

# FEATURE ENHANCEMENT BY BIDIRECTIONAL LSTM NETWORKS FOR CONVERSATIONAL SPEECH RECOGNITION IN HIGHLY NON-STATIONARY NOISE

Martin Wöllmer<sup>1</sup>, Zixing Zhang<sup>2</sup>, Felix Weninger<sup>2</sup>, Björn Schuller<sup>2</sup>, Gerhard Rigoll<sup>2</sup>

<sup>1</sup>BMW Group, Munich, Germany

<sup>2</sup>Institute for Human-Machine Communication, Technische Universität München, Germany

`martin.woellmer@bmw.de`

## ABSTRACT

The recognition of spontaneous speech in highly variable noise is known to be a challenge, especially at low signal-to-noise ratios (SNR). In this paper, we investigate the effect of applying bidirectional Long Short-Term Memory (BLSTM) recurrent neural networks for speech feature enhancement in noisy conditions. BLSTM networks tend to prevail over conventional neural network architectures, whenever the recognition or regression task relies on an intelligent exploitation of temporal context information. We show that BLSTM networks are well-suited for mapping from noisy to clean speech features and that the obtained recognition performance gain is partly complementary to improvements via additional techniques such as speech enhancement by non-negative matrix factorization and probabilistic feature generation by Bottleneck-BLSTM networks. Compared to simple multi-condition training or feature enhancement via standard recurrent neural networks, our BLSTM-based feature enhancement approach leads to remarkable gains in word accuracy in a highly challenging task of recognizing spontaneous speech at SNR levels between -6 and 9 dB.

**Index Terms**— feature enhancement, Long Short-Term Memory, recurrent neural networks, non-negative matrix factorization

## 1. INTRODUCTION

Spontaneous, conversational speaking styles and interfering noise sources corrupting the speech signal can be seen as two major challenges that have to be faced by modern systems for automatic speech recognition (ASR). Aiming to increase the robustness of ASR systems, various techniques affecting different stages and system components within a speech recognizer have been proposed in recent years. Such techniques comprise for example speech enhancement, feature enhancement and normalization, probabilistic feature extraction via neural networks, model adaptation, or multi-condition training, i. e., including noisy speech data in the training corpus [1].

Most studies on noise robust ASR focus on simplistic recognition tasks such as digit recognition [2] or tasks that follow a fixed grammar [3]. In this paper, we evaluate various noise compensation strategies under extremely challenging conditions by considering the Buckeye corpus of spontaneous speech [4] superposed with highly non-stationary noise obtained from the 2011 PASCAL CHiME Challenge data [3] at low signal-to-noise ratios (SNR). Building on our previous work [5] in which we show word accuracy improvements obtained by combination of semi-supervised sparse non-negative matrix

factorization (NMF) [6, 7], probabilistic feature extraction via bidirectional Long Short-Term Memory (BLSTM) neural networks [8], and multi-condition training, this study investigates whether additional performance improvements can be obtained by applying BLSTM networks that are trained to map from noisy speech features to clean features.

Motivated by experiments which have shown that recurrent neural network (RNN) architectures can be applied for feature enhancement in noisy conditions [9], we show that more effective feature enhancement can be achieved if RNNs are replaced by BLSTM networks which are known to outperform conventional RNNs for tasks in which long-range temporal context has to be considered. The BLSTM architecture [10, 11] allows a more accurate mapping from noisy observations to enhanced speech features as it is able to model the temporal evolution of speech and noise over a longer period of time. Furthermore, instead of exploiting this context-sensitivity for mapping directly from speech features to phoneme estimates (as in earlier work [8]), we now propose to apply separate BLSTM networks for estimating clean features and phoneme likelihoods, and show that this leads to superior ASR accuracy. Finally, we demonstrate that our ASR architecture can exploit both BLSTM feature enhancement and NMF speech enhancement in a complementary fashion.

The overall architecture of our ASR system is depicted in Figure 1: We use NMF to enhance the noisy speech signal (Section 2) before we extract features that are enhanced by a BLSTM network (Section 3). Finally, a Bottleneck-BLSTM network (see Section 4) is employed to generate a tandem feature vector that is processed by a Hidden Markov Model (HMM) system.

## 2. SPEECH ENHANCEMENT BY NMF

NMF-based techniques for monaural speech enhancement, such as the ones used in this study, are generally based on the assumption that the wanted speech signal is corrupted by addition of interfering noise in the magnitude spectral domain. Furthermore, it is assumed that both the speech spectrogram and the noise spectrogram can in turn be approximated as the product of non-negative speech and noise dictionaries with non-negative coefficients (activations). The number of atoms in the speech and noise dictionaries will be denoted by  $R^{(s)}$  and  $R^{(n)}$ , respectively. In our semi-supervised NMF approach, we estimate a fixed speech dictionary from training data as detailed in Section 6. In contrast, the noise dictionary is estimated for each utterance along with the activations of the speech and noise atoms. To this end, the Kullback-Leibler divergence of the observed spectrogram given the product of dictionary atoms and activations is minimized, and an additive sparsity constraint corresponding to the L1 norm of the activations is added. For this minimization, the standard multi-

This research has been supported by the German Research Foundation (DFG) through grant nos. SCHU 2508/4 and /2.

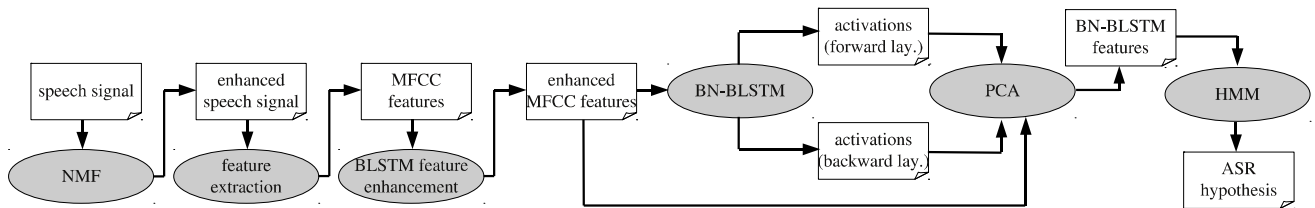


Fig. 1: ASR system architecture.

plicative update NMF algorithm is applied, with a straightforward extension to include the sparsity constraint (cf. [12, 13]). Similar semi-supervised NMF approaches have been proven to be highly efficient for speech enhancement, e. g., in [7, 14]. Informally, the purpose of sparsity is to force that only a few basis vectors can be active at a given time, which is a reasonable assumption if the basis vectors correspond to, e. g., phonemes, or spectra originating from different noise sources. The update rules are applied for a fixed number of iterations. For speech enhancement, we use a simple soft-masking approach where each time-frequency bin of the observed spectrogram is weighted by the contributions of speech and noise according to the NMF decomposition, as in our previous work [5]. All experiments for this paper are based on the NMF implementations found in our open-source toolkit openBliSSART [15] to enforce reproducibility of our results.

### 3. FEATURE ENHANCEMENT USING BLSTM NETWORKS

The basic architecture of Long Short-Term Memory (LSTM) networks was introduced in [10]. LSTM networks can be seen as an extension of conventional recurrent neural networks that enables the modeling of long-range temporal context for improved sequence labeling. They are able to store information in linear memory cells over a longer period of time and can learn the optimal amount of contextual information relevant for the regression or classification task. An LSTM hidden layer is composed of multiple recurrently connected subnets (so-called *memory blocks*). Every memory block consists of self-connected *memory cells* and three multiplicative *gate* units (input, output, and forget gates). Since these gates allow for write, read, and reset operations within a memory block, an LSTM block can be interpreted as (differentiable) memory chip in a digital computer. Further details on the LSTM principle can be found in [11]. Note that the initial version of the LSTM architecture contained only input and output gates. Forget gates were added later [16] in order to allow the memory cells to reset themselves whenever the network needs to *forget* past inputs. In our experiments we exclusively consider the enhanced LSTM version including forget gates.

Standard RNNs have access to past but not to future context. To exploit both, past and future context, RNNs can be extended to *bidirectional* RNNs (BRNN), where two separate recurrent hidden layers scan the input sequences in opposite directions [17]. The two hidden layers are connected to the same output layer, which therefore has access to context information in both directions. Bidirectional modeling can also be applied within an LSTM framework, which results in bidirectional LSTM.

Exploiting the context-sensitivity of the BLSTM technique, we aim to build a network that learns how clean speech features can be generated from noisy features. Hence, our feature enhancement (FE)

network has one input node for each noise corrupted input feature vector component and one output node for each regression target representing the clean feature vector. This means that we require a clean and a noisy version of the training and development set (see Section 5). In our experiments, we use 39 MFCC features (including deltas and double deltas) which are extracted from the NMF-enhanced speech signal every 10 ms using a window size of 25 ms. Prior to network training, we compute the global means and variances of the clean and the noisy training set feature vectors and perform mean and variance normalization of the network inputs and the network outputs using the means and variances from the noisy training set and the clean set, respectively. The applied network has three hidden layers consisting of 78, 128, and 78 memory blocks. Each memory block contains one memory cell. During training we use a learning rate of  $10^{-5}$  and a momentum of 0.9. Zero mean Gaussian noise with standard deviation 0.1 is added to the input activations in the training phase in order to improve generalization. Prior to training, all weights are randomly initialized in the range from -0.1 to 0.1. Input and output gates use tanh activation functions, while the forget gates have logistic activation functions. In the training phase, we evaluate the overall root mean square error on the development set after every fifth epoch. Training is aborted as soon as no improvement on the development set can be observed during the last 25 epochs, and the network that achieved the best root mean square error on the development set is chosen as the final network.

### 4. BOTTLENECK-BLSTM FRONT-END

In addition to the feature enhancement BLSTM network, we employ a secondary BLSTM network, trained to map from (enhanced) speech features to phonemes. As shown in [8], combining BLSTM-based probabilistic feature generation with the ‘bottleneck’ (BN) idea proposed in [18] leads to lower error rates in spontaneous speech recognition. The bottleneck principle allows to generate tandem feature vectors of arbitrary size by using the activations of a narrow hidden (bottleneck) layer as features – rather than the logarithmized output activations corresponding to the estimated phoneme or phoneme state posteriors.

Since we focus on *bidirectional* processing, we have two bottleneck layers: one within the network processing the speech sequence in forward direction and one within the network for backward processing (see Figure 1). The features enhanced according to the procedure described in Section 3 serve as input for a BN-BLSTM network that is trained on framewise phoneme targets. During BN-BLSTM feature extraction, the activations of the output layer are ignored; only the activations of the forward and backward bottleneck layer are processed (i. e., the memory block outputs of the bottleneck layers). Together with the enhanced MFCC features, the forward and backward bottleneck layer activations are concatenated to one large feature vector

which is then decorrelated and dimensionality reduced by Principal Component Analysis (PCA) as shown in Figure 1.

## 5. DATABASE

The evaluation database applied in this study is identical to the speech corpus employed in [5]: We use the Buckeye corpus [4] recorded in clean conditions, mixed with the CHiME noise corpus [3] to simulate spontaneous speech encountered in a noisy domestic environment. The Buckeye corpus contains recordings of interviews with 40 speakers. The speech is highly spontaneous and contains a variety of non-linguistic vocalizations. The segmentation into utterances and the speaker-independent subdivision into training, development, and test set (stratified by speaker age and gender) exactly corresponds to the ASR experiments reported in [19].

The additive noise considered in this study is taken from the corpus of the 2011 PASCAL CHiME Challenge [3]. This corpus contains genuine recordings from a domestic environment obtained over a period of several weeks. Most of the noise is highly non-stationary due to abrupt changes such as appliances being turned on/off, impact noises such as banging doors, and interfering speakers; more details can be found in [3]. To create the noisy version of our evaluation database, we followed the protocol which was used to create the CHiME Challenge ASR task [3]: In the development and test set, we employ six signal-to-noise ratios (SNRs) ranging from 9 dB down to -6 dB in steps of 3 dB. After normalizing the speech signals to -6 dB maximum amplitude to avoid clipping after mixing with noise, we chose for each speech signal six noise segments from the CHiME development/test noise matching the different SNRs. As proposed in [3], the noisy utterances are not constructed by artificial scaling of the speech or noise amplitudes, but by choosing noise segments as they were recorded in a real life situation. This means that noisy utterances at low SNRs occur in noise that naturally has high energy, such as broad band impact noise. The SNRs were measured on first order differences of speech and noise signals.

In addition, we created a multi-condition training set by mixing clean training speech with random segments of the six hours of training noise (disjoint from development and test noise) provided with the CHiME Challenge corpus. For this multi-condition training set, we added random segments of noise to the normalized speech utterances; this provides a good coverage of SNRs while not assuming any knowledge about the exact SNRs occurring in the test conditions.

## 6. EXPERIMENTS AND RESULTS

BLSTM-based feature enhancement was performed as described in Section 3. For tandem feature generation (see Section 4), we trained a BN-BLSTM network consisting of three hidden layers (per input direction) on framewise phoneme targets obtained via HMM based forced alignment of the clean Buckeye training set. All network and training parameters, including the size of the hidden layers, learning rate, etc. were set exactly as in [8]. Only the first 39 principal components of the PCA-transformed BN-BLSTM feature vector were used as final features for tandem ASR. In the HMM system applied for processing the BN-BLSTM features, each phoneme is represented by three emitting states (left-to-right HMMs) with 16 Gaussian mixtures. Tied-state cross-word triphone models with shared state transition probabilities were applied. Both, acoustic models and a back-off bigram language model were trained on the training set of the Buckeye corpus.

In order to apply NMF on the development and test set, spectrograms of the signals were calculated by short-time Fourier Trans-

form using Hann windows of 25 ms length at 10 ms frame shift, in conformance with [5]. To build a phoneme-dependent yet speaker-independent speech model for NMF, for each phoneme, the corresponding spectrograms were extracted from the Buckeye training set according to a forced alignment with the recognizer described in [8]. These concatenated phoneme spectrograms were reduced to a single dictionary atom by a 1-component NMF. The column-wise concatenation of these atoms constitutes the speech dictionary. Thus, the number of speech atoms  $R^{(s)}$  in semi-supervised NMF was equivalent to the number of phonemes (39). The number of noise atoms  $R^{(n)}$ , the sparsity constant  $\lambda$ , and the number of NMF iterations  $K$  were optimized as in [5].

In Table 1, the word accuracies (WA) on the Buckeye test set are shown for different SNR levels as well as for clean speech. The upper half of the table contains the results obtained with a recognizer processing (standard or enhanced) MFCC features, while the lower half shows the results achieved applying the Bottleneck-BLSTM front-end explained in Section 4. For both experimental setups, we examine the effect of speech enhancement via NMF (Section 2), multi-condition training (MCT), and feature enhancement via BLSTM networks (BLSTM-FE). Applying a standard MFCC-based recognizer without any enhancement techniques (first line in Table 1), we observe a drastic decrease of recognition performance from 50.97 % WA for clean speech to 21.21 % for noise corrupted speech at -6 dB. When including noisy speech in the training set, word accuracies can be improved for all SNR levels: On average, multi-condition training increases the WA by 3.8 % absolute. Yet, for clean speech, a lower WA of 43.84 % has to be tolerated if MCT is employed. The third line of Table 1 reveals that it seems to be much more effective to apply BLSTM feature enhancement than to simply include noisy speech in the training corpus. Compared to a system only using MCT, a recognizer including a BLSTM network, that is trained to map from noisy to clean features, can increase the average WA from 30.92 % to 37.80 % at SNR levels between -6 and 9 dB. Interestingly, also the recognition of clean speech is slightly improved, when applying BLSTM-FE. Note that for a fair evaluation of the effect of BLSTM-FE, we have to compare the obtained results with the corresponding MCT results, as in both cases (MCT and BLSTM-FE) the training noise is ‘seen’ during training, while the baseline recognizer (line 1 in Table 1) is exclusively built from clean training material. Additionally applying NMF for speech enhancement prior to feature extraction consistently leads to a further WA improvement of around 1 to 2 % absolute in noisy conditions.

A notable performance gain can also be observed when employing the BN-BLSTM front-end: For the best system configuration (NMF, BLSTM-FE, BN-BLSTM, see last line of Table 1), we get word accuracies between 38.24 and 52.94 % while for a comparable recognizer without BN-BLSTM feature generation, word accuracies of between 32.98 and 44.28 % are obtained. An interesting observation is that improvements via NMF, BLSTM-based feature enhancement, and BLSTM-based probabilistic bottleneck feature extraction seem to be partly complementary. Depending on whether NMF speech enhancement or the BN-BLSTM front-end is applied or not, we achieve average WA improvements of between 6.9 and 1.9 % absolute (compared to MCT) if we additionally include a BLSTM network for feature enhancement into the system architecture. Furthermore, we can conclude that it is more effective to train two separate BLSTM networks, i. e., one mapping from noisy to enhanced features and one mapping from enhanced features to phonemes, than to train just one network mapping from noisy features to phonemes (see last two rows of Table 1). Comparing speech enhancement by NMF and feature enhancement by BLSTM on their own, we note

**Table 1:** Word accuracies [%] on Buckeye test set at SNRs from -6 to 9 dB, on average across these SNRs, and for clean speech. NMF: speech enhancement via non-negative matrix factorization; MCT: multi-condition training; BLSTM-FE: feature enhancement via BLSTM networks.

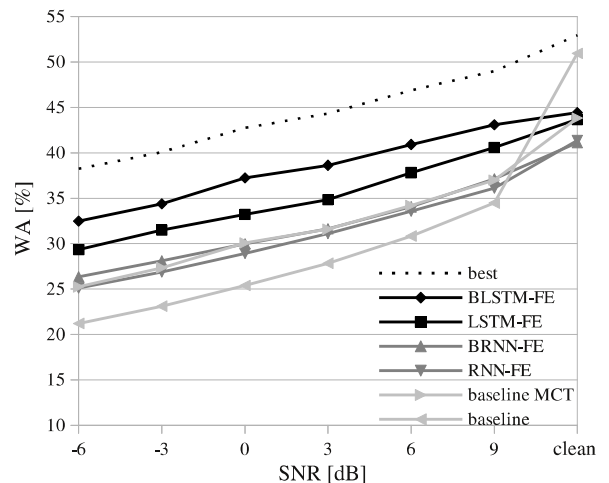
Front-end	NMF	MCT	BLSTM-FE	SNR							clean
				-6 dB	-3 dB	0 dB	3 dB	6 dB	9 dB	avg	
MFCC	✗	✗	✗	21.21	23.11	25.40	27.85	30.85	34.48	27.15	50.97
MFCC	✗	✓	✗	25.25	27.36	30.09	31.59	34.20	37.00	30.92	43.84
MFCC	✗	✓	✓	32.50	34.39	37.24	38.62	40.93	43.10	37.80	44.42
MFCC	✓	✗	✗	23.06	25.32	27.17	29.65	32.56	36.48	29.04	50.54
MFCC	✓	✓	✗	26.51	28.82	30.85	32.85	35.13	37.95	32.02	43.83
MFCC	✓	✓	✓	32.98	35.02	37.42	38.76	41.30	43.32	38.13	44.28
BN-BLSTM	✗	✗	✗	22.73	25.08	28.13	30.51	35.16	39.04	30.11	58.21 [8]
BN-BLSTM	✗	✓	✗	34.93	37.58	40.04	41.71	44.60	46.87	40.96	51.12
BN-BLSTM	✗	✓	✓	37.45	39.46	42.36	43.87	46.43	48.53	43.02	53.14
BN-BLSTM	✓	✗	✗	24.47	26.79	29.75	32.18	36.53	40.74	31.74	57.94
BN-BLSTM	✓	✓	✗	35.74	38.45	40.49	42.45	45.27	47.29	41.62	50.91 [5]
BN-BLSTM	✓	✓	✓	38.24	40.09	42.75	44.33	46.88	49.00	43.55	52.94

that BLSTM feature enhancement seems to be vastly superior in case of the MFCC front-end (37.80 % vs. 32.92 % average accuracy on noisy data), while the difference is smaller for the BLSTM front-end (43.02 % vs. 41.62 %). We believe that this is due to the non-linear and context-sensitive modeling in the BLSTM-MFCC framework, which is not captured by the simple frame-wise spectral NMF model. Still, adding NMF enhancement to the best BLSTM system (43.02 %) yields a slight gain of 0.53 % absolute WA.

To investigate the effect of applying the BLSTM technique rather than unidirectional LSTM and standard (B)RNNs for feature enhancement, we repeated the baseline recognition experiments (no NMF, standard MFCC front-end) using different neural network architectures for feature enhancement. Figure 2 compares feature enhancement with RNNs, BRNNs, LSTM, and BLSTM networks. As a reference, also the results corresponding to the best system configuration (NMF, BN-BLSTM, see last line of Table 1) is indicated as a dotted line. Further, the performance of the baseline system without feature enhancement is shown with and without MCT. We see that feature enhancement with RNN or BRNN is comparable to simply using MCT. A notable performance gain is reached via exploitation of long-range contextual information within the network for feature enhancement: Employing Long Short-Term Memory significantly increases the word accuracy at all SNR levels. If LSTM context is considered along both input directions, we see a further performance improvement.

## 7. CONCLUSION AND OUTLOOK

In this paper we have shown how BLSTM networks can be applied for noisy speech feature enhancement if they are trained to map from noisy to clean features. Compared to standard RNNs, the Long Short-Term Memory architecture allows for a more efficient exploitation of temporal context which leads to improved feature enhancement. We integrated BLSTM-based feature enhancement into an ASR system featuring speech enhancement via NMF and a Bottleneck-BLSTM front-end as introduced in [8]. After evaluating several variants of our system, we found that the proposed feature enhancement technique leads to increased word accuracies in all cases and is much more effective than simple multi-condition training. Furthermore, the combination of our technique with the BN-BLSTM feature extractor shows that using two separate networks for subsequently estimating clean features and phonemes, respectively, leads to better results than applying a network that maps directly from noisy speech features



**Fig. 2:** Word accuracies [%] on Buckeye test set at SNRs from -6 to 9 dB and for clean speech using different network architectures for feature enhancement (BLSTM, LSTM, BRNN, and RNN).

to phonemes as done in [5]. Future work will concentrate on the evaluation of alternative input features for BLSTM-based feature enhancement, such as spectral or PLP features, and enhanced speech and context modeling for NMF (cf., e. g., [6, 20]).

## 8. RELATION TO PRIOR WORK

LSTM has been applied for various pattern recognition tasks, including phoneme classification [11], handwriting recognition [21], emotion recognition [22], and driver distraction detection [23]. In the field of ASR, (bidirectional) LSTM was shown to improve both keyword spotting [24] and continuous speech recognition [8]. Since the exploitation of temporal context within an RNN architecture is known to lead to improved results for feature enhancement [9], this study combines the ideas of RNN-based speech feature enhancement and context-sensitive sequence labeling by BLSTM. Experimental settings as well as the baseline recognition systems are adopted from our previous work reported in [5].

## 9. REFERENCES

- [1] B. Schuller, M. Wöllmer, T. Moosmayr, and G. Rigoll, "Recognition of noisy speech: A comparative survey of robust model architecture and feature enhancement," *Journal on Audio, Speech, and Music Processing*, 2009, ID 942617.
- [2] B. Mesot and D. Barber, "Switching linear dynamic systems for noise robust speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 6, pp. 1850–1858, 2007.
- [3] H. Christensen, J. Barker, N. Ma, and P. Green, "The CHiME corpus: a resource and a challenge for Computational Hearing in Multisource Environments," in *Proc. of Interspeech*, Makuhari, Japan, 2010, pp. 1918–1921.
- [4] M. A. Pitt, L. Dilley, K. Johnson, S. Kiesling, W. Raymond, E. Hume, and E. Fosler-Lussier, *Buckeye Corpus of Conversational Speech (2nd release)*, Department of Psychology, Ohio State University (Distributor), Columbus, OH, USA, 2007, [www.buckeyecorpus.osu.edu].
- [5] F. Weninger, M. Wöllmer, and B. Schuller, "Combining Bottleneck-BLSTM and Semi-Supervised Sparse NMF for Recognition of Conversational Speech in Highly Instationary Noise," in *Proc. of Interspeech*, Portland, Oregon, USA, 2012.
- [6] G. J. Mysore and P. Smaragdis, "A Non-Negative Approach to Semi-Supervised Separation of Speech from Noise with the Use of Temporal Dynamics," in *Proc. of ICASSP*, Prague, Czech Republic, 2011, pp. 17–20.
- [7] F. Weninger, J. Feliu, and B. Schuller, "Supervised and Semi-Supervised Suppression of Background Music in Monaural Speech Recordings," in *Proc. of ICASSP*, Kyoto, Japan, 2012, pp. 61–64.
- [8] M. Wöllmer, B. Schuller, and G. Rigoll, "A novel Bottleneck-BLSTM front-end for feature-level context modeling in conversational speech recognition," in *Proc. of ASRU*, Waikoloa, Big Island, Hawaii, 2011, pp. 36–41.
- [9] S. Parveen and P. Green, "Speech enhancement with missing data techniques using recurrent neural networks," in *Proc. of ICASSP*, Montreal, Canada, 2004, pp. 733–736.
- [10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [11] 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.
- [12] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. of NIPS*, Vancouver, Canada, 2001, pp. 556–562.
- [13] 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.
- [14] A. Ozerov, C. Févotte, and M. Charbit, "Factorial scaled hidden Markov model for polyphonic audio representation and source separation," in *Proc. of WASPAA*, Mohonk, NY, United States, 2009, pp. 121–124.
- [15] F. Weninger, A. Lehmann, and B. Schuller, "openBLISSART: Design and Evaluation of a Research Toolkit for Blind Source Separation in Audio Recognition Tasks," in *Proc. of ICASSP*, Prague, Czech Republic, 2011, pp. 1625–1628.
- [16] F. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.
- [17] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, pp. 2673–2681, 1997.
- [18] F. Grezl, M. Karafiat, K. Stanislav, and J. Cernocky, "Probabilistic and bottle-neck features for LVCSR of meetings," in *Proc. of ICASSP*, Honolulu, Hawaii, 2007, pp. 757–760.
- [19] F. Weninger, B. Schuller, M. Wöllmer, and G. Rigoll, "Localization of non-linguistic events in spontaneous speech by non-negative matrix factorization and Long Short-Term Memory," in *Proc. of ICASSP*, Prague, Czech Republic, 2011, pp. 5840–5843.
- [20] J. Gemmeke, T. Virtanen, and A. Hurmalainen, "Exemplar-Based Speech Enhancement and its Application to Noise-Robust Automatic Speech Recognition," in *Proc. of CHiME Workshop*, Florence, Italy, 2011, pp. 53–57.
- [21] M. Liwicki, A. Graves, S. Fernandez, H. Bunke, and J. Schmidhuber, "A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks," in *Proc. of ICDAR*, Curitiba, Brazil, 2007, pp. 367–371.
- [22] M. Wöllmer, B. Schuller, F. Eyben, and G. Rigoll, "Combining long short-term memory and dynamic bayesian networks for incremental emotion-sensitive artificial listening," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 5, pp. 867–881, 2010.
- [23] M. Wöllmer, C. Blaschke, T. Schindl, B. Schuller, B. Färber, S. Mayer, and B. Trefflich, "On-line driver distraction detection using long short-term memory," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 2, pp. 574–582, 2011.
- [24] M. Wöllmer, B. Schuller, and G. Rigoll, "Keyword spotting exploiting Long Short-Term Memory," *Speech Communication*, 2012.