Proceedings of the Institute of Acoustics

STOCHASTIC MODELLING OF TEMPORAL INFORMATION IN SPEECH WITH FUZZY VECTOR QUANTISATION

Jianing Dai, Jon E.M. Tyler, and Iain G. MacKenzie

School of Computer and Mathematical Sciences, Robert Gordon's Institute of Technology, St. Andrew Street, Aberdeen AB1 1HG, UK

1. INTRODUCTION

Hidden Markov modelling (HMM) of speech signals inherits the property of temporal information ignorance of statistical modelling techniques. As a result, the time ordering of successive acoustic observations is partially disregarded by the HMM, such that a number of different observation sequences may share the same model trained only by one of them [5]. More seriously, we found that HMMs failed to recognise a considerable proportion of their own training utterances [1, 5]. A solution, the temporal Markov model (TMM) [5], has been proposed to compensate for the information loss. By adding the parameters of the TMM to the HMM, the conventional hidden Markov model can be extended to a time-ordered HMM. The model can recognise its training utterances (3997 utterances of 52 speakers) with accuracy of 99.2% compared with the HMM's 89.4%, with the codebook size of 256. However, the model has little tolerance to unseen data, which resulted in a relatively minor improvement (from 78.5% to 80.4%, tested on 3978 utterances of 52 unknown speakers) in speaker-independent recognition. In this paper, we present a method to improve the recognition of unseen data by applying fuzzy vector quantisation [10] to the model training in order to smooth parameters of the temporal Markov model. The smoothed model has been tested on the same data as before, and the results are compared with those of the unsmoothed one.

2. THE HMM'S TEMPORAL MODELLING PROBLEM

A hidden Markov model with discrete output distributions is defined on a state set, $S=\{s_1, s_2, \ldots, s_N\}$, and an output symbol set, $V=\{v_0, v_1, \cdots, v_{M-1}\}$. The symbol set, V, can be the codebook in modelling of speech signals. An observation sequence, denoted by $O=\{O_1,O_2, \cdots, O_T\}$, is assumed to be generated by its underlying Markov chain, $X=\{X_1,X_2, \cdots, X_T\}$, where $X_t \in S$ and $O_t \in V$. If a left-to-right structure is applied to the Markov chain, the Markov process is forced to start at s_1, and to end in s_N, while its activity is restricted to either staying in the present state, or moving on to the next state. If the process occupies state s_j for D_j times, the number of observations generated by s_j is D_j as well. Since observations are state-dependent, an observation sequence, O, emitted by N states can be divided into N segments accordingly, and expressed as $O=\{O_1,O_2, \cdots, O_N\}$ without altering the temporal ordering of its components. With this notation, the properties of a left-to-right structured hidden Markov model are described by a set of parameters, an initial distribution vector $\pi=[1,0,\cdots,0]$, a state transition matrix $A=[a_{ij}]$, and a set of output distributions , $B=[b_j(k)]$. A word model consisting of the parameters is denoted by $\lambda=\{\pi, A, B\}$. The probability of model λ given sequence O can be written as [1]

$$P_O(\lambda) = \sum_Q P_O(\lambda,Q_i) = \sum_Q [\pi_1 b_1(O_1) \prod_{t=1}^{T-1} a_{j_t j_{t+1}} b_{j_{t+1}}(O_{t+1})] , \qquad (1)$$

where subscript j_n indicates that the hidden Markov process occupies state s_j at time n, and Q stands for the collection of all possible state sequences, $\{Q_1,Q_2, \cdots, Q_J\}$.

[1] We adopt the notation of [2, 3] throughout the paper , since we are only interested in finding an existing model, $\bar{\lambda}$, which maximises $P_O(\lambda)$.

To be specific, we rewrite output distributions in the form of

$$b_j(k) = P(O_t=v_k \,|\, X_t=s_j)\,, \quad 1\le j\le N,\ 0\le k\le M-1, \tag{2}$$

$$= P(O_t=v_k \,|\, O_t\in \mathbf{O}_j)\,.$$

From Eq.(2), we can see that the temporal ordering between observations within an acoustic segment, $\mathbf{O}_j$, is disregarded, although the segmental ordering is represented by π and A. As we know, a state usually generates a number of observations. These observations may not be the same, i.e., their codewords are different. If we alter the ordering in which they were generated, a new pattern is produced by the same state. In other words, state s_j does not uniquely emit its observation patterns. To clarify the problem, we denote the number of different codewords observed in $\mathbf{O}_j$ by n_j, and the number of observations belonging to the kth category of the n_j codewords by I_{jk}. So, the number of total observations in $\mathbf{O}_j$ is

$$D_j = \sum_{k=1}^{n_j} I_{jk}\,. \tag{3}$$

The number of different permutations based on the segment is

$$I_j = \frac{D_j!}{I_{j1}!I_{j2}!\cdots I_{jn_j}!}\,. \tag{4}$$

Since the hidden Markov chain has N states, it may generate

$$I = \prod_{j=1}^{N} I_j \tag{5}$$

different observation sequences. In other words, these different sequences share the same model. Obviously, this property is not desirable for the task of speech recognition, since we only expect a model to represent the properties of specific utterances. The problem is more serious when a larger number of utterances are used to train a model. The number of possible utterances represented by the model is much bigger than the actual number of training utterances (although this does not follow !!!). If some of these sequences happen to be the same as some of the sequences used to train another model, there may be little distinction between the two models. The situation is more likely to happen in large vocabulary speaker-independent recognition than in small vocabulary speaker-dependent recognition.

According to Eqs.(4, 5), the condition for a model to uniquely represent its training pattern, $\mathbf{O}$, is $I=1$, which implies $I_j=1$ for all j. However, the condition for $I_j=1$, in turn, is $n_j=1$. In such case, a state is allowed only to generate the same observations. This directly leads to a solution of using a large number of states. However, adding more states does not guarantee to increase the recognition accuracy [8,9,11]. Rabiner *et al* [9] pointed out that there is very little gain in using HMMs with more than five or six states when the left-to-right structure with double skips is used. An explanation for the limited number of states is that features and duration of different utterances vary enormously. For instance, the duration of an utterance in the database used for following experiments varies from 18 to 126 vectors. When using a larger amount of utterances from a number of speakers to generate speaker-independent word models, it is unlikely that an algorithm could be found which is able to assign the same observations to a state in order to make $n_j=1$ for all j, since a fixed N is commonly used for all words of the vocabulary in hidden Markov modelling.

3. THE TIME-ORDERED HIDDEN MARKOV MODEL

The time-ordered hidden Markov model (TOHMM) consists of the parameters of a left-to-right hidden Markov model generating $\mathbf{O}$, and the parameters of a temporal Markov model (TMM) designed to model the temporal

ordering of O_ts. The temporal Markov chain is a stationary 1-step Markov chain, and defined directly on the observation sequence, O. Therefore, codebook V is the state set, and each codeword is a TMM state. To distinguish states of the TMM from those of the HMM, we call them codeword states, or codewords for convenience. If a v_i is observed in the sequence at time t, i.e., $O_t=v_i$, we say that codeword v_i is visited. The temporal Markov model is described by an initial probability distribution, $\theta = [\theta_0,\theta_1, \cdots, \theta_{M-1}]$, where

$$\theta_i = P(O_1=v_i) , \tag{6}$$

and a transition probability matrix, $E=\{\xi_{ij}\}$, where

$$\xi_{ij} = P(O_{t+1}=v_j \,|\, O_t=v_i) . \tag{7}$$

To be complete, we define a null codeword, v_M, to make $\sum_j \xi_{ij} = 1$, in case that a codeword, v_i, does not appear in training utterances of a word, where $\xi_{iM} = 1$. For convenience, we use $\lambda'=\{\theta,E\}$ to denote the model. We also use the notations, $\theta(O_1)$ and $\xi(O_t,O_{t+1})$ to represent the function relationship in Eqs.(6, 7) respectively, when we are not interested in a particular state, v_i, which O_t equals to. The probability of model λ', given O, is

$$P_O(\lambda') = \theta(O_1)\prod_{t=1}^{T-1}\xi(O_t,O_{t+1}) . \tag{8}$$

Now, we denote the time-ordered HMM by a set of parameters, $M = \{\pi, A, B, \theta, E\}$. The probability of O being emitted by a hidden state sequence, Q_i, of the TOHMM is

$$\begin{aligned} P_O(M,Q_i) &= \pi_1 b_1(O_1)\theta(O_1)\prod_{t=1}^{T-1} a_{j_t j_{t+1}} b_{j_{t+1}}(O_{t+1})\xi(O_t,O_{t+1}) \qquad (9)\\ &= \left[\pi_1 b_1(O_1)\prod_{t=1}^{T-1} a_{j_t j_{t+1}} b_{j_{t+1}}(O_{t+1})\right]\cdot\left[\theta(O_1)\prod_{t=1}^{T-1}\xi(O_t,O_{t+1})\right] \\ &= P_O(\lambda,Q_i)\cdot P_O(\lambda') . \end{aligned}$$

Since all of possible hidden state sequences, $Q=\{Q_i\}$, should be considered, the probability of O being generated by a TOHMM is written as

$$P_O(M) = \sum_Q P_O(\lambda,Q_i)\cdot P_O(\lambda') = P_O(\lambda)\cdot P_O(\lambda') . \tag{10}$$

Eq.(10) states that $P_O(M)$ can be computed from the product of two independent probabilities, $P_O(\lambda)$ and $P_O(\lambda')$. It also indicates that the parameter estimation of the TOHMM can be carried out by estimating a HMM and a TMM separately.

4. FUZZY VECTOR QUANTISATION APPLIED TO SMOOTHING MODELS

We denote the acoustic vector sequence of an utterance by $x=\{x_1, x_2, \cdots, x_T\}$. An x_t can be vector-quantised to a codeword, v_i, in codebook V. We use $O=\{O_1,O_2, \cdots, O_T\}$ to denote the quantised sequence of x. The conventional vector quantiser assigns an unique codeword, v_i, to an acoustic vector, x_t. The distortion between x_t and v_i can be denoted by $d(x_t,v_i)$. Instead of assigning only one codeword to x_t, the fuzzy vector quantiser quantises the acoustic vector to L nearest codewords each of which is associated with a distortion to x_t. The degree of attachment of x_t to codeword v_i is defined as [13]

$$\gamma(v_i \mid x_t) = \left\{ \sum_{k=1}^{L} [d(x_t, v_i)/d(x_t, v_k)]^{1/(F-1)} \right\}^{-1} \quad . \qquad (11)$$

where F is a constant called fuzziness, and L is the number of codewords nearest to x_t. To simplify computation, we choose $F=2$.

Owing to very limited data available for training models, statistical models are less accurate in predicting unseen data than in predicting data already seen. To get around the problem, we can smooth the models with additional information which is closely related to training utterances. By assuming that the acoustic features of an incoming utterance may be close to the features of one of our training utterances, we can vector-quantise x to L nearest observation sequences

$$\mathbf{O}_m = \{O_1^{(m)}, O_2^{(m)}, \ldots, O_T^{(m)}\} \quad , \quad m = 1, 2, \ldots, L \, ,$$

instead of one which is $\mathbf{O}_1 = \mathbf{O}$. At time t, an observation, $O_t^{(m)}$, of each of the sequences is associated with its affinity, $\gamma_m(t) = \gamma(O_t^{(m)} = v_i \mid x_t)$. By taking the mean of the affinity of each of the sequences over time t, we can assign a weighting factor, γ_m, to each of the sequences.

To smooth the temporal Markov model, we would like to define following probabilities. The initial distribution of the m th sequence is

$$\theta_i^{(m)} = P(O_1^{(m)} = v_i)\gamma_m \quad , \quad 0 \le i \le M-1, \text{ and } 1 \le m \le L. \qquad (12)$$

The transition probability from v_i of $\mathbf{O}^{(m)}$ at time t to v_j of $\mathbf{O}^{(n)}$ at time $t+1$ is defined as

$$\xi_{ij}^{(m,n)} = P(O_{t+1}^{(n)} = v_j \mid O_t^{(m)} = v_i)\gamma_m \gamma_n \quad , \quad 0 \le i, j \le M-1, \text{ and } 1 \le m, n \le L \, . \qquad (13)$$

With these probabilities, parameters of the temporal Markov model defined by Eqs.(6, 7) can be replaced by following definitions, respectively

$$\theta_i = \sum_{m=1}^{L} \theta_i^{(m)} \, , \text{ and} \qquad (14)$$

$$\xi_{ij} = \sum_{m=1}^{L} \sum_{n=1}^{L} \xi_{ij}^{(m,n)} \, . \qquad (15)$$

The resulting parameters then are smoothed by a floor, $\varepsilon = 10^{-4}$. Parameters of a HMM can be smoothed by the interpolated co-occurrence smoothing [12] in addition to the floor smoothing.

5. EXPERIMENTS

5.1. Speech Data and Model Estimation

Preliminary experiments have been conducted on speaker-independent isolated utterance recognition to compare performances of the HMM, TMM, and TOHMM, as well as the fuzzied TMM (FTMM) and TOHMM (FTOHMM), using discrete probability distributions. The speech data used in the experiments are the BT Alphabetic database provided by British Telecom Research Laboratories, consisting of 7975 isolated spoken utterances of the letters of the British English Alphabet. These utterances were spoken by 104 talkers each of whom provided 3 repetitions of each letter. After being sampled at 20kHz using a 16bit A/D converter and endpointed by visual inspection, the utterances were further processed by a 27-channel filter bank using a Hamming window of 20ms with 50% overlap. In experiments, the data were divided into two sets: a training set and a testing set. The training set contains 3997 utterances from 52 talkers (26 male and 26 female). The testing set gives 3978 utterances from the remaining 52 talkers (27 male and 25 female).

We used the K-means clustering algorithm to generate a codebook of 256 entries. All of 66530 vectors for the codebook generation were picked from the training set at random. In other words, the codebooks are speaker-independent. A 5-state left-to-right model structure was designated to HMMs. The training of HMMs adopted the standard forward-backward algorithm. Then, parameters of HMMs were smoothed by a floor, $\varepsilon=10^{-4}$, since using the interpolated co-occurrence smoothing yielded a slightly higher error rate in recognition. Parameters of TMMs were estimated by accumulating the counts of different codewords. Then, the counts were normalised to satisfy the constraint of the stochastic matrix, and then smoothed by a floor, $\varepsilon=10^{-4}$. To compare performance of the FTOHMM with that of the TOHMM, we trained two temporal Markov models; one by a single observation sequence of each training utterance, and the other by 3 possible sequences of each training token, i.e., $L=3$. The number of training utterances for each model is 154 on average.

5.2. Results

The recognition task was carried out on both the training and testing sets, as well as the E-set (B, C, D, E, G, P, T, V). The E-set consists of 1219 utterances from the testing set. We believe that the test on the training set is essential for comparing robustness of different models, since all utterances in this set are predictable, i.e., they satisfy the assumption of Statistics as far as the models concerned. If a model performs badly on this set, the robustness of its modelling is questionable. The testing set is designed to test the generality of a model. The tolerance of a model to unseen data will be shown by the test on this set. We used the Viterbi algorithm for computing $P_O(\lambda)$. The results are shown in Table 1.

Table 1 consists of recognition results from both the training data set (TRN) and the testing set (TST), each of which contains results from HMM, TMM, TOHMM, FTMM, and FTOHMM. In the table we can see that HMMs failed in recognising 10.6% of their own training utterances, i.e., HMMs made 424 recognition errors out of 3997 training tokens. In contrast, the performance of TMMs is surprisingly good. Their failure rate is much lower than HMMs. For instance, TMMs only have less than 0.1% failure on the whole training set, which corresponds to 2 errors. This figure shows how important the temporal ordering of observations is for recognising spoken utterances. Not surprisingly, TOHMMs give a reasonable error rate of 0.8%, which is slightly inferior to TMMs. The error rates for fuzzied temporal Markov models (FTMMs) and fuzzied time-ordered hidden Markov models (FTOHMMs) are 1.7% and 4.6%, respectively. Both of them are lower than HMMs, and higher than that of TMMs and TOHMMs.

As expected, recognition results of all models on the testing set are not as good as the ones on the training set. Due to little tolerance to unseen data, TMMs performed worse than HMMs. However, TOHMMs still give higher recognition accuracy than both HMMs and TMMs. For example, the error rate of TOHMMs is 19.6%, 1.9% lower than that of HMMs, and 8.1% lower than that of TMMs. Moreover, FTOHMMs give an even lower error rate of 17.4%, which is 2.2% lower than TOHMMs and 4.1% lower than HMMs. Obviously, the

Table 1.
Comparison of recognition accuracy between different models.
In the table, 'TRN' stands for the training set, and 'TST' for the testing set.

Recognition Error Rate (%)					
Vocabulary	HMM	TMM	TOHMM	FTMM	FTOHMM
Alphabet (TRN)	10.6	0.1	0.8	1.7	4.6
Alphabet (TST)	21.5	27.7	19.6	21.1	17.4
E-set (TST)	34.0	39.5	29.9	28.4	25.6

better performance of FTOHMMs is mainly due to fuzzied TMMs, which can be observed from FTMMs' error rate of 21.1%, 6.6% lower than TMMs. The test on the E-set confirms the superior performance of FTOHMMs with FTOHMMs' error rate of 25.6% compared to HMMs' 34.0%, which indicates a greater distinction between different models can be shown by a more difficult recognition task.

5.3. Comparison with Other Approaches

There are similar approaches [4, 6, 14] which intend to cope with the correlation between observations. Brown [4] proposed a model with Conditional Gaussian distributions (CGHMM), while Kenny *et al* [6] suggested a Linear Predictive HMM (LPHMM). We compare our results as well as the testing condition with those of the approaches except [14] which did not mention related experiments. We listed both test conditions and results in Table 2. We use names of the first authors to stand for related experiments. From this table, we can see that our test condition is more difficult that those of others. First of all, the testing utterances are from both male and female speakers, instead of male speakers only. Secondly, the test was conducted in speaker-independent mode, instead of speaker-dependent or multi-speaker mode. Finally, our vocabulary is relatively larger than the one of [4].

Now, let us look at the performance of these different models. Since test conditions of these approaches are different, we could not compare their performance simply by their recognition rates. Instead, we listed the recognition rate of a modified model against that of its reference model. Then, we calculated the recognition improvement made by the new model. Since both [4] and [6] provided two sets of results using different dimensional observation vectors, we listed all of them in the table to avoid any misunderstanding. For example, we use CGHMM/GHMM to represent the conditional Gaussian HMM proposed in [4] against its reference model, the Gaussian HMM. The corresponding recognition rates, in percentage, are 84.1/88.7(12-d) and 73.1/89.1(22-d) respectively. The number in the brackets represents the number of dimensions of observation vectors used in the experiment. So using 12-dimensional vectors as input, the recognition rate of the CGHMM is 84.1% against 88.7% of the GHMM, which indicates that the improvement of the CGHMM, in terms of recognition rate, is −4.6%. Similarly, we have −16.0% improvement of the CGHMM with 22-dimensional

Table 2.
Comparison of experimental results given by three similar approaches.
In the table, GHMM stands for the HMM with Gaussian distributions.

Test Condition	Brown [4]	Kenny [6]	Dai
Vocabulary	E-set [2]	60000 words	Alphabet set
No. of words	886	399	3978
No. of speakers	100 (Male only)	1 (Male)	27 (M) + 25 (F)
Test mode	multi-speaker	speaker-dependent	speaker-independent
Performance	CGHMM / GHMM	LPHMM / GHMM	TOHMM / HMM
Recognition rate (%)	84.1 / 88.7 (12-d)	81.0 / 79.4 (8-d)	82.6 / 78.5
	73.1 / 89.1 (22-d)	78.7 / 85.5 (15-d)	
Improvement (%)	−4.6 (12-d) or −16 (22-d)	1.6 (8-d) or −6.8 (15-d)	4.1

[2] This E-set consists of 9 letters (B, C, D, E, G, P, T, V, Z).

vectors as input. Disappointingly, both sets of results are in favour of the GHMM, rather than the CGHMM.

The results provided by [6] seem inconsistent under different signal processing conditions. The LPHMM gave a minor improvement of 1.6% as 8-d vectors (7 cepstral coefficients and 1 differenced loudness) were used as input, while it gave poorer performance than its reference model under a higher precision of signal processing, e.g. 15-d feature vectors (7 cepstral coefficients, 7 differenced cepstral coefficients, and 1 differenced loudness). However, differenced parameters have been proven to be a very useful information for recognition [7]. The improvement based on disregarding this information cannot be regarded significant. Therefore, we can only have the impression that the LPHMM performed worse than the HMM with conventional multivariate Gaussian distributions. One thing we have noticed from the results is that both the CGHMM and LPHMM performed worse when using relatively more accurate information provided by the signal processing than they did when using less accurate information, which is just the opposite of the performance of their reference model, the GHMM. We believe that at least the performance of a robust model should not deteriorate as more useful information is provided. Since the experimental results of both [4,6] were disappointing, we can only conclude that correlation may not be an appropriate representation for the temporal information existing between successive observations.

In contrast to [4,6], the TOHMM has improve the recognition rate by 4.1% on the most difficult task among the three. Its performance is consistent under different vocabularies and testing data. All of its results in Table 1 have shown that its performance is considerably better than its reference HMM. Furthermore, the number of testing utterances for our experiments is much bigger than those of [4,6], which indicates that the results provided a rather convincing evidence of the TOHMM's performance.

6. CONCLUSIONS

The temporal information embedded in speech signals plays a significant role in speech recognition. The discriminating power of the HMM can be enhanced by incorporating the parameters of the temporal Markov model into the HMM, which results in the time-ordered hidden Markov model. However, the temporal Markov model gives little tolerance to unseen data. As a result, a minor improvement on recognition accuracy was made on utterances from unknown speakers, e.g. the testing set, in comparison with a significant improvement on the training utterances.

Owing to limited training data, a model-smoothing method using fuzzy vector quantisation has been suggested to make the temporal Markov model more tolerant to unseen data. The effectiveness of the smoothing has been demonstrated by experiments, which have shown a superior performance of fuzzied TOHMMs over other models. Meanwhile, the discriminating power of TOHMMs reduces in testing of recognition accuracy on the training data, which indicates that a compromise must be made between a model's high discriminating power and its tolerance to unseen data. Nevertheless, the smoothed time-ordered hidden Markov model has shown a significant superiority over the conventional hidden Markov model by improvement of 4.1% on recognition accuracy in modelling of speech signals.

7. ACKNOWLEDGEMENT

The authors wish to thank British Telecom Research Laboratories for providing the BT Alphabetic database for the experiments.

8. REFERENCES

[1] L. R. Bahl, P. F. Brown, P. V. de Souza, and R. L. Mercer, "A New Algorithm for the Estimation of Hidden Markov Model Parameters," *Proceedings of IEEE ICASSP-88* 1 pp. 493-96 (Apr., 1988).

[2] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains," *The Annals of Mathematical Statistics* 41, No. 1 pp. 164-171 (1970).

[3] L. E. Baum, "An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes," *Inequalities* 3 pp. 1-8 (1972).

[4] P. F. Brown, "The Acoustic-Modeling Problem in Automatic Speech Recognition," Ph.D. Thesis, Department of Computer Science, Carnegie-Mellon University (1987).

[5] J. Dai, I. G. MacKenzie, and J. E. M. Tyler, "The Temporal Modelling Problem of Hidden Markov Models for Speech Recognition," *presented at the Int. Conf. on Signal Processing '90/Beijing*, (Oct., 1990).

[6] P. Kenny, M. Lennig, and P. Mermelstein, "A Linear Predictive HMM for Vector-Valued Observations with Applications to Speech Recognition," *IEEE Trans. on Acoustics, Speech, and Signal Processing* 38, No. 2 pp. 220-225 (Feb., 1990).

[7] K. F. Lee, "Large-Vocabulary Speaker-Independent Continuous Speech Recognition: The SPHINX System," Ph.D thesis, Carnegie-Mellon University, Pittsburgh (1988).

[8] K. M. Ponting, "A Statistical Approach to the Determination of Hidden Markov Model Structure," Technical report, Speech Research Unit, RSRE (1988).

[9] L. R. Rabiner, S. E. Levinson, and M. M. Sondhi, "On the Application of Vector Quantization and Hidden Markov Model to Speaker Independent, Isolated Word Recognition," *The Bell System Technical Journal* 62 pp. 1075-1105 AT & T, (Apr., 1983).

[10] E. H. Ruspini, "Numerical Methods for Fuzzy Clustering," *Information Sciences* 2 pp. 319-350 (1970).

[11] M. J. Russell and A. E. Cook, "Experiments in Speaker-Dependent Isolated Digit Recognition Using Hidden Markov Models," Technical Report, Speech Research Unit, RSRE (1986).

[12] R. Schwartz, O. Kimball, F. Kubala, M. W. Feng, Y. L. Chow, C. Barry, and J. Markhoul, "Robust Smoothing Methods for Discrete Hidden Markov Models," *Proc. of IEEE ICASSP-89* 1 pp. 548-551 (May, 1989).

[13] H. P. Tseng, M. J. Sabin, and E. A. Lee, "Fuzzy Vector Quantization Applied to Hidden Markov Modeling," *Proceedings of ICASSP-87*, pp. 641-644 (April, 1987).

[14] C. J. Wellekens, "Explicit Time Correlation in Hidden Markov Models for Speech Recognition," *Proceedings of IEEE ICASSP-87* 1 pp. 384-386 (Apr., 1987).