DBN BASED MULTI-STREAM MODELS FOR SPEECH

Yimin Zhang, Qian Diao, Shan Huang, Wei Hu *Chris Bartels, Jeff Bilmes*
Intel China Research Center, Beijing Univ. of Washington, Seattle, WA98195
{yimin.zhang, qian.diao, shan.huang, wei.hu}@intel.com {bartels, bilmes}@crow.ee.washington.edu

ABSTRACT

We propose dynamic Bayesian network (DBN) based synchronous and asynchronous multi-stream models for noise-robust automatic speech recognition. In these models, multiple noise-robust features are combined into a single DBN to obtain better performance than any single feature system alone. Results on the Aurora 2.0 noisy speech task show significant improvements of our synchronous model over both single stream models and over a ROVER based fusion method.

1. INTRODUCTION

The task of noise-robust automatic speech recognition (ASR) has become an active research topic in recent years. In this endeavor, various kinds of noise robust feature-extraction methodologies have been developed in an attempt to produce much better performance than standard mel-frequency cepstral coefficients (MFCCs). In general, methodologies that employ only a single feature stream have not performed nearly as well as those which in some way combine multiple systems together. Indeed, much research on such multi-stream models that aims to take advantage of the complementary information in multiple information streams [1,2] has occurred. These streams might be multi-modal (audio and visual information) [3], or simply different sets of features extracted from the same speech data.

How best to combine multiple features is the one of the key problems in multi-stream modeling. Previous work on combining multiple features, be it audio-visual speech recognition (AVSR), multi-band, or multi-stream, can be divided into three categories: feature fusion (or early integration), decision fusion (or late integration), and model fusion. In the *feature fusion* method [3], multiple features are concatenated into a large feature vector that is subjected to dimensionality reduction, and the resulting features are modeled by a conventional HMM. This method, however, cannot easily represent any loose asynchrony between different features. In the *decision fusion* method, independent HMMs are trained using different features, and decoding is also done independently on each HMM. The final results are obtained by combining the results (via likelihood scores and an n-best list or lattice) from each HMM using either ROVER [4], word graph based hypothesis combination [5], or posterior combination [6]. This method might not capture any direct correlation between feature vector elements. The most common *model fusion* methods are product HMMs or multi-stream HMMs [7]. These methods typically require various heuristic-based combination strategies to form a unified HMM model from the original separately trained HMMs. Often, multi-stream HMMs impose some form of state synchronicity constraint, while product HMMs allow only for limited synchrony. In some cases, the number of states in the unified HMM model will be the Cartesian product of the states in the component HMMs. With only a modest number of streams, the resulting unified HMM can become intractable because of such a large state space. Also, if any *stream exponents* are used (i.e., exponential weights on the stream probabilities), it might cause the emission probability densities to be improper (e.g., not summing to unity), and standard EM training algorithms cannot be directly used to estimate the stream exponents.

In this work, we propose the use of general dynamic Bayesian network (DBN) models to naturally combine multiple features. We call this a *multi-stream DBN model*. Our model can represent both the various relationships between different feature streams and also any asynchrony that might exist between these streams. Using a DBN greatly simplifies statistical modeling issues, and the method can immediately be varied to use any number of feature streams. In section 2, we describe the multi-stream DBN models that we developed. In section 3, experimentation results and a comparison with a standard decision combination method (ROVER) is given. Finally in section 4 we provide further discussion and then conclude.

2. MULTI-STREAM DBN MODELS FOR SPEECH

A dynamic Bayesian network (DBN), a generalization of HMMs, is a statistical model that can represent multiple collections of random variables as they evolve over time. Coupled HMMs and factorial HMMs are just special cases of the much more general DBN. Indeed, DBN models have been proposed in recent years for speech recognition [8, 9]. In this work, we use the graphical

model structure for continuous speech given in [9] as our baseline model. In some sense, this particular DBN is equivalent to a traditional HMM because it emulates what an HMM does. One major difference is that it explicitly represents the hierarchical structure consisting of sentence, word, phone and sub-phone states. Our goal in this work is to extend this baseline model by introducing structure so as to better combine multiple feature streams.

2.1. Synchronous multi-stream model

Our new model is given in Fig. 1. The graph shows a whole word model, i.e. a word is composed of fixed number of states, and no intermediate phoneme level. The upper part of the model is similar to [9]. To the lower part of this model, our model adds multiple observation variables, each corresponding to a different acoustic feature, e.g. MFCC, PLP, etc. All the observation variables in this case share one state variable, so in this model all streams are synchronized at the state level.

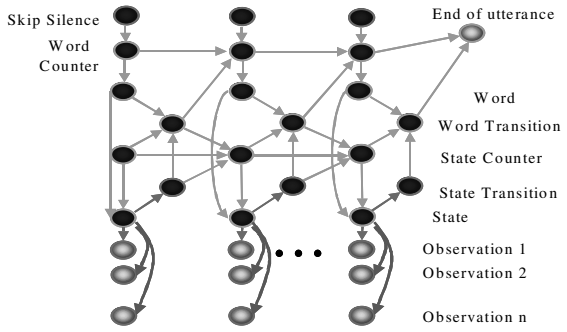

Fig.1 Synchronous multi-stream model

The meaning of each variable is as follows:
Skip Silence, a binary random variable, denoted as SS_t, where t means the t-th time slice.
End of utterance, assigned to 1 to mean end of an utterance, denoted as E.
Word counter, the position in current sentence, denoted as WC_t.
Word, the current word, determined by word counter and the current sentence (represented implicitly in the model during training and decoding), denoted as W_t.
Word transition, indicating when the current word ends and a transition to next word occurs, denoted as WT_t.
State counter, the index of the current state in the whole word model, denoted as SC_t.
State, the current state, it is determined by a word and the state counter, denoted as S_t.
Observation 1-n, each of the n streams of observations, the m-th observation node denoted as O_t^m.
For a better understanding of this model, we precisely describe each node's CPD (conditional probability distribution).

$$P(WC_t = j \mid WC_{t-1} = i, SS_t = b, WT_{t-1} = f) = \begin{cases} 1 & if \quad j = i+1 \quad and \quad b = 0 \quad and \quad f = 1 \\ 1 & if \quad j = i \quad and \quad b = 0 \quad and \quad f = 0 \\ 1 & if \quad j = SIL \quad and \quad b = 1 \\ 0 & otherwise \end{cases}$$

$$P(W_t = \omega \mid WC_t = i) = \begin{cases} 1 & if \quad \omega = words[i] \\ 0 & otherwise \end{cases}$$

$$P(SC_t = j \mid WT_{t-1} = b, ST_{t-1} = f, SC_{t-1} = i) = \begin{cases} 1 & if \quad j = i+1 \quad and \quad b = 0 \quad and \quad f = 1 \\ 1 & if \quad j = i \quad and \quad b = 0 \quad and \quad f = 0 \\ 1 & if \quad i = 1 \quad and \quad b = 1 \quad and \quad f = 1 \\ 0 & otherwise \end{cases}$$

$$P(WT_t = f \mid W_t = \omega, SC_t = q, ST_t = b) = \begin{cases} 1 & if \quad b = 0 \quad and \quad f = 0 \\ 1 & if \quad b = 1 \quad and \quad f = 1 \quad and \quad laststate(q, \omega) \\ 1 & if \quad b = 1 \quad and \quad f = 0 \quad and \quad \sim laststate(q, \omega) \\ 0 & otherwise \end{cases}$$

$$P(ST_t = b \mid S_t = i) = \begin{cases} A_{ii} & if \quad b = 0 \\ 1 - A_{ii} & if \quad b = 1 \end{cases}$$

$$P(S_t = j \mid W_t = \omega, SC_t = i) = \begin{cases} 1 & if \quad j = state(\omega, i) \\ 0 & otherwise \end{cases}$$

$$P(O_t^m \mid S_t = j) = b_j^m$$

In the above, *SIL* denotes the pause between words, *words[i]* denotes the *i*-th word in current utterance, *laststate(q,ω)* is true iff *q* is that last state of word *ω*, A_{ii} denotes the probability of staying at state *i*, *state(ω,i)* gives the *i*-th state in word *ω*, b_j^m is the emission probability of the m-th stream. Here we use strictly left-to-right HMMs, so $(1-A_{ii})$ is the transition probability from state *i* to state *i+1*. This multi-stream model is similar to HTK synchronous multi-stream models [10], but in our case we are using a unified DBN. Therefore, our model does not require special algorithmic support and can be extended easily.

2.2. Asynchronous multi-stream model

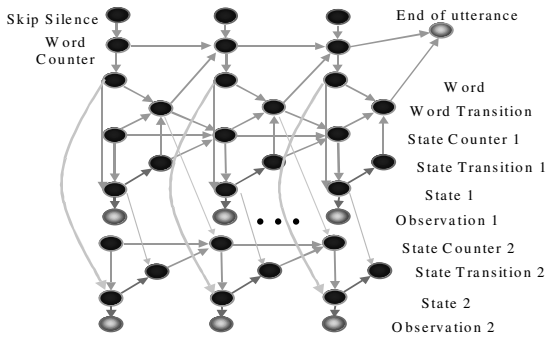

Fig.2 An asynchronous multi-stream model

The synchronous model assumes multiple streams are strictly synchronized at the lowest state level. In this section, we investigate a model that relaxes this restriction, allowing for limited asynchrony between streams at the state level. This is done by having each stream depend only on its own hidden state variable. Specifically, each stream uses two variables, namely a state counter and state transition variable to model the relation between word and state, and the transition between states. Fig. 2 shows an example of this model. It is an asynchronous 2-stream model. It can be seen that the two streams share one word variable, thereby requiring that the two streams be synchronized at the word level. When a word transition occurs, it will reset the state variables of both streams to their initial value, and this is realized through the edges from word transition to state counter. Each stream has different state variable, so there is some asynchrony between the two streams in a word. The edges from node "state-1" to "state-transition-2" denote the correlation

System	Test	-5dB	0dB	5dB	10dB	15dB	20dB	Clean	0-20dB
HTK Single-stream PLP	TSA	11.46	20.72	42.89	76.39	92.09	96.94	99.17	65.81
	TSB	13.55	26.12	53.66	83.15	94.36	97.69	99.17	75.69
	TSC	12.38	21.46	45.33	77.18	92.12	96.92	99.24	66.6
HTK ROVER All	TSA	10.32	19.93	44.58	74.07	90.57	96.34	99.24	65.1
	TSB	11.36	24.16	53.11	79.29	92.27	96.76	99.24	74.14
	TSC	10.89	21.35	47.01	76.03	90.75	96.17	99.26	66.26
HTK ROVER Best2+worst1	TSA	10.49	19.11	41.9	71.13	89.36	95.93	99.4	63.49
	TSB	11.09	22.62	50.13	76.71	90.9	96.37	99.29	72.67
	TSC	10.48	20.24	44.18	73.41	89.68	95.71	99.48	64.64
GMTK Single-stream PLP	TSA	2.88	18.4	52.42	79.75	92.89	97.31	99.39	68.15
	TSB	6.45	27.02	61.49	85.44	95.17	97.98	99.39	73.42
	TSC	3.9	21.73	54.67	80.22	92.82	97.19	99.31	69.33
GMTK Multi-stream All	TSA	9.16	25.18	61.69	84.98	94.01	97.26	99.38	72.62
	TSB	10.58	29.09	64.45	86.33	94.48	97.29	99.38	74.33
	TSC	9.94	23.62	56.43	81.45	93.33	97.04	99.43	70.37
GMTK Multi-stream Best2+worst1	TSA	11.96	34.39	66.83	86.17	94.24	97.21	99.39	75.77
	TSB	13.58	36	67.89	87.25	94.28	97.19	99.39	76.52
	TSC	11.88	27.68	59.54	82.45	93.1	96.9	99.46	71.93

Table 1 Word recognition accuracies for the HTK and GMTK systems averaged across noise types

between two streams. The definition of the CPD for state-transition-2 now becomes

$$P(ST_t^m = b \mid S_t^m = i, S_t^{m-1} = j) = \begin{cases} A_{ii}^{m,j} & if \quad b = 0 \\ 1 - A_{ii}^{m,j} & if \quad b = 1 \end{cases}$$

Where $A_{ii}^{m,j}$ denotes probability of staying at state i for stream m in condition that stream $m-1$ is in state j.

We can see here that for a stream m, whether the state stays the same or moves to the next state is determined by the current states of both the current stream and also the other streams.

This model has many more variables than the synchronous model. As is well known, the computational complexity and memory requirements of exact DBN inference are exponential in the number of nodes in the largest clique. Therefore, to make the model more tractable, we currently make stream m correlated only with stream $m-1$. So the model we currently evaluate is something akin to a coupled HMM. For more complex models, approximate inference might be required.

3. EXPERIMENTAL RESULTS

3.1. Setup

Our evaluations are conducted on the Aurora 2.0 continuous noisy digit speech recognition task [11]. This corpus allows for both matched (multi-condition training) and mis-matched (clean-training) training/testing conditions. Although multi-condition training generally produces better performance, it is somewhat less useful for evaluating more real-world testing conditions, where the training and testing conditions are quite mis-matched. Therefore, we evaluate using only the clean training data (mis-matched conditions). The testing is done on all the Aurora 2.0 test data (sets A, B, C), including clean speech, and noisy speech at different SNRs.

The baseline systems include single-stream models for 5 widely used features: MFCC, PLP [10], RASTA, JRASTA [12], Wide-band MFCC [4], using both HTK and GMTK [13]. We also tested a ROVER-based [4] decision fusion method to combine n-best results from the individual HTK single-stream systems. Finally we built several corresponding multi-stream models using GMTK. Because there are many possible combinations of different features, we chose the following multi-stream combinations based on the constituent single-stream performances:

All: jrasta, mfcc, plp, rasta, wbmfcc

Best4: mfcc, plp, rasta, wbmfcc

Best3: mfcc, plp, rasta

Best2+worst1: mfcc, plp, jrasta

We used a toolkit from ICSI/UC Berkeley to generate rasta and jrasta features [12], and made wide-band MFCC features by modifying the overlap of each triangular filter in the filterbank to be 75% rather than 50%. All features were processed in the HTK format. The detailed parameter settings for each feature extraction are: MFCC_D_Z_0, Jrasta_E_D_A, Rasta_E_D_A, PLP_D_Z_A_0, and WMFCC_D_Z_0. In this format, _E indicates having energy, _D means having delta coefficients, _A indicates having acceleration coefficients, _Z means having zero mean static coefficients, and _0 means having 0'th cepstral coefficients. In all models, each stream is modeled by 16-states per word, 4-mixture per state model.

3.2. Results

The experimental results for single-stream, ROVER combination and synchronous multi-stream models are shown in Table 1. For single stream models, we only give the results for PLP features, which got the best performance with both HTK and GMTK. For the multi-stream case, we report the results of the two best performing combinations: All, Best2+ worst1.

The results show that ROVER combination is not good in this case for combining multiple acoustic features; actually it is even worse than the HTK-based single stream system.

Note that all four of the DBN-based synchronous multi-stream models outperformed all of the single stream models both on average and on most of the individual tests. Interestingly, the Best2+worst1 GMTK case performed the best and got a 16% relative WER reduction over the GMTK single-stream model based on PLP features alone (averaged across 0-20dB of all three test sets). The other three in order of best to worst were: all, best 4, and best 3.

We also performed preliminary experiments using the asynchronous mutli-stream model. We compared two HTK-based single-stream models for two different feature streams (jrasta and wbmfcc) with a DBN-based asynchronous two-stream model (using the same two feature streams). Because of time limitations, we tested models using only of 8-states per word (rather than 16, as in the previous experiments). We present averaged results across the 5-10dB SNR cases of test set A only:

DBN/GMTK asynchronous multi-stream	32.93
HTK single-stream (jrasta)	22.26
HTK single-stream (wbmfcc)	57.06

As can be seen, the asynchronous multi-stream model does not yet produce a benefit, as in the synchronous case (and as further discussed in the next section).

4. CONCLUSIONS AND DISCUSSION

In this paper, we described the results of our initial work on multi-stream DBN models for speech. Specifically, using GMTK, we implemented and tested several multi-stream DBN models for speech that incorporated multiple acoustic features. Results show that these models can get significant WER improvement over both single stream models and ROVER combination on the Aurora 2.0 noisy digits recognition task. We believe, therefore, that the DBN approach is a simple and effective way to combine multiple noise-robust features to improve performance of speech recognition system under various types of noisy environments. Two highlights of this work are that we use general DBN models, so no special-purpose complex algorithms need be developed besides standard DBN inference. Second, our combination methods achieved good performance for noisy speech even using full-band acoustic features that are believed to be highly correlated. Of course, our asynchronous model still needs further research. One promising direction is to discover model structure automatically [9] so as to better model the degree of asynchrony between features. Also we plan to explore loosely coupled relations and approximate methods to make these complex models more tractable. Another possible reason for the asynchronous model's poor performance is that there might not be much actual asynchrony required when representing different acoustic features. We therefore plan to utilize complementary feature sets that are believed to possess more asynchrony. These might include lip-reading, multi-rate features, non-standard microphones (throat microphones), and features from EMGs and other sensors.

5. REFERENCES

[1] A. Janin, D. Ellis and N. Morgan, "Multi-stream speech recognition: Ready for prime time", *Proc. Eurospeech*, Budapest, 1999.

[2] H.J. Nock, S.J. Young, "Loosely Coupled HMMs for ASR", Proc. ICSLP, Beijing, China, 2000.

[3] S.Nakamura,et al., "Improved bimodal speech recognition using tied-mixture HMMs and 5000 word Audio-Visual Synchronous database", *Proc. Eurospeech*, 1997

[4] J. Fiscus, "A Post-Processing System To Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduc-tion (ROVER)," *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997.

[5] R. Singh, M. L. Seltzer, B. Raj, and R. M. Stern, "Speech in Noisy Environments: Robust Automatic Segmentation, Feature Extraction, and Hypothesis Combination," *Proc. IEEE Conf. on Acoustics, Speech, and Sig. Proc.,* Salt Lake City, Utah, May, 2001.

[6] S. Okawa, E. Bocchieri and A. Potamianos, "Multi-band speech recognition in noisy environments", *Proc. ICASSP*, Seattle, 2:641-644, May 1998.

[7] C. Neti, G. Potamianos, J. Luettin, I. Matthews, D. Vergyri, J. Sison, A. Mashari, and J. Zhou. "Audio visual speech recognition", *Final Report: JHU 2000 Summer Workshop*, 2000.

[8] G. Zweig, *DBN based speech recognition*, Ph.D. thesis, U.C. Berkeley, 1998.

[9] J. Bilmes, G. Zweig etc. "Discriminatively structured dynamic graphical models for speech recognition", *Final Report: JHU 2001 Summer Workshop*, 2001.

[10] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book*. Entropic Ltd., Cambridge, 1999.

[11] H. G. Hirsch and D. Pearce, "The aurora experimental framework for the performance evaluations of speech recognition systems under noisy conditions", *ICSA ITRW ASR2000*, September 2000.

[12] H., Morgan, N., Bayya, A, and Kohn, P. Hermansky, "Rasta-PLP Speech Analysis", *ICSI Technical Report TR-91-069*, Berkeley, California, 1991.

[13] J. Bilmes and G. Zweig "The Graphical Models Toolkit: An Open Source Software System for Speech and Time-Series Processing", *Proc. ICASSP*, Orlando Florida, 2002.