Subword-based Stochastic Segment Modeling for Offline Arabic Handwriting Recognition

Krishna Subramanian, Vasant Manohar, Huaigu Cao, Rohit Prasad, Prem Natarajan
Raytheon BBN Technologies, 10 Moulton St, Cambridge, MA 02138
{ksubrama, vmanohar, hcao, rprasad, prem}@bbn.com

Abstract

In this paper, we describe several experiments in which we use a stochastic segment model (SSM) to improve offline handwriting recognition (OHR) performance. We use the SSM to re-rank (re-score) multiple decoder hypotheses. Then, a probabilistic multi-class SVM is trained to model stochastic segments obtained from force aligning transcriptions with the underlying image. We extract multiple features from the stochastic segments that are sensitive to larger context span to train the SVM. Our experiments show that using confidence scores from the trained SVM within the SSM framework can significantly improve OHR performance. We also show that OHR performance can be improved by using a combination of character-based and Parts-of-Arabic-Words (PAW)-based SSMs.

1. Introduction

Offline handwriting recognition (OHR) continues to be a challenging research problem due to a variety of reasons. Most recognition approaches that require accurate segmentation of the text into smaller units do not perform well on handwritten text. There are two primary causes for poor performance of segmentation-based approaches on real-world handwritten text. First, segmenting handwritten text for connected scripts such as Arabic is very difficult. Second, most real-world images are prone to degradations that result in breaks and merges in glyphs. This phenomenon creates new connected components that are not observed in training data, and therefore the character classifier is unable to accurately recognize the glyphs.

In our earlier work [1], we noted that the HMM-based systems have several advantages over other systems, primarily because they are *segmentation-free*, i.e. no pre-segmentation of word/line images into smaller units such as sub-words or characters is required, making it viable to quickly and cheaply incorporate large amounts of data for experimental use. However, there are well known limitations with HMM-based approaches [2]. These limitations are due to two reasons: (a) the assumption of conditional independence of the observations given the state sequence, and (b) the restrictions on feature extraction imposed by frame-based observations. The limitations noted in [2] are also relevant to OHR systems as they use pixel-level features from narrow slices of the text. Specifically, the narrow windows provide very little contextual information making the conditional independence assumption in these systems unrealistic.

In [1], we presented a novel framework for combining structural matching and HMM-based recognition, which has more discriminative power than simply combining the structural and short span features at each frame. Structural matching was done by extracting structural or longer span shape features such as Gradient, Structure, and Concavity (GSC) [3] from *stochastic segments* and using a support vector machine (SVM) classifier trained on these features to match the decoder hypotheses against the stochastic segments. The SVM provides confidence scores that are used to re-rank the decoder hypotheses and improve the overall system word error rate (WER).

In [1], we only used the GSC features extracted from stochastic character segmentations and showed improved performance. In this paper, we expand on our earlier work and experiment with GSC features, in combination with two other features – Gabor and 2-D percentile, each of which having a known capacity to extract information from larger context. In addition to character-based SSMs, we also work with Parts-of-Arabic-Words (PAW)-based SSMs. In this paper, we define a PAW to be a combination of two or more characters that are part of at least one naturally occurring Arabic word.

The rest of this paper is organized as follows. In Section 2, we describe the corpus of annotated Arabic handwritten text that is used in our experiments. In Section 3, we provide an overview of the Raytheon BBN HMM-based OHR system. In Section 4, we provide a procedural description of the SSM framework, more details of which can be found in [1]. In Section 5, we describe the three features that we use in our experiments. In Section 6, we describe our experimental setup followed by experimental results in Section 7. We conclude in Section 8 with our closing remarks.

2. Corpus Description

We used two sets of corpora in our experiments – one corpus is from the Applied Media Analytics (AMA), which we refer to as the AMA corpus and the second one is from the Linguistic Data Consortium (LDC), which we refer to as the LDC corpus. The AMA corpus that we use in our experiments consists of Arabic handwritten documents provided by a diverse body of writers. The collection is based on a set of 200 documents with a variety of formats and layout styles. The final collection contains a scanned TIFF image of each page, an XML file for each page which contains writer and page metadata, the bounding box for each word in the page in pixel coordinates, and a set of offsets representing PAWs. We used a subset of the images, scanned at 300dpi for our experiments. This data set is used in our PAW classification experiments.

The LDC corpus consisted of scanned image data of handwritten Arabic text from newswire articles, weblog posts, and newsgroup posts along with the corresponding ground truth annotations including tokenized Arabic transcriptions and their English translations. It consists of a total of 39361 images scanned at 600 dpi written by 357 different authors for training, development, and testing purposes. The partitioning of images into training, development, and test sets ensures that no document with the same content appears in two or more sets. Additionally, we also ensure that the proportion of authors common to training in both the development and test sets is approximately the same.

3. Baseline HMM-based OHR System

We use the Raytheon BBN Byblos OHR system [4, 5] as our baseline OCR system. The system was trained on 37K pages of handwritten text documents. 868 pages were used for development and 885 pages were used for validation. Feature extraction involves horizontal segmentation of the line image into frames followed by feature vector computation for each frame. The features used in the current baseline configuration include: Percentile of intensities, Angle, Correlation, and Energy, which we refer to as the PACE [4, 5] features, in combination with Gradient-Structure-Concavity (GSC) features [3]. Raw feature dimensionality was 129, which results in 387 features after three frame concatenation. We then perform Linear Discriminate Analysis (LDA) [6] to reduce the overall feature dimensions to 15. The training module estimates multi-state, left-to-right HMMs for each character using the Expectation Maximization (EM) algorithm for maximum likelihood training. Position Dependent Tied Mixture (PDTM) HMMs [7] were trained for each character. PDTMs are HMMs where a separate set of Gaussians is estimated for each state of all the context-dependent HMMs associated with a particular character. In total, we used 2723K Gaussians to model 181 Arabic character glyphs. The recognition module uses an efficient 2-pass n-best decoder [4]. Unsupervised adaptation was performed on each page using the best hypothesis from an initial pass of recognition. The overall WER for the baseline system is 26.5%.

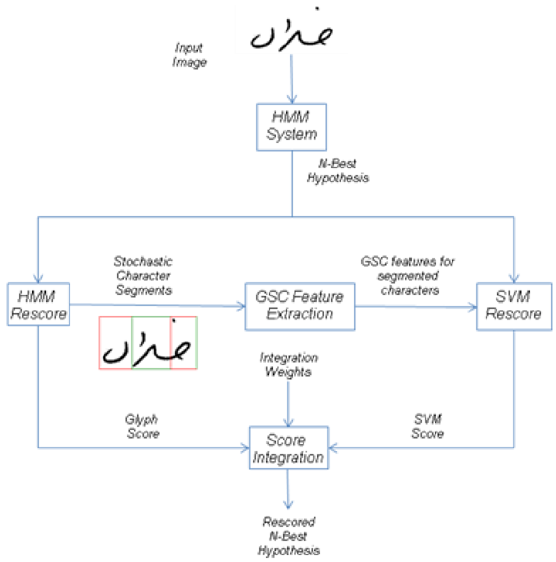

Figure 1. **Illustration of the rescoring procedure in which the SVM scores are combined with the glyph scores.**

In this paper, we use glyph models that are trained to recognize characters. Ligatures are considered as independent characters and are modeled as such.

4. Design for Stochastic Segment Modeling

The stochastic segment modeling framework involves the following key steps for performing recognition:

1. *Stochastic Segment Generation*: First, we generate a set of recognition hypotheses using the HMM system trained on short span features. Then, for each hypothesis, we extract *stochastic segments*

(2-D character images) using the character segmentation provided by the HMM.

2. *Segmental Classifier/Scorer*: We extract structural features that represent shape characteristics of the character. Then, we compute a score for each character in the hypothesis using a classifier trained on the stochastic segments from the training data. For generating the composite score for each hypothesis from the segmental model, we compute the geometric mean of the SVM scores from each character and use the logarithm of this score as the final SVM score. In this paper, we use support vector machines (SVM) as the *segmental* classifier.
3. *Score Combination from HMM and Segmental Model*: We use the score from the HMM and the SVM for each hypothesis to generate the best hypothesis.

A block diagram illustrating these steps is shown in Figure 1.

5. Longer Span Features

In our experiments, we explore the use of three different types of features that have a known capacity to capture structural and broad-based glyph characteristics. In our experiments, we first extract stochastic segment images using segment boundaries provided by the Byblos recognition engine. We then tighten the image to crop white space around the borders and then resize the cropped image to a 64x64 image. The image is then binarized. The binary image is used to extract GSC, Gabor, and percentile features described below.

5.1. Gradient-Structure-Concavity (GSC) Features

GSC features are symbolic, multi-resolution features that combine three different attributes of the shape of a character – the gradient representing the local orientation of strokes; structural features that extend the gradient to longer distances and provide information about stroke trajectories; and concavity that captures stroke relationships at long distances. The GSC features have been successfully applied in handwritten digit and character recognition. More details about this feature can be found in [3, 8].

In our experiments, for each stochastic segment image, we first segment the input image in to a 4x4 grid and extract 64 GSC features from each grid resulting in a total of 512 GSC features.

5.2. Gabor Features

Gabor filters have been applied to face recognition [9, 10], speech recognition, and OCR [11, 12]. Sung et al. [11] extracted hierarchical Gabor features (HGFs) in such a way that these features represent different levels of structured information. Then they constructed a Bayesian network classifier to encode the hierarchical dependence among HGFs. Another work using Gabor features is by Wang et al. [12], where they make use of both positive and negative values in the real part of Gabor filtering results and construct histogram feature vectors for classification.

A 2-D Gabor filter could be considered as a complex sinusoidal plane modulated by a Gaussian function in spatial domain,

$$h(x, y, \lambda, \phi, \sigma_x, \sigma_y) = \frac{1}{2\pi\sigma_x\sigma_y} \times \exp\{ -\frac{1}{2}(\frac{R_1^2}{\sigma_x^2} + \frac{R_2^2}{\sigma_y^2})\} \times \exp\{ i\frac{2\pi R_1}{\lambda}\},$$

$$R1 = x\cos\phi + y\sin\phi,$$
$$R2 = y\cos\phi - x\sin\phi.$$

Here, λ and ϕ are the wavelength and orientation of the sinusoidal plane wave; $f=1/\lambda$ is the frequency. Due to these two independent parameters, Gabor filters have selectivity in both the spatial and frequency domains.

The feature extraction is based on the convolution of the original image with Gabor filters with specific spatial and frequency orientation. The convolved image has strong responses at specific orientations. The procedure that we use to compute Gabor features is as follows:

```
For each frequency f in flist:
  For each orientation $\phi$ in the orientation list olist:
    -  Construct a Gabor filter g(f, $\phi$ ).
    -  Convolve g(f, $\phi$ ) with original image I, get response image R,
    -  Compute the mean response in R, denote as m,
    -  Count # of pixels that have a larger value than m, denote as Nr,
    -  Divide R into n by m frames,
         o  For i = 1 : n:
              For j = 1: m
                - Count the # of strong responses $N_{i,j}$
                  and compute the ratio r =$N_{i,j}$/ Nr;
              - Append r to the feature vector x
```

Using this procedure, the total number of feature vectors obtained is equal to |flist|*|olist|*m*n. In our

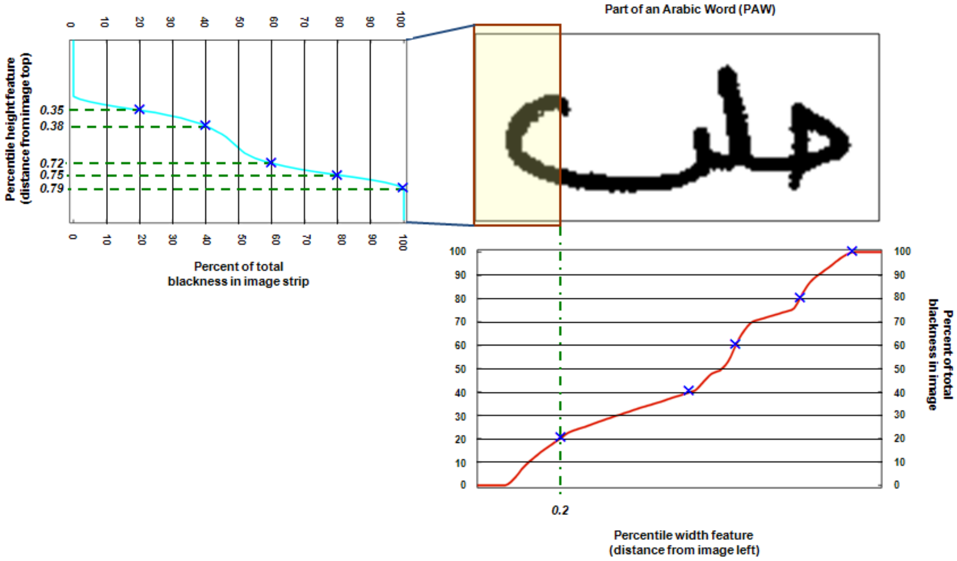

Figure 2. **Illustration of the procedure for extracting 2-D percentiles from a PAW. In this example, N=5. The following features will be added to the overall 2-D percentile feature vector: {X_1=0.2,{Y_m^1=0.35,0.38,0.72,0.75,0.79}}**

experiments, we use two different frequencies {0.03, 0.10} and four different orientations {0, π /4, π /2, 3π /4}. Besides, to construct Gabor features, we divided each stochastic segment image into 10 by 10 frames to get a total of 800 Gabor features.

5.3. 2-D Percentile Features

1-D percentile features have proven to be extremely successful in modeling short-term context. They are the core set of features use in our Byblos OHR system and have been successfully used to recognize a multiplicity of scripts, including machine and handwritten text from Arabic, Chinese, and various other languages. In this paper, we introduce the 2-D percentile features.

The procedure to extract the 2-D percentile features as used in this paper is depicted in Figure 2. We first project the stochastic segment image onto the x-axis and compute the cumulative projection profile along the x-axis. Using the x-projection profile, we compute the location, X_n, at which i_nth percentile occurs, where i_n = 100/(N_x-1)*n, where N_x is the total number of percentile features we want to extract from the x-axis. We then segment the image such that the nth segment, S_n, is bounded between {X_{n-1},X_n}, $n \in \{1,N_x\}$ where X_0 and X_{Nx} are the left and right boundaries of the stochastic segment. For each segment, S_n, we compute the cumulative projection profile along the y-axis and perform a similar analysis we performed on the x-projection profile on the y-projection profile to obtain {Y_m^n}, $m \in \{1,N_y\}$, where N_y is the total number of percentile points we want to extract from the y-axis. Next, we invert the order of finding projection profiles by first computing the y-projection profile and then the x-projection profile. The final set of features used within the SSM framework for rescoring are {{{X_n,{Y_m^n}}, {Ym,{X_n^m}}}. The total number of features are given by N_x*(N_y+1)+N_y*(N_x+1). If N_x and N_y are equal, the total number of features is given by $2N(N+1)$.

In our experiments, we use N=20 to get a total of 840 2-D percentile features for each stochastic segment image.

6. Experimental Setup

For each of our SSM experiment, we used the libsvm [13] tool to train a classifier and to provide confidence scores for classification and re-scoring experiments. We trained a multi-class C-SVC SVM using the Radial Basis Function (RBF) kernel because it gave the best results in our internal tests. We compared it with the nu-SVC SVM and linear, polynomial, and sigmoid kernels. We setup the training so that libsvm computes probability estimates. For each trained model, we also performed 5-fold cross validation to measure classification accuracy. For experiments using character-based SSMs, we build a multi-class C-SVC classifier using 165 Arabic characters occurring at least 500 times in the training corpus, resulting in a total of 82500 training instances. For experiments using PAW-based SSMs, we build a multi-class C-SVC classifier

Table 1. **Segment classification accuracy SVM classifier.**

Features	Classification Accuracy (%)
GSC	82.1
Gabor	82.2
2-D Percentile	74.6
GSC+Gabor	84.2
GSC+2-D Per	84.8
GSC+Gabor+2-D Per	87.4

using 466 most frequently occurring PAWs in Arabic language that occur at least 500 times in the training corpus resulting in a total of 233000 training instances. The character-based and PAW-based stochastic segments that were used in training were obtained by force aligning reference transcriptions instead of n-best hypotheses.

7. Experiments Results using SSM

In the first experiment, we used manually annotated PAW images and the corresponding PAW labels to train a SVM classifier. The PAW images and labels were randomly chosen from the AMA corpus. We used the entire PAW image to extract features. A total of 6498 training samples from 34 PAW classes were used to train the classifier. A C-SVC SVM using the RBF kernel was trained on features extracted from each of the training sample. The test set consists of 848 PAW images from the same set of 34 PAW classes. The trained SVM model was then used to classify the test images. The accuracy using the GSC, Gabor, and 2-D percentile features is shown in Table 1. From Table 1, we see that all of the three features are successful at extracting context sensitive information. The combination of GSC, Gabor, and 2-D Percentile features gives the best results.

In our second experiment, we perform closed-set classification using classifiers trained on character-based and PAW-based stochastic segments. The classification results from using features extracted from character-based stochastic segments are shown in Table 2. The classification results from using features extracted from PAW-based stochastic segments are shown in Table 3. From Table 2, we see that the GSC feature set performs the best, followed by Gabor, and then 2-D Percentiles. It is surprising that although 2-D Percentile features performed very well in our PAW-based experiments as seen in Table 1, we do not see the same results when they were used on stochastic segments. On comparing the classification performance using char-based and PAW-based stochastic segments from Table 2 and 3, it is gratifying to note that although the number of PAW classes was much larger than the number of character classes (466 v/s 165), the classification accuracy on PAWs was much better. It demonstrates that the discriminative ability of our features increases with the amount of context present in the input image.

Table 2. **Classification accuracy using char-based stochastic segments.**

Features	Acc(%) (Char)
GSC	60.5
Gabor	56.9
2-D Percentile	49.6
GSC+Gabor	61.6
GSC+Gabor+2-D Per	61.2

Table 3. **Classification accuracy using PAW-based stochastic segments.**

Features	Acc(%) (PAW)
GSC	74.8
2-D Percentile	63.4
GSC+Gabor	78.4

In our third experiment, we use the three features within the SSM framework for rescoring the n-best hypotheses produced by the baseline Byblos OHR system. In Table 4, character-based SSM is used to provide confidence scores for each n-best hypotheses. From Table 4, we note that the SSMs trained using all of the three feature sets performs best. It improves overall system performance by 0.9% absolute over the baseline. The single best performing feature set is the GSC. The additive value of the other features to GSC is marginal.

Encouraged by the results obtained in our classification

Table 4. **WER after rescoring with char-based stochastic segment models.**

Features	WER(%) (Char)
Baseline	26.5
GSC	25.7
Gabor	26.0
2-D Percentile	26.1
GSC+Gabor	25.7
GSC+Gabor+2-D Per	25.6

Table 5. **WER after rescoring with PAW-based stochastic segment models. If a hypothesis contains character sequences that were not modeled as PAWs, scores from the character-based stochastic segment model is used as a back-off.**

Features	WER(%) (PAW+Char)
Baseline	26.5
GSC	25.6
2-D Percentile	26.0
GSC+Gabor	25.6

experiments reported in Table 3 in which PAW-based classification had better accuracy than character-based classification, we tried to re-score the n-best hypothesis using a combination of PAW-based and character-based stochastic segment scores. Given an n-best hypothesis, we do a longest match search using all the PAWs and characters that were modeled. If a PAW that is modeled exists in the hypothesis, a single confidence score for all characters in the PAW are obtained from the PAW-based SSM. For all the other characters in the hypothesis that were modeled using the character-based SSM, the scores were obtained from the character-based SSM. The logarithm of the geometric mean of scores for all the characters in the hypothesis is used as the composite SSM score. Results using a combination of PAW and character-based stochastic segment models for rescoring are shown in Table 5. Comparing Table 4 and 5, we see that using a combination of PAW-based and character-based scores performs better than the character-based scores alone.

8. Conclusions and Future Work

In this paper, we experimented with longer span features that capture structure and texture from a wider context. We showed that these features scale well, providing improved classification accuracy when presented with wider context. We also showed that the wider span contextual information provided by these features can be combined with a HMM-based OHR system to significantly improve overall OHR performance. Of the three features that we used in our experiments, GSC provides the best performance individually. But the combination of all three features provides the best overall performance.

Given that the oracle WER for the baseline OHR system is 14.1% and the current baseline WER is 26.5, there is a lot to be gained from improving rescoring performance through incorporation of new and external sources of information. The SSM framework provides a robust and flexible platform over which we can build these newer technologies. Our future directions are to use the SSM framework to develop newer features and classifiers that are more sensitive to wider context.

References

[1] P. Natarajan, K. Subramanian, A. Bhardwaj, and R. Prasad, “Stochastic Segment Modeling for Offline Handwriting Recognition,” International Conference on Document Analysis and Recognition (ICDAR 2009), Barcelona, Spain, July 2009.

[2] M. Ostendorf, V. V. Digalakis and O. A. Kimball. From HMM’s to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition. IEEE Transactions on Speech and Audio Processing, 4(5):360-378, 1996.

[3] S. Tulyakov, V Govindaraju, “Probabilistic model for segmentation based word recognition with lexicon,” International Conference on Document Analysis and Recognition, 2001.

[4] P. Natarajan, S. Saleem, R. Prasad, E. MacRostie, K. Subramanian, “Multi-lingual Offline Handwriting Recognition Using Hidden Markov Models: A Script-Independent Approach ,” Springer Book Chapter on Arabic and Chinese Handwriting Recognition, ISSN: 0302-9743, Vol. 4768, pp. 231-250, March 2008.

[5] S. Saleem, K. Subramanian, M. Kamali, R. Prasad, P. Natarajan, “Improvements in BBN’s HMM-based Offline Arabic Handwritten Recognition System”, submitted to ICDAR 2009.

[6] Duda, R. O.; Hart, P. E.; Stork, D. H. (2000). Pattern Classification (2nd ed.)., Wiley Interscience.

[7] R. Prasad, S. Saleem, M. Kamali, R. Meermeier, P. Natarajan, “Improvements in Hidden Markov Model Based Arabic OCR”, International Conference on Pattern Recognition, Tampa, U.S.A, December 2008.

[8] Favata J., Srikantan G. A multiple feature/resolution approach to handprinted digit and character recognition. International Journal of Imaging Systems and Technology, 1996.

[9] Liu, C., Wechsler H. abor Feature Based Classification Using the Enhanced Fisher Linear Discriminant Model for Face Recognition, IEEE Trans. Image Processing, 11(2002), 467-476.

[10] Kämäräinen, J., Kyrki V., Hamouz, M., Kittler, J., Kälviäinen, H. Invariant Gabor Features for Face Evidence Extraction. Proc. of the IAPR Workshop on Machine Vision Applications, (2002). 228-231.

[11] Sung, J., Bang, S, Choi, S. A Bayesian network classifier and hierarchical Gabor features for handwritten numeral recognition, Pattern Recognition Letters, 27(2006)-1, 66-75.

[12] Wang, X., Ding, X, Liu, C. Gabor filter-based feature extraction for character recognition. Pattern Recognition, 38(2005). 369-379.

[13] Web site: http://www.csie.ntu.edu.tw/~cjlin/libsvm