

Cepstrum bias adaptation for the switchboard database in unsupervised mode

Yasuhiro Minami

Summary

This paper describes cepstrum bias adaptation for the switchboard task. Adaptation was performed in unsupervised mode. To do this, an all-phoneme HMM is created from the entire database, that accepted any phoneme event. The input speech signal is matched to the all-phoneme HMM. Then the EM algorithm is performed to estimate cepstrum biases. In this paper, two biases, for speech and noise, are calculated. We performed an experiment for the switch board task using our bias estimation method. However, these results showed little improvement. Investigations the results, we found that our method showed a worse error rate for short sentences than CMN. So we combine the two methods. CMN is performed only for short sentences. Final experimental results showed the method improved 0.2 % in the error rate.

1. Introduction

Robust speech recognition for speech distorted simultaneously by line distortion and noise was studied [1]. Figure 1 shows speech corrupted by noise and channel distortion. I wanted to try to work on this study at the work shop. However, when I looked at and listened to some switchboard data, I found that noise is not a major problem in degradation of speech recognition accuracy. So I decided to work on line adaptation first. Figure 2 shows the speech signal corrupted by a microphone and telephone line. The problem is how to estimate this channel distortion. Cepstrum Mean Subtraction or CMS is a simple and powerful adaptation technique for this [2]. It is widely used in many recognition systems. Of course in the switchboard task, CMS was used. However, when we evaluated digit speech data recorded through the telephone line, we found that CMS is not very effective in recognizing telephone speech data. Similar results have been reported by AT&T[3]. What I did here is basically apply a simple Cepstrum bias estimation method using a maximum likelihood procedure [3][4]. In my experiments, bias estimation is performed on a per-utterance basis. This means that the utterances' transcription is not perfectly known. To solve this problem, some people use the recognition result as the transcription. However, as is well known, in the switchboard task, the recognition accuracy when making the transcription is not that good. Several people have reported that in speaker adaptation, the transcription accuracy influences the final

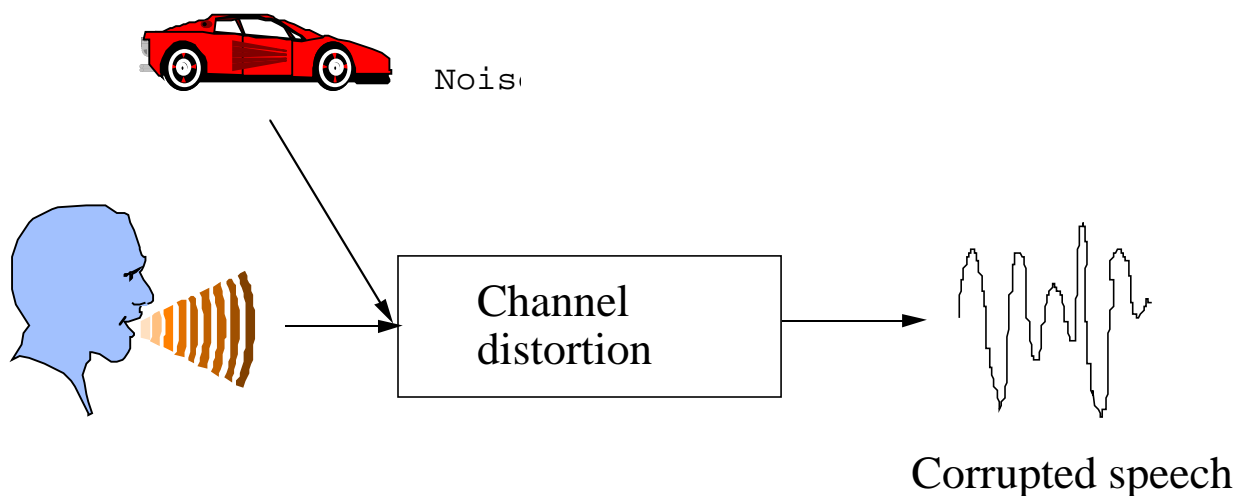


Figure 1. Speech corrupted by channel and noise.



Figure 2. Speech corrupted by channel.

recognition accuracy. Here, I proposed a simple adaptation method.

2. Formulation of cepstral bias adaptation

Considering only a linear transform, the situation in Fig. 2 is modeled by the process in Fig. 3. Y is a signal produced by speech HMMs. B is the bias. There are several methods of estimating the bias. Here, we used cepstral bias estimation by maximum likelihood. The process in Fig. 3 is written as the following random-variable equation:

$$\mathbf{X} = \mathbf{Y} + \mathbf{B} . \quad (1)$$

This can be rewritten by the time domain equation.

$$\mathbf{x}_t = \mathbf{y}_t + \mathbf{b} . \quad (2)$$

Here, as we suppose that b is not a random value, B and b are identical.

The maximum likelihood procedure can be defined as follows.

$$\mathbf{b} = \underset{b}{\operatorname{argmax}} p(\mathbf{X} | \mathbf{b}, \mathbf{M}) . \quad (3)$$

X is the sequence of the input vector, M is a set of HMMs. The maximization is attained by the EM algorithm. The following equation shows the Q function to calculate the EM procedure:

$$Q(b', b) = \sum_{t=1}^T \sum_{n=1}^N \sum_{m=1}^M \gamma_t(n, m) \cdot \left[- \sum_{i=1}^D \frac{(y_{t,i} - b'_{n,i} - \mu_{n,m,i})^2}{2\sigma_{n,m,i}^2} \right] . \quad (4)$$

The formulation of the maximization can be obtained using a similar process to the process that calculates the mean vectors of Gaussian distributions in HMMs. The iterative estimation formula for the i th component of a fixed bias b for speech is

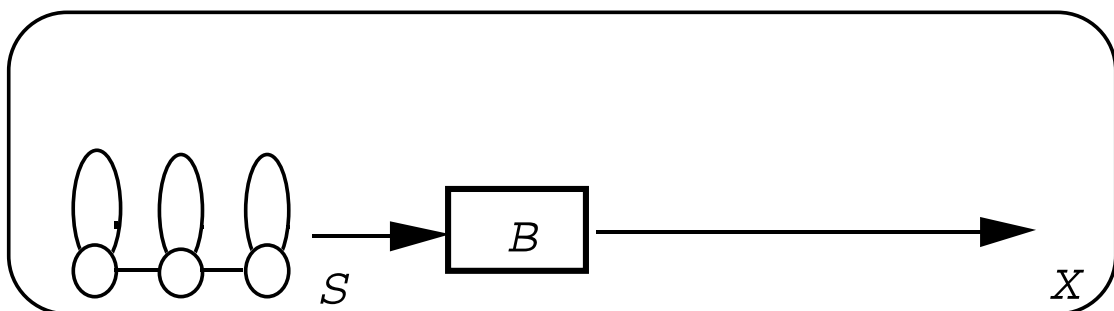


Figure 3. Modeling for cepstrum bias adaptation.

$$b'_i = \frac{\sum_{t=1}^T \sum_{n=1}^N \sum_{l=1}^L \sum_{m=1}^M \gamma_t(n,l,m) \frac{y_{t,i} - \mu_{n,l,m,i}}{\sigma_{n,l,m,i}^2}}{\sum_{t=1}^T \sum_{n=1}^N \sum_{l=1}^L \sum_{m=1}^M \frac{\gamma_t(n,l,m)}{\sigma_{n,l,m,i}^2}}. \quad (5)$$

where t is time, n and m are state numbers, l indicates the l th mixture component, and i indicates the i th element of the vector; $\gamma_t(n, m, l)$ is the joint probability of observing Y and moving from state n to state m from time t to time $t+1$ and taking the l th mixture component to produce x_t . $\gamma_t(n, m, l)$ can be calculated using the forward-backward algorithm.

3. Unsupervised training with all-phoneme model

In this paper, we want to do on-line adaptation for each sentence. Under this condition, as adaptation data is the sentence that should itself be recognized by the recognizer the sentence transcription is unknown. An unsupervised adaptation method is required. Several unsupervised methods have been proposed. However, in the switchboard task, the recognition accuracy is not very good. Several

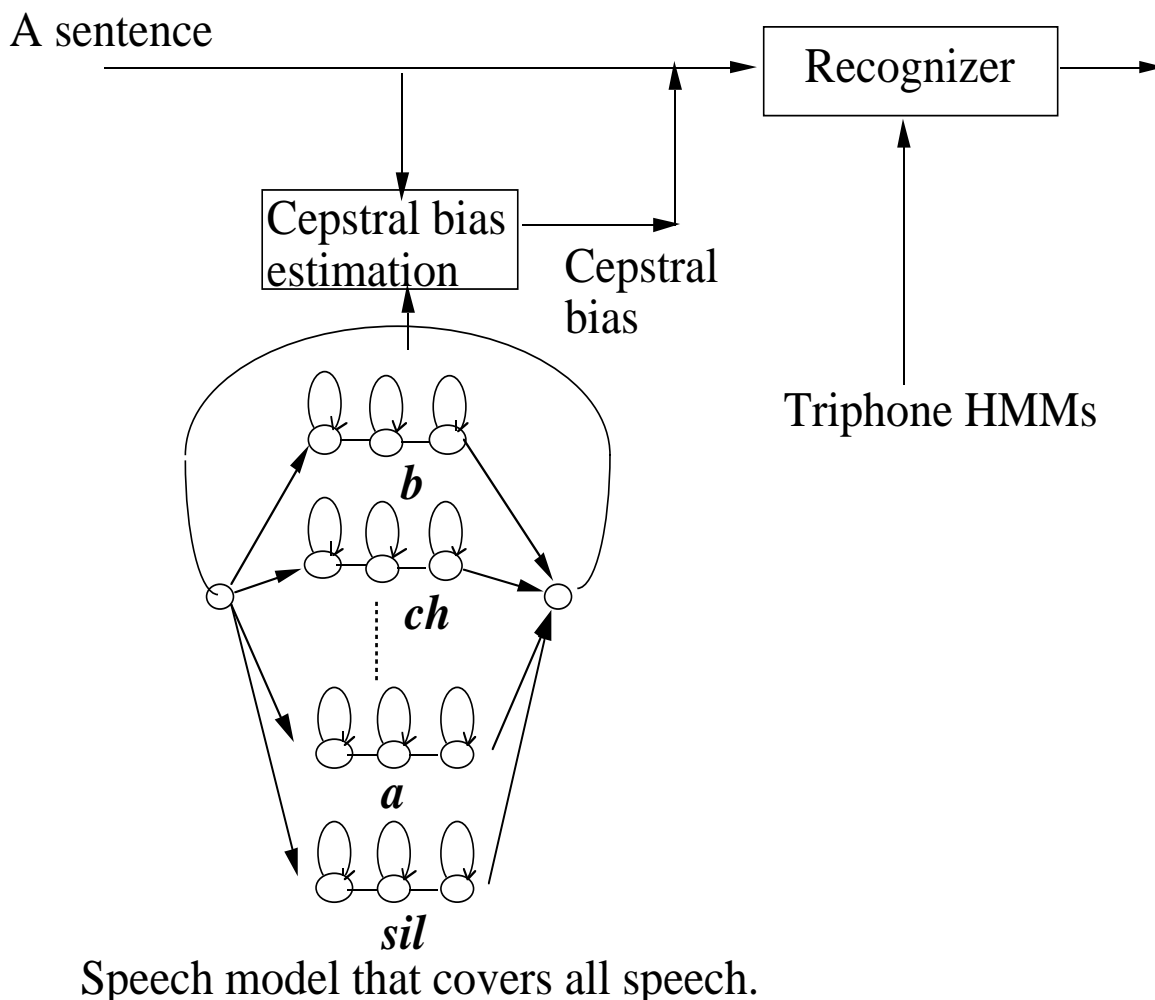


Figure 4. Block diagram of cepstral bias adaptation

people have reported that in speaker adaptation, the transcription accuracy influences the recognition accuracy.

Figure 4 shows the recognition process using the bias estimation. The cepstrum bias is estimated by the EM algorithm using a big HMM. This HMM was made from monophone models. All of the initial states of the monophones are merged. All of the final states of the monophones are merged. The final state is connected to the initial state to make a big loop. This big HMM accepts any phonemes. Using this HMM the cepstral bias is calculated. Then input speech is modified using this cepstral bias. Hvite decodes the input speech with triphone HMMs. Figure 5 shows a more precise procedure of bias estimation. To attain robust and fast estimation, we use these models as I mentioned before. In the first step, the sentence is decoded by the decoder using the HMM. The decoder outputs the transcription of the sentence and identifies all-phoneme and silence. All shows the phoneme part and sil shows the silence part. After that, the bias is calculated using this transcription and the HMM. This method takes one minute to estimate the biases per sentence. Two cepstral biases; cepstrum bias for speech and cepstral bias for silence are estimated.

4. Experimental result

A recognition experiment for the switchboard was performed using the cepstral bias estimation. The evaluation task was the WS96 development set. A baseline result was obtained with WS96 word internal triphone models and the bigram language model. These models were trained using 60 hours of training data. All monophone models that were used to make a all-phoneme model were also trained from the same database. Each state of the monophone models has only a single

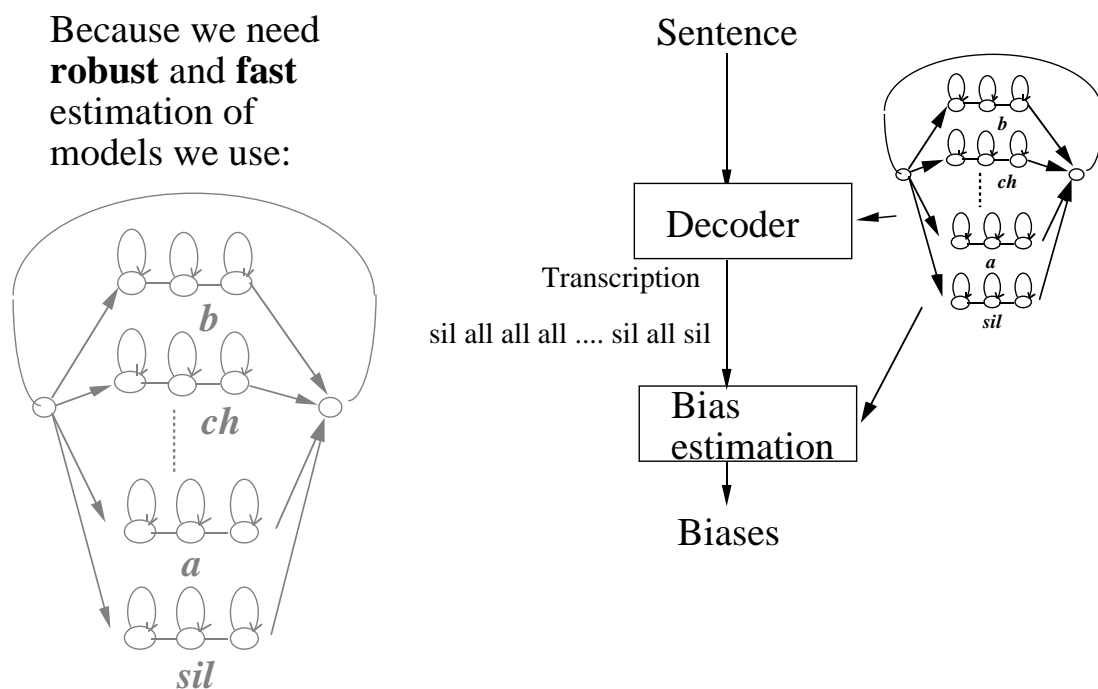


Figure 5. Adaptation process using a all phonemodell.

Gaussian distributions Sentences are decoded by the Hvite. Table 1 shows the results of our experiment. We evaluate our method with two test sets. One is a short test set, the other is a complete test set. The short test set is a subset of the complete test set. In short test set, our method produces result that was better the baseline result. However in the complete test set, there isn't that much difference in the results, so the advantage offered is small in this case. To investigate the reason for this, we checked how the recognition accuracy depends on the length of sentences in seconds. Table 2 shows the word error rates of baseline and adaptation for each sentence length. From this table, for short sentences, the recognition rate of bias adaptation is worse than that of the baseline result. If CMS is used for sentences under 0.5 secs, our method improves the error rate by 0.1%.

5. Discussion

Our method works well in the laboratory. However some differences between our experiments and the SWITCHBOARD task exist. The HMMs that I used in our laboratories are digit HMMs. The number of HMMs is 10, and the number of states is 6 to 9. At the workshop, we used phoneme-based HMMs, and the number of states was 3. The number of HMMs was 44. Therefore I think phoneme based HMMs are more confused more easily than digit HMMs. I found that in some utterances, there is not enough data to train the bias for silence, so the bias for utterances might be unreliable. In future work, I plan on doing Delta cepstrum adaptation and Delta cepstrum adaptation. I also found that some power biases for silence are very high. This means this data has noise. So I think we should work on noise adaptation as well.

Table 1. Recognition result.

	Short test set	Complete test set
Baseline	52.2%	52.3%
Cepstral bias adaptation	50.5%	52.2%

- [1] B. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification", *Journal of the Acoustic Society of America*, vol. 55, June 1974, pp. 1304-1312.
- [2] A. Sankar and C.-H. Lee, "Robust speech recognition based on stochastic matching", *Proc. Int. Conf. Acoust. Speech, Signal Process.*, May 1995, pp. 121-124.
- [3] M. G. Rahim and B.-H. Juang, "Signal bias removal for robust telephone-based speech recognition in adverse environments", *Proc. Int. Conf. Acoust. Speech, Signal Process.*, April 1994, pp. 445-448.
- [4] Y. Minami and S. Furui, "A maximum likelihood procedure for a universal adaptation method based on HMM composition", *Proc. Int. Conf. Acoust. Speech, Signal Process.*, May 1995, pp. 129-132.

Table 2. Word error rate depending on sentence length in se

	Overall number of sentences	Overall number of words	Word error rate	
			Baseline	Adaptation
0 sec. \leq sentence length \leq 0.5 sec.	141	159	40.9%	47.8%
0.5 sec. < sentence length \leq 1.0 sec.	502	898	56.1%	55.8%
1.0 sec. < sentence length \leq 2.0 sec.	512	2472	56.4%	56.8%
2.0 sec. < sentence length \leq 3.0 sec.	362	3090	52.4%	51.1%
3.0 sec. < sentence length	602	10850	51.2%	51.2%