Pletschacher, D. Bridson, and C. Papadopoulos, "Page segmentation competition," *ICDAR*, pp. 1370–1374, 2009.

[2] L. O'Gorman, "The document spectrum for page layout analysis," *TPAMI*, pp. 1162–1173, 1993.

[3] G. Nagy and S. Seth, "Hierarchical representation of optically scanned documents," *ICPR*, pp. 347–349, 1984.

[4] Jean-Marc Vauthier and Abdel Belaid, "Segmentation et classification des zones d'une page de document," *CIFED*, pp. 1–16, 2012.

[5] Christoph Goller, Joachim Lning, Thilo Will, and Werner Wolff, "Automatic document classification - a thorough evaluation of various methods," *ISI*, pp. 145–162, 2000.

[6] Daniel Keysers, Faisal Shafait, and Thomas M. Breuel, "Document image zone classification - a simple high-performance approach," *VISAPP*, pp. 44–51, 2007.

[7] Wael Abd-Almageed, Mudit Agrawal, Wontaek Seo, and David S. Doermann, "Document-zone classification using partial least squares and hybrid classifiers," *ICPR*, pp. 1–4, 2008.

[8] Nawei Chen and Dorothea Blostein, "A survey of document image classification: problem statement, classifier architecture and performance evaluation," *IJDAR*, pp. 1–16, 2007.

[9] Joshi Prachi and Parag Kulkarni, "Incremental learning: Areas and methods - a survey," *IJDKP*, 2012.

[10] Y. Wang, I. Phillips, and R. Haralick, "Document zone content classification and its performance evaluation," *Pattern Recognition*, pp. 57–73, 2006.

[11] Y. Zheng, "Machine printed text and handwriting identification in noisy document images," *TPAMI*, pp. 337–353, 2004.

[12] Liangxiao Jiang, Zhihua Cai, Dianhong Wang, and Siwei Jiang, "Survey of improving k-nearest-neighbor for classification," *FSKD*, pp. 679–683, 2007.

[13] Abdel Belaid, Vincent D'Andecy, Hatem Hamza, and Yolande Belaid, "Administrative document analysis and structure," *Learning Structure and Schemas from Documents*, pp. 51–71, 2011.