An Information Extraction model for unconstrained handwritten documents

Simon Thomas, Clément Chatelain, Laurent Heutte, Thierry Paquet
Université de Rouen, LITIS EA 4108, BP 12 76801-Saint-Etienne du Rouvray, FRANCE
{*FirstName.LastName*}*@univ-rouen.fr*

Abstract

In this paper, a new information extraction system by statistical shallow parsing in unconstrained handwritten documents is introduced. Unlike classical approaches found in the literature as keyword spotting or full document recognition, our approch relies on a strong and powerful global handwriting model. A entire text line is considered as an indivisible entity and is modeled with Hidden Markov Models. In this way, text line shallow parsing allows fast extraction of the relevant information in any document while rejecting at the same time irrelevant information. First results are promising and show the interest of the approach.

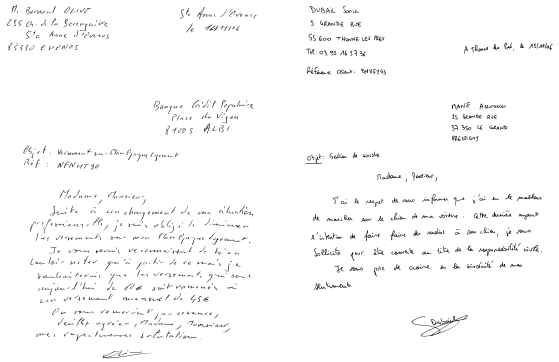

Figure 1. Two incoming letters from RIMES database [5].

1. Introduction

Since the early 2000, the processing of incoming mail is an active research area. Administrations and large businesses have to deal with a large number of handwritten documents (cf. figure 1). The automatic processing of handwritten incoming mail would generate high gains in efficiency and productivity but the automatic reading of an unconstrained handwritten document is still a very challenging task due to its complexity (free layout, open vocabulary, very large language model, inter-writer variability ...). Thus, more specific approaches based on the research of a particular information should be used. For instance, it can be easier to only seek the relevant information in a letter. Main literature approaches in this field can be classified regarding the nature of their input.

On the one hand, systems taking a word image as input like in [2] can be considered. First a document image is segmented into lines then into words. A distance measure between the input image and every image in the document is computed. A threshold allows one to decide whether an image corresponds to the input or not. These systems are often mono-writer and need a lot of samples for different queries. On the other hand, systems questioned with textual queries can be considered as keyword spotting [10, 4] or information extraction based on full recognition [7, 11]. While the first ones try to isolate word images and accept or reject them regarding their normalized recognition scores, the second kind of system aims at fully recognizing a letter and then filtering information sought. It seems irrelevant to try to segment a priori lines into words without recognition like in [10, 4] since it yields irreversible segmentation errors. Moreover, using a threshold to reject or not a hypothesis is very data-dependent [2]. Likewise, full recognition of a document for the only purpose of extracting a specific content seems to be too rigid and heavy [7, 11]. Considering entire text lines and trying to model them efficiently could therefore be a good alternative.

In this paper, an innovating information extraction system is introduced, based on a text line model able to handle relevant (words of a lexicon) and irrelevant (everything else) information. This paper is organized as follows. In section 2, the text line model is introduced. Implementation issues are described in section 3. We present in section 4 the experimental results on a French incoming mail database [5]. Conclusion and

future works are drawn in section 5.

2. Text line model

In an information extraction strategy, the content is viewed as a mixture of two types of information: the relevant one and the irrelevant one. In order to provide the best handwriting model, all the a priori knowlegde i.e. character models, language model, proportion of relevant information has to be gathered. In this section, we present our global text line model able to describe both relevant and irrelevant information for an application in keyword extraction.

For that purpose, our modeling choices are text line oriented. HMMs (*Hidden Markov Models*), recognized as one of the most interesting tools in sequence modeling [9] are chosen for this purpose. They are particularly used for the recognition of handwritten words [6] or sentences [11]. Given the keyword extraction problem, two types of information are to be discriminated in a line of text:

(i) Keywords belonging to a lexicon (reasonably sized). They are modeled by the concatenation of their HMM characters.

(ii) Irrelevant information made of Out-Of-Vocabulary (OOV) words, numerical information, punctuation, spaces and noise, all represented by a shallow parsing model that consists of a language model whose transitions are learned on a training set.

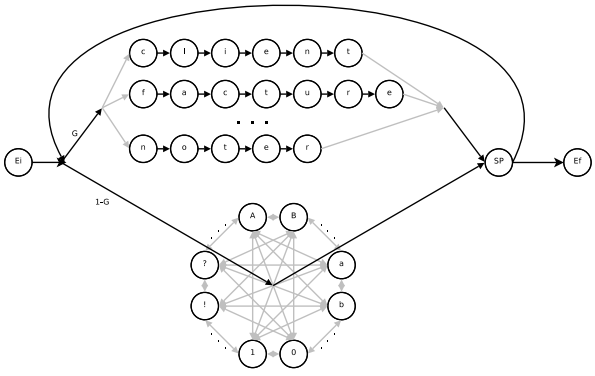

Figure 2. Global line model

The global line model is represented in figure 2. It highlights a competition between the two types of information modeled: on the one hand keywords, on the other hand our shallow parsing model regrouping uppercase and lowercase letters, digits, punctuation and spaces whose purpose is to represent everything but keywords. The hyperparameter G can be seen as a controller of the system behaviour. Also, it can be viewed as the proportion of relevant information that could appear within a line and is therefore dependent on the lexicon size. Thus, a line can be seen as a succession of keywords and irrelevant information separated by spaces.

Learning the model

Given the global line model, two kinds of information have to be learned: character models and transitions between them. This is done on a part of the database devoted to learning. The elementary entities of a line are the character models (HMMs) and the space model. In order to learn all of them in an efficient way, the well known Baum-Welch algorithm is used [9] on annotated lines of text including spaces. The main advantage of this embedded learning is that every instance of a given character is considered to learn its corresponding HMM model. As the learning is done on manually annotated lines, even the HMM space model is learned in an embedded way. The second item to be learned is the transitions between characters within the global model (cf. figure 2). These transitions are equal to 1 between characters in a keyword model and equal to the probabilities of the language model previously learned representing the shallow parsing model.

The recognition stage using this model is described in the next section.

3. Our information extraction system

In this section we describe the whole recognition process including preprocessing, feature extraction and HMM decoding constrained by our model.

Preprocessing

Standard preprocessing is integrated in a sequential way. First, incoming letter images are binarized. Lines are segmented thanks to a state of the art method. Then, in order to reduce the writing variability between writers, it is common to correct the skew and the slant of text lines. The skew angle is basically corrected by rotation while the slant angle is corrected by a shear transformation. In a last step, writing baselines are detected. For further details on this step, see [3].

Feature extraction

HMMs used during the recognition step expect a sequence of feature vectors as input for each unknown preprocessed line to be recognized. To extract such a sequence of feature vectors, a sliding window is used. A window of d pixels width with o pixels overlap between two consecutive windows is moved from left to right over the current line. For every position of the window, a feature vector inspired by [1] is computed.

Each vector contains $20 + d$ features. Taking a look at the ICDAR 2009 word recognition competition shows the power of this feature vector [6]. Furthermore, it has been designed especially for its use with HMMs.

Recognition phase

During the recognition phase, text lines are decoded regarding the model. The recognition problem can be written according to equation 1, L_{opt} representing the best sequence of words for a given observation sequence O:

$$L_{opt} = \arg\max_{L}\{P(L|O)\} \quad (1)$$

$$= \arg\max_{L}\left\{\frac{P(O|L)P(L)}{P(O)}\right\} \quad (2)$$

$$= \arg\max_{L}\{P(O|L)P(L)\} \quad (3)$$

where $P(O)$ is the probability of a sequence of observations O, $P(L)$ the probability of a sequence of words L and $P(O|L)$ the probability of a sequence of observations O given the sequence of words L.

Although $P(O)$ is difficult to estimate, it does not affect the search for the best state sequence regarding observations and can be thus deleted from the computations. In equation 3, $P(L)$ is the prior probability of a word sequence. It allows one to accept or not particular word sequences. The probabilities are learned on the training dataset regarding our model. The term $P(O|L)$ is estimated using the *Time Synchronous Beam Search* algorithm introduced in [8]. If K_i stands for a given word in the lexicon, W_j for a given word in OOV words and S_k for a given space between two words in the word sequence L, $P(O|L)$ can be computed as follows:

$$P(O|L) = \prod_{i}^{N} P(O_i|K_i) \prod_{j}^{M} P(O_j|W_j) * \prod_{k}^{K} P(O_k|S_k) \quad (4)$$

$$P(O|L_{opt}) = \max_{i,j,k} P(O_i^*|K_i)P(O_j^*|W_j) * P(O_k^*|S_k) \quad (5)$$

4. Experiments and results

We now present the database used for experiments, our experimental protocol and the results obtained.

Database

RIMES database including 1150 French incoming mail from different writers is used [5]. 950 of them containing 36000 words are used for training the character models and for learning the global line model transitions. As this database is partially annotated at the word level, a complete annotation work has been done on 20 different letters. These 20 documents are used as test database.

Experimental protocol

In order to evaluate information extraction in a database of D documents, recall and precision measures must be computed. Given an incoming letter of index i, let $N_{ok}(i)$ be the number of well detected words, $N_{fa}(i)$ the number of false alarms and $N(i)$ the number of words to extract, Recall $R(i)$ and Precision $P(i)$ in a piece of mail are computed as follows:

$$R(i) = \frac{N_{ok}(i)}{N(i)} \quad (6)$$

$$P(i) = \frac{N_{ok}(i)}{N_{ok}(i) + N_{fa}(i)} \quad (7)$$

In order to compute relevant results, one part of the information sought should belong to the current piece of mail. Thus, 10 words are randomly picked in it. Different lexicons of size N are then generated by picking randomly $N - 10$ words that appear in the other documents but not in the document under investigation. Doing so for every piece of mail allows one to compute recall and precision means and variances on the entire test database as follows :

$$R_{mean} = \frac{1}{D}\frac{\sum_i N_{ok}(i)}{\sum_i N(i)} \quad (8)$$

$$R_{variance} = \frac{1}{D}\sum_{i}\left(R_{mean} - \frac{N_{ok}(i)}{N(i)}\right)^2 \quad (9)$$

Precision mean and variance are computed like equations 8 and 9 regarding 7. Varying the hyperparameter G allows one to obtain different operating points of the system and thus enables to better describe the Recall/Precision curve: a value of G close to 1 gives an advantage to words from the lexicon at the expense of the shallow parsing model and thus will favor recall and vice versa for values of G close to 0 that will result in improving precision of the system. In case of deployment of such a system, the value of G can be chosen depending on industrial needs and expectations.

Results

The pair (d, o) giving the best results in word recognition have been chosen on the learning database i.e. $(8, 5)$. In this configuration, the optimal number of

states for each character model is 4 and the number of gaussians for each state has been fixed to 5. The Recall/Precision curve for a 10 word lexicon is shown in figure 3.

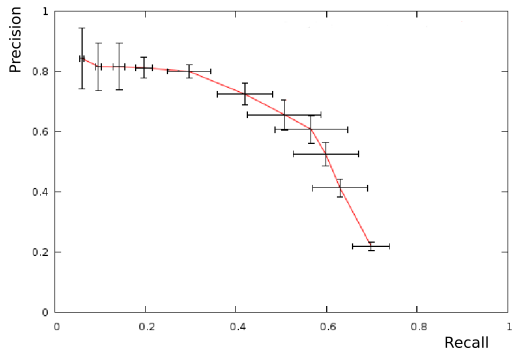

Figure 3. Recall/Precision for a 10 word lexicon on the RIMES database

Note that mean and variance are given for each operating point since each experience has been repeated for each document. We can note a break-even point at 60% which seems to be an interesting result. It should be noticed that we are the first ones to give results on the field of information extraction on this database, therefore unfortunately no comparison can be done with other works. Let us now evaluate our system with a larger lexicon size. Corresponding results for a lexicon size of 10 to 200 words are given on figure 4.

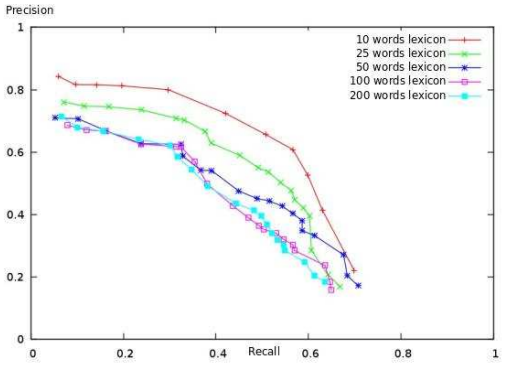

Figure 4. Recall/Precision for different lexicon sizes on the RIMES database

We can note that the system is able to handle reasonable lexicon size thus allowing document classification or categorization applications. We may want to test its scalability with larger lexicons even if such applications do not require open lexicon.

5. Conclusion and future works

A new handwritten information extraction system in handwritten uncontrained documents has been introduced. It relies on a generic line modelisation based on HMMs allowing a dual representation of the relevant and the irrelevant information. First results illustrate the potential of our approach.

As a short term objective, it will be useful to compare our approach to a full recognition one. Then, a part of the focus will regard the feature vector enhancement and testing this extraction method on numerical fields such as phone numbers or more complex sequences like dates and other alphanumeric fields using jokers like '?' and '*' in the words of the lexicon.

References

[1] R. Al-Hajj, C. Mokbel, and L. Likforman-Sulem. Combination of hmm-based classifiers for the recognition of arabic handwritten words. *Proc. ICDAR*, 1:959–963, 2007.

[2] H. Cao and V. Govindaraju. Template-free word spotting in low-quality manuscripts. *Proc. ICDAR*, pages 392–396, February 2007.

[3] C. Chatelain, L. Heutte, and T. Paquet. A syntax-directed method for numerical field extraction using classifier combination. *IWFHR*, pages 93–98, 2004.

[4] C. Choisy. Dynamic handwritten keyword spotting based on the nshp-hmm. *Proc. ICDAR*, 1:242–246, 2007.

[5] E. Grosicki and H. El-Abed. Icdar 2009 handrwriting recognition competition. *Proc. ICDAR*, 1:1398–1402, 2009.

[6] Y. Kessentini, T. Paquet, and A. Benhamadou. Off-line handwritten word recognition using multi-stream hidden markov models. *Pattern Recognition Letters*, 31:60–70, 2010.

[7] U. Marti and H. Bunke. Text line segmentation and word recognition in a system for general writer independent handwriting recognition. *In Sixth International Conference on Document Analysis and Recognition*, 1:159–163, 2001.

[8] D. Moore. TODE: A Decoder for Continuous Speech Recognition. *IDIAP Research Report 02-09*, 2002.

[9] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Readings in speech recognition*, pages 267–296, 1990.

[10] J. A. Rodríguez-Serrano, F. Perronnin, and J. Llados. A similarity measure between vector sequences with application to handwritten word image retrieval. *CVPR 09*, August 2009.

[11] A. Vinciarelli, S. Bengio, and H. Bunke. Offline recognition of unconstrained handwritten texts using hmms and statistical langage models. *IEEE Trans. on PAMI*, 26(6):709–720, 2004.