A DEEP NEURAL NETWORK INTEGRATED WITH FILTERBANK LEARNING FOR SPEECH RECOGNITION

Hiroshi Seki, Kazumasa Yamamoto, and Seiichi Nakagawa

Department of Computer Science and Engineering
Toyohashi University of Technology, Japan
{seki, kyama, nakagawa}@slp.cs.tut.ac.jp

ABSTRACT

Deep neural networks (DNN) have achieved significant success in the field of speech recognition. One of the main advantages of the DNN is automatic feature extraction without human intervention. Therefore, we incorporate a pseudo-filterbank layer to the bottom of DNN and train the whole filterbank layer and the following networks jointly, while most systems take pre-defined mel-scale filterbanks as acoustic features to DNN. In the experiment, we use Gaussian functions instead of triangular mel-scale filterbanks. This technique enables a filterbank layer to maintain the functionality of frequency domain smoothing. The proposed method provides an 8.0% relative improvement in clean condition on ASJ+JNAS corpus and a 2.7% relative improvement on noise-corrupted ASJ+JNAS corpus compared with traditional fully-connected DNN. Experimental results show that the frame-level transformation of filterbank layer constrains flexibility and promotes learning efficiency in acoustic modeling.

Index Terms— automatic speech recognition, deep neural networks, acoustic models, filterbank learning, data-driven filterbank

1. INTRODUCTION

Deep neural networks (DNNs) have been applied to automatic speech recognition (DNN-HMM; deep-neural-network hidden Markov models) and have outperformed conventional Gaussian mixture model (GMM) based methods [1]. Recently, there has been a focus on the front-end learning based on DNNs, such as speech enhancement and filterbank-learning which takes low-level acoustic features [2, 3]. These works show better performance compared to those based on a hand-crafted procedure. This deep hierarchical feature extraction under a simple objective function is one of the main advantages of the DNNs. In [4] and [5], an analysis is conducted to evaluate the difference between hand-crafted filterbanks and learned filterbanks. These analyses show that there is a similarity between the center frequency of mel-scale filterbanks and learned filterbanks. However, learned center frequency of filterbanks in [4] and [5] do not show consistency. These results suggest that the shapes of learned filterbanks depend on the task, especially in the presence of background noise.

Earlier works on data-driven filterbank learning go back to shallow neural networks. Biem, et. al. [6] learned filterbanks and classifier jointly under the condition that the filterbank is parameterized by a Gaussian function. Accuracy was improved by training filterbanks and classifier jointly. To our knowledge, this method was not applied to a state-of-the-art DNN based system. However, there are some studies that take over the joint training of filterbanks and classifier. Sainath, et. al. [7] learned filterbanks and classifier (DNN) jointly under the restriction that the element of the filterbank is always positive by introducing the exponential of weights, $\exp(W)$. This weak restriction does not give a function explicitly which is the original purpose of hand-crafted triangular filterbanks. That is to say, the parameters of pseudo-filterbanks overfit to the given data and shape of pseudo-filterbanks lead to multiple peaks. Such pseudo-filterbanks do not have the ability to apply frequency-domain smoothing.

Unlike a fully connected layer, a convolutional layer of CNN (Convolutional Neural Networks) focuses on small localized regions of the input speech [8]. CNN can extract shift-invariant features and further improve recognition accuracy. In addition, weight sharing greatly reduces the number of parameters and increases learning efficiency. Several studies further reduce the parameters by using Gabor filters as a convolutional layer which takes Power-Normalized Spectrum as the input feature [9].

In this paper, we introduce pseudo-filterbanks to a bottom of DNN. In reference to [6], we use the Gaussian function as pseudo-filterbanks. The Gaussian function has an advantage over a convolutional layer in the number of free parameters and in the fast adaptation of pseudo-filters. In the experiment, the filterbanks and classifier (DNN) are trained jointly and evaluated under a standard hybrid system and Weighted Finite-State Transducer (WFST) decoder.

The rest of the paper is organized as follows. The related works are summarized in Section 2. The architecture and training algorithm of the proposed system are presented

in Section 3. Experimental setup and results are presented in Section 4. Finally, Section 5 concludes the paper and discusses future work.

2. RELATED WORKS TO FILTERBANK LEARNING

Even though the performance of a DNN-HMM significantly outperforms a GMM-HMM, a mismatch between the training and testing conditions still deteriorates the performance of a DNN-HMM. To solve this problem, several adaptation methods have been proposed for DNN acoustic models. Learning of fitlerbanks can be approximated by some of adaptation methods, though there are differencies with regard to expressiveness and constraint. We summarize related works to filterbank learning in the next paragraph.

Adaptation techniques can be classified roughly into two types: feature space adaptation[10] and model adaptation. Feature space adaptation such as fMLLR (feature-space Maximum Likelihood Linear Regression) and VTLN (Vocal Tract Length Normalization) try to extract invariant features in a GMM-derived manner [11]. Model adaptation of DNN tries to fine-tune a large number of parameters directly using limited adaptation data. Li, et. al. [12] inserted an additional linear layer to the bottom of DNN (LIN; Linear Input Network). The parameters of this affine transformation are trained discriminatively. Seide, et. al [11] also inserted an additional linear layer which was tied across neighbor frames (fDLR; feature-space discriminative linear regression). By introducing a block-diagonal matrix (layer) and ignoring the connections between the external frames, the same transformation is applied to individual frames. Likewise, VTLN is approximately represented as a tridiagonal matrix [13]. This indicates that the expressiveness of VTLN is included in the fDLR. The fDLR is again included in the LIN.

3. DISCRIMINATIVE TRAINING OF GAUSSIAN FILTERBANKS

In this section, we describe the overview of the proposed system. We first describe the pseudo-filterbanks to be incorporated into DNN in Section 3.1. Then we present how we train the whole network jointly in Section 3.2.

3.1. Gaussian filterbanks

For acoustic feature extraction, triangular filters like HTK tool [14] are commonly used to compute mel-scale filterbank features. However, a triangular filter is not differentiable and unable to be incorporated into a scheme of a backpropagation algorithm. In addition, the filter to be incorporated should be computationally simple because the summation over frequency bins is required in every feedforward computation of neural networks. Therefore, a function of filter must be differentiable and simple.

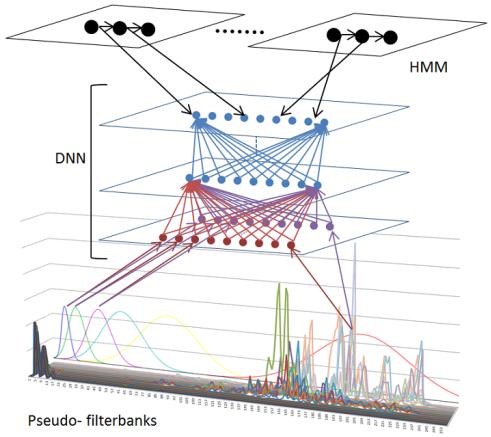

Fig. 1. Overview of the filterbank-incorporated DNN. The horizontal axis is for the frequency bin, and the vertical axis is for the power spectrum. In the experiment, input power spectra are concatenated from several consecutive frames (depth).

In the experiment, pseudo-filterbanks are modeled using Gaussian function [6]:

$$\theta_n(f) = \varphi_n \exp\left\{-\beta_n(p(\gamma_n) - p(f))\right\}^2 \quad (1)$$

where $\theta_n(f)$ is the n-th filter at frequency f. φ_n is the gain parameter, β_n is the bandwidth parameter, and γ_n is the center frequency. Linear frequency f is mapped on the Mel scale by the function $p(f)$. Trainable parameters are as follows: φ_n (gain), β_n(bandwidth), and γ_n(center frequency). Both a traditional triangular filter and a Gaussian filter maintain the functionality of frequency domain smoothing. A main difference between the filters is the coverage of frequency bin. A Gaussian filter focuses on all frequency bins while a triangular filter zeroes out the frequency bins outside a certain distance of bins.

3.2. Training algorithm

The overview of the Gaussian-filterbank-incorporated DNN is shown in Figure 1. Power spectra $x(f)$ are concatenated and fed into the pseudo-filterbanks. These features are multiplied by the corresponding filter gain by Equation 1 and summed across the frequency bin. Then applying a log-compression gives log-mel (pseudo-) filterbank feature:

$$h_n = \log(\sum_{f}^{256} \theta_n(f)x(f)) \quad (2)$$

h_n are fed into the following DNN. The parameters of filterbanks are trained within the framework of backpropagation. The update rule of φ_n, for example, is as follows:

$$\varphi_n^{new} = \varphi_n^{old} - \eta \frac{\partial L}{\partial \varphi_n} \quad (3)$$

where L is an objective function, and η is a learning rate. β_n(bandwidth) and γ_n(center frequency) are updated in the same manner.

When we focus on the gain parameter, it corresponds to fMLLR. On the other hand, when we focus on the shift of center frequency, our proposed system has an ability similar to VTLN. VTLN is implemented by warping the frequencies of Mel-scale filterbanks. This frequency warping is also accomplished by a shift of the center frequency in the proposed system. In addition, there are some advantages compared to conventional VTLN and fDLR. First, the proposed system can apply VTLN in a discriminative manner based on backpropagation. Second, pseudo-filterbanks can be constructed using a small number of parameters by assuming the Gaussian function. This technique dramatically decreases the number of parameters, and is advantageous in terms of model adaptation compared to other data driven feature extractors.

The proposed system was trained in two stages. First, DNN except for pseudo-filterbanks are fine-tuned (hereinafter, referred to as "fixed model"). Then the pseudo-filterbanks and following DNN are trained jointly (hereinafter, referred to as "trained model").

4. EXPERIMENTS

4.1. Experimental Setup

We used ASJ [15]+JNAS [16] to train acoustic models and learn data-driven center frequencies. ASJ+JNAS consists of 20,337 (≈33 h) and 25,056 (≈44 h) newspaper sentences uttered by 133 male speakers and 164 female speakers, respectively. In addition to the above corpora, we further evaluated the proposed system under noisy speech to learn data-driven gain parameters. We added noises from a NOISEX-92 database [17] to a quarter of the speech of ASJ+JNAS uttered by male speakers while varying the signal-to-noise ratio (SNR). Noise types of speech, car, F16, and Lynx with 10 dB, 15 dB, and 20 dB SNRs are used to deteriorate the speech. In total, the second training data (noise corrupted ASJ+JNAS) consists of 81,357 sentences (20337 utters. + 5085 utters. $\times$ 4 noise types $\times$ 3 dB types, ≈134 h). The speech was analyzed using a 25-ms Hamming window with a pre-emphasis coefficient of 0.97 and shifted with a 10-ms frame advance.

For an evaluation set, we used an IPA 100 test set consisting of 100 utterances uttered by 23 male speakers. As with the training data, we added the noise of speech, a car, an F16, and Lynx for the closed noise set, and machine gun, STITEL, a factory, and an operation room for the open noise set (≈5.3 h). An evaluation is conducted using Word Error Rate (WER).

We used a standard DNN-HMM hybrid system and rectified linear unit for an activation function [18]. Triphone HMMs were trained using an HTK toolkit [14]. The number of senones were set to 2087 for male speakers and 2562 for female speakers.

Table 1. WERs of baseline fully connected DNN and filterbank-incorporated DNN (clean trained).

system	WER [%]	
	Male	Female
baseline (triangle)	4.2	4.6
Gaussian (fixed)	4.6	5.0
Gaussian (trained)	3.8	4.3

A tri-gram based language model was trained on the Mainichi newspaper corpus (11,533,739 words in total, vocabulary size of 20,000 words) [19]. As the decoder, we used SPOJUS++(SPOken Japanese Understanding System) WFST version [20]. Next, we describe the experimental setup with respect to each acoustic model.

Baseline DNN

As a baseline system, we trained a fully connected DNN. The DNN has five hidden layers with 2,048 rectified linear units. Its input was 11 continuous frames of 40-dimensional log mel-scale filterbanks extracted by the HTK toolkit [14]. The features were normalized to zero mean and unit variance.

Incorporation of pseudo-filterbanks

We trained a fully connected DNN with pseudo-filterbanks. In our system, the system has a filterbank layer composed of 440 (log-) units at the bottom and five hidden layers with 2,048 rectified linear units at the middle. Its input was 11 continuous frames of 256-dimensional power spectra. The number of pseudo-filterbanks was set to 40 the same as in the baseline system ($n = 1, 2, ..., 40$). Initial values of pseudo-filterbanks are set as follows: Gains are set to 1.0, center frequencies are spaced equally along a Mel-scale, and bandwidths are set so that Two Sigma is equal to the corresponding bandwidth of the Mel-scale filterbank.

DNN with fDLR

We also trained DNN with fDLR. In this model, we inserted fDLR to the bottom of a baseline DNN. We used identity matrix as an initial value of fDLR. Except for the additional linear layer, the experimental setup of this model is as the same as the baseline DNN.

For network training, we used an annealing strategy with stochastic gradient descent (SGD) and momentum. When the frame level accuracy of the development set decreased, the learning rate was halved until it reached a minimum value for stopping. A 1% of training data were subtracted for the development set.

4.2. Evaluation Results

We first evaluated the proposed system under a clean ASJ+JNAS. Table 1 shows the WER of a baseline (triangle) DNN and

Table 2. Results on noisy ASJ+JNAS in WER under optimum condition (multi-condition trained).

system	WER [%]									
		closed noise set				open noise set				
	clean	speech	car	F16	Lynx	machine gun	STITEL	factory	operation room	Avg.
baseline (triangle)	6.6	10.6	7.9	11.0	10.1	21.4	43.4	38.1	48.0	21.9
Gaussian (trained)	5.8	10.6	7.8	12.1	10.9	24.1	41.1	36.4	42.9	21.3

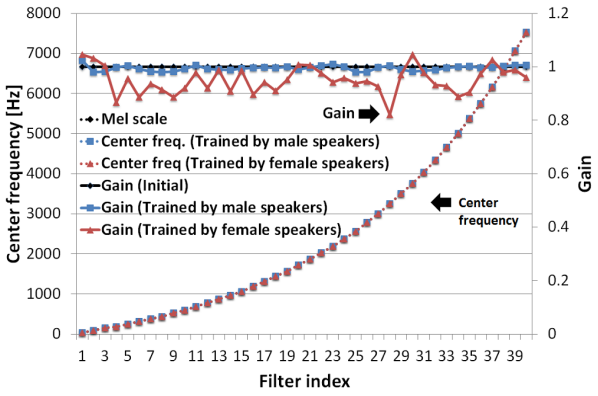

Fig. 2. Tuned parameters of Gaussian filters.

Table 3. WERs of DNN which includes fLDR in a training stage (clean trained).

system	WER [%]	
	Male	Female
baseline (triangle) + fDLR	3.7	4.5

DNN with pseudo-filterbanks. A baseline fully-connected DNN, which takes triangular filterbanks, achieved a WER of 4.2% for male speakers and 4.6% for female speakers. By introducing Gaussian filterbanks to the DNN, the trained model showed the best performance though the fixed Gaussian filterbanks were worse than the fixed triangular filterbanks. These results indicate that the discriminatively trained filterbanks improve performance. The shapes of the initial filterbanks and the learned filterbanks are depicted in Figure 2. From this figure, the center frequencies of the trained models do not differ from the mel scale and the tuning of gains contributes to performance. These learned gains showed different values dependent on male and female.

We also trained DNNs with the baseline (triangle) filterbank which contained fDLR to mimic the modeling ability of a framewise operation of a Gaussian filterbank. Table 3 shows the WER of fDLR. By introducing fDLR, WERs show the same performance as the trained model of the Gaussian filterbank-incorporated DNN. We can find that the frame-level transformation of the filterbank layer and the fDLR constrain flexibility and promote learning efficiency. The fDLR is mainly used to apply model adaptation. However, the fDLR is also beneficial during the training stage.

Next, we evaluated our system using noise corrupted ASJ+JNAS. Results are shown in Table 2. As same as the previous experiments on the clean ASJ+JNAS, we trained fully-connected baseline DNN and filterbank-incorporated DNNs. In this table, we summarized a WER with respect to clean speech and noise types. WERs of noisy speech are averaged over SNRs. The WERs over a whole test set are presented in the last column. The baseline system obtained a 21.9% average WER on the test set. Also, the trained model provides a WER of 21.3%, which gives a 2.7% relative improvement in WER over the baseline DNN. When we apply data-driven filterbanks, it appears to fit into the given training data in an excessive manner. However, improvements over the baseline DNN were obtained under the open noise set except for "machine gun". By introducing data-driven filterbanks composed of a Gaussian function, the filterbank layer can act as a more robust feature extractor under unknown conditions compared with handcrafted triangular filterbanks.

5. CONCLUSIONS

In this paper, we introduced a pseudo-filterbank layer to the bottom of DNN and trained whole networks jointly. The proposed systems are evaluated on ASJ+JNAS corpora and noise-added ASJ+JNAS corpora. The experimental results showed the effectiveness of the proposed method. The DNN with fDLR also improved recognition accuracy. We can find that the frame-level transformation of the proposed system and the fDLR promote improved learning efficiency.

One of our proposed future works is an adaptation of Gaussian filterbanks. In such a situation, it is important to devise which layers are to serve which functions. In other words, a filterbank layer should concentrate on speaker- and noise-specific functions such as VTLN to efficiently apply adaptation. It should be noted that there is a big difference between the center frequencies ($0 \sim 8000$) and other weights of the DNN. Therefore, the usage of same learning rate restrains further shift of center frequencies. Indeed, when we trained our proposed system with Adam[21], we obtained a larger shift of center friquencies compared with ones of SGD. In the future, we will also investigate optimization methods and relation to optimized filterbank parameters.

6. REFERENCES

[1] G. Hinton, L. Deng, D. Yu, G.E. Dahl, A.R. Mohamed, N. Jitaly, A. Senior, V. Vanhoucke, P. Ngyyen, T.N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[2] Z.Q. Wang and D. Wang, "Joint training of speech separation, filterbank and acoustic model for robust automatic speech recognition," *Proc Interspeech*, pp. 2839–2843, 2015.

[3] Z. Chen, S. Watanabe, H. Erdogan, and J.R. Hershey, "Speech enhancement and recognition using multi-task learning of long short-term memory recurrent neural networks," *Proc Interspeech*, pp. 3274–3278, 2015.

[4] T.N. Sainath, R.J. Weiss, A. Senior, K.W. Wilson, and O. Vinyals, "Learning the speech front-end with raw waveform CLDNNs," *Proc Interspeech*, pp. 1–5, 2015.

[5] H.B. Sailor and H.A. Patil, "Filterbank learning using convolutional restricted boltzmann machine for speech recognition," *Proc ICASSP*, pp. 5895–5899, 2016.

[6] A. Biem, S. Katagiri, E. McDermott, and B.H. Juang, "An application of discriminative feature extraction to filter-bank-based speech recognition," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 9, no. 2, pp. 96–110, 2001.

[7] T.N. Sainath, B. Kingsbury, A.R. Mohamed, and B. Ramabhadran, "Learning filter banks within a deep neural network framework," *Proc ASRU*, pp. 297–302, 2013.

[8] O. Abdel-Hamid, A.R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 22, no. 10, pp. 1533–1545, 2014.

[9] S.Y. Chang and N. Morgan, "Robust CNN-based speech recognition with Gabor filter kernels," *Proc Interspeech*, pp. 905–909, 2014.

[10] D.H. Nguyen, X. Xiao, E.S. Chen, and H. Li, "Feature adaptation using linear spectro-temporal transform for robust speech recognition," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 24, no. 6, pp. 1006–1019, 2016.

[11] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," *Proc ASRU*, pp. 24–29, 2011.

[12] B. Li and K.C. Sim, "Comparison of discriminative input and output transformations for speaker adaptation in the hybrid NN/HMM system," *Proc Interspeech*, pp. 526–529, 2010.

[13] D. Saito, N. Minematsu, and K. Hirose, "Rotational properties of vocal tract length difference in cepstral space," *Journal of Research Institute of Signal Processing*, vol. 15, no. 5, pp. 363–374, 2011.

[14] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book*, University of Cambridge.

[15] "ASJ continuous speech corpus for research (ASJ-JIPDEC)," http://research.nii.ac.jp/src/en/ASJ-JIPDEC.html.

[16] K. Itou, M. Yamamoto, K. Takeda, T. Takezawa, T. Matsuoka, T. Kobayashi, K. Shikano, and S. Itahashi, "JNAS: Japanese speech corpus for large vocabulary continuous speech recognition research," *The Journal of the acoustical society of Japan(E)*, vol. 20, no. 3, pp. 199–206, 1999.

[17] A. Varga and H.J.M. Steeneken, "Assessment for automatic speech recognition II: NOISEX-92: a database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Communication*, vol. 12, no. 3, pp. 247–251, 1993.

[18] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," *Aistats*, vol. 15, no. 106, 2011.

[19] "The Mainichi Newspapers," http://www.nichigai.co.jp/sales/mainichi/mainichi-series.html.

[20] Y. Fujii, K. Yamamoto, and S. Nakagawa, "Large vocabulary speech recognition system: SPOJUS++," *Proc WSEAS International Conference MUSP*, pp. 110–118, 2011.

[21] D.P. Kingma and J.L. Ba, "ADAM: A method for stochastic optimizatin," 2014, arXiv preprint arXiv:1412.6980.