ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414649

REPRESENTATION LEARNING FOR SPEECH RECOGNITION USING FEEDBACK BASED RELEVANCE WEIGHTING

Purvi Agrawal and Sriram Ganapathy

Learning and Extraction of Acoustic Patterns (LEAP) lab,
Electrical Engineering, Indian Institute of Science, Bangalore, India.

ABSTRACT

In this work, we propose an acoustic embedding based approach for representation learning in speech recognition. The proposed approach involves two stages comprising of acoustic filterbank learning from raw waveform, followed by modulation filterbank learning. In each stage, a relevance weighting operation is employed that acts as a feature selection module. In particular, the relevance weighting network receives embeddings of the model outputs from the previous time instants as feedback. The proposed relevance weighting scheme allows the respective feature representations to be adaptively selected before propagation to the higher layers. The application of the proposed approach for the task of speech recognition on Aurora-4 and CHiME-3 datasets gives significant performance improvements over baseline systems on raw waveform signal as well as those based on mel representations (average relative improvement of 15% over the mel baseline on Aurora-4 dataset and 7% on CHiME-3 dataset).

Index Terms— Speech representation learning, feedback of acoustic embeddings, raw speech waveform, 2-stage relevance weighting, speech recognition.

1. INTRODUCTION

Representation learning deals with the broad set of methods that enable the learning of meaningful representations from raw data. Similar to machine learning, representation learning can be carried out in an unsupervised fashion like principal component analysis (PCA), t-stochastic neighborhood embeddings (tSNE) proposed by [1] or in supervised fashion like linear discriminant analysis (LDA). Recently, deep learning based representation learning has drawn substantial interest. While a lot of success has been reported for text and image domains (for eg., word2vec embeddings [2]), representation learning for speech and audio is still challenging.

One of the research directions pursued for speech has been the learning of filter banks operating directly on the raw waveform [3–7], mostly in supervised setting. Other efforts attempting unsupervised learning of filterbank have also been investigated. The work in [8] used restricted Boltzmann machine while the efforts in [9] used variational autoencoders. The wav2vec method recently proposed by [10] explores unsupervised pre-training for speech recognition by learning representations of raw audio. There has been some attempts to explore interpretability of acoustic filterbank recently, for eg. SincNet filterbank by [11] and self-supervised learning by [12]. However, compared to vector representations of text which have shown to embed meaningful semantic properties, the interpretability of speech representations from these approaches has often been limited.

Subsequent to acoustic filterbank processing, modulation filtering is the process of filtering the 2-D spectrogram-like representation using 2-D filters along the time (rate filtering) and frequency (scale filtering) dimension. Several attempts have been made to learn the modulation filters also from data. The earliest approaches using LDA explored the learning of the temporal modulation filters in a supervised manner [13, 14]. Using deep learning, there have been recent attempts to learn modulation filters in an unsupervised manner [15, 16].

In this paper, we extend our previous work [17] on joint acoustic and modulation filter learning in the first two layers of a convolutional neural network (CNN) operating on raw speech waveform. The novel contribution of our approach is the incorporation of acoustic embeddings as feedback in the relevance weighting approach. In particular, the relevance weighting network is driven by the acoustic/modulation filter outputs along with the embedding of the previous one-hot targets. The output of the relevance network is a relevance weight which multiplies the acoustic/modulation filter [17]. The rest of the architecture performs the task of acoustic modeling for automatic speech recognition (ASR). The approach of feeding the model outputs back to the neural network is also previously reported as a form of recurrent neural network (RNN) called the teacher forcing network [18]. However, in this work, the embeddings of the model outputs are fed back only to the relevance weighting network and not as a RNN architecture.

The ASR experiments are conducted on Aurora-4 (additive noise with channel artifact) dataset [19], CHiME-3 (additive noise with reverberation) dataset [20] and VOiCES (additive noise with reverberation) dataset [21]. The experiments show that the learned representations from the proposed framework provide considerable improvements in ASR results over the baseline methods.

2. RELEVANCE BASED REPRESENTATION LEARNING

The block schematic of the senone embedding network is shown in Figure 1. The entire acoustic model using the proposed relevance weighting model is shown in Figure 3.

2.1. Step-0: Embedding network pre-training

The embedding network (Figure 1) is similar to the skip-gram network of word2vec models as proposed in [2]. In this work, the one-hot encoded senone (context dependent triphone hidden Markov model (HMM) states modeled in ASR) target vector at frame t, denoted as $\boldsymbol{h}_t$, is fed to a network whose first layer outputs the embedding denoted as $\boldsymbol{e}_t$. This embedding predicts the one-hot target vec-

This work was partly funded by grants from the Department of Atomic Energy (DAE/34/20/12/2018-BRNS/34088) project, and the Ministry of Human Resource and Development (MHRD), Government of India.

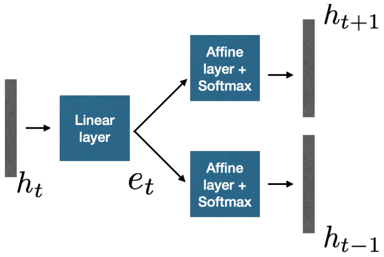

Fig. 1: Block schematic of senone embedding network used in the proposed model.

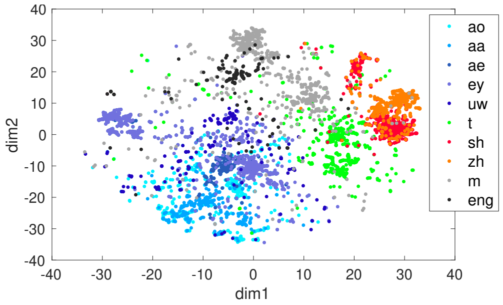

Fig. 2: t-SNE plot of the senone embeddings for TIMIT dataset.

tors for the preceding and succeeding time frames $\boldsymbol{h}_{t-1}$ and $\boldsymbol{h}_{t+1}$. This model is trained using the ASR labels for each task before the acoustic model training. Once the model is trained, only the embedding extraction part (first layer outputs) is used in the final ASR model. We use embeddings of 200 dimensions. During the ASR testing, the embeddings are derived by feeding the softmax outputs from the acoustic model (similar to teacher forcing network by [18]).

For the analysis, the TIMIT test set [22] consisting of 1344 utterances is used. The dataset is hand labelled for phonemes. The t-SNE visualization of the embeddings is shown in Fig. 2 for phonemes from TIMIT test set for a group of vowel phonemes {/ao/, /aa/, /ae/, /ey/, /uw/} and a group of plosives {/t/}, fricatives {/sh/, /zh/}, and nasals {/em/, /eng/}. As seen in the t-SNE plot of embeddings, the embeddings while being trained on one-hot senones, provides segregation of different phoneme types such as vowels, nasals, fricatives and plosives.

2.2. Step-1: Acoustic Filterbank representation [23]

The input to the neural network are raw samples windowed into S samples per frame with a contextual window of T frames. Each block of S samples is referred to as a frame. This input of size $S \times 1$ raw audio samples are processed with a 1-D convolution using F kernels (F denotes the number of sub-bands in filterbank decomposition) each of size L. The kernels are modeled as cosine-modulated Gaussian function [9, 23],

$$g_i(n) = \cos 2\pi\mu_i n \times \exp\left(-n^2\mu_i^2/2\right) \qquad (1)$$

where $g_i(n)$ is the i-th kernel ($i = 1, .., F$) at time n, μ_i is the center frequency of the ith filter (in frequency domain). The mean parameter μ_i is updated in a supervised manner for each dataset. The convolution with the cosine-modulated Gaussian filters generates F feature maps which are squared, average pooled within each frame and log transformed. This generates $\boldsymbol{x}$ as F dimensional features for each of the T contextual frames, as shown in Figure 3. The $\boldsymbol{x}$ can be interpreted as the "learned" time-frequency representation (spectrogram).

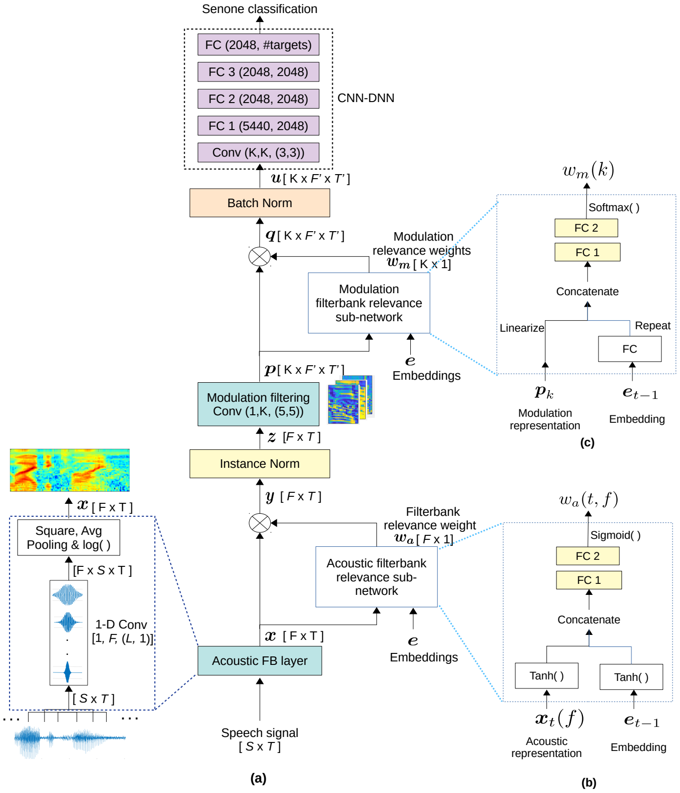

Fig. 3: (a) Block diagram of the proposed representation learning approach from raw waveform, (b) expanded acoustic FB relevance sub-network. Here, $\boldsymbol{x}_t(f)$ denotes the sub-band trajectory of band f for all frames centered at time t, $\boldsymbol{e}_{t-1}$ denotes the acoustic embedding vector for previous time step, (c) expanded modulation filterbank relevance sub-network.

2.3. Acoustic FB relevance weighting

The relevance weighting paradigm for acoustic FB layer is implemented using a relevance sub-network fed with the $F \times T$ time-frequency representation $\boldsymbol{x}$ and embeddings $\boldsymbol{e}$ of the previous time step. Let $\boldsymbol{x}_t(f)$ denote the vector containing sub-band trajectory of band f for all T frames centered at t (shown in Figure 3(b)). Then, $\boldsymbol{x}_t(f)$ is concatenated with embeddings of the previous time step $\boldsymbol{e}_{t-1}$ with $tanh()$ non-linearity. This is fed to a two layer deep neural network (DNN) with a sigmoid non-linearity at the output. It generates a scalar relevance weight $\boldsymbol{w}_a(t, f)$ as the relevance weight corresponding to the input representation at time t for sub-band f. This operation is repeated for all the F sub-bands which gives a F dimensional weight vector $\boldsymbol{w}_a(t)$ for the input $\boldsymbol{x}_t$.

The F dimensional weights $\boldsymbol{w}_a(t)$ multiply each column of the "learned" spectrogram representation $\boldsymbol{x}_t$ to obtain the relevance weighted filterbank representation $\boldsymbol{y}_t$. The relevance weights in the proposed framework are different from typical attention mechanism [24]. In the proposed framework, relevance weighting is applied on the representation as soft feature selection weights without performing a linear combination. We also process the first layer outputs ($\boldsymbol{y}$) using instance norm [25, 26].

In our experiments, we use $T = 101$ whose center frame is the senone target for the acoustic model. We also use $F = 80$ sub-bands and acoustic filter length $L = 129$. This value of L corresponds to 8 ms in time for a 16 kHz sampled signal. The value of S is 400 (25 ms window length) with frame shift of 10ms.

2.4. Step-2: Relevance Weighting of Modulation Filtered Representation

The representation $\boldsymbol{z}$ from acoustic filterbank layer is fed to the second convolutional layer which is interpreted as modulation filtering layer (shown in Figure 3). The kernels of this convolutional layer are 2-D spectro-temporal modulation filters, learning the rate-scale characteristics from the data. The modulation filtering layer generates K parallel streams, corresponding to K modulation filters $\boldsymbol{w}_K$. The modulation filtered representations $\boldsymbol{p}$ are max-pooled with window of 3×1, leading to feature maps of size $F' \times T'$. These are weighted using a second relevance weighting sub-network (referred to as the modulation filter relevance sub-network in Figure 3, expanded in Figure 3(c)).

The modulation relevance sub-network is fed with feature map $\boldsymbol{p}_k$; where $k = 1, 2, ..., K$, and embeddings e of the previous time step. The embedding $\boldsymbol{e}$ is linear transformed and concatenated with the input feature map. This is fed to a two-layer DNN with softmax non-linearity at the output. It generates a scalar relevance weight $w_m(k)$ corresponding to the input representation at time t (t as center frame) for kth feature map. The weights $\boldsymbol{w}_m$ are multiplied with the representation $\boldsymbol{p}$ to obtain weighted representation $\boldsymbol{q}$. The resultant weighted representation $\boldsymbol{q}$ is fed to the batch normalization layer [27]. We use the value of $K = 40$ in the work. Following the acoustic filterbank layer and the modulation filtering layer (including the relevance sub-networks), the acoustic model consists of series of CNN and DNN layers with sigmoid nonlinearity.

3. EXPERIMENTS AND RESULTS

The speech recognition system is trained using PyTorch [28] while the Kaldi toolkit [29] is used for decoding and language modeling. The models are discriminatively trained using the training data with cross entropy loss and Adam optimizer [30]. A hidden Markov model - Gaussian mixture model (HMM-GMM) system is used to generate the senone alignments for training the CNN-DNN based model. The ASR results are reported with a tri-gram language model or using a recurrent neural network language model (RNN-LM).

For each dataset, we compare the ASR performance of the proposed approach of learning acoustic representation from raw waveform with acoustic FB (A) with relevance weighting (A-R) and modulation FB (M) with relevance weighting (M-R) denoted as (A-R,M-R), traditional log mel filterbank energy (MFB) features (80 dimension), power normalized filterbank energy (PFB) features [31], mean Hilbert envelope (MHE) features [32], and excitation based (EB) features [33]. We also compare performance with the SincNet method proposed in [11]. Note that the modulation filtering layer (M) is part of the baseline model, and hence notation M is not explicitly mentioned in the discussion. The neural network architecture shown in Figure 3 (except for the acoustic filterbank layer, the acoustic FB relevance sub-network and modulation filter relevance sub-network) is used for all the baseline features.

3.1. Aurora-4 ASR

This database consists of read speech recordings of 5000 words corpus, recorded under clean and noisy conditions (street, train, car, babble, restaurant, and airport) at $10 - 20$ dB SNR. The training data has 7138 multi condition recordings (84 speakers) with total 15 hours of training data. The validation data has 1206 recordings for multi condition setup. The test data has 330 recordings (8 speakers) for each of the 14 clean and noise conditions. The test data are classified into group A - clean data, B - noisy data, C - clean data with channel distortion, and D - noisy data with channel distortion.

Table 1: Word error rate (%) for different configurations of the proposed model for the ASR task on Aurora-4 dataset.

Features	**ASR (WER in %)**				
	A	B	C	D	Avg.
Baseline Raw Waveform (A,M)	4.1	6.8	7.3	16.2	10.7
Acoustic Relevance					
A-R,M [Softmax, no embedding] [17]	3.6	6.4	8.1	15.1	10.0
A-R,M [Sigmoid, no embedding]	3.4	6.4	6.7	15.5	9.9
A-R,M [Sigmoid, with senone embedding]	3.4	6.2	6.7	14.5	**9.6**
Acoustic Relevance & Mod. Relevance					
A-R,M-R [Softmax, no embedding] [17]	3.6	6.1	**6.0**	14.8	9.6
A-R,M-R [Sigmoid, no embedding]	3.4	6.0	6.5	14.5	9.5
A-R,M-R [Sigmoid, with senone embeddings]	**3.0**	**5.8**	6.2	**14.4**	**9.1**

Table 2: Word error rate (%) in Aurora-4 database with various feature extraction schemes with decoding using trigram LM (and RNN-LM in paranthesis).

Cond	MFB	PFB	MHE	EB	Sinc	MFB-R	S-R,M-R	A-R,M-R
A. Clean with same Mic								
Clean	4.2	4.0	3.8	3.7	4.0	3.9	3.8	**3.0 (2.9)**
B: Noisy with same Mic								
Airport	6.8	7.1	7.3	-	6.9	6.7	6.2	**5.7**
Babble	6.6	7.4	7.4	-	6.7	6.5	6.1	**5.7**
Car	4.0	4.5	4.3	-	4.0	4.1	3.9	**3.6**
Rest.	9.4	9.6	9.1	-	9.4	9.6	8.4	**7.0**
Street	8.1	8.1	7.6	-	8.4	8.4	7.5	**6.3**
Train	8.4	8.6	8.6	-	8.3	8.2	7.4	**6.8**
Avg.	7.2	7.5	7.4	6.0	7.3	7.2	6.6	**5.8 (5.3)**
C: Clean with diff. Mic								
Clean	7.2	7.3	7.3	**5.0**	7.3	7.1	6.8	6.2 (5.9)
D: Noisy with diff. Mic								
Airport	16.3	18.0	17.6	-	16.2	16.2	**13.9**	14.0
Babble	16.7	18.9	18.6	-	17.6	16.9	16.0	**15.0**
Car	8.6	11.2	9.6	-	9.0	8.9	7.9	**8.0**
Rest.	18.8	21.0	20.1	-	19.0	18.8	19.2	**18.5**
Street	17.3	19.5	18.8	-	17.3	17.8	16.6	**15.8**
Train	17.6	18.8	18.7	-	18.1	17.9	16.6	**15.3**
Avg.	15.9	17.9	17.3	15.8	16.2	16.1	15.1	**14.4 (13.7)**
Avg. of all conditions								
Avg.	10.7	11.7	11.4	9.9	10.8	10.8	9.9	**9.1 (8.7)**

The ASR performance on the Aurora-4 dataset is shown in Table 1 for various configurations of the proposed approach and Table 2 for different baseline features. In order to observe the impact of different components of the proposed model, we tease apart the components and measure the ASR performance (Table 1). The fifth row (A-R,M-R, softmax with no-embedding) refers to the previous attempt using the 2-stage filter learning reported in [17]. In this paper, we explore the variants of the proposed model such as use of softmax nonlinearity instead of sigmoid in both relevance weighting sub-networks, sigmoid in both relevance weighting sub-networks, without and with senone embedding, and the 2-stage approach (both relevance weighting sub-networks). Among the variants with acoustic relevance weighting alone, the A-R [sigmoid with senone embeddings] improves over the softmax nonlinearity. With joint A-R,M-R case, again the sigmoid with senone embeddings provides the best result.

While comparing with different baseline features in Table 2, it can be observed that most of the noise robust front-ends do not improve over the baseline mel filterbank (MFB) performance. The raw waveform acoustic FB performs similar to MFB baseline features on average while performing better than the baseline for Cond. A and B. The ASR system with MFB-R features, which denote the application of the acoustic FB relevance weighting over the fixed mel filterbank features, also does not yield improvements over the system with baseline MFB features. We hypothesize that the learning of the relevance weighting with learnable filters allows more freedom in learning the model compared to learning with fixed mel fil-

Table 3: Word error rate (%) in CHiME-3 Challenge database for multi-condition training.

Test Cond	MFB	PFB	RAS	MHE	A-R	A-R,M-R
Sim_dev	12.9	13.3	14.7	13.0	12.4	**11.9**
Real_dev	9.9	10.7	11.4	10.2	9.9	**9.5**
Avg.	11.4	12.0	13.0	11.6	11.2	**10.7**
Sim_eval	19.8	19.4	22.7	19.7	19.0	**18.7**
Real_eval	18.3	19.2	20.5	18.5	17.2	**17.0**
Avg.	19.1	19.3	21.6	19.1	18.1	**17.8**

Table 4: WER (%) for cross-domain ASR experiments.

Filters	**ASR Trained and Tested on**	
Learned on	Aurora-4	CHiME-3
Aurora-4	9.1	14.3
CHiME-3	9.2	14.2

ters. The proposed (A-R,M-R) representation learning (two-stage relevance weighting) provides considerable improvements in ASR performance over the baseline system with average relative improvements of 15% over the baseline MFB features. Furthermore, the improvements in ASR performance are consistently seen across all the noisy test conditions and with a sophisticated RNN-LM. In addition, the performance achieved is also considerably better than the results such as excitation based features (EB) reported by [33].

For comparison with the SincNet method by [11], our cosine modulated Gaussian filterbank is replaced with the sinc filterbank as kernels in first convolutional layer (acoustic FB layer in Fig. 3). The ASR system with sinc FB (Sinc) is trained jointly without any relevance weighting keeping rest of the architecture same as shown in Fig. 3. From results, it can be observed that the parametric sinc FB (without relevance weighting) performs similar to MFB and also our learned filterbank A. In addition, the relevance weighting with Sinc filterbank (S-R,M-R) results show that the relevance weighting is also applicable to other prior works on learnable front-ends.

3.2. CHiME-3 ASR

The CHiME-3 corpus for ASR contains multi-microphone tablet device recordings from everyday environments, released as a part of 3rd CHiME challenge [20]. Four varied environments are present - cafe (CAF), street junction (STR), public transport (BUS) and pedestrian area (PED). For each environment, two types of noisy speech data are present - real and simulated. The real data consists of 6-channel recordings of sentences from the WSJ0 corpus spoken in the environments listed above. The simulated data was constructed by artificially mixing clean utterances with environment noises. The training data has 1600 (real) noisy recordings and 7138 simulated noisy utterances, constituting a total of 18 hours of training data. We use the beamformed audio in our ASR training and testing. The development (dev) and evaluation (eval) data consists of 410 and 330 utterances respectively. For each set, the sentences are read by four different talkers in the four CHiME-3 environments. This results in 1640 (410 × 4) and 1320 (330 × 4) real development and evaluation utterances.

The results for the CHiME-3 dataset are reported in Table 3. The ASR system with SincNet performs similar to baseline MFB features. The initial approach of raw waveform filter learning with acoustic FB relevance weighting (A-R) improves over the baseline system as well as the other multiple noise robust front-ends considered here. The proposed approach of 2-stage relevance weighting over learned acoustic and modulation representations provides significant improvements over baseline features (average relative improvements of 7% over MFB features in the eval set).

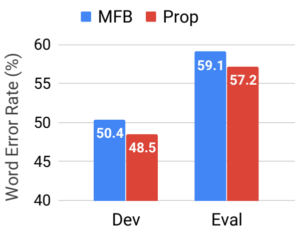

Fig. 4: ASR performance in WER (%) for VOiCES database.

3.3. Representation transfer across tasks

In a subsequent analysis, we perform a cross-domain ASR experiment, i.e., we use the acoustic filterbank learned from one of the datasets (either Aurora-4 or CHiME-3 challenge) to train/test ASR on the other dataset. The results of these cross-domain filter learning experiments are reported in Table 4. The rows in the table show the database used to learn the acoustic FB and the columns show the dataset used to train and test the ASR (all other layers in Figure 3 are learned in the ASR task). The performance reported in this table are the average WER on each of the datasets. The results shown in Table 4 illustrate that the filter learning process is relatively robust to the domain of the training data, suggesting that the proposed approach can be generalized for other "matched" tasks.

3.4. VOiCES ASR

The Voices Obscured in Complex Environmental Settings (VOiCES) corpus is a creative commons speech dataset being used as part of VOiCES Challenge [21]. The training data set of 80 hours has 22, 741 utterances sampled at 16kHz from 202 speakers, with each utterance having $12 - 15$s segments of read speech. We performed a 1-fold reverberation and noise augmentation of the data using Kaldi [29]. The ASR development set consists of 20 hours of distant recordings from the 200 VOiCES dev speakers. It contains recordings from 6 microphones. The evaluation set consists of 20 hours of distant recordings from the 100 VOiCES eval speakers and contains recordings from 10 microphones. The ASR performance on VOiCES dataset with baseline MFB features and our proposed approach (A-R,M-R) of 2-step relevance weighting is reported in Figure 4. These results suggest that the proposed model is also scalable to relatively larger ASR tasks where consistent improvements can be obtained with the proposed approach.

4. SUMMARY

The summary of the work is as follows.

- Extending the previous efforts in 2-stage relevance weighting approach with the use of embeddings feedback from past prediction.
- Incorporating the feedback in the form of word2vec style senone embedding for the task of learning representations.
- Performance gains in terms of word error rates for multiple ASR tasks.

5. REFERENCES

[1] Laurens van der Maaten and Geoffrey Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[2] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, "Efficient estimation of word representations in vector space," *Proc. of International Conference on Learning Representations (ICLR), arXiv preprint arXiv:1301.3781*, 2013.

[3] Dimitri Palaz, Ronan Collobert, and Mathew Magimai Doss, "Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks," *Proceedings of Interspeech*, pp. 1766–1770, 2013.

[4] Tara N Sainath, Brian Kingsbury, Abdel Rahman Mohamed, and Bhuvana Ramabhadran, "Learning filter banks within a deep neural network framework," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2013, pp. 297–302.

[5] Zoltán Tüske, Pavel Golik, Ralf Schlüter, and Hermann Ney, "Acoustic modeling with deep neural networks using raw time signal for LVCSR," in *Proc. of Interspeech*, 2014, pp. 890–894.

[6] Yedid Hoshen, Ron J Weiss, and Kevin W Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4624–4628.

[7] Tara N Sainath, Ron J Weiss, Andrew Senior, Kevin W Wilson, and Oriol Vinyals, "Learning the speech front-end with raw waveform CLDNNs," in *Proc. of Interspeech*, 2015, pp. 1–5.

[8] Hardik B Sailor and Hemant A Patil, "Filterbank learning using convolutional restricted boltzmann machine for speech recognition.," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5895–5899.

[9] Purvi Agrawal and Sriram Ganapathy, "Unsupervised raw waveform representation learning for ASR," *Proc. of Interspeech 2019*, pp. 3451–3455, 2019.

[10] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli, "wav2vec: Unsupervised pre-training for speech recognition," *Proc. of Interspeech*, pp. 3465–3469, 2019.

[11] Mirco Ravanelli and Yoshua Bengio, "Interpretable convolutional filters with SincNet," *in Proc. of Neural Information Processing Systems (NIPS)*, 2018.

[12] Santiago Pascual, Mirco Ravanelli, Joan Serrà, Antonio Bonafonte, and Yoshua Bengio, "Learning problem-agnostic speech representations from multiple self-supervised tasks," *Proc. of Interspeech*, pp. 161–165, 2019.

[13] Sarel Van Vuuren and Hynek Hermansky, "Data-driven design of RASTA-like filters.," in *Eurospeech*, 1997, vol. 1, pp. 1607–1610.

[14] Jeih-Weih Hung and Lin-Shan Lee, "Optimization of temporal filters for constructing robust features in speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 3, pp. 808–832, 2006.

[15] Hardik B Sailor and Hemant A Patil, "Unsupervised learning of temporal receptive fields using convolutional rbm for asr task," in *European Signal Processing Conference (EUSIPCO)*. IEEE, 2016, pp. 873–877.

[16] Purvi Agrawal and Sriram Ganapathy, "Modulation filter learning using deep variational networks for robust speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 244–253, 2019.

[17] Purvi Agrawal and Sriram Ganapathy, "Interpretable representation learning for speech and audio signals based on relevance weighting," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2823–2836, 2020.

[18] Ronald J Williams and David Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation*, vol. 1, no. 2, pp. 270–280, 1989.

[19] Hans-Günter Hirsch and David Pearce, "The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW)*, 2000.

[20] Jon Barker, Ricard Marxer, Emmanuel Vincent, and Shinji Watanabe, "The third 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," in *IEEE Workshop on ASRU*, 2015, pp. 504–511.

[21] Mahesh Kumar Nandwana, Julien Van Hout, Colleen Richey, Mitchell McLaren, Maria Alejandra Barrios, and Aaron Lawson, "The VOiCES from a distance challenge 2019," *Proc. of Interspeech*, pp. 2438–2442, 2019.

[22] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus," *NASA STI/Recon technical report*, vol. 93, 1993.

[23] Purvi Agrawal and Sriram Ganapathy, "Robust raw waveform speech recognition using relevance weighted representations," in *Proc. of Interspeech*, 2020, pp. 1649–1653.

[24] Yu Zhang, Pengyuan Zhang, and Yonghong Yan, "Attention-based LSTM with multi-task learning for distant speech recognition," *Proc. of Interspeech*, pp. 3857–3861, 2017.

[25] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533, 1986.

[26] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[27] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *Proc. of ICML*, pp. 448–456, 2015.

[28] Adam Paszke, Sam Gross, Soumith Chintala, and Gregory Chanan, "PyTorch," *Computer software. Vers. 0.3*, vol. 1, 2017.

[29] Daniel Povey et al., "The KALDI speech recognition toolkit," in *IEEE ASRU*. IEEE Signal Processing Society, 2011, number EPFL-CONF-192584.

[30] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *Proc. of ICLR, arXiv preprint arXiv:1412.6980*, 2015.

[31] Chanwoo Kim and Richard M Stern, "Power-normalized cepstral coefficients (PNCC) for robust speech recognition," in *ICASSP*, 2012, pp. 4101–4104.

[32] Seyed Omid Sadjadi, Taufiq Hasan, and John HL Hansen, "Mean Hilbert envelope coefficients (MHEC) for robust speaker recognition," in *Proc. of Interspeech*, 2012.

[33] Thomas Drugman, Yannis Stylianou, Langzhou Chen, Xie Chen, and Mark JF Gales, "Robust excitation-based features for automatic speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4664–4668.