A Confidence-Based Method for Keyword Spotting in Online Chinese Handwritten Documents

Heng Zhang, Da-Han Wang, Cheng-Lin Liu
National Laboratory of Pattern Recognition (NLPR),
Institute of Automation, Chinese Academy of Sciences,
95 Zhongguancun East Road, Beijing 100190, P.R. China
{*hzhang,dawang,liucl*}*@nlpr.ia.ac.cn*

Abstract

In keyword spotting from handwritten documents, the word similarity is usually computed by combining character similarities. Converting similarity to probabilistic confidence is beneficial for context fusion and threshold selection. In this paper, we propose to directly estimate the posterior probability of candidate characters based on the N-best paths from the segmentation-recognitioin candidate lattice. The N-best path scores are converted to confidence measure (CM) using soft-max, and the posterior probability of candidate characters is the summation of confidence measures of paths that pass the candidate character. The parameter for CM is optimized using the binary cross-entropy criterion. Experimental results on database CASIA-OLHWDB demonstrate the effectiveness of the proposed method.

1. Introduction

The methods for keyword spotting from handwritten documents [3] can be grouped into image matching-based ones and word/character model-based ones. Character model-based approach can take advantage of the character classification ability, handle large vocabulary and overcome the out-of-vocabulary problem [7]. In character model-based retrieval, word similarity is obtained by combining similarities of candidate characters [3]. However, similarities output by the character model are usually not probabilities, making it difficult to fuse with contexts and to set the threshold for retrieval.

In this paper, we propose to directly estimate the posterior probabilities of candidate characters based on Chinese handwritten text recognition under the integrated segmentation-recognition framework [8]. In the framework, a text line is over-segmented into primitive segments. Consecutive segments are combined to generate candidate patterns, which are recognized by a character classifier and assigned candidate classes. The candidate patterns and classes form the segmentation-recognition candidate lattice. Paths in the candidate lattice are scored by combining the classification scores, linguistic and geometric contexts. The scores of the N-best paths are converted to confidence measures [11] using soft-max, based on which the posterior probabilities of candidate characters (a candidate character is the candidate pattern assigned a class) are computed. The parameter for confidence measure is estimated on the N-best paths of a training dataset of text lines using the binary cross-entropy (CE) criterion.

The proposed method is different from the OCR-based method in that the latter one only retained the 1-best recognition result for retrieval while our method is based on N-best paths and gives the posterior probabilities of candidate characters. Some works [9] have used forward-backward algorithm to compute the path probabilities and also use these probabilities to produce the word confidence. But in our over-segmentation based handwriting recognition system, the output is a distance measure, unlike that in HMM-based speech and handwriting recognition where the path score is already a probability. We evaluated the proposed method on the online Chinese handwriting database CASIA-OLHWDB [1], and demonstrated the effectiveness of the proposed method compared to OCR based method.

2. System overview

Fig.1 illustrates the proposed document retrieval system, comprising two main stages: indexing and keyword spotting. The online documents are first segmented into text lines [12] using temporal and spatial information, after which each text line is over-segmented into primitive segments. Candidate patterns are generated by combining consecutive segments and are assigned candidate classes by the character classifier. The candidate patterns and classes, linguistic context and geometric context are represented in a segmentation-recognition candidate lattice, which contains many paths each corresponding to a recognition result. Each path is evaluated by the scoring function as follows:

$$\begin{aligned} d(C,\vec{X}) = -\sum_{i=1}^{n}\{&k_i \log P(c_i|\mathrm{x}_i) + \lambda_1 \log P(c_i|c_{i-1}) \\ &+\lambda_2 \log P(c_i|\mathrm{g}_i^{uc}) + \lambda_3 \log P(z_i^p = 1|\mathrm{g}_i^{ui}) \\ &+\lambda_4 \log P(c_{i-1}, c_i|\mathrm{g}_i^{bc}) + \lambda_5 \log P(z_i^g = 1|\mathrm{g}_i^{bi})\}, \end{aligned} \quad (1)$$

where k_i is the number of primitives composing the candidate character, the probabilities are the character classification score, bi-gram linguistic score, unary class-dependent and class-independent geometric scores, binary class-dependent and class-independent geometric scores, respectively. Except the linguistic score, all these constituent probabilities are approximated from the outputs of classifiers on character features and geometric features.

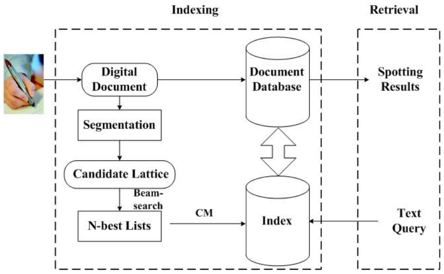

Figure 1. Block diagram of the keyword spotting system.

The candidate lattice is searched for the N-best paths by the beam-search algorithm and the path scores are converted to posterior probabilities using the soft-max function. After measuring the character confidence from the path probabilities, the N-best paths are rearranged into lattice structure for smaller size and computed confidence of candidate characters are stored for indexing. In retrieval, each word is compared with the candidate characters by the dynamic search algorithm similar to [3].

3. Character probability

In the dynamic word matching progress, the word similarity is obtained by combining the confidence of candidate characters as introduced in our previous work [3]. So, how to compute the character probability is the key point of keyword spotting.

3.1 Probabilities of individual candidate characters

We estimate the character confidence from the probabilities of N-best paths. Let b and e denote the beginning and end segments of the candidate character c, respectively. The i-th path can be represented as $(c,b,e)_i = (c_1,b_1,e_1),...,(c_M,b_M,e_M)$. Where M is the number of candidate characters on the path. Then given a text line $\vec{X}$, the posterior probability of a candidate character (c,b,e) can be computed by summing up the posterior probabilities of all paths which contain this candidate character [11]:

$$P((c,b,e)|\vec{X}) = \sum_{(c,b,e)_i:(c,b,e)\in(c,b,e)_i} P((c,b,e)_i|\vec{X}) \quad (2)$$

The posterior probability $P((c,b,e)_i|\vec{X})$ can be approximated using path scores of the N-best paths. Under the assumption of Gaussian distribution with equal identity covariance matrix, the posterior probability is proportional to the exponential of the path score:

$$P((c,b,e)_i|\vec{X}) \propto exp[\frac{-d((c,b,e)_i,\vec{X})}{\theta}] \quad (3)$$

where $d((c,b,e)_i,\vec{X})$ is the path score as in Eq.(1) and the parameter θ is optimized on training samples to overcome the deviation from distribution assumption.

Yet, the path score depends upon the length of the text line. To overcome the variable scale of path score on text lines of different lengths, [10] used the length of the line for normalization:

$$d_i = \frac{d((c,b,e)_i,\vec{X})}{len} \quad (4)$$

Here, we define the len as the number of the primitive segments after over-segmentation. We also consider another way of normalizing the scale of path scores using the difference of path score from the top-rank path:

$$d_i = d((c,b,e)_i,\vec{X}) - d((c,b,e)_0,\vec{X}) \quad (5)$$

The probabilities of the remaining paths beyond the N-best paths are viewed as zero. In this way, we can get the posteriori probability by the normalization of Eq.(3), resulting in the so-called soft-max:

$$P((c,b,e)_i|\vec{X}) = \frac{exp[-\alpha \cdot d_i]}{\sum_{j=1}^{N} exp[-\alpha \cdot d_j]} \quad (6)$$

where $\alpha = \frac{1}{\theta}$.

3.2 Parameter estimation

The estimation of confidence parameter α is expected to optimally fit the recognition performance on a dataset of text lines. A proper criterion for this purpose is the cross-entropy (CE) loss, which was shown to be effective for training the classifier for retrieval [4]. We use the candidate characters on the N-best paths of the training text lines (preferably different from the data set for training handwriting recognition parameters) as the training set, defined as:

$$\{(c,b,e), y = \delta(T,R) \in 0,1\} \tag{7}$$

where (c,b,e) is the candidate character on the N-best paths, T and R are the true class and recognition result of the candidate character, respectively, and $y = \delta(T,R)$ is the binary class label assigned to the candidate character. For the candidate characters that are mis-segmentations, they will be labeled as 0.

Similarly to [4], the CE loss is obtained:

$$MinCE = -\sum_{k=1}^{m}[y_k log P_k + (1-y_k) log(1-P_k)] \tag{8}$$

where m is the number of all the candidate characters on the N-best paths of training set and P_k is defined as in Eq.(2). We minimize the empirical loss by stochastic gradient descent to estimate the parameter α.

3.3 Accumulated probability

A candidate character can be correctly recognized even if it contains only a part of primitive segments of the truth character. In this situation, its posterior probability mass may be split among different paths. To avoid this undesirable effect which produces an underestimation of posterior probabilities, the split posterior probability should be re-joined by summing up the posterior probabilities of the intersected candidate characters with identical class label. So, given a character which occurs at a specific segment s, its accumulated posterior probability at segment s, $A((c,b,e),s)$, is computed by summing the posterior probabilities over all candidate characters which have the same character label and intersect with segment s:

$$A((c,b,e),s) = \sum_{(c,b,e)_i : s \in (c,b,e)_i \cap (c,b,e)} P((c,b,e)_i | \vec{X}) \tag{9}$$

In order to compute the posterior probability of a character which is recognized in the successive segments (b,e), different methods based on the accumulated posterior probabilities have been proposed in [11]. The method which yields the best performance is based on the maximum (best-case) $A((c,b,e),s)$ within the character boundaries:

$$PMax((c,b,e)) = \max_{s\in[b,e]} A((c,b,e),s) \tag{10}$$

4. Experimental results

We evaluated the performance of the proposed method on an online Chinese handwriting database CASIA-OLHWDB [1] ,which includes three isolated data sets 1.0-1.2 (DB1), and three text data sets 2.0-2.2 (DB2), and are divided into standard training and test subsets. We used a commonly used classifier named modified quadratic discriminant function (MQDF) and trained classifier parameters using the training set of all the isolated characters and the characters segmented from the training set of all the text pages. The geometric model was trained using the training set of DB2 and the binary language model is the same as [8]. The training set of DB2.0 was used for estimating the combining weights in text line recognition system for contexts integration, and the training set of DB2 was used to estimate the parameter α. In testing, we used the high-frequency 2-character words in the lexicon of the Sogou labs [2] as query words and performed keyword spotting on CASIA-OLHWDB2.0. The spotting performance is measured by recall, precision and F-measure as in [2].

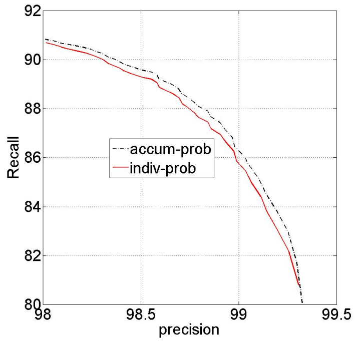

Figure 2. ROC curves of the individual (indiv-prob) and accumulated probabilities (accum-prob).

To justify the accumulation of the character probability over paths, we first compare the performance of the individual candidate probability (Eq.2) and the accumulated character probability (Eq.10). With variable thresholds, we plotted the ROC curves of the two types of probabilities. Fig.2 shows the spotting results and we can draw a conclusion that the accumulated probability outperforms the individual probability, and can overcome the probability loss due to probability split in different paths.

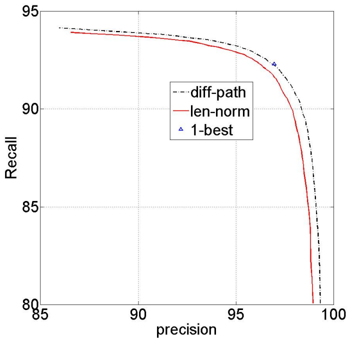

Figure 3. ROC curves of the proposed method with length normalization (len-norm) and difference of path score (diff-path), and the OCR (1-best)-based method.

Using the accumulated probabilities, we compare the performance of length-normalized path score (Eq.4) and difference of path score (Eq.5). The performance is also compared with the handwritten text recognition (OCR) based method. For OCR based method, we adopt the character string recognition method implemented in [8] to recognize the whole documents and store the recognition results (1-best) for search. The results are shown in Fig.3. From the results, we can see that the proposed method with difference of path score performs much better than the variation with length normalization. The 1-best (ROC) based method can get a high recall (92.30%) and precision (96.96%) comparable with the proposed method, but it cannot vary the tradeoff between the recall rate and precision by adjusting thresholds, while our method can get a ROC curve for good ranking of the spotting results. It raises the recall rate and sacrifices the precision but this is preferred in many retrieval applications.

Our experiments were implemented on a PC with Intel(R) Core(TM)2 Duo CPU E8400 3.00 GHz processor and 2GB RAM. The average time of searching for a query word in all the test documents (after indexing) is 6.5ms.

5. Conclusions

We proposed a confidence-based method for keyword spotting from handwritten documents. Based on proper estimation of the parameter for path confidence and accumulating character probability, the proposed method yields promising performance. Compared to OCR-based retrieval, it gives more flexible tradeoffs between recall and precision rates.

Acknowledgment

This work was supported in part by the National Natural Science Founcation of China (NSFC) Grants 60825301 and 60933010.

References

[1] C.-L.Liu, F.Yin, D.-H.Wang, and Q.-F.Wang. Casia online and offline chinese handwriting databases. In *Proc.ICDAR*, pages 37–41, 2011.

[2] H.Zhang and C.-L.Liu. A lattice-based method for keyword spotting in online chinese handwriting. In *Proc.ICDAR*, pages 1064–1068, 2011.

[3] H.Zhang, D.-H.Wang, and C.-L.Liu. Keyword spotting from online chinese handwritten documents using one-vs-all trained character classifier. In *Proc. ICFHR*, pages 271–276, 2010.

[4] C.-L. Liu. One-vs-all training of prototype classifier for pattern classification and retrieval. In *Proc.ICPR*, pages 3328–3331, 2010.

[5] C.-L. Liu and M. Nakagawa. Precise candidate selection for large character set recognition by confidence evaluation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(6):636–642, 2000.

[6] R. Manmatha, C. Han, and E. Riseman. Word spotting: a new approach to indexing handwriting. In *Proc. CVPR*, pages 631–637, 1996.

[7] N.R.Howe, S.Feng, and R.Manmatha. Finding words in alphabet soup: inference on freeform character recognition for historical scripts. *Pattern Recognition*, 42(7):1445–1457, 2009.

[8] Q.-F.Wang, F.Yin, and C.-L.Liu. Handwritten chinese text recognition by integrating multiple contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, in press, 2012.

[9] S.Quiniou and E.Anquetil. Use of a confusion network to detect and correct errors in an on-line handwritten sentence recognition system. In *Proc.ICDAR*, pages 382–386, 2007.

[10] V.Frinken, A.Fischer, R.Manmatha, and H.Bunke. A novel word spotting method based on recurrent neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(2):211–224, 2012.

[11] F. Wessel, R. Schlter, K. Macherey, and H. Ney. Confidence measures for large vocabulary continuous speech recognition. *IEEE Trans.Speech Audio Process.*, 9(3):288–298, 2001.

[12] X.-D. Zhou, D.-H. Wang, and C.-L. Liu. A robust approach to text line grouping in online handwritten japanese documents. *Pattern Recognition*, 42(9):2077–2088, 2009.