RECALL NEURAL NETWORK FOR SOURCE SEPARATION

Jen-Tzung Chien *Kai-Wei Tsou*

Department of Electrical and Computer Engineering, National Chiao Tung University, Taiwan

ABSTRACT

This paper presents a novel memory-augmented neural network for single-channel source separation. We propose a recall neural network (RCNN) where a couple of external memories are realized for sequence-to-sequence learning based on an encoder and a decoder. These memories are learned in a two-pass sensing procedure where the mixed signal is encoded and then decoded (or recalled) as context vectors by using a bidirectional long short-term memory (LSTM) and a LSTM, respectively. These context vectors are integrated in a gating layer. A set of attention weights are calculated to attend the hidden state of decoder to implement a recurrent neural network for source separation. A gated attention mechanism is carried out to fulfill a specialized memory network. The regression errors due to two passes of sensing procedure and one pass of gated attention are jointly minimized to estimate the weight parameters of different components in different layers. The experiments on multi-speaker speech enhancement show that the proposed RCNN consistently outperforms LSTM and neural Turing machine in different settings.

Index Terms— long short-term memory, sequence-to-sequence learning, recall neural network, source separation

1. INTRODUCTION

Monaural source separation aims to separate a single-channel mixed signal into the corresponding source signals. Speech enhancement is a special case of monaural source separation which is realized to demix or enhance a noisy speech signal into its clean source speech. The demixing or enhancement system is treated as a regression problem which can be solved by deep learning based on recurrent neural network (RNN) [1–6]. However, RNN suffers from the problem of gradient explosion or vanishing in stochastic gradient descent (SGD) optimization. Long short-term memory (LSTM) [7] or gated recurrent unit [8] was proposed to mitigate this problem based on the gating mechanism and memory cell. Although LSTM is powerful for sequential data learning, a key limitation is the capability of information storage due to an internal memory caused by rapid transition of hidden states. In [9,10], the memory augmented neural networks based on neural Turing machine (NTM) and memory network were proposed by enhancing the storage based on external memories. Memorization was substantially expanded to catch various nonstationary patterns in long sequential data. NTM realized an addressing mechanism to read and write external memory. Useful information was dynamically retrieved while useless information was overwritten based on an attention mechanism. More recently, NTM was extended and successfully developed for monaural source separation [11].

In general, source separation is seen as a sequence-to-sequence learning problem [12, 13] for sequence mapping between mixed signals and source signals. Sequence-to-sequence learning has been extensively developed for machine translation [14], speech recognition [15] and image caption [16]. This study presents a new memory augmented neural network for source separation where the external memory is built based on a sequence-to-sequence neural network. An encoder-decoder network with a pair of memories is exploited to run a two-pass sensing procedure. The first pass encodes the mixed signal into a context vector using a bidirectional LSTM (BLSTM) while the second pass recalls the mixed signal again via a LSTM decoder. The context vectors of encoder and decoder are combined to estimate a set of weights to attend the decoder outputs for source separation by using the third LSTM. Three LSTMs are constructed for encoder, decoder and separator. A recall neural network (RCNN) is built for monaural source separation. Different from NTM addressing over a limited size of memory, the proposed RCNN adopts the gated attention method without the constraint on memory size. RCNN directly stores all historical information in hidden states of two LSTMs which are attended at separation phase. A series of experiments are conducted to show the merit of RCNN over LSTM and NTM for single-channel speech separation.

2. BACKGROUND SURVEY

2.1. Single-channel source separation

In a single-channel source separation system, the mixed signal at time t is represented by a magnitude spectrum vector $\mathbf{x}_t^{\text{mix}}$ calculated by short-term Fourier transform. A sequence of mixed signal $\{\mathbf{x}_t^{\text{mix}}\}_{t=1}^T$ is separated into two sequences of source signals $\{\hat{\mathbf{x}}_t^1\}_{t=1}^T$ and $\{\hat{\mathbf{x}}_t^2\}_{t=1}^T$. Such a mixing system with a pair of sources is seen as an underdetermined problem which is challenging in many tasks including singing-

voice separation [17] and speech enhancement which identify the singing voice and the clean speech in presence of background music and distortion noise, respectively. Source separation can be solved by a supervised regression model based on deep neural network (DNN). In [1, 18], DNN has been elevated to RNN which learned the temporal dynamics to improve the system performance for speech separation or speech enhancement. In addition, RNN is extended to LSTM which tackles the difficulty in gradient calculation as well as handle the speech enhancement in presence of unseen speakers. A meaningful reason is that LSTM can memorize the long-term context and improve the masking functions at each time step. Memorizing the history information is helpful to make sure the quality of the estimated masks for reconstruction of clean speech. However, the internal memory in LSTM, represented by hidden state, is updated too rapid to store sufficient information for source separation. In [11], the memory-augmented neural network based on NTM was proposed to improve the performance of LSTM for speech enhancement.

2.2. Neural Turing machine

NTM is a differentiable computer which was proposed to relax the limitation of memorization in LSTM [9]. A memory matrix $\mathbf{M}_t$ with N memory slots is additionally built and continuously updated at each time t based on an addressing mechanism. A controller network, implemented by using a LSTM variant and driven by a hidden state $\mathbf{h}_t$, is developed to carry out the addressing mechanism. This controller provides a set of addressing parameters to decide where to read and write at each time. The addressing parameters of memory key, key strength, interpolation gate, shift weight, sharpening factor, erase vector and add vector are used to fulfill four steps of addressing procedure including content addressing, interpolation, convolutional shift and sharpening. A set of attention weights $\mathbf{w}_t = \{w_{r,t}(i), w_{w,t}(i)\}_{i=1}^{N}$ are accordingly estimated to obtain a read vector by $\mathbf{r}_t = \sum_{i=1}^{N} w_{r,t}(i)\mathbf{M}_t(i)$ and update the memory matrix from $\mathbf{M}_{t-1}$ to $\mathbf{M}_t$ based on an erase vector $\mathbf{e}_t$ and an add vector $\mathbf{a}_t$ via $\mathbf{M}_t(i) = \mathbf{M}_{t-1}(i) \odot [\mathbf{1} - w_{w,t}(i)\mathbf{e}_t] + w_{w,t}(i)\mathbf{a}_t$ where $\odot$ denotes the element-wise product. Different from LSTM only updating hidden state $\mathbf{h}_t$, the recurrent layer in NTM is run by continuously updating $\mathcal{H}_t = \{\mathbf{h}_t, \mathbf{r}_t, \mathbf{M}_t\}$. The extended hidden state parameters $\mathcal{H}_t$, consisting of state vector $\mathbf{h}_t$, read vector $\mathbf{r}_t$ and memory matrix $\mathbf{M}_t$, are formed in NTM. Instead of using memory cell $\mathbf{c}_t$ in standard LSTM, NTM adopts the read vector $\mathbf{r}_t$ to update four gates in this new LSTM variant. A systematic procedure for reading and writing is implemented.

3. RECALL NEURAL NETWORK

Although the memorization capability in NTM is upgraded, there are still two issues remained to affect its practicality in real world. The first one is the high computation cost due to the intensive calculation for read and write with external memory. The second one is the limited size of memory which causes the fast expiration of useful information due to the erase operation at each time. To deal with these issues, we propose a new memory-augmented neural network where the external memory is constructed according to a sequence-to-sequence learning based on an encoder-decoder network. There is no writing of memory matrix. The frequent interaction with external memory is mitigated. More importantly, we implement the recall neural network (RCNN) to effectively preserve and utilize useful information for a period of time without fast expiration due to a limited size of memory.

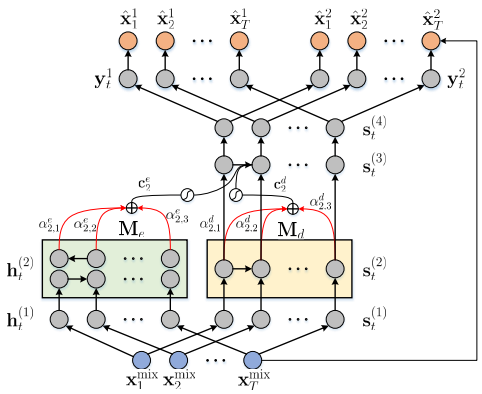

Fig. 1: A recall neural network containing a bidirectional LSTM on the left as an encoder, a LSTM on the right as an decoder and a LSTM on the top as a separator.

3.1. System architecture

Figure 1 depicts the overall architecture of a deep sequence-to-sequence model for monaural source separation from $\{\mathbf{x}_t^{\text{mix}}\}_{t=1}^{T}$ to $\{\hat{\mathbf{x}}_t^1\}_{t=1}^{T}$ and $\{\hat{\mathbf{x}}_t^2\}_{t=1}^{T}$. The proposed RCNN, consisting of three LSTMs, is established to conduct a two-pass sensing procedure before running the separation task. The first layer is a fully-connected (FC) layer which transforms an input $\mathbf{x}_t^{\text{mix}}$ into two hidden states $\{\mathbf{h}_t^{(1)}, \mathbf{s}_t^{(1)}\}$ at each frame t. One is for encoder on the left and the other one is for decoder on the right. The second hidden layer of the encoder is run to obtain hidden units $\mathbf{h}_t^{(2)}$ by using a BLSTM [19, 20] which is composed of a memory matrix for forward and backward directions $\mathbf{M}_e = \{\mathbf{h}_{f,t}^{(2)}, \mathbf{h}_{b,t}^{(2)}\}$ which is shown by green plate. At the same time, the second hidden layer of the decoder is formed by a LSTM layer which produces the hidden units to form the memory matrix $\mathbf{M}_d = \{\mathbf{s}_t^{(2)}\}$ displayed by orange plate. The context vectors of encoder $\mathbf{c}_t^e$ and decoder $\mathbf{c}_t^d$ are then obtained and used to carry out the gated attention mechanism to calculate the hidden unit in the third hidden layer $\mathbf{s}_t^{(3)}$ which is built as a LSTM layer and run for source separation. Basically, the mixed signal $\{\mathbf{x}_t^{\text{mix}}\}_{t=1}^{T}$

is perceived two times before going to separation task. The second-time sensing, run by the decoder, is seen as as a *recall* over the information learned in the first-time sensing, run by the encoder. In case of speech separation, three LSTMs in RCNN are learned to perform *listening*, *listening* and *separating*. This procedure is fitted to how human enhances a noisy speech. Human usually asks to listen the noisy speech again to make sure the result of the separated speech. We therefore call this model as the *recall neural network*.

In the implementation, the fourth hidden layer is formed by a FC layer with parameters $\mathbf{w}^{(4)}$ to find hidden unit $\mathbf{s}_t^{(4)}$ before stacking with the reconstruction layer to estimate the masking function of each source $\mathbf{y}_t^i = \frac{|\mathbf{W}_i^{(5)}\mathbf{s}_t^{(4)}|}{|\mathbf{W}_1^{(5)}\mathbf{s}_t^{(4)}|+|\mathbf{W}_2^{(5)}\mathbf{s}_t^{(4)}|}$ [2, 11] where $\mathbf{W}_i^{(5)}$ denotes the weight parameters at the fifth hidden layer corresponding to two sources $i \in \{1, 2\}$, and $\mathbf{y}_t^i$ is the estimated mask function corresponding to source i. The output signals are then obtained by $\hat{\mathbf{x}}_t^i = \mathbf{y}_t^i \odot \mathbf{x}_t^{\text{mix}}$. We accordingly develop a new sequence-to-sequence model where the attention mechanism allows the model to spotlight on complementary features from input signal based on a two-pass sensing procedure and use these latent features for demixing. Such an attention using weights $\{\alpha_{t,i}^e, \alpha_{t,i}^d\}$ for encoder and decoder is similar to a dropout operation in DNN which picks up useful information and discards redundant information over two external memories $\{\mathbf{M}_e, \mathbf{M}_d\}$. Different from traditional sequence-to-sequence models [12, 14–16], RCNN performs the gated attention over the external memory. In addition to the decoder loss, the encoder loss is considered as a regularizer to our model. The same mixed signal is fed into RCNN twice via the encoder LSTM and the decoder LSTM. Detailed descriptions of encoder-decoder network and gated attention mechanism are addressed below.

3.2. Encoder-decoder network

An encoder-decoder network is introduced to carry out a pair of external memories in RCNN in accordance with a sequence-to-sequence learning. These memories provide the external latent codes and context vectors to demix an observed signal into two source signals. First, the encoder is formed by a FC layer followed by a BLSTM layer. The hidden units in FC layer and in forward and backward directions of BLSTM layer are expressed by $\mathbf{h}_t^{(1)} = \text{FC}(\mathbf{x}_t^{\text{mix}}, \mathbf{w}_e^{(1)})$ and $\mathbf{h}_t^{(2)} = \{\mathbf{h}_{f,t}^{(2)}, \mathbf{h}_{b,t}^{(2)}\} = \text{BLSTM}(\mathbf{h}_t^{(1)}, \mathbf{h}_{t-1}^{(2)}, \mathbf{h}_{t+1}^{(2)}, \mathbf{w}_e^{(2)})$ where $\mathbf{w}_e^{(1)}$ and $\mathbf{w}_e^{(2)}$ denote the parameters in FC and BLSTM layers, respectively. The encoder memory $\mathbf{M}_e = \{\mathbf{h}_{f,t}^{(2)}, \mathbf{h}_{b,t}^{(2)}\}$ is obtained. Owing to the bidirectional sensing, the *whole* mixed signal should be encoded before moving to the next layer to combine with the information learned from decoder. Then, the decoder is built by a FC layer followed by a LSTM layer to obtain hidden codes $\mathbf{s}_t^{(1)} = \text{FC}(\mathbf{x}_t^{\text{mix}}, \mathbf{w}_d^{(1)})$ and $\mathbf{s}_t^{(2)} = \text{LSTM}(\mathbf{s}_t^{(1)}, \mathbf{s}_{t-1}^{(2)}, \mathbf{w}_d^{(2)})$ using decoder parameters $\mathbf{w}_d^{(1)}$ and $\mathbf{w}_d^{(2)}$, respectively. The decoder memory $\mathbf{M}_d = \{\mathbf{s}_t^{(2)}\}$ is stored. Next, a gated attention method is implemented to calculate the context vectors $\{\mathbf{c}_t^e, \mathbf{c}_t^d\}$ and the gating weights $\{\mathbf{g}_t^e, \mathbf{g}_t^d\}$ to carry out the attention-based LSTM layer $\mathbf{s}_t^{(3)} = \text{LSTM}(\mathbf{s}_t^{(2)}, \mathbf{s}_{t-1}^{(3)}, \mathbf{g}_t^e \odot \mathbf{c}_t^e, \mathbf{g}_t^d \odot \mathbf{c}_t^d, \mathbf{w}^{(3)})$ using the parameters $\mathbf{w}^{(3)}$. We build one BLSTM for encoder, one LSTM for decoder and one LSTM for separator. In LSTMs, we use the hidden states from previous time $\{\mathbf{s}_{t-1}^{(2)}, \mathbf{s}_{t-1}^{(3)}\}$ as inputs. In BLSTM, the hidden states in two directions $\{\mathbf{h}_{t-1}^{(2)}, \mathbf{h}_{t+1}^{(2)}\}$ are used as inputs. The gating weights $\{\mathbf{g}_t^e, \mathbf{g}_t^d\}$ reflect how much information in context vectors $\{\mathbf{c}_t^e, \mathbf{c}_t^d\}$ is attended or is ignored.

3.3. Gated attention mechanism

We present a gated attention mechanism which allows RCNN to select useful information for source separation. Using this mechanism, the context vectors of encoder and decoder at each time t are calculated by linearly combining all entries in external memories of encoder and decoder

$$\mathbf{c}_t^e = \sum_{i=1}^{N_e} \alpha_{t,i}^e \mathbf{M}_e(i), \qquad \mathbf{c}_t^d = \sum_{i=1}^{N_d} \alpha_{t,i}^d \mathbf{M}_d(i). \tag{1}$$

In Eq. (1), memories $\mathbf{M}_e = \{\mathbf{M}_e(i)\}$ and $\mathbf{M}_d = \{\mathbf{M}_d(i)\}$ have N_e and N_d vectors, respectively, and $\alpha_{t,i}^e$ and $\alpha_{t,i}^d$ denotes the attention weights given by

$$\alpha_{t,i}^e = \frac{\exp(a_{t,i}^e)}{\sum_{j=1}^T \exp(a_{t,j}^e)}, \quad \alpha_{t,i}^d = \frac{\exp(a_{t,i}^d)}{\sum_{j=1}^T \exp(a_{t,j}^d)} \tag{2}$$

where $a_{t,i}^e = (\mathbf{v}_e)^\top \tanh\left(\mathbf{W}_{ea}\mathbf{s}_{t-1}^{(3)} + \mathbf{U}_{ea}\mathbf{M}_e(i)\right)$ and $a_{t,i}^d = (\mathbf{v}_d)^\top \tanh\left(\mathbf{W}_{da}\mathbf{s}_{t-1}^{(3)} + \mathbf{U}_{da}\mathbf{M}_d(i)\right)$. Given context vectors $\{\mathbf{c}_t^e, \mathbf{c}_t^d\}$ and previous hidden state $\mathbf{s}_{t-1}^{(3)}$, the gating units are computed by

$$\mathbf{g}_t^e = \sigma(\mathbf{W}_{eg}\mathbf{s}_{t-1}^{(3)} + \mathbf{U}_{eg}\mathbf{c}_t^e + \mathbf{b}_{eg}),\ \mathbf{g}_t^d = \sigma(\mathbf{W}_{dg}\mathbf{s}_{t-1}^{(3)} + \mathbf{U}_{dg}\mathbf{c}_t^d + \mathbf{b}_{dg}) \tag{3}$$

where $\sigma(\cdot)$ denotes the sigmoid function. In this study, RCNN parameters $\mathbf{\Theta} = \{\mathbf{w}_e^{(1)}, \mathbf{w}_d^{(1)}, \mathbf{w}_e^{(2)}, \mathbf{w}_d^{(2)}, \mathbf{w}^{(3)}, \mathbf{v}_e, \mathbf{v}_d, \mathbf{W}_{ea}, \mathbf{W}_{da}, \mathbf{U}_{ea}, \mathbf{U}_{da}, \mathbf{W}_{eg}, \mathbf{W}_{dg}, \mathbf{U}_{eg}, \mathbf{U}_{dg}, \mathbf{b}_{eg}, \mathbf{b}_{dg}, \mathbf{w}^{(4)}, \mathbf{W}^{(5)}\}$ are estimated by minimizing the sum-of-squares error function between clean spectra and the estimated spectra $\mathcal{L}(\mathbf{\Theta}) = \frac{1}{2}\sum_{t=1}^T \left(||\mathbf{x}_t^1 - \hat{\mathbf{x}}_t^1|| + ||\mathbf{x}_t^2 - \hat{\mathbf{x}}_t^2|| + ||\mathbf{x}_t^1 - \tilde{\mathbf{x}}_t^1|| + ||\mathbf{x}_t^2 - \tilde{\mathbf{x}}_t^2||\right)$. This loss function is not only calculated from separation outputs $\{\hat{\mathbf{x}}_t^1(\mathbf{\Theta}), \hat{\mathbf{x}}_t^2(\mathbf{\Theta})\}$ ($\mathcal{L}_{\text{sep}}(\mathbf{\Theta})$) but also from the encoder outputs $\{\tilde{\mathbf{x}}_t^1(\mathbf{\Theta}), \tilde{\mathbf{x}}_t^2(\mathbf{\Theta})\}$ ($\mathcal{L}_{\text{enc}}(\mathbf{\Theta})$).

4. EXPERIMENTS

4.1. Experimental setup

Monaural source separation was evaluated using the task of single-channel speech enhancement. Speech signals were

sampled from 83 speakers in WSJ0 SI-84 and noise signals were collected with 88 noise types from http://www.freesfx.co.uk/soundeffects/ and http://www.audiomicro.com/free-sound-effects/. 77 speakers were chosen as training speakers and the remaining 6 speakers were treated as unseen test speakers. The utterances from different speakers were randomly mixed with various nonstationary noises. There were 7768 training utterances which were mixed by using 86 noise types. To evaluate the robustness of different methods, the noisy training data were generated by SNR of -5 dB while the noisy test data were generated by SNR of -5 dB, 0 dB and 5 dB with two unseen noise types (cafeteria and bus). There were 300 noisy test utterances collected from unseen speakers and unseen noisy types. In the implementation, 1024-point FFT was calculated for mixed signals, i.e. $\mathbf{x}_t^{\text{mix}} \in \mathbb{R}^{513}$. DNN, LSTM and NTM were carried out by referring [11]. NTMs with memory size N=8, 32, 64 were examined. Using RCNN, the encoder and decoder with topologies 513-1000 (FC)-800 (BLSTM)-600 (FC)-$\{$513-513$\}$ and 513-1000 (FC)-800 (LSTM)-700 (LSTM)-600 (FC)-$\{$513-513$\}$ were implemented, respectively. All models were trained by using SGD algorithm with Xavier initialization and mini-batch size of 50 frames. Step size of 20 frames was used in backpropagation through time. Adam optimizer was used. Speech enhancement was evaluated by using short-time objective intelligibility (STOI) (higher is better).

(a) (b)

Fig. 2: An example of estimated attention weights (a) $\alpha_{t,i}^e$ and (b) $\alpha_{t,i}^d$ over encoder $\mathbf{M}_e$ and decoder memories $\mathbf{M}_d$, respectively.

Model	-5 dB	0 dB	5 dB	Elaps.
Baseline	0.647	0.746	0.829	–
DNN	0.678	0.785	0.844	3.9
LSTM	0.718	0.815	0.872	12.9
NTM (N=8)	0.725	0.819	0.872	21.2
NTM (N=32)	0.731	0.825	0.877	25.3
NTM (N=64)	0.730	0.824	0.876	31.1
RCNN w/o $\mathbf{c}_t^d$ & $\mathcal{L}_{\text{enc}}$	0.738	0.833	0.883	14.0
RCNN w/o $\mathcal{L}_{\text{enc}}$	0.742	0.835	0.887	14.5
RCNN w/o $\mathbf{c}_t^d$	**0.747**	**0.838**	0.886	14.8
RCNN	0.744	0.837	**0.888**	14.9

Table 1: Comparison of STOIs under different SNRs by using DNN, LSTM and different variants of NTM and RCNN. Elapsed time per learning epoch (in minutes) is provided.

4.2. Experimental results

Figure 2 shows an example of attention weights $\{\alpha_{t,i}^e, \alpha_{t,i}^d\}$ over encoder and decoder memories. It is meaningful that RCNN likely attends the information in specific locations over encoder memory $\mathbf{M}_e$ but attends different locations over decoder memory $\mathbf{M}_d$ to learn for source separation. Table 1 reports the STOIs of using DNN, LSTM, NTM and RCNN under different SNRs of test samples. Baseline result corresponds to that without doing separation. STOI is increased by applying DNN, LSTM, NTM and RCNN. Different RNN variants using LSTM, NTM and RCNN outperform DNN. NTM consistently performs better than LSTM under different SNRs and memory sizes N. Improvement is larger in case of -5 dB but smaller in cases of 0 dB and 5 dB. External memory does work for RNN. Too large memory size (N=64) does not always help. Nevertheless, the improvement of RCNN over NTM is obvious in different SNRs. This is partially because RCNN is *not* constrained by memory size. RCNN stores the information without erasing and adding. To evaluate different functions in RCNN, three simplified variants are realized for comparison. RCNN without attending decoder memory $\mathbf{M}_d$ corresponds to the RCNN without using context vector $\mathbf{c}_t^d$. RCNN without $\mathcal{L}_{\text{enc}}$ expresses the RCNN without including loss function from encoder. RCNN can be also reduced by disregarding both $\mathbf{c}_t^d$ and $\mathcal{L}_{\text{enc}}$. Attending the decoder memory and including the encoder loss do improve the separation performance. In particular, including encoder loss enables the RCNN to speed up convergence while attending the decoder memory helps little in terms of STOI. The information in context vector $\mathbf{c}_t^d$ contains redundancy. RCNN without considering $\mathbf{c}_t^d$ and $\mathcal{L}_{\text{enc}}$ consistently worse than the other RCNN variants. In addition, the elapsed time per learning epoch is shown in minutes by using multi-GPU GTX GeForce GTX 980 device. Attractively, RCNN runs much faster than NTM but insignificantly slower than LSTM. Source codes of RCNN is accessible at https://github.com/NCTUMLlab/Kai-Wei-Tsou-RecallNet.

5. CONCLUSIONS

We have addressed a novel sequence-to-sequence learning to implement the external memories in a recall neural network which imitated the human perceiving for source separation. A recall function was realized by an encoder-decoder network which extracted the complementary information from external memories. A gated attention method was implemented to attend the information of encoder and decoder for speech enhancement. A bidirectional LSTM for encoder, a LSTM for decoder and a LSTM for separator were constructed to *listen*, *listen* and *separate* for source separation. Experiments on speech enhancement showed the superiority of RCNN to other methods under different conditions. RCNN ran significantly faster than neural Turing machine.

6. REFERENCES

[1] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Deep learning for monaural speech separation," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 1562–1566.

[2] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proc. of International Conference on Acoustics, Speech, and Signal Processing*, 2013, pp. 7092–7096.

[3] G.-X. Wang, C.-C. Hsu, and J.-T. Chien, "Discriminative deep recurrent neural networks for monaural speech separation," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 2544–2548.

[4] J.-T. Chien and K.-T. Kuo, "Variational recurrent neural networks for speech separation," in *Proc. of Annual Conference of International Speech Communication Association*, 2017, pp. 1193–1197.

[5] J.-T. Chien and K.-T. Kuo, "Spectro-temporal neural factorization for speech dereverberation," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018.

[6] J.-T. Chien and Y.-C. Ku, "Bayesian recurrent neural network for language modeling," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 2, pp. 361–374, 2016.

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

[8] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[9] A. Graves, G. Wayne, and I. Danihelka, "Neural Turing machines," *arXiv preprint arXiv:1410.5401*, 2014.

[10] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "End-to-end memory networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 2440–2448.

[11] K.-W. Tsou and J.-T. Chien, "Memory augmented neural network for source separation," in *Proc. of IEEE International Workshop on Machine Learning for Signal Processing*, 2017, pp. 1–6.

[12] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems*, 2014, pp. 3104–3112.

[13] J.-T. Chien and H.-L. Hsieh, "Nonstationary source separation using sequential and variational Bayesian learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 5, pp. 681–694, 2013.

[14] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[15] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 4960–4964.

[16] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. of International Conference on Machine Learning*, 2015, pp. 2048–2057.

[17] J.-T. Chien and P.-K. Yang, "Bayesian factorization and learning for monaural source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 1, pp. 185–195, 2016.

[18] J. Chen and D. Wang, "Long short-term memory for speaker generalization in supervised speech separation," *The Journal of the Acoustical Society of America*, vol. 141, no. 6, pp. 4705–4714, 2017.

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

[20] A. Graves, N. Jaitly, and A. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013, pp. 273–278.