DEEP LEARNING FOR MONAURAL SPEECH SEPARATION

Po-Sen Huang[†], *Minje Kim*[‡], *Mark Hasegawa-Johnson*[†], *Paris Smaragdis*[†‡§]

[†]Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, USA
[‡]Department of Computer Science, University of Illinois at Urbana-Champaign, USA
[§]Adobe Research, USA

{huang146, minje, jhasegaw, paris}@illinois.edu

ABSTRACT

Monaural source separation is useful for many real-world applications though it is a challenging problem. In this paper, we study deep learning for monaural speech separation. We propose the joint optimization of the deep learning models (deep neural networks and recurrent neural networks) with an extra masking layer, which enforces a reconstruction constraint. Moreover, we explore a discriminative training criterion for the neural networks to further enhance the separation performance. We evaluate our approaches using the TIMIT speech corpus for a monaural speech separation task. Our proposed models achieve about 3.8∼4.9 dB SIR gain compared to NMF models, while maintaining better SDRs and SARs.

Index Terms— Monaural Source Separation, Time-Frequency Masking, Deep Learning

1. INTRODUCTION

Source separation of audio signals is important for several real-world applications. For example, separating noise from speech signals enhances the accuracy of automatic speech recognition (ASR) [1, 2]. Separating singing voices from music enhances the accuracy of chord recognition [3]. Current separation results are, however, still far behind human capability. Monaural source separation is even more difficult since only one single channel signal is available.

Recently, several approaches have been proposed to address the monaural source separation problem [4, 5, 6, 7]. The widely used non-negative matrix factorization (NMF) [4] and probabilistic latent semantic indexing (PLSI) [5, 6] factorize time-frequency spectral representations by learning the non-negative reconstruction bases and weights.

NMF and PLSI models are linear models with nonnegative constraints. Each can be viewed as one linear neural network with non-negative weights and coefficients. Moreover, NMF and PLSI usually operate directly in the spectral domain. In this paper, in order to enhance the model expressibility, we study source separation based on nonlinear models, specifically, deep neural networks (DNNs) and recurrent neural networks (RNNs) [8, 9, 10]. Instead of using a spectral representation for separation directly, the networks can be viewed as learning optimal hidden representations through several layers of nonlinearity, and the output layer reconstructs the spectral domain signals based on the learnt hidden representations.

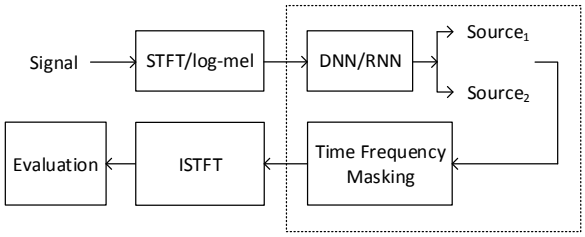

Fig. 1: Proposed framework

In this paper, we explore the use of a DNN and the use of an RNN for monaural speech separation in a supervised setting. We propose the joint optimization of the network with a soft masking function. Moreover, a discriminative training objective is also explored. The proposed framework is shown in Figure 1.

The organization of this paper is as follows: Section 2 discusses the relation to previous work. Section 3 introduces the proposed methods, including the joint optimization of deep learning models and a soft time-frequency masking function, and a discriminative training objective. Section 4 presents the experimental setting and results using the TIMIT speech corpus. We conclude the paper in Section 5.

2. RELATION TO PREVIOUS WORK

Deep learning approaches have yielded many state of the art results by representing different levels of abstraction with multiple nonlinear layers [8, 11, 12]. Recently, deep learning techniques have been applied to related tasks such as speech enhancement and ideal binary mask estimation [2, 13, 14].

A 2-stage framework for predicting an ideal binary mask using deep neural networks was proposed by Narayanan and

This research was supported by U.S. ARL and ARO under grant number W911NF-09-1-0383.

Wang [13] and by Wang and Wang [14]. The authors first try K neural networks to predict each feature dimension separately, where K is the feature dimension, and then train another classifier (one layer perceptron [13] or an SVM [14]) using neighboring time-frequency predictions in the first stage as the input. The approach of training one DNN per output dimension is not scalable when the output dimension is high. For example, if we want to use spectra as targets, we would have 513 dimensions for a 1024-point FFT. Training such large neural networks is often impractical. In addition, there are many redundancies between the neural networks in neighboring frequencies. In our approach, we propose a general framework that can jointly train all feature dimensions at the same time using one neural network, and we also propose a method to jointly train the masking function with the network directly.

Maas et al. [2] proposed using an RNN for speech noise reduction in robust automatic speech recognition. Given the noisy signal $\mathbf{x}$, the authors apply an RNN to learn clean speech $\mathbf{y}$. In the source separation scenario, we found that directly modeling one target source in the denoising framework is suboptimal compared to the framework that models all sources. In addition, we can use the information and constraints from different prediction outputs to further perform masking and discriminative training.

3. PROPOSED METHODS

3.1. Architecture

We explore using a deep neural network and a recurrent neural network for learning the optimal hidden representations to reconstruct the target spectra. Figure 2 presents an example of the proposed framework using an RNN. At time t, the training input, $\mathbf{x}_t$, of the network is the concatenation of features (spectral or log-mel filterbank features) from a mixture within a window. The output predictions, $\hat{\mathbf{y}}_{\mathbf{1}_t}$ and $\hat{\mathbf{y}}_{\mathbf{2}_t}$, of the network are the spectra of different sources. In an RNN, the l^{th} hidden layer, $l > 1$, is calculated based on the current input $\mathbf{x}_t$ and the hidden activation from the previous time step $h^{(l)}(\mathbf{x}_{t-1})$,

$$h^l(\mathbf{x}_t) = f\left(\mathbf{W}^l h^{l-1}(\mathbf{x}_t) + \mathbf{b}^l + \mathbf{U}^l h^l(\mathbf{x}_{t-1})\right) \quad (1)$$

where $\mathbf{W}^l$ and $\mathbf{U}^l$ are weight matrices, and $\mathbf{b}^l$ is the bias vector. For a DNN, the temporal weight matrix $\mathbf{U}^l$ is zero. The first hidden layer is computed as $h^1(\mathbf{x}_t) = f(\mathbf{W}^1\mathbf{x}_t + \mathbf{b}^1)$. The function $f()$ is a nonlinear function, and we explore using the rectified linear unit $f(\mathbf{x}) = max(0, \mathbf{x})$ [15] in this work. The output layer is a linear layer and is computed as:

$$\hat{\mathbf{y}}_t = \mathbf{W}^l h^{l-1}(\mathbf{x}_t) + \mathbf{c} \quad (2)$$

where $\mathbf{c}$ is a bias vector and $\hat{\mathbf{y}}_t$ is the concatenation of two predicted sources $\hat{\mathbf{y}}_{\mathbf{1}_t}$ and $\hat{\mathbf{y}}_{\mathbf{2}_t}$.

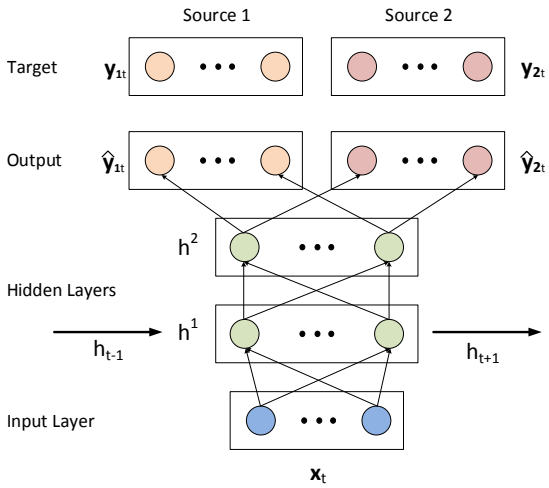

Fig. 2: An example of the proposed architecture using a recurrent nerual network

3.2. Time-Frequency Masking

Directly training the previously mentioned networks does not have the constraint that the sum of the prediction results is equal to the original mixture. One possible way to enforce the constraint is by time-frequency masking of the original mixture. To enforce the constraint, two commonly used masking functions are explored in this paper: binary (hard) and soft time-frequency masking methods.

Given a mixture $\mathbf{x}_t$, we obtain the output predictions $\hat{\mathbf{y}}_{\mathbf{1}_t}$ and $\hat{\mathbf{y}}_{\mathbf{2}_t}$ through the network. The binary time-frequency mask $\mathbf{M_b}$ is defined as follows:

$$\mathbf{M_b}(f) = \begin{cases} 1 & |\hat{\mathbf{y}}_{\mathbf{1}_t}(f)| > |\hat{\mathbf{y}}_{\mathbf{2}_t}(f)| \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where $f = 1 \ldots F$, represent different frequencies.
We can also define the soft time-frequency mask $\mathbf{M_s}$ as follows:

$$\mathbf{M_s}(f) = \frac{|\hat{\mathbf{y}}_{\mathbf{1}_t}(f)|}{|\hat{\mathbf{y}}_{\mathbf{1}_t}(f)| + |\hat{\mathbf{y}}_{\mathbf{2}_t}(f)|} \quad (4)$$

where $f = 1 \ldots F$, represent different frequencies.

Once a time-frequency mask $\mathbf{M}$ ($\mathbf{M_b}$ or $\mathbf{M_s}$) is computed, it is applied to the spectra $\mathbf{X}_t$ of the mixture $\mathbf{x}_t$ to obtain the estimated separation spectra $\hat{\mathbf{s}}_{\mathbf{1}_t}$ and $\hat{\mathbf{s}}_{\mathbf{2}_t}$, which correspond to sources 1 and 2, as follows:

$$\begin{aligned} \hat{\mathbf{s}}_{\mathbf{1}_t}(f) &= \mathbf{M}(f)\mathbf{X}_t(f) \\ \hat{\mathbf{s}}_{\mathbf{2}_t}(f) &= (1 - \mathbf{M}(f))\,\mathbf{X}_t(f) \end{aligned} \quad (5)$$

where $f = 1 \ldots F$, represent different frequencies.

Moreover, in addition to taking the outputs from the network and computing the masking results, we can integrate the masking function into the neural network directly. Since the binary mask function is not smooth, we propose the integration of the soft time-frequency masking function directly. We add an extra layer to the original output of the neural network as follows:

$$\tilde{\mathbf{y}}_{\mathbf{1}_t} = \frac{|\hat{\mathbf{y}}_{\mathbf{1}_t}|}{|\hat{\mathbf{y}}_{\mathbf{1}_t}| + |\hat{\mathbf{y}}_{\mathbf{2}_t}|} \odot \mathbf{X}_t$$
$$\tilde{\mathbf{y}}_{\mathbf{2}_t} = \frac{|\hat{\mathbf{y}}_{\mathbf{2}_t}|}{|\hat{\mathbf{y}}_{\mathbf{1}_t}| + |\hat{\mathbf{y}}_{\mathbf{2}_t}|} \odot \mathbf{X}_t \quad (6)$$

where the operator $\odot$ is the element-wise multiplication (Hadamard product). In this way, we can integrate the constraints to the network and optimize the network with the masking function jointly. Note that although this extra layer is a deterministic layer, the network weights are optimized for the error metric between and among $\tilde{\mathbf{y}}_{\mathbf{1}_t}$, $\tilde{\mathbf{y}}_{\mathbf{2}_t}$ and $\mathbf{y}_{\mathbf{1}_t}$, $\mathbf{y}_{\mathbf{2}_t}$, using back-propagation. To further smooth the predictions, we can apply masking functions to $\tilde{\mathbf{y}}_{\mathbf{1}_t}$ and $\tilde{\mathbf{y}}_{\mathbf{2}_t}$, as in Eqs. (3), (4), and (5), to get the estimated separation spectra $\tilde{\mathbf{s}}_{\mathbf{1}_t}$ and $\tilde{\mathbf{s}}_{\mathbf{2}_t}$. The time domain signals are reconstructed based on the inverse short time Fourier transform (ISTFT) of the estimated spectra.

3.3. Discriminative Training

Given the output predictions $\hat{\mathbf{y}}_{\mathbf{1}_t}$ and $\hat{\mathbf{y}}_{\mathbf{2}_t}$ (or $\tilde{\mathbf{y}}_{\mathbf{1}_t}$ and $\tilde{\mathbf{y}}_{\mathbf{2}_t}$) of the original sources $\mathbf{y}_{\mathbf{1}_t}$ and $\mathbf{y}_{\mathbf{2}_t}$, we can optimize the neural network parameters by minimizing the squared error,

$$||\hat{\mathbf{y}}_{\mathbf{1}_t} - \mathbf{y}_{\mathbf{1}_t}||_2^2 + ||\hat{\mathbf{y}}_{\mathbf{2}_t} - \mathbf{y}_{\mathbf{2}_t}||_2^2 \quad (7)$$

where $||\cdot||_2$ is the l_2 norm between the two vectors.

Furthermore, minimizing Eq. (7) is equivalent to increasing the similarity between the prediction and the target. For a source separation problem, one of the goals is to have a high signal to interference ratio (SIR); that is, we do not want signals from other sources in the current source prediction. Therefore, we propose a discriminative objective function that takes into account the similarity between the prediction and other sources, and between the prediction and the current target.

$$||\hat{\mathbf{y}}_{\mathbf{1}_t}-\mathbf{y}_{\mathbf{1}_t}||_2^2-\gamma||\hat{\mathbf{y}}_{\mathbf{1}_t}-\mathbf{y}_{\mathbf{2}_t}||_2^2+||\hat{\mathbf{y}}_{\mathbf{2}_t}-\mathbf{y}_{\mathbf{2}_t}||_2^2-\gamma||\hat{\mathbf{y}}_{\mathbf{2}_t}-\mathbf{y}_{\mathbf{1}_t}||_2^2 \quad (8)$$

where γ is a constant chosen by the performance on the development set.

4. EXPERIMENTS

4.1. Setting

We evaluate the performance of the proposed approaches for monaural speech separation using the TIMIT corpus. Eight TIMIT sentences from a male and a female speaker, respectively, are used for training. With the remaining sentences, one sentence from the male and one from the female are used as the development set and the others are used as the test set. Test sentences are added up to form a mixed signal at 0 dB SNR. For neural network training, in order to increase the variety of training samples, we circularly shift (in the time domain) the signals of the male speaker and mix them with utterances from the female speaker.

4.1.1. Features

In the experiments, we explore two different input features: spectral and log-mel filterbank features. The spectral representation is extracted using a 1024-point short time Fourier transform (STFT) with 50% overlap. In the speech recognition literature [16], the log-mel filterbank is found to provide better results compared to mel-frequency cepstral coefficients (MFCC) and log FFT bins. The 40-dimensional log-mel representation and the first and second order derivative features are also explored in the experiments. Empirically we found that using a 32 ms window with a 16 ms frame shift performs the best. The input frame rate corresponds to the output spectra which are extracted using a 512-point STFT.

4.1.2. Metric

The source separation evaluation is measured by using three quantitative values: Source to Interference Ratio (SIR), Source to Artifacts Ratio (SAR), and Source to Distortion Ratio (SDR), according to the BSS-EVAL metrics [17]. Higher values of SDR, SAR, and SIR represent better separation quality. The suppression of interference is reflected in SIR. The artifacts introduced by the separation process are reflected in SAR. The overall performance is reflected in SDR.

4.2. Experimental Results

We use the standard NMF with the generalized KL-divergence metric using 512-point and 1024-point STFT as our baselines. We first train a set of basis vectors, $\mathbf{W_m}, \mathbf{W_f}$ from male and female training data, respectively. After solving coefficients, $\mathbf{H_m}$ and $\mathbf{H_f}$, the binary and soft time-frequency masking functions are applied to the predicted magnitude spectrogram. Figure 3 shows the NMF results with respect to different numbers of basis vectors (10, 30, 50) and different STFT window sizes using binary and soft masks. The results are averaged across 10 different random initializations. For our proposed neural networks, we optimize our models by back-propagating the gradients with respect to the training objectives. The limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm is used to train the models from

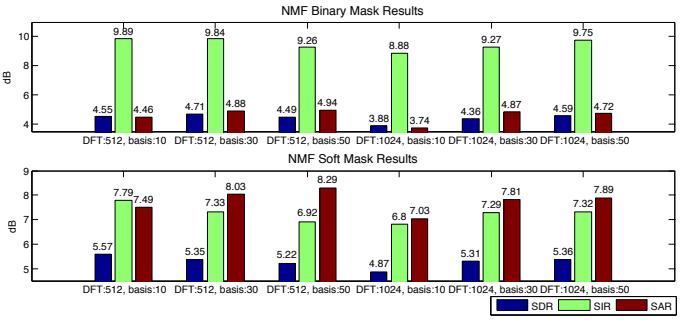

Fig. 3: NMF results with the 512-point and 1024-point STFT and basis vector sizes (10, 30, 50) using binary and soft time-frequency masking

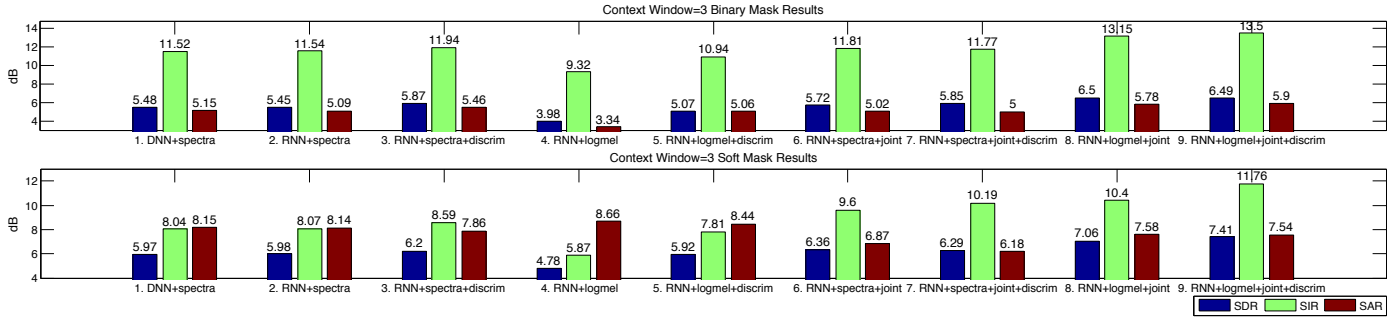

Fig. 4: Neural network results with concatenating neighboring 1 frame as input, where "joint" indicates the joint training between the network and the soft masking function, and "discrim" indicates the training with discriminative objectives

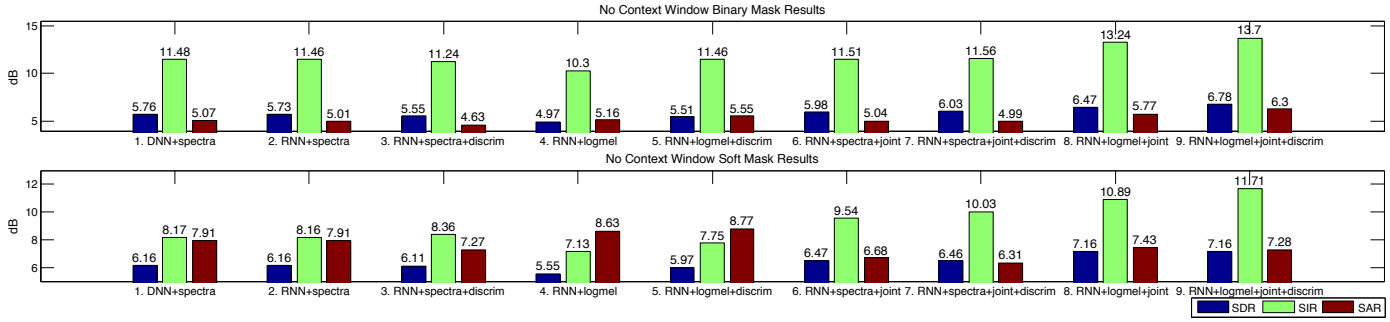

Fig. 5: Neural network results without concatenating neighboring frames as input, where "joint" indicates the joint training between the network and the soft masking function, and "discrim" indicates the training with discriminative objectives

random initialization. We train the models with two hidden layers of 150 hidden units. To further understand the strength of the models, we compare the experimental results in several aspects.

To examine the effectiveness of using input with and without neighboring frames, we report the results in Figure 4 and 5, respectively. The differences between the two cases are not significant. The top and bottom rows of Figure 4 and 5 show the results with binary and soft time-frequency masking, respectively. Similar to the results in NMF, as shown in Figure 3, a binary mask makes hard decisions to enforce the separation and hence results in higher SIRs, but also leads to artifacts with lower SARs. Soft mask, conversely, achieves better SDRs and SARs, but with lower SIRs. In the first two columns, we compare the results between the DNN and the RNN using spectra as features. We found that the differences between the DNN and the RNN are small. The differences in using other features or other training criteria are also insignificant. Due to the space limit, we only report the results of the RNNs here. Between columns 2, 3, 6, and 7, and columns 4, 5, 8, and 9, we make comparisons using spectra and log-mel filterbank as input features. In the cases without joint training, columns 2, 3, 4, and 5, spectral features perform better than log-mel filterbank features. On the other hand, in the joint training cases, columns 6, 7, 8, and 9, log-mel filterbank features achieve better results. Between columns 2 and 3, columns 4 and 5, columns 6 and 7, and columns 8 and 9, we compare the effectiveness of using the discriminative training criterion, i.e., $\gamma > 0$ in Eq. (8). In most cases, SIRs are improved. The results match our expectation when we design the objective function. However, it also leads to some artifacts which result in slightly lower SARs in some cases. Empirically, the value γ is in the range of $0.05 \sim 0.2$ in order to achieve SIR improvements and maintain SAR and SDR. Comparing columns 2, 3, 4, and 5 and columns 6, 7, 8, and 9, we can observe that jointly training the network with the masking function achieves large improvements. Since the standard NMF is trained without concatenating neighboring features, finally, we compare the NMF results with the results in Figure 5. Our best model achieves $3.8 \sim 4.8$ dB and $3.9 \sim 4.9$ dB SIR gain with binary and soft time-frequency masking, respectively, while the model achieves better SDRs and SARs. The sound examples and more details of this work are available online.[1]

5. CONCLUSION

In this paper, we propose using deep learning models for monaural speech separation. Specifically, we propose the joint optimization of a soft masking function and deep learning models (DNNs and RNNs). With the proposed discriminative training criterion, we further improve the SIR. Overall, our proposed models achieve $3.8 \sim 4.9$ dB SIR gain compared to the NMF baseline, while maintaining better SDRs and SARs. For future work, it is important to explore longer temporal information with neural networks. Our proposed models can also be applied to many other applications such as robust ASR.

[1] https://sites.google.com/site/deeplearningsourceseparation/

6. REFERENCES

[1] O. Vinyals, S. V. Ravuri, and D. Povey, "Revisiting recurrent neural networks for robust ASR," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2012, pp. 4085–4088.

[2] A. L. Maas, Q. V Le, T. M O'Neil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *INTERSPEECH*, 2012.

[3] P.-S. Huang, S. D. Chen, P. Smaragdis, and M. Hasegawa-Johnson, "Singing-voice separation from monaural recordings using robust principal component analysis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 57–60.

[4] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[5] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 50–57.

[6] P. Smaragdis, B. Raj, and M. Shashanka, "A probabilistic latent variable model for acoustic modeling," *Advances in models for acoustic processing, NIPS*, vol. 148, 2006.

[7] Ron J Weiss, *Underdetermined source separation using speaker subspace models*, Ph.D. thesis, Columbia University, 2009.

[8] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504 – 507, 2006.

[9] S. Parveen and P. Green, "Speech enhancement with missing data techniques using recurrent neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2004, vol. 1, pp. I–733.

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

[11] P.-S. Huang, L. Deng, M. Hasegawa-Johnson, and X. He, "Random features for kernel deep convex network," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[12] P.-S. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. Heck, "Learning deep structured semantic models for web search using clickthrough data," in *ACM International Conference on Information and Knowledge Management (CIKM)*, 2013.

[13] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2013.

[14] Yuxuan Wang and DeLiang Wang, "Towards scaling up classification-based speech separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1381–1390, 2013.

[15] Xavier Glorot, Antoine Bordes, and Yoshua Bengio, "Deep sparse rectifier neural networks," in *JMLR W&CP: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2011)*, 2011.

[16] J. Li, D. Yu, J.-T. Huang, and Y. Gong, "Improving wideband speech recognition using mixed-bandwidth training data in CD-DNN-HMM," in *IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2012, pp. 131–136.

[17] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 4, pp. 1462 –1469, July 2006.