PAPER *Special Issue on Speech Information Processing*

A Hybrid HMM/BN Acoustic Model for Automatic Speech Recognition

Konstantin MARKOV[†], ***Nonmember and*** **Satoshi NAKAMURA**[†], ***Member***

SUMMARY In current HMM based speech recognition systems, it is difficult to supplement acoustic spectrum features with additional information such as pitch, gender, articulator positions, etc. On the other hand, Bayesian Networks (BN) allow for easy combination of different continuous as well as discrete features by exploring conditional dependencies between them. However, the lack of efficient algorithms has limited their application in continuous speech recognition. In this paper we propose new acoustic model, where HMM are used for modeling of temporal speech characteristics and state probability model is represented by BN. In our experimental system based on HMM/BN model, in addition to speech observation variable, state BN has two more (hidden) variables representing noise type and SNR value. Evaluation results on AURORA2 database showed 36.4% word error rate reduction for closed noise test which is comparable with other much more complex systems utilizing effective adaptation and noise robust methods.

key words: *automatic speech recognition, bayesian networks, acoustic modeling, hmm*

1. Introduction

For many years, since the introduction of the HMM for speech recognition [1], [2], observations conditional distributions $P(y|Q)$ for each state Q have been modeled most often by mixture of parametric probability density functions (pdf). Gaussian as well as Laplacian pdfs are commonly used for this purpose. Later, a hybrid HMM/NN systems were proposed [3] where Neural Networks (NN) are used to estimate HMM state likelihoods given input observation. In most of the cases, features extracted from speech spectrum form these observations. However, research in speech recognition has shown that using only these features is not enough to achieve high system performance. Thus, many researchers have tried to include additional features representing some other knowledge into their HMM systems. For example, in [4] multi-space probability distribution is proposed for modeling additional pitch information. But, in almost all the cases, different approach is taken depending on the properties of the additional feature. There is no common, flexible enough framework to deal with this problem.

Recently, the Bayesian Networks (BN) have attracted researchers' attention as an alternative to the HMM. BN are well known and studied in Artificial Intelligence research field, but in speech recognition, they are relatively new research topic. Bayesian Networks can model complex joint probability distributions of many different (discrete and/or continuous) random variables in well structured and easy to represent way. Especially suitable for modeling temporal speech characteristics are the Dynamic BN (DBN)[5]. In some of the first reports on DBN in speech recognition, they were used as word models in isolated word recognition tasks [6], [7]. In these works, DBN are regarded as generalization of the HMM, which in addition to speech spectral information can easily incorporate additional knowledge, such as articulatory features, sub-band correlation, speaking style, etc. In [8], acoustic features are easily supplemented with pitch information within the framework of DBN. Another advantage of the Bayesian Networks is that additional features which are difficult to estimate reliably during recognition may be left hidden, i.e. unobservable. Despite these attractive properties of BN, their application in speech recognition is still limited to small, isolated word recognition tasks. The reason is that existing algorithms for BN parameter learning and inference are not practically suitable for continuous speech recognition (CSR) and especially large vocabulary CSR tasks. Although, an extension of the DBN word model allowing recognition of continuously spoken digits was reported in [9], increasing task vocabulary even to a few hundred words would be computationally prohibitive.

The method we are proposing in this paper aims at utilizing advantages of both HMM and BN while being free from their drawbacks described above. In our approach, HMM and BN are combined together in one hybrid HMM/BN model. In this model, temporal characteristics of speech signal are modeled by HMM state transitions and the BN is used to model HMM state distributions. There is a two level hierarchy in which the BN is at the lower level and the HMM stays at the top level. The advantage of this is that existing recognition algorithms can be used without any modification since this model behaves as a conventional HMM and can be used to model both word and sub-word units which is essential for large vocabulary systems.

This paper is organized as follows. Section 3 describes in detail our hybrid HMM/BN model and several possible BN structures. In Sect.5, we show how to include additional information about noise type and noise SNR using HMM/BN framework and in Sect.6 we

Manuscript received June 30, 2002.

[†]The authors are with the ATR Spoken Language Translation Research Labs.

describe the evaluation of our system on AURORA2 task. Section 7 offers discussion about our approach and some conclusions are drawn in Sect.8.

2. HMM based speech recognition

2.1 Background

The basic problem of speech recognition is to be able to transcribe the sequence of words corresponding to a spoken utterance. The statistical approach to this problem is to find the most probable word sequence given the acoustic data. Thus, we choose the word sequence $W = w_1, w_2, \ldots, w_m$ for which the probability $P(W|X)$ is maximum, where $X = x_1, x_2, \ldots, x_T$ is a sequence of data feature vectors extracted from the acoustic signal. Direct estimation of this probability is difficult, but using Bayes' rule it can be expressed as[†]:

$$P(W|X) = \frac{p(X|W)P(W)}{p(X)} \tag{1}$$

This divides the probability estimation into two parts: *acoustic* modeling, where the data dependent $p(X|W)/p(X)$ is estimated[††], and *language* modeling, in which the prior probability of word sequence $P(W)$ is estimated. This allows us to treat acoustic modeling and language modeling independently. When acoustic modeling is based on HMM, then word sequence is represented by a particular sequence of HMMs. In small vocabulary speech recognition tasks, one HMM model is used per vocabulary word. However, for large vocabularies sub-word unit HMM is adopted since word level HMM models require very large training data set. Commonly used sub-word units are phonemes, although syllables or demisyllables are also often used.

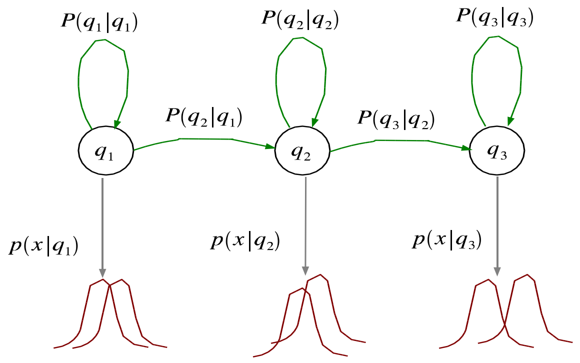


Fig. 1 Three state left-to-right HMM model

Fig.1 shows typical three state left-to-right HMM model commonly used for phoneme representation. Each state $q_i, i = 1, 2, 3$ is associated with probability distribution $p(x|q_i)$ which is usually modeled by mixture of Gaussians. Temporal transitions from one state to another are governed by probabilities $P(q_i|q_j)$.

2.2 HMM training and recognition

Two different algorithms - the Baum-Welch and the Viterbi algorithms, are used for the training of HMM's [10], [11]. In the first one, the probability of producing an acoustic vector sequence is maximized, while the Viterbi algorithm uses only the best path through the model.

The Baum-Welch algorithm iteratively provides HMM parameter estimates that maximize the likelihood of the data $p(X|M)$, where M represents parameter set of HMM [2], [10]. $p(X|M)$ can be computed in terms of joint state and data probability densities:

$$p(X|M) = \sum_{q_1^T \in Q} p(X, q_1^T|M) \tag{2}$$

where Q is the set of all possible state sequences. Direct calculation of Eq.(2) is intractable, but it can be effectively computed by so called forward $\alpha_t(j)$ and backward $\beta_t(i)$ variables:

$$\begin{aligned} \alpha_t(j) &= p(x_1^t, q_j(t)|M) && (3)\\ &= \left[\sum_i^S \alpha_{t-1}(i)a_{ij}\right] p(x_t|q_j(t), M) \\ \beta_t(i) &= p(x_{t+1}^T|q_i(t), M) && (4)\\ &= \sum_j^S a_{ij}p(x_{t+1}|q_j(t+1), M)\beta_{t+1}(j) \end{aligned}$$

where $a_{ij} = P(q_j|q_i, M)$ is the probability of transition from state q_i to state q_j and S is the number of states. Then,

$$p(X|M) = \sum_i^S \alpha_t(i)\beta_t(i) = \sum_i^S \alpha_T(i) \tag{5}$$

When the form of the state probability distributions $p(x|q)$ is parametric, i.e. Gaussian (or Laplacian), its parameters can be computed using forward-backward variables.

The Viterbi algorithm[12] is also sometimes used for training. In that case, the parameters are updated so as to increase the probability of the most probable path and $p(X|M)$ is no longer maximized. An explicit formulation of the Viterbi criterion is obtained by replacing all summations by a "max" operator. Thus,

$$p^{Vit}(X|M) = \max_{q_1^T \in Q} p(X, q_1^T|M) \tag{6}$$

[†] We use P to represent probability and p to represent probability density

[††] If we use maximum likelihood criterion, then estimation of the acoustic model reduces to estimating $p(X|W)$ as $p(X)$ is assumed equal across the models.

and in correspondence to the forward variable $\alpha_t(j)$ new variable $\delta_t(i)$ is defined as:

$$\begin{aligned}\delta_t(j) &= \max_{q_1^{t-1}} p(x_1^t, q_j(t)|M) \qquad (7)\\ &= \left[\max_i \delta_{t-1}(i)a_{ij}\right] p(x_t|q_j(t), M)\end{aligned}$$

and then,

$$p^{Vit}(X|M) = \max_i \delta_T(i) \qquad (8)$$

The path associated with $p^{Vit}(X|M)$, i.e. the optimal state sequence, can be recovered by backtracking. Each training vector x_t is then uniquely associated (aligned) with only one state. In this case, state probability distribution parameters can be estimated directly from the data aligned to this state by Maximum Likelihood criterion (or Expectation-Maximization algorithm for mixture of distributions).

The Viterbi algorithm is the main tool for the recognition task. It is much faster than the Baum-Welch algorithm as it is a simplified version of the latter. The Viterbi algorithm essentially traces the minimum cost (or maximum probability) path through a time-state lattice subject to the constraints imposed by the acoustic and language models.

3. Speech recognition with Bayesian Networks

3.1 Definition

A Bayesian network represents a joint probability distribution of a set of random variables $Z_1, \ldots, Z_n$ and is expressed by a directed acyclic graph (DAG) where each node corresponds to unique variable and arcs between the nodes correspond to conditional dependencies between variables. Depending on the type of variables (discrete or continuous) conditional probability distributions can be represented by tables or mixture of Gaussians. The immediate predecessors of a variable Z_i are called its *parents* and referred to as $Pa(Z_i)$. The joint probability distribution is factored as:

$$P(Z_1, \ldots, Z_n) = \prod_{i=1}^{n} P(Z_i|Pa(Z_i)) \qquad (9)$$

Temporal processes (as speech) are modeled with a variant referred to as Dynamic Bayesian Network (DBN). In DBN, a set of variables is associated with each frame, the graph structure is repeated for all frames and the conditional probabilities associated with analogous variables in different time frames are tied [13].

A representation of the standard HMM as a DBN is shown in Fig. 2 where Q_t is the state variable and Y_t is the continuous observation variable at time $t = 1, 2, 3, 4, \ldots$. Arcs between state instances represent HMM transition probabilities and arcs between state and observation instances represent HMM state conditional distributions.

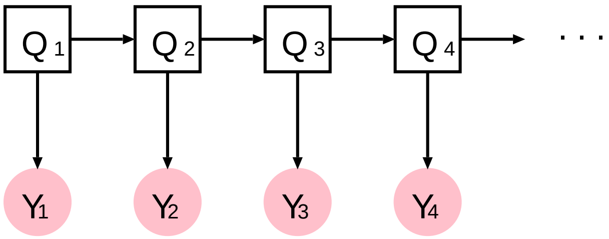


Fig. 2 Representation of HMM as DBN

3.2 Bayesian Network learning and inference

As for the HMM case, we are interested in BN parameter learning as well as in estimating the probability of observation data. For both tasks, there exist a number of algorithms. The most simple algorithms apply to tree-structured Bayesian networks with only discrete variables [14], [15].

The observed variables values, also called evidence, are partitioned into three sets - e_i^+, e_i^- and e_i^0, for each variable Z_i. The values for the variables which occur in sub-trees below particular Z_i form the e_i^- set. Those which are above Z_i form e_i^+ set and e_i^0 is the observed value for Z_i. If Z_i is hidden variable, then $e_i^0 = \emptyset$. Thus, union of the evidence sets e includes all observations. Joint probability of observations and Z_i can be factorized as:

$$\begin{aligned}&P(e, Z_i = j) =\\ &\quad P(e_i^+, e_i^-, e_i^0, Z_i = j) = \qquad (10)\\ &\quad P(e_i^-, e_i^0|Z_i = j, e_i^+)P(e_i^+, Z_i = j) = \qquad (11)\\ &\quad P(e_i^-, e_i^0|Z_i = j)P(e_i^+, Z_i = j) \qquad (12)\end{aligned}$$

Eq.(12) follows from Eq.(11) because of BN property (9). The two factors of Eq.(12) are known as $\lambda_i(j)$ and $\pi_i(j)$ and are key quantities of this procedure:

$$\begin{aligned}\lambda_i(j) &= P(e_i^-, e_i^0|Z_i = j) \qquad (13)\\ \pi_i(j) &= P(e_i^+, Z_i = j) \qquad (14)\end{aligned}$$

They are analogous to the $\beta_t(i)$ and $\alpha_t(j)$ of the Baum-Welch algorithm for the HMM. Also, the evidence sets e_i^+ and e_i^- correspond to x_1^{t-1} and x_{t+1}^T and e_t^0 corresponds to x_t. Computation of λ and π parameters is done in two passes as well: bottom-up pass for λ's and top-down pass for π's which correspond to the backward β-recursion and forward α-recursion in HMM. It follows from the above definitions that for every variable Z_i,

$$P(e) = \sum_j P(e, Z_i = j) = \sum_j \lambda_i(j)\pi_i(j) \qquad (15)$$

$$P(Z_i = j|e) = \frac{P(e, Z_i = j)}{P(e)} = \frac{\lambda_i(j)\pi_i(j)}{\sum_j \lambda_i(j)\pi_i(j)} \qquad (16)$$

The probability $P(e)$[†] corresponds to $p(X|M)$ from Eq.(5) and is the quantity we are interested in during recognition. On the other hand, $P(Z_i = j|e)$ gives the probability of any variable given the observation data, which is especially useful if Z_i is hidden. The inference procedures for Bayesian networks are essentially identical to those for HMMs when the underlying graph is a chain. However, when the graph is a real tree, the two become different.

So far, the above procedure has been defined for tree-structured BN. In order to apply it for dynamic BN as of Fig.2, network structure is transformed into a tree using "junction tree" (or JLO) algorithm [16], [17]. As with HMMs, the cost of exact inference for DBNs, i.e. calculation of Eqs.(15) and (16), scales with the square of the number of hidden states and exponentially in the number of state variables. Therefore, more complex networks may become infeasible for exact inference. In such cases, approximate inference algorithms can be applied and they include *variational* algorithms [18] or Monte Carlo sampling methods [19].

For BN parameter estimation, there exist number of methods. For the simplest case, when all variables are observable, maximum likelihood (ML) estimates can be computed in closed form. In partially observed case, i.e. when some variables are hidden, EM algorithm can be applied [20]. In this case, sufficient statistics for the E-step are calculated from the marginal posterior probabilities using Eq.(16). It is apparent that BN algorithms are analogous to HMM algorithms and this relation has been explored in depth [21]. Learning the structure (topology) of BN is a problem much more complicated and is less well developed [22].

4. The hybrid HMM/BN model

4.1 HMM/BN model structure

We will introduce our hybrid HMM/BN model in several steps. First, let's consider the DBN from Fig.2 and imaginary break arcs between state nodes. Then we get multiple, independent BN as shown in Fig. 3 corresponding to each time t. Next, if we let the time transitions (broken arcs) be governed by conventional HMM, and assign those BNs to appropriate HMM states we can drop the time index and since all BNs have the same structure we can represent them as single BN shown in Fig. 4 where the variable Q takes values of state indexes (q_{ij}) of all HMMs in the acoustic model and the state probability distributions $P(Y|Q = q_{ij})$ are represented by the arc.

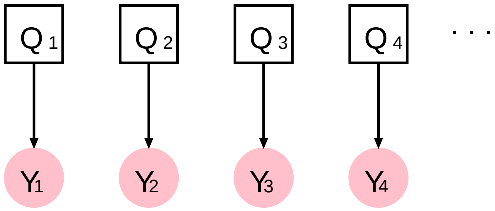


Fig. 3 Multiple BN for each time t

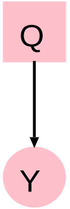


Fig. 4 State BN

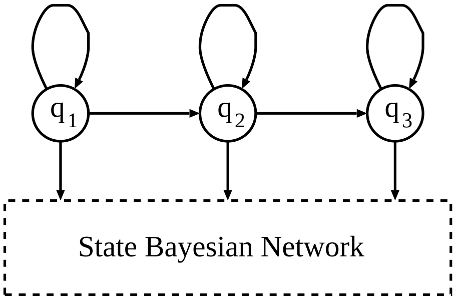


Fig. 5 HMM/BN model structure.

In other words, we modified the conventional HMM to have a BN as state distribution model instead of mixture of Gaussians. This we call the *hybrid HMM/BN* model. Combining HMM and BN in this manner makes the HMM/BN model hierarchical, where BN is at the bottom level and HMM is at the top level as shown in Fig. 5.

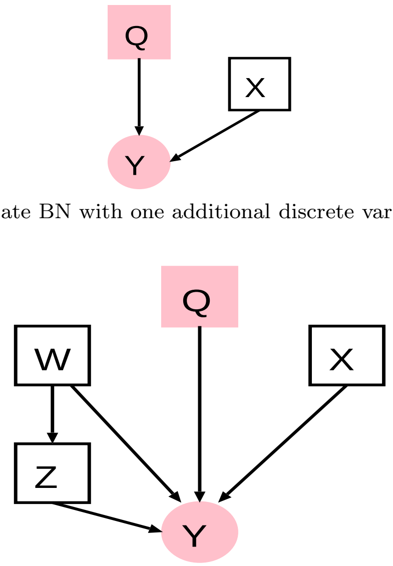
Note that, the state variable Q (Fig. 4) has become observable for the BN, but at the upper HMM level it is still hidden.

The state BN, can easily be extended to include other random variables representing additional knowledge. The graphical structure of the extended BN can be imposed according to our knowledge of the relationship between variables, rather than be learned from data, which is not a trivial task. Some possible structures of extended sate BN are shown in Fig. 6. For example, the variable X in this figure can represent the environment noise type and the other W and Z variables can represent speaker id and his/her native language.

4.2 Comparison with the hybrid HMM/NN model

The hybrid HMM/BN model is analogous to the hybrid HMM/NN model [3], [23]. In both cases, HMM is used to model temporal speech characteristics. The

[†] To be more precise, we should refer to $P(e)$ as $P(e|M)$ where M denotes BN parameter set, but we have dropped this dependency for the sake of notational simplicity.



a) State BN with one additional discrete variable.

b) State BN with more complex structure.

Fig. 6 Possible state BN structures.

difference is in the state probability distribution modeling. In the HMM/NN model, it is modeled by Neural network, while in our case it is modeled by Bayesian network. The advantage of using BN is that it offers much greater flexibility in modeling probabilistic dependencies and combines both continuous and discrete random variables (features) in simple and consistent way. The hybrid HMM/NN model had gained popularity in speech recognition community because of the strong classification abilities of the NN which is naturally trained in discriminative fashion and is better than Maximum Likelihood (ML) trained Gaussian classifiers. However, it has been found difficult to build context dependent (triphone) model based on the HMM/NN, so it has been used mainly in such systems where whole word or monophone models are good enough. In contrast, the HMM/BN model can be used instead of HMM in every system since it behaves just like HMM and the state BN can also be shared across the models in the same way as in the tied state HMM systems.

4.3 Training and recognition with HMM/BN model

For HMM/BN model training, the same approach as for HMM/NN training [24] can be adopted. It is based on the Viterbi training algorithm introduced in Sec. 2.2. First, we choose topologies of HMMs and state Bayesian network. Then models are initialized and Viterbi alignment is performed using bootstrap recognizer. This gives a time-aligned state segmentation. The state segmentation is used to produce training data for the state Bayesian network which is then trained as described in Sec. 3.2. In this Viterbi training scheme, the temporal and static parts of the training are separated. The process may be iterated, alternating between BN training and re-estimating the transition probabilities, which is an embedded training process.

When doing recognition with this HMM/BN model, as in the case of conventional HMM, the usual Viterbi decoding algorithm is used. Here, we need to calculate the $P(y|Q)$ for each state $Q = q_{ij}$ where i is the HMM index and j is the state index of the i^{th} HMM. We can infer this value from the BN probability model using standard inference algorithms (like Eq.(15)).

For simple BN, as that of Fig. 6.a, even "brute force" inference method is applicable. The joint probability model for this BN can be expressed by chain rule as follows:

$$P(Y, X, Q) = P(Y|X, Q) * P(X|Q) * P(Q) \qquad (17)$$

and since X and Q are independent variables (there are no arcs linking them), above equation can be rewritten as:

$$P(Y, X, Q) = P(Y|X, Q) * P(X) * P(Q) \qquad (18)$$

Then, probability of interest $P(Y|Q)$ is calculated by marginalization over X:

$$\begin{aligned} P(Y|Q) &= \frac{P(Y,Q)}{P(Q)} = \frac{\sum_x P(Y, X=x, Q)}{P(Q)} \\ &= \frac{\sum_x P(Y|X=x, Q) * P(X=x) * P(Q)}{P(Q)} \\ &= \sum_x P(Y|X=x, Q) * P(X=x) \qquad (19) \end{aligned}$$

In many practical cases, we can assume that $P(X)$ is the same for all $X = x$ and then Eq.(19) reduces to:

$$P(Y|Q) = \frac{1}{N(x)} \sum_x P(Y|X=x, Q) \qquad (20)$$

where $N(x)$ is the number of values X can take.

5. HMM/BN model in noisy speech recognition system

When speech is contaminated by noise, speech feature vectors change their distributions and this change depends on the noise type as well as on the SNR value. Therefore, we can express these dependencies with a state BN of the type shown in Fig. 7.

Here, N and S are hidden discrete variables representing noise type and SNR value. In this case, the state likelihood can be expressed analytically in the same way as we derived Eq.(19). In most cases, prior probabilities $P(N)$ and $P(S)$ can reasonably be assumed equal for each type of noise and each SNR value and then:

$$P(Y|Q) = \frac{1}{N(n,s)} \sum_{n,s} P(Y|N=n, S=s, Q) \qquad (21)$$

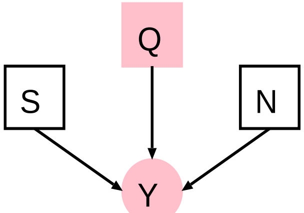


Fig. 7 State BN with noise and SNR variables

Above equation can be interpreted as an average of likelihoods from environment (noise and SNR pair) dependent models which itself may not be new acoustic modeling approach. However, we have to point out that in this paper, Eq. (21) follows naturally from the state BN structure of Fig. 7 and thus has can be explained easily. Obviously, different BN topologies will result in different state likelihood formulae (provided there is an analytical solution to the BN inference problem) which can be difficult to motivate otherwise.

HMM/BN based word models as well as sub-word models are made in the same way as in the conventional HMM case. Decoding also can be performed as in standard HMM based systems without any changes in the decoder.

6. Evaluation on AURORA2 task

In these experiments, we followed closely the evaluation scenario suggested by the official AURORA2 task [25]. The source speech for AURORA2 task is the TIdigits, consisting of connected digits spoken by American English talkers. A selection of 8 different real-world noises is added over a range of signal to noise ratios. Training set consists of clean and noisy (multi-condition) data from 4 different noises (train, babble, car, exhibition hall) and 4 SNR values (20dB, 15dB, 10dB and 5dB). As test data we used Test set A and Test set B. Test set A contains the same noises as multi-condition training data, while Test set B contains 4 different noises. In both test sets, in addition to the SNR values of the training data noises are added at 0dB and -5dB as well. There are 8440 training utterances and about 1000 utterances per each test condition.

Of primary interest for us was to compare the HMM/BN system with multi-condition trained HMM system. Each word in the baseline HMM system is modeled by 16 state HMM with 3 mixtures per state. Only the silence model uses 3 states with 6 mixtures each. Speech data are processed in standard manner with 10ms frame rate and 25ms frame length. 12 MFCC coefficients, power and their first and second deltas are used as 39 dimensional feature vector. When training the HMM/BN state conditional distributions, we divided the training data by noise type and by SNR value and used HTK to train parameters (3 mixtures per state) for each condition separately. All other system parameters as feature vectors, word model state number and experimental conditions are kept the same. The main functional difference between the two systems is that HMM/BN system explores the hidden dependencies of speech features and noise.

Recognition results for test set A (same noise types as in training data) and test set B (different noises) are summarized in Table 1. As can be seen, the HMM/BN system performance is much higher for the closed noise condition test (A set) approaching the state-of-the-art results for this task obtained by much more complex systems. Especially big difference in performance is observed for the low SNR conditions and, in average, the relative improvement over the baseline HMM system is 36.4%. As for the B set condition, there is degradation of the performance. This can be explained by the fact that no knowledge of dependencies for the new noises is available to the HMM/BN system in addition to the mismatch in the speech spectrum feature distributions. On the other hand, in the multi-condition HMM system, state Gaussian mixtures clearly do not model very well the complex distribution from multiple noise and SNR conditions. However, this mismatch between data and model distributions has some smoothing effect which increases the model abilities to generalize over unseen data.

Table 1 HMM and HMM/BN systems performance (%)

SNR	Test set A		Test set B	
	HMM	HMM/BN	HMM	HMM/BN
Clean	98.54	98.83	98.54	98.83
20 dB	97.52	98.12	96.96	97.26
15 dB	96.94	97.65	95.38	95.05
10 dB	94.59	96.04	92.58	90.27
5 dB	87.51	91.70	83.50	78.00
0 dB	59.84	76.11	58.91	48.70
-5 dB	23.46	35.79	23.86	3.18
Average*	87.29	91.92	85.46	81.85

* Calculated over values from 20dB to 0dB.

Another difference between the baseline HMM and the hybrid HMM/BN model is that latter has 17 times (4 noise types times 4 SNR values plus clean condition) more parameters. In order to prove that the better performance of the HMM/BN model on test set A is not only due to increased number of parameters, we trained HMM model with the same number of parameters by increasing the mixture number. The overall average word accuracy rates of the three types of models is shown in Fig. 8 where the newly trained model is denoted by HMM+.

This comparison clearly shows that the hybrid HMM/BN model is still better than the HMM+ for the known environments case which is due to better modeling of the environment-observation dependency which is learned explicitly. In contrast, the conventional HMM learns it implicitly. This advantage comes, however, at the expense of lesser generalization ability.

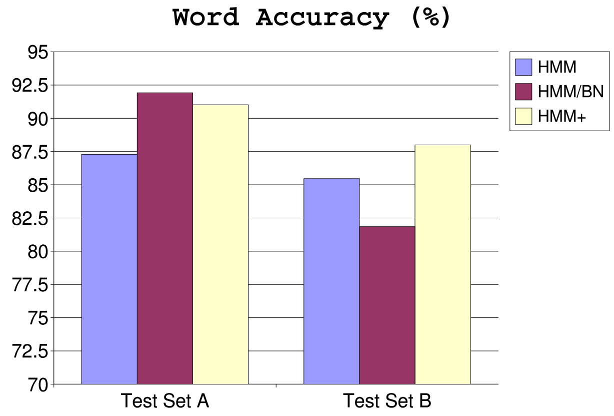


Fig. 8 Comparison between baseline HMM, HMM/BN and HMM+ with the same number of parameters as the hybrid model.

7. Discussion

Obviously, the proposed hybrid HMM/BN model is applicable not only in noisy speech recognition systems, but in many other cases, where performance can benefit from additional observable or hidden features. The number of possible state BN topologies is enormous which results in very flexible modeling and better match between models and data. System modeling capabilities increase because of combining together features from different spaces and exploring dependencies between them.

Simple state BN structures, as the one we used in our experiments, allow for analytical solution to the likelihood calculation problem and sometimes may lead to known acoustic modeling approaches which are often heuristically motivated. Conventional HMM is also a special case of HMM/BN model with state BN topology as of Fig. 4.

Especially interesting is the possibility to infer the probabilities of the hidden variables of the state BN. This way, HMM/BN system can be used for recognition of those additional parameters. For example, if an additional hidden variable X represents language in a multi-lingual system, we can calculate $P(X|Q)$ for each frame and accumulate these probabilities over the input utterance. Then, $x = \arg\max_x P(x|Q_S)$, where Q_S is the best hypothesis state sequence, shows the most probable language the utterance has been spoken in. Thus, in addition to recognizing multi-lingual speech, such system can perform language recognition as well.

8. Conclusion

We have proposed a method for combining HMM and BN in a single model which benefits from strengths of both HMM and BN. The hybrid HMM/BN model allows for easy addition of other information in the speech recognition systems increasing their performance at minimal cost. Furthermore, HMM/BN model can represent sub-word phonetic units like the conventional HMM. This way, it becomes possible to use the BN framework in large vocabulary continuous speech recognition. Experimental evaluation of the method in noisy speech recognition task showed that adding noise type and SNR values as additional parameters and exploring dependency between them and the spectrum feature parameters resulted in 36.4% less errors in the AURORA2 task.

9. Acknowledgment

The research reported here was supported in part by a contract with the Telecommunications Advancement Organization of Japan entitled, "A study of speech dialogue translation technology based on a large corpus".

References

[1] S. E. Lavinson, L. R. Rabiner, and M. M. Sondhi, "An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition," *The Bell System Technical Journal*, vol. 62, pp. 1035–1074, Apr. 1983.

[2] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, pp. 257–285, Feb. 1989.

[3] Herve Bourlard and Nelson Morgan, "A continuous speech recognition system embedding MLP into HMM," in *Advances in Neural Information Processing 2* (D. Touretzky, ed.), pp. 186–193, Morgan Kaufmann, 1990.

[4] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Hidden Markov Models based on multi-space probability distribution for pitch pattern modeling," in *Proc. ICASSP*, pp. 229–232, 1999.

[5] T. Dean and K. Kanazawa, "Probabilstic temporal reasoning," in *AAAI*, pp. 524–528, 1988.

[6] G. Zweig and S. Russell, "Probabilistic modeling with Bayesian Networks for automatic speech recognition," in *Proc. ICSLP*, pp. 3010–3013, 1998.

[7] K. Daoudi, D. Fohr, and C. Antoine, "A new approach for multi-band speech recognition based on probabilistic graphical models," in *Proc. ICSLP*, vol. I, pp. 329–332, 2000.

[8] T. Stephenson, M. Mathew, and H. Bourlard, "Modeling auxiliary information in Bayesian Network based ASR," in *Proc. Eurospeech*, pp. 2765–2768, 2001.

[9] K. Daoudi, D. Fohr, and C. Antoine, "Continuous multiband speech recognition using Bayesian Networks," in *Proc. ASRU*, 2001.

[10] L. Bahl, F. Jelnek, and R. Mercer, "A Maximum Likelihood Approach to Continuous Speech Recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, no. 2, pp. 179–190, 1983.

[11] H. Bourlard, Y. Kamp, H. Ney, and C. Wellekens, "Speaker Dependent Connected Speech Recognition via Dynamic Programming and Statistical Methods," in *Speech and Speaker Recognition* (M. Schroeder, ed.), Karger, 1985.

[12] G. Forney, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[13] G. Zweig and S. Russell, "Probabilistic modeling with

Bayesian Networks for Automatic Speech Recognition," *Australian Journal of Inteligent Information Processing Systems*, vol. 5(4), pp. 253–260, 1999.

[14] Judea Pearl, *Probabilistic Reasoning in Intellident Systems: Networks of Plausible Inference*. San Mateo, California: Morgan Kaufmann, 1988.

[15] Mark Peot and Ross Shachter, "Fusion and Propagation with Multiple Observations," *Artificial Intelligence*, vol. 48(3), pp. 299–318, 1991.

[16] F. Jensen, S. Lauritzen, and K. Olsen, "Bayesian updating in recursive graphical models by local comutations," *Computational Statistics and Data Analysis*, vol. (4), pp. 269–282, 1990.

[17] Finn V. Jensen, *An Introduction to Bayesian Networks*. London: UCL Press, 1996.

[18] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul, "An Introduction to Variational Methods for Graphical Models," in *Learning in Graphical Models* (M. Jordan, ed.), pp. 175–204, MIT Press, 1999.

[19] D. McKay, "Introduction to Monte Carlo Methods," in *Learning in Graphical Models* (M. Jordan, ed.), pp. 175–204, MIT Press, 1999.

[20] Steffen L. Lauritzen, "The EM algorithm for graphical association models with missing data," *Computational Statistics & Data Analysis*, vol. 19, pp. 191–201, Feb. 1995.

[21] P. Smyth, D. Heckerman, and M. Jordan, "Probabilistic independence networks for hidden Markov probability models," *Neural Computation*, vol. 9(2), pp. 227–269, 1997.

[22] N. Friedman, K. Murphy, and S. Russel, "Learning the structure of dynamic probabilistic networks," in *Unsertainty in Artificial Intelligence: Proceedings of the 14th Conference*, Madisson, Wisconsin: Morgan Kaufmann, July 1998.

[23] H. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*. Boston: Kluwer Academic Publishers, 1994.

[24] S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist Probability Estimators in HMM Speech Recognition," *IEEE Trans. SAP*, vol. 2, pp. 161–173, Jan. 1994.

[25] H. Hirsch and D. Pearce, "The AURORA Experimental Framework for the Performance evaluations of Speech Recognition Systems under Noisy Conditions," in *ISCA ITRW ASR2000 "Automatic Speech Recognition: Challenges for the Next Millennium"*, Sept. 2000.

Konstantin Markov was born in Bulgaria. He received his B.E. degree in Electrical Engineering from Department of Cybernetics, St. Petersburg Technical University, Russia in 1984. In 1996 and 1999 he received his M.E. and D.E. in Computer Science from Toyohashi University of Technology, Japan. From 1999 to 2000, he was a research engineer in ATR Research and development center and from 2000 he joined ATR Spoken Language Translation Research Laboratories as invited researcher. Dr. Markov is member of Acoustic Society of Japan. His main research interests include automatic speech recognition noise robustness, speaker identification and statistical pattern recognition in general.

Dr.Satoshi Nakamura was born in in Japan on August 4, 1958. He received the B.S.degree in electronics engineering from Kyoto Institute of Technology in 1981 and the Ph.D. degree in information science from Kyoto University in 1992. Between 1981-1986 and 1990-1993, he worked with the Central Research Laboratory, Sharp Corporation, Nara, Japan, where he was engaged in speech recognition research. During 1986-1989, he was a researcher of the speech processing department at ATR Interpreting Telephony Research Laboratories. From 1994-2000, he was an associate professor of the graduate school of information science, Nara Institute of Science and Technology, Japan. In 1996, he was a visiting research professor of the CAIP center of Rutgers, the state university of New Jersey USA. He is currently the head of Department 1 in ATR Spoken Language Translation Laboratories, Japan. He also serves as a guest professor for Toyohashi University of Technology and Ritsumeikan University from April, 2002. His current research interests include speech recognition, speech translation, spoken dialogue systems, stochastic modeling of speech, and microphone arrays. He received the Awaya Award from the Acoustical Society of Japan in 1992, and the Interaction2001 best paper award from the Information Processing Society of Japan in 2001. He is a member of the Acoustical Society of Japan, Institute of Electrical and Electronics Engineers (IEICE), Information Processing Society of Japan, and IEEE. He is currently a member of the Speech Technical Committee of the IEEE Signal Processing Society and an editor for the Journal of the IEICE Information and System Society.