

Brief Paper:

Augmentation of Hidden Markov Chain for Complex Sequential Data in Context

Bong-Kee Sin^{1*}

Abstract: The classical HMM is defined by a parameter triple $\lambda = (\pi, A, B)$, where each parameter represents a collection of probability distributions: initial state, state transition and output distributions in order. This paper proposes a new stationary parameter $e = (e_1, e_2, \dots, e_N)$ where N is the number of states and $e_t = P(x_t = i, y)$ for describing how an input pattern y ends in state $x_t = i$ at time t followed by nothing. It is often said that all is well that ends well. We argue here that all should end well. The paper sets the framework for the theory and presents an efficient inference and training algorithms based on dynamic programming and expectation-maximization. The proposed model is applicable to analyzing any sequential data with two or more finite segmental patterns are concatenated, each forming a context to its neighbors. Experiments on online Hangul handwriting characters have proven the effect of the proposed augmentation in terms of highly intuitive segmentation as well as recognition performance and 13.2% error rate reduction.

Key Words: HMM, Segmentation, Model Inference, Training Algorithm

I. INTRODUCTION

Hidden Markov Model or HMM is a stochastic modeling tool for sequential signals [1]. Real world signals are constantly being generated and streaming in for us to capture and make sense of. There are no explicit starting points and there are no explicit end points as well. For modeling convenience, however, we assume they start at a certain point in time and space. Similarly they are assumed to end their manifestations in due time completing one pattern. This is particularly true when we consider segmental patterns that occur in sequential context; one is followed by another as well as it is preceded by still another, which will be called the context for the target pattern. Then there is a problem of boundary

detection for practical analysis of sequential patterns in context.

The study on improving the HMM has been around the two processes in the model, Markov chain and observation process. The majority of the study in the literature has been around the improving the observation. The most notable among the greatest inventions would be the continuous density modeling using Gaussian mixtures and semi-continuous HMM [2]. On the other hand ideas for improving the Markov chain have been relatively rare. There are a few variations of semi-Markov models in this category. They introduce additional parameters that explicitly model the state duration of the Markov chain [3].

This paper proposes adding a new set of HMM parameters augmenting the underlying Markov chain that control the end of Markov processes. They help the model terminate the random process at the right time. Experimental results in online Hangul handwriting recognition have shown highly intuitive letter boundaries as well as higher recognition performance, upward of 90%.

The rest of the paper consists as follows: Section 2 presents a formal definition of the model based on a modeling assumption. Section 3 provides the theory for model inference and then a set of formulae for estimating the model parameters using the expectation-maximization algorithm. Section 4 describes experimental results based on a network of HMMs for online handwriting recognition.

II. MODELING ASSUMPTION

Many real world signals are dynamic and change over time. But we often catch common patterns recurring often making them look familiar. Thus it is natural to view such a signal as a sequence of local noisy patterns out of finite set of patterns. Hence an input signal is modeled as a concatenation of random segmental patterns which themselves are highly variable. In real world signals the boundaries of the segments are not clear. But each pattern has distinctive characteristics, starts and ends giving some

Manuscript received February 22, 2021; Revised March 13, 2021; Accepted March 23, 2021. (ID No. JMIS-21M-02-005)

Corresponding Author (*): Bong-Kee Sin, Yongso-ro 45 Nam-ku, Busan, Republic of Korea, +82-51-629-6256, bkshin@pknu.ac.kr.

¹Department IT Convergence and Applications, Pukyong National University, Buasn, Republic of Korea, bkshin@pknu.ac.kr

clues to the locations of boundaries. Each segmental sequence concludes its characteristic pattern at a certain point in time.

The classical HMM is defined by a parameter triple $\lambda = (\pi, A, B)$ where $\pi = (\pi_1, \pi_2, \dots, \pi_N)$ is the initial state distribution describing how a process (here, a typical target pattern) starts probabilistically[2] while satisfying the constraints $\pi_i \geq 0$ and $\sum_{i=1}^N \pi_i = 1$. A and B are state transition and state output distributions respectively.

This paper proposes a new probabilistic parameter $e_t = P(x_t, y)$ for describing how a typical pattern y ends in state $x_t = i$ at time t followed by nothing. It also satisfies the stochastic constraints:

$$e_i \geq 0 \quad \text{and} \quad \sum_{i=1}^N e_i = 1. \quad (1)$$

Alternative definition based on a different modeling assumption would be an exit from state i to an external sink. In this case the final transition in each is assigned a probability that satisfies $\sum_{i=1}^N a_{ij} + e_i = 1$. This extension is similar to the model with constraint in (1), but is more likely to lead to less discriminative model particularly when Viterbi algorithm that computes the most likely state sequence is used for classification.

Historically Markov chains have been studied a lot with a view to improving the accuracy of HMM modeling [3]. But this feature of ending behavior has never been studied. This paper sets the framework for the theory and presents an efficient inference and a training algorithms based on dynamic programming and expectation-maximization. In order to justify the proposed model augmentation, a set of experiments are conducted and analyses are made. The proposed model is also applied to Hangul characters in which two or three alphabets are written sequence.

III. MODEL INFERENCE

Given an observation sequence $O = o_1 o_2 \dots o_T$, the likelihood of model λ is computed as follows:

$$\begin{aligned} P(O|\lambda) &= \sum_X P(O, X|\lambda) \\ &= \sum_X \prod_{t=1}^{T-1} P(x_{t+1} | x_t, \lambda) e_{x_T} \prod_{t=1}^T P(o_t | x_t, \lambda) \end{aligned} \quad (2)$$

where $X = x_1 x_2 \dots x_T$ is a Markov chain. Note that there is a probability factor e_{x_T} at the end of the chain of transitions. The latter factor explicitly concludes the process inside the model.

Now let us consider a set of HMMs concatenated into a sequence where a model k is concatenated to a preceding model $k-1$ and a trailing model $k+1$. Refer to Fig. 1 where HMMs are connected via dummy nodes as shown in big circles. Dummy nodes take the role of a sink for the preceding HMM and the environment for the ensuing

HMMs. Either way they do not belong to any model and do not generate any symbol. They are introduced just for computational convenience.

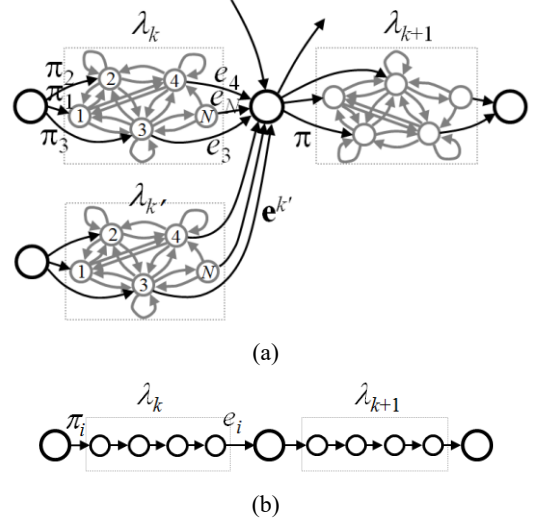


Fig. 1. Network of HMMs. (a) General network, (b) a simplified linear network for ease of explanation.

Then we can compute the forward and backward probabilities of Baum et al.'s [1] as follows:

$$\tilde{\alpha}_t^k(i) = \alpha_t^k(i) + \left[\sum_h \alpha_{t-1}^{k-1}(h) e_h^k \right] \pi_i^k b_i^k(o_t), \quad (3)$$

$$\tilde{\beta}_t^k(i) = \beta_t^k(i) + e_i^k \sum_j \pi_j^{k+1} b_j^{k+1}(o_{t+1}) \tilde{\beta}_{t+1}^{k+1}(j), \quad (4)$$

where the first factors in the right hand side are the standard forward and backward probabilities: $\alpha_t^k(i)$ and $\beta_t^k(i)$. They are the well-known forward and backward variables introduced for efficient computation free of repeated calculation [2].

The EM algorithm for the model is based on the following posterior estimators with regard to the latent state variables:

$$\gamma_t^k(i) = \Pr(x_t = i | O, \lambda_k) = \frac{1}{P} \tilde{\alpha}_t^k(i) \tilde{\beta}_t^k(i), \quad (5)$$

$$\xi_t^k(i, \$) = \frac{1}{P} \tilde{\alpha}_t^k(i) e_i^k \sum_j \pi_j^{k+1} b_j^{k+1}(o_{t+1}) \tilde{\beta}_{t+1}^{k+1}(j), \quad (6)$$

where '\$' denotes the sink or the outside of the model. Following the logic of parameter transformation[2], we can easily derive the resulting formulae. Here only the formula for the new parameter will be explicitly presented:

$$\hat{e}_i^k = \frac{\xi_{T_k}^k(i, \$)}{\sum_i \xi_{T_k}^k(i, \$)}, \quad (7)$$

where T_k is the time after which a Markov chain exits the model k .

[Theorem] The algorithm presented through Equations (3)

$\sim (7)$ is guaranteed to converge.

Proof. Following the reasoning of Baum et al. and using Jensen's inequality, it is straightforward to show that $P(O|\lambda^{(n+1)}) \geq P(O|\lambda^{(n)})$ with equality when $\lambda^{(n+1)} = \lambda^{(n)}$.

When multiple samples are used for training, the maximum likelihood estimate is simply given by the following relation:

$$\hat{e}_i^k = \frac{\sum_t \xi_t^k(i, \$)}{\sum_t \sum_i \xi_t^k(i, \$)}, \quad i = 1, \dots, N_k. \quad (8)$$

IV. EXPERIMENTS AND DISCUSSION

4.1. Experimental set up and Dataset

On-line handwriting is often the right tool for data entry using a stylus on a digitizer tablet [4], [5]. The design of the proposed model for Hangul syllable handwriting has been borrowed from the earlier work [6]. But all the component HMMs have been redeveloped and all the inference algorithms have been modified to include the proposed augmentation.

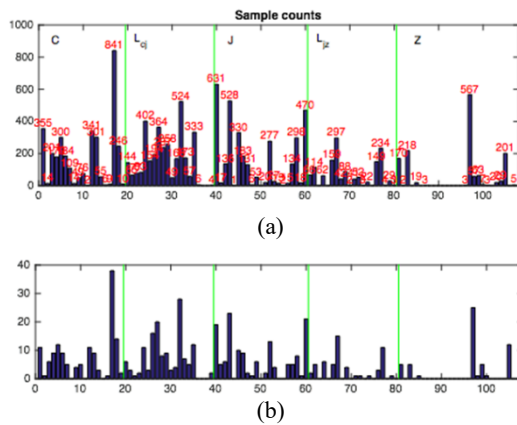


Fig. 2. The histogram of the letters in the training set. (a) Severe imbalance is apparent. Half the models lack samples. (b) Prior samples. There are five groups of handwriting components; C, J, Z and two-type of ligatures in between.

The training set consists of 2886 samples for about 260 character classes. Each character consists of two or three alphabets: initial consonant(C), vowel (J), and with or without a final consonant (Z). Notationwise, a character is organized as either C-J or C-J-Z. But handwriting often involves ligatures that link between strokes. Explicit modeling of them with separate HMMs provides us a great advantage in designing a

model for cursive scripts. There are over 100 HMMs including 20 ligature types each with different context. See Fig. 2(b). They were trained with whole character samples. Although the HMMs in the network were designed, they were first trained together in a network and learned the component boundaries simultaneously. With a small training set, this, however, didn't turn out well [7]. So the model has been primed by prior samples, about 10% of the training set selected at random from the training set. See Fig. 2(c). Their letter boundaries were manually added.

4.2. Model Inference and Handwriting Recognition

The first set of tests is about the behavior of dynamic programming-based inference algorithms. Fig. 3(a) shows the Viterbi pass probabilities for the character '한' (C-g, Ligature, J-k, another Ligature, and Z-q). The component models are aligned left-to-right (space), and the time sequence proceeds from top to bottom (time). The initial states are at the top-left corner and are brightest implying a high probability, where the intensity is rendered in the log scale. It also shows the most likely state sequence in knotted curves given an input handwriting. On the other hand Fig. 3(b) presents the 'best' posterior path for reference. It is based on the posterior probabilities $\gamma_t(i)$ in Eq. (5) that takes both the left and right context into account. Note that the Viterbi path in Fig. 3(a), although computed using only the left (or past) context, is very close to and indistinguishable from the best one. This is often the case in practice, thus providing a justification for the use of Viterbi algorithm for path-based recognition.

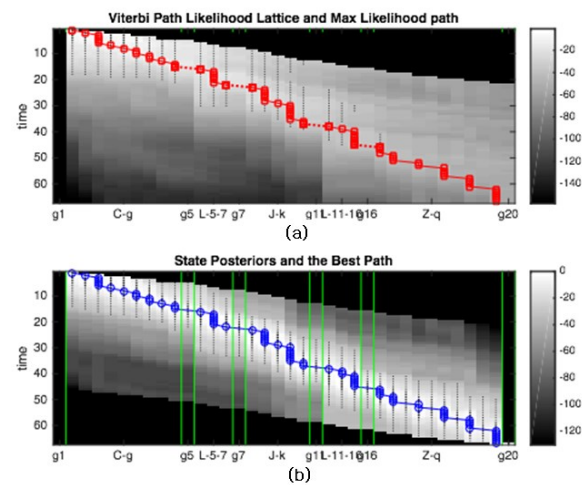


Fig. 3. The histogram of the characters in the training set.

Handwriting recognition performance has been

measured over a set of 3,292 characters and recorded 90.2% with a single best hypothesis at twenty training iterations. When up to five best candidates are considered, the figure rose to 98.7%. Refer to Table 1. Considering a limited number and unbalanced distribution of samples, the performance is deemed promising. But it is a bit disappointing in that the character samples were based on simple math-related texts and the number of character classes is small. In the current experiments, only grapheme-level bigram language model was used. But we expect that the performance will make a big jump once we employ character-level statistics and dictionaries.

Table 1. Online Hangul handwriting recognition compared. Performance with a single (five best) candidates (%).

Models	#training iterations	
	10	20
Pronosed model (LR*, w exit arcs)	89.25 (98.40)	90.22 (98.70)
Standard model (LR, w/o exit arcs)	87.85 (97.51)	88.73 (97.93)
Standard model (ergodic, w/o exit arcs)	32.87 (50.89)	51.34 (66.50)

* LR : left-right topology

2.3. System Behavior

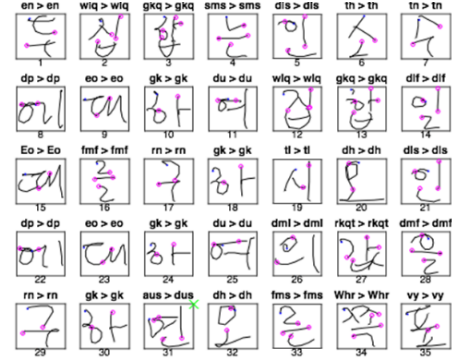
Fig. 4(a) shows a score of character samples tested on the recognizer developed above. The small circles mark the component boundaries proposed by the system. Some cursive samples (numbered 8, 9, 11, and 20) show two or more strokes connected into one making boundary detection trickier. However, all were correctly recognized except for the 31st sample.

Fig. 4(b) show a selection of noteworthy samples returned by the recognizer based on the proposed HMMs. Upper row shows samples recognized correctly while the lower row represents failures on tough cases. Some are ambiguous while others imply simple mistakes possibly due to insufficient training. Nevertheless, the segmentation points look highly intuitive regardless of misrecognition.

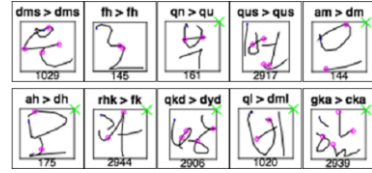
V. CONCLUSION

This paper presents an idea of enhancing the HMM behaviors in networked HMMs for modeling concatenated sequence of patterns with ambiguous boundaries. The concept of sinks as distinct from states is not new. But it is formally modeled in a different way rendering the HMMs more discriminative. Just like the initial distribution, the explicit modeling of exiting arcs is conceptually viable.

Experiment has confirmed that the proposed model (1) enables the detection of highly intuitive pattern boundaries and (2) leads to a significant error reduction of 13.22%.



(a)



(b)

Fig. 4. Recognition and segmentation examples: (a) simple cases, (b) difficult samples.

Acknowledgement

This work was supported by a Research Grant of Pukyong National University (2019).

REFERENCES

- [1] L.E. Baum and T. Petrie, "Statistical Inference for probabilistic functions of finite state Markov chains," *Ann. Math. Statist.*, vol. 37, pp.1554-1563, 2009.
- [2] L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol.77, pp.257-285, 1989.
- [3] B.-K. Sin and J.H. Kim, "Nonstationary hidden Markov model," *Signal Processing*, vol. 46, pp.1857-1869, 1995.
- [4] D. Keysers, T. Deselaers, H. A. Rowley, Li-Lun Wang, and V. Carbune, "Multi-Language Online Handwriting Recognition," *IEEE Trans. PAMI*, vol. 39, no. 6, pp.1180-1194, June 2017.
- [5] A. Graves et al. "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. PAMI*, vol. 31, no. 5, pp.855-868, 2009.
- [6] B.-K. Sin and J. H. Kim, "Ligature Modeling for Online Cursive Script Recognition," *IEEE Trans. PAMI*, vol. 19, no. 6, pp.623-633, June 1997.
- [7] N. Fabrice, S. Gang, and J. J. Lee, "Training Data Sets Construction from Large Data Set for PCB Character Recognition," *Journal of Multimedia Information System*, vol. 6, no. 4, pp.225-234, 2019.