Investigating NMF Speech Enhancement for Neural Network based Acoustic Models

Jürgen T. Geiger[1], *Jort F. Gemmeke*[2], *Björn Schuller*[3,1], *Gerhard Rigoll*[1]

[1]Institute for Human-Machine Communication, Technische Universität München, Munich, Germany
[2]Department ESAT, KU Leuven, Leuven, Belgium
[3]Department of Computing, Imperial College London, London, U.K.

geiger@tum.de

Abstract

In the light of the improvements that were made in the last years with neural network-based acoustic models, it is an interesting question whether these models are also suited for noise-robust recognition. This has not yet been fully explored, although first experiments confirm this question. Furthermore, preprocessing techniques that improve the robustness should be re-evaluated with these new models. In this work, we present experimental results to address these questions. Acoustic models based on Gaussian mixture models (GMMs), deep neural networks (DNNs), and long short-term memory (LSTM) recurrent neural networks (which have an improved ability to exploit context) are evaluated for their robustness after clean or multi-condition training. In addition, the influence of non-negative matrix factorization (NMF) for speech enhancement is investigated. Experiments are performed with the Aurora-4 database and the results show that DNNs perform slightly better than LSTMs and, as expected, both beat GMMs. Furthermore, speech enhancement is capable of improving the DNN result.

Index Terms: robust speech recognition, long short-term memory, speech enhancement

1. Introduction

Automatic speech recognition in realistic acoustic conditions (e. g. involving room reverberation and interfering noise sources) is still a major research challenge. System robustness can be achieved by several strategies at different levels [1]: speech/feature enhancement, robust features, or robust acoustic models. On the one hand, the speech signal can be enhanced using de-noising algorithms. Monaural signal separation techniques like non-negative matrix factorization (NMF) [2] are especially useful for cases where multi-channel audio with a specified microphone placement is not available. On the other hand, robust models and decoding methods are often employed. Such approaches addressing the robustness of the back-end of the recognition system were mostly developed for conventional systems using Gaussian mixture models (GMMs) for acoustic modelling.

Recently, deep neural networks (DNNs) gained popularity in speech recognition due to the improved acoustic modelling performance compared to GMMs [3], although the underlying methods had already been developed years ago [4]. In [5, 6], the potential of DNNs for robust recognition was demonstrated. It was shown that DNNs could be improved with noise-aware training, where the network exploits approximate knowledge of the noise. In addition, recurrent neural networks (RNNs) using the long short-term memory (LSTM) architecture [7] have become popular in speech recognition [8, 9, 10, 11, 12]. In [13], it was shown how LSTM networks can also be used for feature enhancement preprocessing. Because of their deep topology, DNN acoustic models can learn higher-level representations of the features by themselves. In this way, they also learn to process context information that is either introduced through feature frame stacking (for DNNs) or is inherently incorporated in the model topology (for LSTMs). Exploiting such context is helpful to improve noise robustness, for example in cases where a portion of frames within a longer window is spectrally masked by noise.

It is unclear whether methods for speech enhancement, such as NMF, that were successfully applied with GMMs are still useful for DNN acoustic models. In this work we investigate the influence of NMF speech enhancement on the performance of GMM, DNN, and LSTM acoustic models in different configurations.

In our contributions to the first and second CHiME challenges, we proposed the application of LSTMs for phoneme prediction in a multi-stream HMM framework in combination with a GMM. The systems were highly effective for small-vocabulary [14] and medium-vocabulary [15] recognition in environments with highly non-stationary noise and reverberation. While the GMM made use of NMF-enhanced speech, the exact interaction between NMF enhancement and LSTM acoustic models was not considered in detail.

There exist only a few studies that investigate the application of speech enhancement prior to DNN acoustic modelling. For example, in the study presented in [16], a speech enhancement method using spatial and spectral cues was capable of improving a DNN system. On the other hand, in [5], a DNN ASR system could not be improved by applying feature enhancement in the front-end. Beyond that, the interplay of clean and multi-condition training with speech enhancement and the influence of artifacts introduced by enhancement algorithms (requiring enhanced training data) are less explored with respect to DNN systems.

The analysis of previous work leaves the following key questions: how do DNN and LSTM acoustic models compare to conventional GMM systems for speech recognition in noisy environments? What are the effects of clean and multi-condition training? Furthermore, what is the influence of NMF speech enhancement on the three different acoustic models? In particular, in which cases is it necessary to perform additional speech enhancement on the training data (which will be referred to as

The research of Jort F. Gemmeke was funded by the IWT-SBO project ALADIN (contract 100049).

retraining)? The contribution of the present work is to address these questions with a series of experiments involving GMM, DNN, and LSTM acoustic models, using NMF speech enhancement for preprocessing.

Next, we describe the employed methods for acoustic modelling and speech enhancement. The experiments use the Aurora-4 experimental framework and are described in Section 3, along with a discussion of the results. Our conclusions follow in Section 4.

2. Methodology

The employed GMM and DNN systems are implemented in the Kaldi toolkit [17] and we used the implementations that are available as "recipes" for download.

2.1. GMM Acoustic Models

The GMM-HMM system is based on context-dependent tied-state triphone models. Each model has three HMM states and in total, there are around 2 000 distinct HMM states. Models are trained with maximum-likelihood parameter estimation. In addition, linear discriminant analysis (LDA) [18] and maximum likelihood linear transform (MLLT) [19] are employed for feature decorrelation. LDA is applied on stacked MFCC feature vectors (13 coefficients over seven consecutive frames), reducing the 91-dimensional vector to 40 dimensions.

2.2. Deep Neural Networks

Simply put, a deep neural network (DNN) is a multi-layer perceptron with more than one hidden layer. Multiple hidden layers are stacked on top of each other, which allows to extract higher-level information in the upper layers.

Such networks usually employ large numbers of parameters, because of the multiple layers with a high number of hidden units. This makes these networks difficult to train, and random initialisation of the parameters can result in a poor local optimum. In order to overcome this problem, pre-training is used to improve the parameter initialisation prior to training. The networks are initialised layer by layer in an unsupervised manner by treating each pair of layers as a restricted Boltzmann machine (RBM) [20].

2.3. LSTM Networks

The third acoustic modelling method employed in this study is the use of long short-term memory (LSTM) recurrent neural networks (RNNs).

Compared to a conventional RNN, the hidden units are replaced by so-called memory blocks. These memory blocks can store information in the cell variable $\boldsymbol{c}_t$. In this way, the network can exploit long-range temporal context. Each memory block consists of a memory cell and three gates: *input*, *output*, and *forget* gate, as depicted in Figure 1. These gates control the behaviour of the memory block. The forget gate can reset the cell variable which leads to 'forgetting' the stored input $\boldsymbol{c}_t$, while the input and output gates are responsible for reading from the input $\boldsymbol{x}_t$ and writing to the output $\boldsymbol{h}_t$, respectively:

$$\boldsymbol{c}_t = \boldsymbol{f}_t \otimes \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \otimes \tanh(\boldsymbol{W}_{xc}\boldsymbol{x}_t + \boldsymbol{W}_{hc}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c) \quad (1)$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \otimes \tanh(\boldsymbol{c}_t), \quad (2)$$

where $\otimes$ denotes element-wise multiplication and $\tanh$ is also applied in an element-wise fashion. The variables $\boldsymbol{i}_t$, $\boldsymbol{o}_t$, and $\boldsymbol{f}_t$ are the outputs of the input gates, output gates and forget gates, respectively, $\boldsymbol{b}_c$ is a bias term, and $\boldsymbol{W}$ is the weight matrix. Each memory block can be regarded as a separate, independent unit.

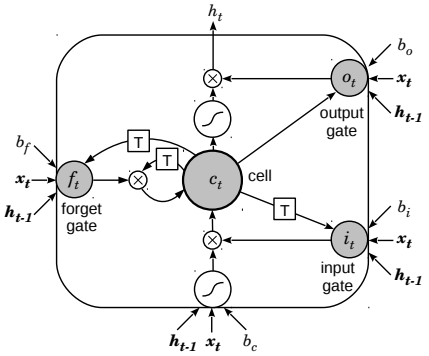

Figure 1: Long short-term memory block, containing a memory cell and the input, output and forget gates. T denotes a delay of one time step.

In addition to LSTM memory blocks, we use bidirectional RNNs [21]. A bidirectional RNN exploits context from both temporal directions, which makes it suitable for speech recognition, utterances are decoded as a whole. This is achieved by processing the input data in both directions with two separate hidden layers. Both hidden layers are then fed to the output layer. The combination of bidirectional RNNs and LSTM memory blocks leads to bidirectional LSTM networks [22]. Similar to the concept of DNNs, employing multiple hidden layers allows the system to "learn" higher-level feature representations, resulting in a deep LSTM.

The LSTM network is trained with on-line gradient descent using backpropagation through time, with cross entropy as an error function. Our GPU-enabled LSTM software is publicly available[1].

2.4. DNN and LSTM Acoustic Modelling

DNNs and LSTM networks are used for acoustic modelling in a hybrid HMM setup. In this method, the network predicts HMM states, resulting in state posterior probabilities. These posteriors are transformed to state likelihoods via Bayes' rule (using state probabilities determined from the forced alignment) and then substitute the GMM likelihoods during HMM decoding.

While DNNs exploit context through feature frame stacking, usually having access to 7-11 frames, context modelling is inherently incorporated in the complex LSTM topology. The LSTM is structured better due to the addition of the different gates. Therefore it is expected that LSTMs require a smaller number of trainable parameters compared to DNNs, and this is why pre-training is presumably not necessary for LSTMs.

2.5. NMF Speech Enhancement

The speech enhancement pre-processing component of our system uses NMF-based spectrogram factorisation algorithms previously employed in noise robust ASR experiments on Aurora-2, SPEECON and CHiME/GRID datasets [23, 24]. In short, noisy Mel-magnitude spectra are decomposed as a sparse, non-negative linear combination of speech and noise dictionary atoms. The speech atom activations are then used to obtain an estimate of the clean speech segments. In order to capture time

[1] https://sourceforge.net/p/currennt

context, the atoms span multiple time frames and utterances are decoded using a sliding-window method:

$$\mathbf{V} \approx \hat{\mathbf{V}} = \mathbf{\Lambda}^{(s)} + \mathbf{\Lambda}^{(n)} = \sum_{j=1}^{J} \mathbf{W}_j^{(s)} h_j + \sum_{k=1}^{K} \mathbf{W}_k^{(n)} h_k, \quad (3)$$

where $\mathbf{V}$ is a $B \times T$ spectrogram representing the current window of the observed noisy speech, B is the number of spectral bands, and T the number of consecutive frames per windowed spectrogram. The spectrograms $\mathbf{\Lambda}^{(s)}$ and $\mathbf{\Lambda}^{(n)}$ are estimates for the speech and noise content of the signal, respectively, $\mathbf{W}$ are $B \times T$ dictionary atoms and h their *activation weights*. We denote the number of speech atoms by J and similarly the noise dictionary size by K. The NMF dictionary atoms are formed by *exemplars*, spectrograms directly extracted from spectrograms [25].

The coefficients h_j and h_k are obtained through supervised NMF by minimizing the KL-divergence between $\mathbf{V}$ and $\hat{\mathbf{V}}$ regularised through a sparsity constraint on the activations [23]. After factorisation, speech and noise spectrogram estimates are generated for full utterances by averaging the frame estimates of overlapping windows. These are used to estimate a time-varying filter in the linear frequency domain to perform speech enhancement of the original noisy speech with the procedure described in [23]. The choice in favor of speech enhancement rather than feature enhancement leaves more freedom in the feature extraction and in the architecture of the back-end recogniser.

3. Experiments

3.1. Aurora-4 Database

The Aurora-4 experimental framework provides a database that is based on the WSJ-0 corpus of read speech. Different testing conditions are evaluated: clean speech (A, with 330 utterances), additive noise (B, 1 980 utterances), microphone variation (C, 330 utterances), and noise + microphone variation (D, 1 980 utterances), all from 8 speakers, summing up to 4 620 utterances. For the noisy test utterances, six noise types (street traffic, train terminals and stations, cars, babble, restaurants, and airports) were artificially added at varying SNR between 5 and 15 dB. For model training, a clean training set as well as a multi-condition training (MCT) set are provided, both containing 7 138 utterances from 83 speakers. The MCT set contains both clean and noisy utterances, all from different microphones. In addition, a development set is provided with the same partition as the evaluation test set (4 620 utterances under different conditions from 10 speakers).

3.2. Experimental Setup

The configuration of the GMM acoustic model was already described in Section 2.1. Instead of MFCC features, the DNN and LSTM models work with Mel filterbank coefficients as input. The DNN uses 40 coefficients, with a context size of 11 frames, summing up to 440 input features in total; the LSTM employs 26 coefficients (plus root-mean-square energy) along with delta coefficients, and the LSTM inputs are globally mean and variance normalised. For the DNN, the number of hidden layers is set to 7, each with 2 048 units, resulting in roughly 30 m weights. Due to its advantageous topology, the LSTM requires less parameters and it is thus comprised of 2 hidden layers, each with 250 units (125 per temporal direction). This sums to roughly 3 m weights. Both the DNN and LSTM systems are trained with an early stopping strategy, using the development set for validation. In the case of clean training, only the clean part of the development set is used for validation. State targets for training are obtained through a forced alignment of the GMM system that uses the same setup.

Table 1: WER (%) for different systems with or without multi-condition training (MCT), using original unenhanced data. Lowest scores are highlighted in bold font.

System	MCT	A	B	C	D	Avg.
GMM	-	5.1	**44.6**	**25.0**	**64.7**	**49.0**
DNN	-	**3.1**	50.9	48.7	69.7	55.4
LSTM	-	5.9	63.2	46.1	85.7	67.5
GMM	✓	8.5	13.4	13.1	27.4	19.0
DNN	✓	**3.6**	**7.9**	**10.4**	22.4	**14.0**
LSTM	✓	5.5	9.7	12.5	**22.0**	14.9

The speech enhancement operates on Mel-magnitude spectra, with $B = 40$ bands, using 25 ms hamming windows with 10 ms hop size. The NMF window length is $T = 15$ frames with a window shift of one frame. The sparsity for the speech was set at 0.075 times the average L_1 norm of the fixed part of the dictionary (speech and noise jointly). The noise sparsity was set at 0.5 times the speech sparsity. The number of iterations was kept constant at 350. These values (except the number of iterations) were taken from earlier work on the CHiME dataset [15].

The speech dictionary consists of 10 000 speech exemplars extracted by random sampling. In contrast to earlier work, the dictionary consists of exemplars extracted from the clean speech in the multicondition training data, thus covering multiple microphone characteristics. Evaluations (not shown) revealed that using only a subset of the speakers for training (implied by the subset of the multicondition training data corresponding to clean speech) does not impact the accuracy.

Two noise dictionaries were used: a fixed noise dictionary of 5 000 exemplars extracted randomly from the multicondition training data (by subtracting the clean speech), and a small noise dictionary extracted from cyclicy shifted versions of the first 15 frames of the noisy utterance that is being decoded (15 exemplars). This results in a total number of 15 015 exemplars in the dictionary.

3.3. Experiments and Results

The first experiment compares the three acoustic modelling methods, with the results given in Table 1. Unsurprisingly, the performance difference between results with clean training and MCT show that MCT is necessary to prepare the models for unseen testing conditions, since no other model adaptation techniques are applied in this experiment. Interestingly, the GMM can cope better with such unseen test conditions compared to the DNN and LSTM systems. When using MCT, the GMM (19.0 %) falls back behind the two neural network-based acoustic models, the DNN (14.0 %) being slightly better than the LSTM (14.9 %).

Next, we apply NMF speech enhancement on the test data, keeping the trained models unchanged (i. e. without retraining). In this way, NMF might improve the speech quality, but at the same time introduce artifacts the models may not be robust against. Table 2 lists the results for this second experiment. With clean models, enhancing the test data improves the performance of the GMM and LSTM systems in noisy conditions. The multi-condition GMM is robust against the NMF artifacts (except in condition C) and the overall performance is slightly

Table 2: WER (%) for different systems with or without multi-condition training (MCT), using original unenhanced data for training and NMF-enhanced data for testing. Lowest scores are highlighted in bold font.

System	MCT	A	B	C	D	Avg.
GMM	-	**5.5**	**22.8**	**25.2**	**43.8**	**30.8**
DNN	-	13.4	45.7	58.3	69.6	54.5
LSTM	-	15.3	44.7	52.4	65.6	52.1
GMM	✓	**8.4**	**11.6**	**20.6**	**26.3**	**18.3**
DNN	✓	9.3	15.8	29.9	36.3	25.1
LSTM	✓	10.7	16.5	23.6	34.0	24.1

Table 3: WER (%) for different systems with or without multi-condition training (MCT), using NMF-enhanced data for training and testing. Lowest scores are highlighted in bold font.

System	MCT	A	B	C	D	Avg.
GMM	-	5.5	**24.2**	**26.7**	**44.9**	**31.9**
DNN	-	**3.1**	32.4	38.4	54.1	40.1
LSTM	-	5.2	50.2	44.2	74.8	57.1
GMM	✓	8.0	11.5	18.3	26.3	18.1
DNN	✓	**3.6**	**7.3**	**11.2**	**20.7**	**13.1**
LSTM	✓	6.0	9.2	12.9	22.8	15.1

improved (18.3 %). Results for multi-condition DNN (25.1 %) and LSTM (24.1 %) acoustic models undergo a deterioration when only the test data are enhanced, indicating that these models cannot cope with the artifacts introduced by the NMF.

Finally, experiments are performed where test and training data are processed with NMF (i. e. using retraining), which enables the models to "learn" the influence of artifacts. The results of this experiment are given in Table 3. This setup leads to an improvement of the clean DNN performance, while the clean GMM undergoes a small deterioration compared to the second experiment. The more interesting results are obtained with MCT. Compared to unprocessed data (Table 1), the GMM performance under the influence of noise is improved (18.1 %). Overall, the effect of retraining is small for the multi-condition GMM (compared to Table 2), except in condition C, where the artifacts introduced by NMF are partly compensated through retraining. What is more, NMF is also capable of improving the DNN system by almost 1 % absolute (13.1 %). The LSTM performance is only improved in noisy environments and in total gets marginally worse (15.1 %).

Regarding the NMF results, the most critical finding is the impact of channel mismatch in test set C and D. The results show that especially for the GMM, but even with a DNN, when using retraining and MCT the results on set C are actually worse than those obtained without preprocessing. This means the artifacts introduced by the channel mismatch of the test data are particularly severe, since the DNN *is* able to successfully learn the impact of artifacts introduced by NMF on clean speech.

Even though the dictionary is extracted from the multicondition training data in an attempt to provide some robustness against channel mismatches, additional evaluations (not shown) revealed that this did in fact not perform any better than using a dictionary extracted from the clean speech training data. Obviously, speech in the multicondition training data is not representative for the mismatch observed in the test data - as a result, the linear additive model underlying NMF is unable to model the convolutive effect of the channel. For practical applications it will therefore be essential that the convolutive effect is modeled explicitly. Preliminary research on this is reported in [26].

Table 4: State prediction frame error rate (%) for different DNN and LSTM systems using multi-condition training, for the training (train) and development (dev) sets. Lowest scores are highlighted in bold font.

System	Layers	# weights	train	dev	avg. WER
LSTM	2x250	3.1 m	**26.3**	**37.6**	14.9
LSTM	3x250	4.6 m	26.6	38.1	-
LSTM	2x125	1.1 m	31.7	39.1	-
DNN	7x2048	30.2 m	44.1	54.7	**14.0**

Finally, we analysed the LSTM performance in more detail. For this purpose, we investigated the frame error rate (on the training and development set) for the state predictions of different LSTM and DNN networks. These results are given in Table 4. The first row shows the same LSTM that was employed for HMM decoding (see Table 1). In order to verify the choice of LSTM configuration, two more setups were tested. First, an additional layer was added, which did not improve frame error rate. The necessity of the size of the hidden layers was verified by halving this size, which resulted in a degradation in performance. For comparison, the frame error rate is also reported for the DNN system. The ability to predict HMM states of the DNN is much worse compared to the LSTM, although the DNN performs better in terms of WER. Our findings support the results presented in [9], where a similar effect was observed. In that work, it was suspected that the main reasons for this are that the frame-wise error does not take into account the language model, and that the LSTM might learn a word-level language model itself, which interferes with the language model during decoding. In fact, our experiments also showed that for LSTM decoding, much higher language model weights are necessary compared to DNN decoding.

4. Conclusions

We compared acoustic models based on GMMs, DNNs, and LSTMs for their robustness and investigated the influence of NMF speech enhancement. The results were obtained with the Aurora-4 experimental framework and showed that DNN and LSTM require MCT, but greatly outperform the GMM in this case. The LSTM performed worse than the DNN, although the frame-wise state prediction error was lower for LSTM. On the one hand, the LSTM requires only one tenth of the number of parameters of the DNN, but on the other hand, the training procedure is more complex (backpropagation through time). Increasing the number of LSTM parameters brought no improvement; network pretraining (as done for the DNN) might help in this case.

NMF enhancement with retraining improves all systems substantially when using clean training data, although the clean LSTM performs better when trained on unprocessed data. Even with multi-condition DNN acoustic models, NMF enhancement still brings improvements, and the best overall results are obtained with NMF. At the same time, NMF is shown to be quite sensitive to the channel mismatch, so future research should focus on this aspect. A possible line of research would be the techniques proposed in [26]. We can conclude that even with MCT and a DNN or similar architecture, the use of additional preprocessing for noise robustness can be effective and warrants further investigation.

5. References

[1] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 4, pp. 745–777, 2014.

[2] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 1–12, 2007.

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

[4] H. A. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach*. Kluwer Academic Publishers, 1994.

[5] M. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. ICASSP*, Vancouver, Canada, 2013, pp. 7398–7402.

[6] C. Weng, D. Yu, S. Watanabe, and B.-H. Juang, "Recurrent deep neural networks for robust speech recognition," in *Proc. ICASSP*, Florence, Italy, 2014, to appear.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] M. Wöllmer, F. Eyben, B. Schuller, and G. Rigoll, "A multi-stream ASR framework for BLSTM modeling of conversational speech," in *Proc. ICASSP*, Prague, Czech Republic, 2011, pp. 4860–4863.

[9] A. Graves, N. Jaitly, and A.-R. Mohamed, "Speech recognition with deep recurrent neural networks," in *Proc. ASRU*. Olomouc, Czech Republic: IEEE, 2013, pp. 273–278.

[10] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," *arXiv:1402.1128 [cs.NE]*, 2014.

[11] C. Plahl, M. Kozielski, R. Schlüter, and H. Ney, "Feature combination and stacking of recurrent and non-recurrent neural networks for LVCSR," in *Proc. ICASSP*, Vancouver, Canada, 2013, pp. 6714–6718.

[12] J. T. Geiger, F. Weninger, J. F. Gemmeke, M. Wöllmer, B. Schuller, and G. Rigoll, "Memory-enhanced neural networks and NMF for robust ASR," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 6, pp. 1037–1046, 2014.

[13] F. Weninger, J. Geiger, M. Wöllmer, B. Schuller, and G. Rigoll, "Feature enhancement by deep LSTM networks for ASR in reverberant multisource environments," *Computer Speech and Language*, vol. 28, no. 4, pp. 888–902, 2014.

[14] ——, "The Munich 2011 CHiME challenge contribution: NMF-BLSTM speech enhancement and recognition for reverberated multisource environments," in *Proc. Int. Workshop on Machine Listening in Multisource Environments (CHiME)*, Florence, Italy, 2011, pp. 24–29.

[15] J. T. Geiger, F. Weninger, A. Hurmalainen, J. Gemmeke, M. Wöllmer, B. Schuller, G. Rigoll, and T. Virtanen, "The TUM+TUT+KUL approach to the 2nd CHiME challenge: Multi-stream ASR exploiting BLSTM networks and sparse NMF," in *Proc. Int. Workshop on Machine Listening in Multisource Environments (CHiME)*, Vancouver, Canada, 2013, pp. 25–30.

[16] M. Delcroix, Y. Kubo, T. Nakatani, and A. Nakamura, "Is speech enhancement pre-processing still relevant when using deep neural networks for acoustic modeling?" in *Proc. INTERSPEECH*, Lyon, France, 2013, pp. 2992–2996.

[17] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlícek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *Proc. ASRU*, Honolulu, HI, USA, 2011.

[18] R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," in *Proc. ICASSP*, San Francisco, CA, USA, 1992, pp. 13–16.

[19] G. Saon, M. Padmanabhan, R. Gopinath, and S. Chen, "Maximum likelihood discriminant feature spaces," in *Proc. ICASSP*, Istanbul, Turkey, 2000, pp. 1129–1132.

[20] G. Hinton, "A practical guide to training restricted Boltzmann machines," University of Toronto, Torondo, Canada, Tech. Rep., 2010.

[21] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[22] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[23] J. Gemmeke, T. Virtanen, and A. Hurmalainen, "Exemplar-based sparse representations for noise robust automatic speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2067–2080, 2011.

[24] H. Kallasjoki, U. Remes, J. F. Gemmeke, T. Virtanen, and K. J. Palomäki, "Uncertainty measures for improving exemplar-based source separation," in *Proc. INTERSPEECH*, Florence, Italy, 2011, pp. 469–472.

[25] T. Sainath, B. Ramabhadran, D. Nahamoo, D. Kanevsky, D. Van Compernolle, K. Demuynck, J. Gemmeke, J. Bellegarda, and S. Sundaram, "Exemplar-based processing for speech recognition: An overview," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 98–113, 2012.

[26] J. F. Gemmeke, T. Virtanen, and K. Demuynck, "Exemplar-based joint channel and noise compensation," in *Proc. ICASSP*, Vancouver, Canada, 2013, pp. 868–872.