Minimizing Speaker Variation Effects for Speaker-Independent Speech Recognition

Xuedong Huang

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

ABSTRACT

For speaker-independent speech recognition, speaker variation is one of the major error sources. In this paper, a speaker-independent normalization network is constructed such that speaker variation effects can be minimized. To achieve this goal, multiple speaker clusters are constructed from the speaker-independent training database. A codeword-dependent neural network is associated with each speaker cluster. The cluster that contains the largest number of speakers is designated as the golden cluster. The objective function is to minimize distortions between acoustic data in each cluster and the golden speaker cluster. Performance evaluation showed that speaker-normalized front-end reduced the error rate by 15% for the DARPA resource management speaker-independent speech recognition task.

1. INTRODUCTION

For speaker-independent speech recognition, speaker variation is one of the major error sources. As a typical example, the error rate of a well-trained speaker-dependent speech recognition system is three times less than that of a speaker-independent speech recognition system [11]. To minimize speaker variation effects, we can use either speaker-clustered models [28, 11] or speaker normalization techniques [2, 24, 3, 25, 7]. Speaker normalization is interesting since its application is not restricted to a specific type of speech recognition systems. In comparison with speaker normalization techniques, speaker-clustered models will not only fragment data, but also increase the computational complexity substantially, since multiple models have to be maintained and compared during recognition.

Recently, nonlinear mapping based on neural networks has attracted considerable attention because of the ability of these networks to optimally adjust the parameters from the training data to approximate the nonlinear relationship between two observed spaces (see [22, 23] for a review), albeit much remains to be clarified regarding practical applications. Nonlinear mapping of two different observation spaces is of great interest for both theoretical and practical purposes. In the area of speech processing, nonlinear mapping has been applied to noise enhancement [1, 32], articulatory motion estimation [29, 18], and speech recognition [16]. Neural networks have been used successfully to transform data of a new speaker to a reference speaker for speaker-adaptive speech recognition [11]. In this paper, we will study how neural networks can be employed to minimize speaker variation effects for speaker-independent speech recognition. The network is used as a nonlinear mapping function to transform speech data between two speaker clusters. The mapping function we used is characterized by three important properties. First, the assembly of mapping functions enhances overall mapping quality. Second, multiple input vectors are used simultaneously in the transformation. Finally, the mapping function is derived from training data and the quality will dependent on the available amount of training data.

We used the DARPA Resource Management (RM) task [27] as our domain to investigate the performance of speaker normalization. The 997-word RM task is a database query task designed from 900 sentence templates [27]. We used word-pair grammar that has a test-set perplexity of about 60. The speaker-independent training speech database consists of 3990 training sentences from 109 speakers [26]). The test set comprises of a total of 600 sentences from 20 speakers. We used all training sentences to create multiple speaker clusters. A codeword-dependent neural network is associated with each speaker cluster. The cluster that contains the largest number of speakers is designated as the golden cluster. The objective function is to minimize distortions between acoustic data in each cluster and the golden speaker cluster. Performance evaluation showed that speaker-normalized front-end reduced the error rate by 15% for the DARPA resource management speaker-independent speech recognition task.

This paper is organized as follows. In Section 2, the speech recognition system SPHINX-II is reviewed. Section 3 presents neural network architecture. Section 4 discusses its applications to speaker-independent speech recognition. Our findings are summarized in Section 5.

2. REVIEW OF THE SPHINX-II SYSTEM

In comparison with the SPHINX system [20], the SPHINX-II system [6] reduced the word error rate by more than 50% through incorporating between-word coarticulation modeling [13], high-order dynamics [9], sex-dependent shared-distribution semi-continuous hidden Markov models [9, 15]. This section will review SPHINX-II, which will be used as our baseline system for this study [6].

2.1. Signal Processing

The input speech signal is sampled at 16 kHz with a pre-emphasized filter, $1 - 0.9Z^{-1}$. A Hamming window with a width of 20 msec is applied to speech signal every 10 msec. The 32-order LPC analysis is followed to compute the 12-order cepstral coefficients. Bilinear transformation of cepstral coefficients is employed to approximate mel-scale representation. In addition, relative power is also computed together with cepstral coefficients. Speech features used in SPHINX-II include (t is in units of 10 msec) LPC cepstral coefficients; 40-msec and 80-msec differenced LPC cepstral coefficients; second-order differenced cepstral coefficients; and power, 40-msec differenced power, second-order differenced power. These features are vector quantized into four independent codebooks by the Linde-Buzo-Gray algorithm [21], each of which has 256 entries.

2.2. Training

Training procedures are based on the forward-backward algorithm. Word models are formed by concatenating phonetic models; sentence models by concatenating word models. There are two stages at training. The first stage is to generate the shared output distribution mapping table. Forty-eight context-independent *discrete* phonetic models are initially estimated from the uniform distribution. Deleted interpolation [17] is used to smooth the estimated parameters with the uniform distribution. Then context-dependent models have to be estimated based on context-independent ones. There are 7549 triphone models in the DARPA RM task when both within-word and between-word triphones are considered. To facilitate training, one codebook discrete models were used, where acoustic feature consists of the cepstral coefficients, 40-msec differenced cepstrum, power and 40-msec differenced power. After the 7549 discrete models are obtained, the distribution clustering procedure [14] is then applied to create 4500 distributions (senones). The second stage is to train 4-codebook models. We first estimate 48 context independent, four-codebook discrete models with the uniform distribution. With these context independent models and the senone table, we then estimate the shared-distribution SCHMMs [9]. Because of substantial difference between male and female speakers, two sets of sex-dependent SCHMMs are are separately trained to enhance the performance.

To summarize, the configuration of the SPHINX-II system has:

- four codebooks of acoustic features,
- shared-distribution between-word and within-word triphone models,
- sex-dependent SCHMMs.

2.3. Recognition

In recognition, a language network is pre-compiled to represent the search space. For each input utterance, the (artificial) sex is first determined automatically as follows [8, 31]. Assume each codeword occurs equally and assume codeword i is represented by a Gaussian density function $N(\mathbf{x}, \mu_i, \Sigma_i)$. Then given a segment of speech $\mathbf{x}_1^T$, $\Pr_{sex}$, the probability that $\mathbf{x}_1^T$ is generated from codebook-*sex* is approximated by:

$$\sum_t \sum_{i \in \eta_t} \log(N(\mathbf{x}_t, \mu_i, \Sigma_i))$$

where η_t is a set that contains the top N codeword indices during quantization for cepstrum data $\mathbf{x}_t$ at time t. If $\Pr_{male} > \Pr_{female}$, then $\mathbf{x}_1^T$ belongs to male speakers. Otherwise, $\mathbf{x}_1^T$ is female speech. After the sex is determined, only the models of the determined sex are activated during recognition. This saves both CPU time and memory requirement. For each input utterance, the Viterbi beam search algorithm is used to find out the optimal state sequence in the language network.

3. NEURAL NETWORK ARCHITECTURE

3.1. Codeword-Dependent Neural Networks (CDNN)

When presented with a large amount of training data, a single network is often unable to produce satisfactory results during training as each network is only suitable to a relatively small task. To improve the mapping performance, breaking up a large task and modular construction are usually required [5, 7]. This is because the nonlinear relationship between two speakers is very complicated, a simple network may not be powerful enough. One solution is to partition the mapping spaces into smaller regions, and to construct a neural network for each region as shown in Figure 1. As each neural network is trained on a separate region in the acoustic space, the complexity of the mapping required of each network is thus reduced. In Figure 1, the switch can be used to select the most likely network or top N networks based on some probability measures of acoustic similarity [10]. Functionally, the assembly of networks is similar to a huge neural network. However, each network in the assembly is learned independently with training data for the corresponding regions. This reduces the complexity of finding a good solution in a huge space of possible network configurations since strong constraints are introduced in performing complex constraint satisfaction in a massively interconnected network.

Vector quantization (VQ) has been widely used for data compression in speech and image processing. Here, it can be used to to partition original acoustic space into different prototypes (codewords). This partition can be regarded as a procedure to perform broad-acoustic pattern classification.

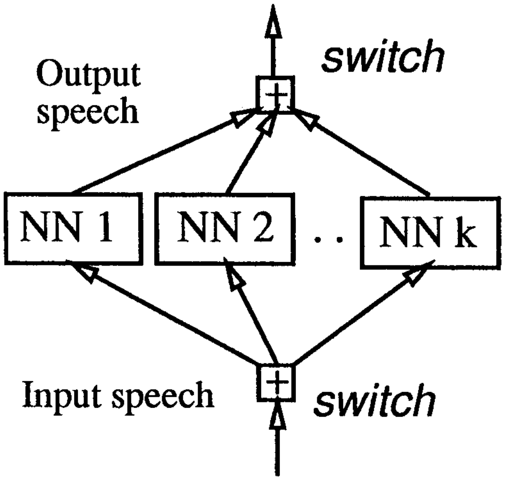


Figure 1: Codeword-dependent neural networks (CDNN).

The broad-acoustic patterns are automatically generated via a self-organization procedure based on the LBG algorithm [21]. When the codeword-dependent neural network (CDNN) was constructed from the data in the corresponding cell, it was found that learning for the CDNN converges very quickly in comparison with a huge neural network. The larger the codebook, the quicker it converges. However, the size of codebook relies on the number of available training data since codeword-dependent structure fragments training data. The size of codebook should be determined experimentally.

Speaker normalization involves acoustic data transformation from one speaker cluster to another. In general, let $\mathcal{X}^a = \mathbf{x}^a_1, \mathbf{x}^a_2, \ldots \mathbf{x}^a_t$ be a sequence of observations (frames) at time 1, 2, .. t of speaker a. Here, each observation at time k, $\mathbf{x}^a_k$, is a multidimensional vector, which usually characterizes some short-time spectral features. For the sequence of speech observations $\mathcal{X}^a$ produced by speaker-cluster a, our goal is to find a mapping function $\mathcal{F}(\mathcal{X}^a)$ such that $\mathcal{F}(\mathcal{X}^a)$ resembles the corresponding sequence of observations produced by speakers in the golden speaker cluster. Speaker variations include many factors such as vocal tract, pitch, speaking speed, intensity, and cultural differences. Unfortunately, given two different speakers, there is no simple mapping function that can account for all these variations. Consequently, we are mainly concerned with spectral normalization. For each frame $\mathbf{x}^a$, we want to find out a mapping function to transform it to $\mathbf{x}^b$, the corresponding phonetic realization produced by speaker b. We believe that $\mathbf{x}^a_t$ can represent most important features produced by the speaker. Thus, our objective functions is to minimize:

$$\sum_{corresponding\,pairs} \mathcal{D}(\mathcal{F}(\mathbf{x}^a) - \mathbf{x}^b) \qquad (1)$$

where $\mathcal{D}(\mathbf{x}, \mathbf{y})$ denotes a predefined distortion measure between frame $\mathbf{x}$ and $\mathbf{y}$, and *corresponding pairs* are constructed to approximate acoustic realizations of different speakers. Even if we are only interested in spectral normalization, there is no analytic mapping solution. Instead, stochastic approach has to be used to study the nonlinear relationship between the two observed spaces. We need to have a set of supervision data (*corresponding pairs* in Equation 1) to extract the nonlinear relationship.

It has been found that dynamic information plays an important role in speech recognition [4, 20, 12]. As frame to frame normalization lacks use of dynamic information, the architecture of normalization network is thus chosen to incorporate multiple neighboring frames. One of such architectures is shown in Figure 2. Here, the current frame and its left and right neighboring frames are fed to the multi-layer neural network as inputs. The network output is a normalized frame corresponding to the current input frame. By using multiple input frames for the network, the important dynamic information can be effectively used in estimating network parameters and in normalization. In Figure 2, there are input layer, hidden layer, and output layer. Each arc k is associated with a weight w_k. In the hidden and output layer, each node is characterized by an internal offset θ. The hidden node is also characterized by a nonlinear sigmoid function. The input to each hidden node and output node is a weighted sum of corresponding inputs with the offset θ. Both the internal offset and arc weights are learned by the backpropagation algorithm [30], which uses a gradient search to minimize the objective function. If the dimension of observation space is d and the number of input frames is m, we will have $d\text{x}m$ input units in the normalization network. If we want to incorporate more neighboring frames, this will definitely increase the number of free parameters in the network. Although the increase in the number of free parameters lead to quick convergence during training, this nevertheless may not lead to improved general-

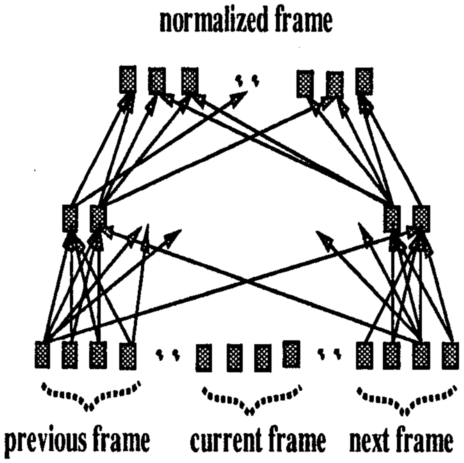


Figure 2: A basic neural network architecture.

ization capability. Since the network is designed to normalize new data from a given speaker to the reference speaker, good generalization capability will be the most important concern. Therefore, a compromise has to be made between generalization capability and the number of free parameters.

3.2. Golden Speaker-Cluster Selection

Speaker-dependent CDNNs have been used successfully for speaker-adaptive speech recognition [7] (speaker-dependent mapping). If we need to map multiple speakers to one golden speaker and simply construct a speaker-independent CDNN, it is unlikely that a single network will do the job. With the same rational as CDNN for speaker-adaptive speech recognition, we can partition multiple speakers into speaker-clusters and construct cluster-dependent CDNN.

For speaker clustering, we first generated 48 phonetic HMM for each speaker in the speaker-independent training database. Thus, for each speaker, we have a set of output distributions. We then merge the two speaker-clusters iteratively that resulted in the least loss of information, and then move elements from cluster to cluster to improve the overall quality. The clustering procedure used here is similar to the one used for generalized triphone clustering [19]. We can continue the clustering process until the specified speaker-clusters are obtained. The golden speaker-cluster is the one that contains the largest number of speakers. We generated two golden clusters for male and female respectively.

4. EXPERIMENTAL EVALUATION

4.1. Experiment conditions

Through this study, only the cepstral vectors are considered for normalization. Once we have the normalized cepstral vector, the first-order and second-order time derivatives can be computed. We first clustered all the speakers in the training set into male and female clusters, and then generated 10 speaker-clusters for male and 7 speaker-clusters for female. We selected two golden speaker-clusters for both male and female. There were 13 and 6 speakers in the male and female golden cluster respectively. To provide learning examples for network learning, we first segmented all the training utterances into triphones using Viterbi alignment and then used the DTW algorithm to warp the data to the corresponding triphone pairs in the golden speaker-cluster. Thus, for a given frame of each training speaker, the desired output frame for network learning is the golden speaker frame paired in the DTW optimal path.

4.2. Benchmark Experiments

As benchmark experiments, speaker-independent speech recognition using SPHINX-II was first evaluated. The word error rate we used here reflects all three types of errors and is computed as

$$100\frac{substitutions + deletions + insertions}{totalwords + insertions} \tag{2}$$

The average error rate was 3.8% for speaker-independent speech recognition.

4.3. Normalization Results

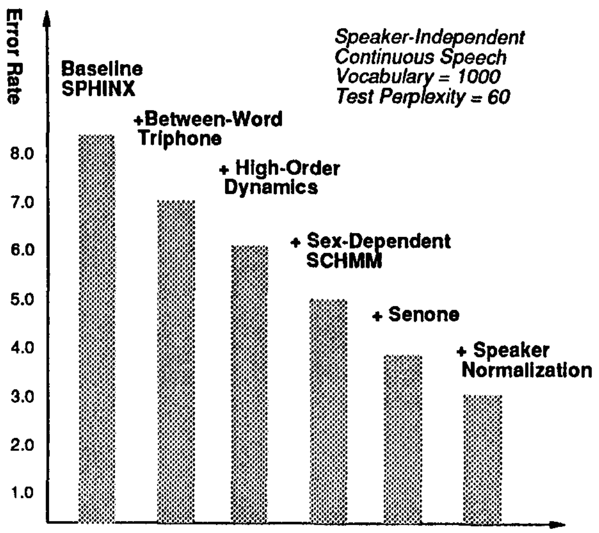
The input of the network consists of three frames from the new speaker. Here, 12 cepstral coefficients and energy are used together. Thus, there are 93 input units in the network. The output of the network has 13 units corresponding the normalized frame, which is made to approximate the frame of the desired reference speaker. The energy output is discarded as it is relative unstable. The objective function for network learning is to minimize the distortion (mean squared error) between the network output and the desired reference speaker frame. The network has one hidden layer with 20 hidden units. Each hidden unit is associated with the generalized $SIGMOID$ function, where α, β and γ are predefined to be 4.0, 1.8, 2.0 respectively. They are fixed for all the experiments conducted here. The weights and offsets in the network were initialized with small random values. The learning step and momentum are controlled dynamically. Experimental experience indicates that 300 to 600 epochs are required to achieve acceptable distortion. We created two golden speaker clusters for male and female respectively. There were seven female clusters and ten male clusters, which are designed according to the available amount of male/female training data. For each speaker cluster, we built a cluster-dependent codebook (size 16). For the input speech signal, joint VQ pdfs are used to select the top 2-5 clusters for normalization. Thus, let λ_i denote the probability that acoustic vector belong to cluster i, and $\mathcal{X}_i$ denote the normalized vector using the ith cluster-dependent CDNN. The normalized vector $\mathcal{Y}$ can then be computed as

$$\mathcal{Y} = \frac{\sum_i \lambda_i \mathcal{X}_i}{\sum_i \lambda_i} \tag{3}$$

With the same training conditions as used in SPHINX-II, when the speaker-normalized front-end is used, we reduced the error rate from 3.8% to 3.3%, which represented 15% error reduction. The modest error reduction indicated the mapping quality still needs to be improved substantially.

5. SUMMARY

In this paper, the codeword-dependent neural network (CDNN) was presented for speaker-independent speech recognition. The network was used as a nonlinear mapping function to transform speech data between speakers in each cluster and the golden speaker cluster. Performance evaluation showed that speaker-normalized front-end reduced the error rate by 15%, as shown in Figure 3, for the DARPA



SPHINX-II System Summary

resource management speaker-independent speech recognition. If we compare the error rate of speaker-dependent and speaker-independent systems, this 15% error reduction is relatively small. We believe that the quality of mapping functions is extremely important if we want to bridge the gap between speaker-dependent and speaker-independent systems.

Acknowledgments

This research was sponsored by the Defense Advanced Research Projects Agency (DOD), Arpa Order No. 5167, under contract number N00039-85-C-0163. The authors would like to express their gratitude to Professor R. Reddy for his encouragement and support.

References

[1] Acero, A. and Stern, R. *Environmental Robustness in Automatic Speech Recognition.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1990, pp. 849–852.

[2] Choukri, K., Chollet, G., and Grenier, Y. *Spectral transformations through cannonical correlation analysis for speaker adapataion in ASR.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1986, pp. 2659–2552.

[3] Class, F., Kaltenmeier, A., Regel, P., and Trottler, K. *Fast speaker adaptation for speech recognition.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1990, pp. 133–136.

[4] Furui, S. *Speaker-Independent Isolated Word Recognition Using Dynamic Features of Speech Spectrum.* **IEEE Transactions on Acoustics, Speech, and Signal Processing**, vol. ASSP-34 (1986), pp. 52–59.

[5] Hampshire, J. and Waibel, A. *The Meta-Pi Network: Connectionist rapid adapatation for high-performance multi-speaker phoneme recognition.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1990, pp. 165–168.

[6] Huang, X., Alleva, F., Hon, H., Hwang, M., and Rosenfeld, R. *The SPHINX-II Speech Recognition System: An Overview.* Technical Report, no. CMU-CS-92-112, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, February 1992.

[7] Huang, X. *Speaker Adaptation Using Codeword-Dependent Neural Networks.* in: **IEEE Workshop on Speech Recognition, Arden House.** 1991.

[8] Huang, X. *A Study on Speaker-Adaptive Speech Recognition.* in: **DARPA Speech and Language Workshop.** Morgan Kaufmann Publishers, San Mateo, CA, 1991.

[9] Huang, X., Alleva, F., Hayamizu, S., Hon, H., Hwang, M., and Lee, K. *Improved Hidden Markov Modeling for Speaker-Independent Continuous Speech Recognition.* in: **DARPA Speech and Language Workshop.** Morgan Kaufmann Publishers, Hidden Valley, PA, 1990, pp. 327–331.

[10] Huang, X., Ariki, Y., and Jack, M. **Hidden Markov Models for Speech Recognition.** Edinburgh University Press, Edinburgh, U.K., 1990.

[11] Huang, X. and Lee, K. *On Speaker-Independent, Speaker-Dependent, and Speaker-Adaptive Speech Recognition.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1991, pp. 877–880.

[12] Huang, X., Lee, K., Hon, H., and Hwang, M. *Improved Acoustic Modeling for the SPHINX Speech Recognition System.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** Toronto, Ontario, CANADA, 1991, pp. 345–348.

[13] Hwang, M., Hon, H., and Lee, K. *Modeling Between-Word Coarticulation in Continuous Speech Recognition.* in: **Proceedings of Eurospeech.** Paris, FRANCE, 1989, pp. 5–8.

[14] Hwang, M. and Huang, X. *Shared-Distribution Hidden Markov Models for Speech Recognition.* Technical Report CMU-CS-91-124, Carnegie Mellon University, April 1991.

[15] Hwang, M. and Huang, X. *Subphonetic Modeling with Markov States - Senone.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1992.

[16] Iso, K. and Watanabe, T. *Speaker-independnet word recognition using a neural prediction model.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1990, pp. 441–444.

[17] Jelinek, F. and Mercer, R. *Interpolated Estimation of Markov Source Parameters from Sparse Data.* in: **Pattern Recognition in Practice**, edited by E. Gelsema and L. Kanal. North-Holland Publishing Company, Amsterdam, the Netherlands, 1980, pp. 381–397.

[18] Kobayashi, T., Yagyu, M., and Shirai, K. *Applications of neural networks to articulatory motion estimation.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1991, pp. 489–4920.

[19] Lee, K. *Context-Dependent Phonetic Hidden Markov Models for Continuous Speech Recognition.* **IEEE Transactions on Acoustics, Speech, and Signal Processing**, April 1990, pp. 599–609.

[20] Lee, K., Hon, H., and Reddy, R. *An Overview of the SPHINX Speech Recognition System.* **IEEE Transactions on Acoustics, Speech, and Signal Processing**, January 1990, pp. 35–45.

[21] Linde, Y., Buzo, A., and Gray, R. *An Algorithm for Vector Quantizer Design.* **IEEE Transactions on Communication**, vol. COM-28 (1980), pp. 84–95.

[22] Lippmann, R. *Neural Nets for Computing.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1988, pp. 1–6.

[23] Lippmann, R. *Review of Research on Neural Nets for Speech.* in: **Neural Computation.** 1989.

[24] Montacie, C., Choukri, K., and Chollet, G. *Speech recognition using temporal decomposition and multilayer feed-forward automata.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1989, pp. 409–412.

[25] Nakamura, S. and Shikano, K. *A comparative study of spectral mapping for speaker adaptation.* **ICASSP**, 1990, pp. 157–160.

[26] Pallett, D., Fiscus, J., and Garofolo, J. *DARPA Resource Management Benchmark Test Results June 1990.* in: **DARPA Speech and Language Workshop**. Morgan Kaufmann Publishers, San Mateo, CA, 1990, pp. 298–305.

[27] Price, P., Fisher, W., Bernstein, J., and Pallett, D. *A Database for Continuous Speech Recognition in a 1000-Word Domain.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1988, pp. 651–654.

[28] Rabiner, L., Lee, C., Juang, B., and Wilpon, J. *HMM Clustering for Connected Word Recognition.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1989, pp. 405–408.

[29] Rahim, M., Kleijn, W., Schroeter, J., and Goodyear, C. *Acoustic to articulatory parameter mapping using an assembly of neural networks.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1991, pp. 485–488.

[30] Rumelhart, D., Hinton, G., and Williams, R. *Learning Internal Representation by Error Propagation.* in: **Learning Internal Representation by Error Propagation**, by D. Rumelhart, G. Hinton, and R. Williams, edited by D. Rumelhart and J. McClelland. MIT Press, Cambridge, MA, 1986.

[31] Soong, F., Rosenberg, A., Rabiner, L., and Juang, B. *A Vector Quantization Approach to Speaker Recognition.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1985, pp. 387–390.

[32] Tamura, S. and Waibel, A. *Noise reduction using connectionist modelsnce Measure for Speech Recognition.* in: **IEEE International Conference on Acoustics, Speech, and Signal Processing.** 1988, pp. 553–556.