INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Variational Recurrent Neural Networks for Speech Separation

Jen-Tzung Chien, *Kuan-Ting Kuo*

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

Abstract

We present a new stochastic learning machine for speech separation based on the variational recurrent neural network (VRNN). This VRNN is constructed from the perspectives of generative stochastic network and variational auto-encoder. The idea is to faithfully characterize the randomness of hidden state of a recurrent neural network through variational learning. The neural parameters under this latent variable model are estimated by maximizing the variational lower bound of log marginal likelihood. An inference network driven by the variational distribution is trained from a set of mixed signals and the associated source targets. A novel supervised VRNN is developed for speech separation. The proposed VRNN provides a stochastic point of view which accommodates the uncertainty in hidden states and facilitates the analysis of model construction. The masking function is further employed in network outputs for speech separation. The benefit of using VRNN is demonstrated by the experiments on monaural speech separation.

Index Terms: recurrent neural network, variational learning, speech separation

1. Introduction

Speech is the vocalized form of communication media which provides the natural interface between human and machine. Nowadays, many speech-related applications and devices have been developed to facilitate our daily lives. However, these speech systems are prone to be degraded in adverse conditions. One of the most challenging tasks in speech technology is to identify the target speech from a mixed speech signal and use the enhanced speech to improve speech recognition system [1, 2]. A typical example of source separation problem is the cocktail-party problem [3] where the target speech is contaminated with a variety of interferences such as ambient noise, competing speech and background music [4]. Over the past few years, a number of single-channel source separation algorithms have been proposed. In particular, *model-based source separation* methods such as nonnegative matrix factorization [5, 6, 7, 8, 9] and deep neural network [10, 11, 12] are current research trends.

In recent years, deep learning has emerged as a powerful learning machine which achieves state-of-the-art performance in many applications ranging from classification to regression tasks. Exploring the solution to speech separation is known as a regression task where the demixed signals are treated as real-valued targets. Basically, deep neural network (DNN) adopts a hierarchical architecture to grasp latent information of demixed signals in different levels of abstraction from the mixed signals. In the literature, DNN was applied to predict the separated spectra from the noisy spectra [13]. Extended from the feedforward neural network, the recurrent neural network (RNN) was proposed to explore the temporal information for source separation [12, 14, 15]. In [16, 11], the long short-term memory was introduced to act as the hidden units to tackle the problem of gradient vanishing or exploding in training procedure of RNN. Long and short-term contextual information was exploited and preserved to improve the performance of monaural source separation.

Traditionally, RNN was constructed with recurrent hidden units which were assumed deterministic. RNN parameters were trained via the deterministic error backpropagation. The estimated parameters in hidden units may not faithfully reflect the uncertainty in model construction which is caused by improper model complexity, noise interference and mismatch between training and test conditions [17, 18]. The data reconstruction from hidden units is not possible. Such a weakness may constrain the representation capability of RNN. In [19], the variational auto-encoder (VAE) was proposed to incorporate a variational distribution to characterize the statistical property of hidden variables. In [20], the generative stochastic network (GSN) was proposed to reconstruct original signal in an unsupervised RNN which was a combination of Markov chain and neural network. In this study, we develop a new variational recurrent neural network (VRNN) for speech separation by integrating the variational learning in VAE and the recurrent property in GSN or RNN. A variational lower bound of log likelihood is maximized to find model parameters of VRNN. A general solution to supervised regression problem is derived. Experiments are conducted to evaluate the performance of the proposed VRNN for single-channel source separation.

2. Background survey

2.1. Recurrent neural network

For single-channel speech separation, RNN is adopted as a nonlinear regression model to predict the magnitude spectra or the masking function of the separated speech signals from two sources $\mathbf{y}_t = \{\mathbf{y}_{1,t}, \mathbf{y}_{2,t}\}$ given by an input magnitude spectra of the mixed signal $\mathbf{x}_t$ from single microphone. A basic RNN is composed of a chain of functional transformations in time horizon as shown in Figure 1(a). The recurrent structure in RNN is crucial to learn temporal dependency from input time-series data such as audio and speech signals. The hidden unit $\mathbf{h}_t$ at time t is obtained from the D-dimensional input $\mathbf{x}_t$ at time t and the hidden unit $\mathbf{h}_{t-1}$ at time $t-1$ using a transformation $\mathcal{F}(\cdot)$ via $\mathbf{h}_t = \mathcal{F}(\mathbf{x}_t, \mathbf{h}_{t-1})$ with weight parameters $\mathbf{w}$. The output $\mathbf{y}_t$ is obtained from hidden unit $\mathbf{h}_t$ at the same time t through a transformation $\hat{\mathbf{y}}_t = \mathcal{F}(\mathbf{h}_t)$. The transformation is composed of an affine function and an activation function. The weight parameters $\mathbf{w}$ for connections from inputs $\{\mathbf{x}_t, \mathbf{h}_{t-1}\}$ to hidden units $\mathbf{h}_t$ and from hidden units $\mathbf{h}_t$ to output $\mathbf{y}_t$ are estimated by minimizing the sum-of-square error function $E_{\mathbf{w}} = \frac{1}{2}\sum_{t=1}^{T}\sum_{i=1}^{2}\sum_{d=1}^{D}(\hat{y}_{i,t,d} - y_{i,t,d})^2$ from a collection of T input-output data pairs $\mathcal{D} = \{\mathbf{x}_t, \mathbf{y}_t\}$ where $\hat{\mathbf{y}}_t = \{\hat{\mathbf{y}}_{1,t}, \hat{\mathbf{y}}_{2,t}\} = \{\{\hat{y}_{1,t,d}\}, \{\hat{y}_{2,t,d}\}\}$ is the target outputs of two sources corresponding to single-channel inputs $\mathbf{x}_t$. Parameters $\mathbf{w}$ are trained by using mini-batches of $\mathcal{D}$ according to the stochastic gradient descent (SGD) algorithm where the

http://dx.doi.org/10.21437/Interspeech.2017-832

gradient of objective function using a mini-batch $\nabla_{\mathbf{w}} \tilde{E}_{\mathbf{w}}$ is calculated for parameter updating. During SGD training, hidden units $\mathbf{h}_t$ are assumed to be deterministic. There is no reconstruction equipped in RNN model.

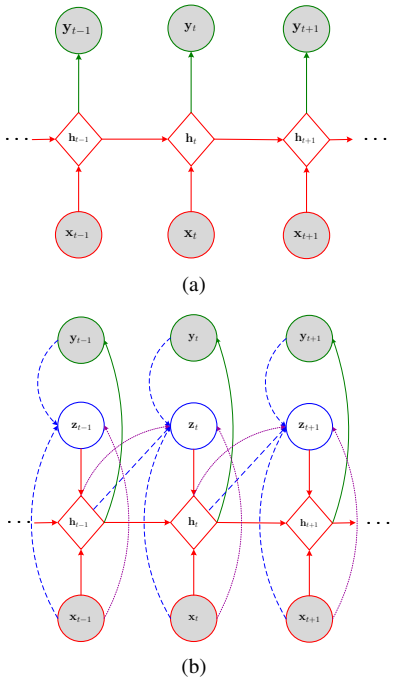

Figure 1: *Graphical representation for (a) RNN and (b) VRNN.*

2.2. Variational auto-encoder

In [19], the variational auto-encoder (VAE) was proposed to estimate the distribution of hidden variables $\mathbf{z}$ and use this information to reconstruct original signal $\mathbf{x}$. This distribution characterizes the randomness of hidden units which provides a vehicle to reconstruct different realizations of output signals rather than a point estimate of outputs in traditional auto-encoder. Accordingly, it makes possible to synthesize the generative samples and analyze the statistics of hidden information of neural network. Figure 2(a) shows how the output $\hat{\mathbf{x}}$ is reconstructed from original input $\mathbf{x}$. The graphical model of VAE is depicted by Figure 2(b) which consists of an encoder and a decoder. Encoder is seen as a recognition model which identifies the stochastic latent variables $\mathbf{z}$ using a variational posterior $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ with parameters $\boldsymbol{\phi}$. Latent variables $\mathbf{z}$ are sampled by using variational posterior. These samples $\mathbf{z}$ are then used to generate or reconstruct original signal $\hat{\mathbf{x}}$ based on the decoder or generative model using likelihood function $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ with parameters $\boldsymbol{\theta}$. The whole model is formulated by using the variational Bayesian expectation maximization algorithm. Variational parameters $\boldsymbol{\phi}$ and model parameters $\boldsymbol{\theta}$ are estimated by maximizing the *variational lower bound* of log likelihood $\log p(\mathbf{x}_{\leq T})$ from a collection of samples $\mathbf{x}_{\leq T} = \{\mathbf{x}_t\}_{t=1}^{T}$. *Stochastic* error backpropagation is implemented for variational learning. This VAE was extended to other unsupervised learning tasks [21] for finding the synthesized images. This study develops a variational recurrent neural network (VRNN) to tackle the supervised regression learning for speech separation.

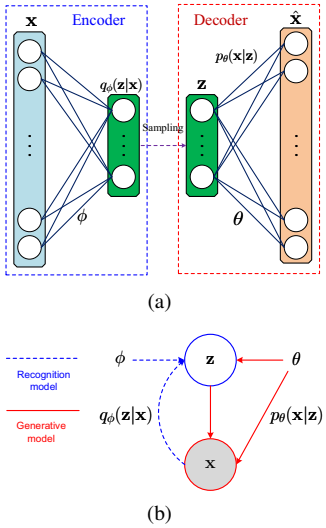

Figure 2: *(a) Encoder and decoder in VAE. (b) Graphical representation for VAE.*

3. Variational recurrent neural network

This paper is motivated by introducing VAE into construction of RNN to implement a stochastic realization of RNN. The resulting VRNN [22] is operated in a way shown in Figure 1(b).

3.1. Model construction

In VRNN, we estimate a set of T time-dependent hidden units $\mathbf{h}_{\leq T}$ corresponding to the observed mixture signals $\mathbf{x}_{\leq T}$ which are used to produce the RNN outputs $\mathbf{y}_{\leq T}$ as the demixed signals for regression task. The hidden units $\mathbf{h}_{\leq T}$ are characterized and generated by hidden variables $\mathbf{z}_{\leq T}$. Similar to VAE, the proposed VRNN is equipped with an encoder and a decoder. This VRNN aims to capture the temporal and stochastic information in time-series observations and hidden features. The encoder in VRNN is designed to encode or identify the distribution $q_{\boldsymbol{\phi}}(\mathbf{z}_t|\mathbf{x}_t, \mathbf{y}_t, \mathbf{h}_{t-1})$ of latent variable $\mathbf{z}_t$ from input-output pair $\{\mathbf{x}_t, \mathbf{y}_t\}$ at each time t and hidden feature $\mathbf{h}_{t-1}$ at previous time $t-1$ as shown by dashed blue lines. Given the random samples $\mathbf{z}_t$ from variational distribution $q_{\boldsymbol{\phi}}(\cdot)$, the decoder in VRNN is introduced to realize the hidden units $\mathbf{h}_t = \mathcal{F}(\mathbf{x}_t, \mathbf{z}_t, \mathbf{h}_{t-1})$ at current time t as shown in solid red lines. Hidden unit $\mathbf{h}_t$ acts as the realization or surrogate of hidden variable $\mathbf{z}_t$. The generative likelihood $p_{\boldsymbol{\theta}}(\cdot)$ is estimated to obtain the random output $\mathbf{y}_t$. Comparable with standard RNN, the hidden units $\mathbf{h}_t$ in VRNN are used to generate the outputs $\hat{\mathbf{y}}_t$ by $\hat{\mathbf{y}}_t \sim p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{h}_t)$ as shown by solid green lines. VRNN pursues the *random* generation of regression outputs guided by the variational learning of hidden features. A stochastic learning of RNN is fulfilled by the following inference procedure.

3.2. Model inference

Figure 3 shows the inference procedure of VRNN. In model inference of supervised VRNN, we maximize the variational lower bound $\mathcal{L}$ of logarithm of conditional likelihood $p(\mathbf{y}_{\leq T}|\mathbf{x}_{\leq T}) = \prod_{t=1}^{T} \sum_{\mathbf{z}_t} p_{\boldsymbol{\theta}}(\mathbf{y}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{\leq t}) p_{\boldsymbol{\omega}}(\mathbf{z}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{<t})$ which is decomposed into two terms containing parameters $\boldsymbol{\theta}$ and $\boldsymbol{\omega}$. The first term is a negative sum-of-squares error function which is obtained due to the Gaussian assumption for re-

gression errors. The second term is an expected Kullback-Leibler (KL) divergence between distributions $q_{\boldsymbol{\phi}}(\cdot)$ and $p_{\boldsymbol{\omega}}(\cdot)$. Lower bound is expressed by

$$\mathcal{L} \triangleq \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T},\mathbf{y}_{\leq T})}\left[\sum_{t=1}^{T}\Big(\log p_{\boldsymbol{\theta}}(\mathbf{y}_t|\mathbf{x}_{\leq t},\mathbf{z}_{\leq t}) - \mathcal{D}_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\mathbf{z}_t|\mathbf{x}_{\leq t},\mathbf{y}_{\leq t},\mathbf{z}_{<t})||p_{\boldsymbol{\omega}}(\mathbf{z}_t|\mathbf{x}_{\leq t},\mathbf{z}_{<t}))\Big)\right]. \quad (1)$$

In maximization of Eq. (1), we first calculate the prior distribution of latent variable $\mathbf{z}_t$ which is a Gaussian $p_{\boldsymbol{\omega}}(\mathbf{z}_t|\mathbf{x}'_t,\mathbf{h}_{t-1}) = \mathcal{N}(\boldsymbol{\mu}_{0,t},\mathrm{diag}(\boldsymbol{\sigma}^2_{0,t}))$ where the mean and variance are calculated by a prior network $[\boldsymbol{\mu}_{0,t},\boldsymbol{\sigma}^2_{0,t}] = \boldsymbol{\psi}^{\mathrm{prior}}_{\boldsymbol{\omega}}(\mathbf{x}'_t,\mathbf{h}_{t-1})$ using encoding weights $\boldsymbol{\omega}$ (shown by dashed purple lines). Then, the variational distribution is calculated at each time frame by using a Gaussian $q_{\boldsymbol{\phi}}(\mathbf{z}_t|\mathbf{x}'_t,\mathbf{y}'_t,\mathbf{h}_{t-1}) = \mathcal{N}(\boldsymbol{\mu}_{z,t},\mathrm{diag}(\boldsymbol{\sigma}^2_{z,t}))$ with mean and variance calculated by an inference network using the encoder weights, i.e. $[\boldsymbol{\mu}_{z,t},\boldsymbol{\sigma}^2_{z,t}] = \boldsymbol{\psi}^{\mathrm{enc}}_{\boldsymbol{\phi}}(\mathbf{x}'_t,\mathbf{y}'_t,\mathbf{h}_{t-1})$. Here, $\mathbf{x}'_t$ and $\mathbf{y}'_t$ denote the encoded features of $\mathbf{x}_t$ and $\mathbf{y}_t$ with reduced dimensions by using feature extractors $\boldsymbol{\psi}^x(\mathbf{x}_t)$ and $\boldsymbol{\psi}^y(\mathbf{y}_t)$ with parameters $\boldsymbol{\phi}^x$ and $\boldsymbol{\phi}^y$, respectively. This is called the recognition or encoding phase with four sets of encoding weights $\{\boldsymbol{\phi}^x,\boldsymbol{\phi}^y,\boldsymbol{\phi}^{\mathrm{enc}},\boldsymbol{\omega}\}$.

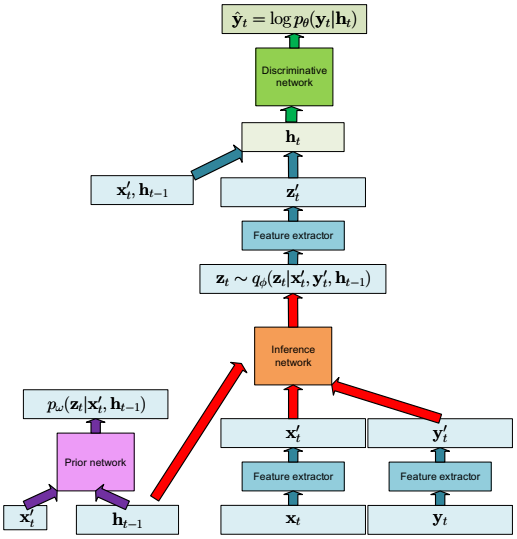

Figure 3: *Inference procedure for VRNN.*

In the generation or decoding phase, we first apply the feature extractor $\boldsymbol{\psi}^z(\mathbf{z}_t)$ with parameters $\boldsymbol{\theta}^z$ to estimate the features $\mathbf{z}'_t$ corresponding to the latent variables $\mathbf{z}_t$. The variables $\mathbf{z}_t$ are sampled from the Gaussian distribution $q_{\boldsymbol{\phi}}(\mathbf{z}_t|\mathbf{x}'_t,\mathbf{y}'_t,\mathbf{h}_{t-1})$ which was obtained in encoding phase. Then, we calculate the conditional likelihood $p_{\boldsymbol{\theta}}(\mathbf{y}_t|\mathbf{h}_t)$ at each time t. This likelihood is estimated from the outputs of a decoder or discriminative network $\boldsymbol{\psi}^{\mathrm{dec}}_{\boldsymbol{\theta}}(\mathbf{h}_t)$ with parameters $\boldsymbol{\theta}^{\mathrm{dec}}$. This likelihood is used to calculate the regression output $\hat{\mathbf{y}}_t$ by using the inputs from $\mathbf{h}_t = \mathcal{F}(\mathbf{x}'_t,\mathbf{z}'_t,\mathbf{h}_{t-1})$ which is function of $\mathbf{x}'_t$, $\mathbf{z}'_t$ and $\mathbf{h}_{t-1}$ with parameters $\boldsymbol{\theta}^h$. There are three sets of parameters $\{\boldsymbol{\theta}^z,\boldsymbol{\theta}^{\mathrm{dec}},\boldsymbol{\theta}^h\}$ in VRNN decoder. Importantly, the expectation in variational lower bound Eq. (1) is calculated by using L samples $\mathbf{z}_t$ obtained via variational distribution $q_{\boldsymbol{\phi}}(\mathbf{z}_t|\mathbf{x}'_t,\mathbf{y}'_t,\mathbf{h}_{t-1})$. However, directly sampling $\mathbf{z}_t$ using the Gaussian distribution with the mean $\boldsymbol{\mu}_{z,t}$ and variance $\boldsymbol{\sigma}^2_{z,t}$ obtained by encoding network $\boldsymbol{\psi}^{\mathrm{enc}}_{\boldsymbol{\phi}}(\mathbf{x}'_t,\mathbf{y}'_t,\mathbf{h}_{t-1})$ is unstable with high variance. We follow [23] and use the reparameterization trick to resolve this problem. Using this trick, we sample $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ and use this sample to determine the sample for latent variable $\mathbf{z}_t \leftarrow \boldsymbol{\mu}_{z,t} + \boldsymbol{\sigma}_{z,t} \odot \boldsymbol{\epsilon}$ where $\odot$ denotes the element-wise multiplication. The stochastic training procedure for encoding weights $\{\boldsymbol{\phi}^x,\boldsymbol{\phi}^y,\boldsymbol{\phi}^{\mathrm{enc}},\boldsymbol{\omega}\}$ and decoding weights $\{\boldsymbol{\theta}^z,\boldsymbol{\theta}^{\mathrm{dec}},\boldsymbol{\theta}^h\}$ is illustrated in Algorithm 1 which is seen as a stochastic gradient variational Bayes estimator.

Algorithm 1: Training procedure for VRNN

Initialize hidden state $\mathbf{h}_0$, parameters $\boldsymbol{\omega}$, $\boldsymbol{\phi}$, $\boldsymbol{\theta}$
for $t = 1, 2, \ldots, T$ **do**
  Feedforward computation
  $\mathbf{x}'_t \leftarrow \boldsymbol{\psi}^x(\mathbf{x}_t)$ with parameter $\boldsymbol{\phi}^x$
  $\mathbf{y}'_t \leftarrow \boldsymbol{\psi}^y(\mathbf{y}_t)$ with parameter $\boldsymbol{\phi}^y$
  $\{\boldsymbol{\mu}_{z,t},\boldsymbol{\sigma}^2_{z,t}\} \leftarrow \boldsymbol{\psi}^{\mathrm{enc}}_{\boldsymbol{\phi}}(\mathbf{x}'_t,\mathbf{y}'_t,\mathbf{h}_{t-1})$ with par. $\boldsymbol{\phi}^{\mathrm{enc}}$
  $\{\boldsymbol{\mu}_{0,t},\boldsymbol{\sigma}_{0,t}\} \leftarrow \boldsymbol{\psi}^{\mathrm{prior}}_{\boldsymbol{\omega}}(\mathbf{x}'_t,\mathbf{h}_{t-1})$ with parameter $\boldsymbol{\omega}$
  $\boldsymbol{\epsilon}$ sampled from $\mathcal{N}(\mathbf{0},\mathbf{I})$
  $\mathbf{z}_t \leftarrow \boldsymbol{\mu}_{z,t} + \boldsymbol{\sigma}_{z,t} \odot \boldsymbol{\epsilon}$
  $\mathbf{z}'_t \leftarrow \boldsymbol{\psi}^z(\mathbf{z}_t)$ with parameter $\boldsymbol{\theta}^z$
  $\mathbf{h}_t \leftarrow \mathcal{F}(\mathbf{x}'_t,\mathbf{z}'_t,\mathbf{h}_{t-1})$ with parameter $\boldsymbol{\theta}^h$
  $\hat{\mathbf{y}}_t \leftarrow \log p_{\boldsymbol{\theta}}(\mathbf{y}_t|\mathbf{h}_t)$ via $\boldsymbol{\psi}^{\mathrm{dec}}_{\boldsymbol{\theta}}(\mathbf{h}_t)$ with par. $\boldsymbol{\theta}^{\mathrm{dec}}$
  Backward computation
  Calculate the variational lower bound $\mathcal{L}$
  Compute the gradients $\frac{\partial\mathcal{L}}{\partial\boldsymbol{\phi}}$, $\frac{\partial\mathcal{L}}{\partial\boldsymbol{\omega}}$, $\frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}}$
  Update the parameters
    $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} + \eta_\phi \odot \frac{\partial\mathcal{L}}{\partial\boldsymbol{\phi}}$
    $\boldsymbol{\omega} \leftarrow \boldsymbol{\omega} + \eta_\omega \odot \frac{\partial\mathcal{L}}{\partial\boldsymbol{\omega}}$
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \eta_\theta \odot \frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}}$
end

4. Experiments

4.1. Experimental setup

In the experiments, we used the mixed speech signals from TIMIT corpus for evaluation of single-channel speech separation performance using different methods [14]. There were 630 speakers in TIMIT corpus. Each speaker had ten sentences. In training phase, eight sentences were randomly chosen from one male speaker and one female speaker for signal mixing. Another sentence was used for cross validation and the remaining sentence was used for testing. All sentences were normalized to be with equal power. In the implementation, the sentences from one speaker were circularly shifted every 10K samples and added to the sentences from the other speaker as the training data. The 1024-point short-term Fourier transform with a 64-ms frame duration and a 32-ms frame shift was calculated to obtain the Fourier spectrograms. The adaptive SGD algorithm using Adam [24] was performed. The measurements (in dB) based on source to distortion ratio (SDR), source to interference ratio (SIR) and source to artifacts ratio (SAR) [25] were evaluated.

For each frame t, the spectral signal of the mixed speech $\mathbf{x}_t$ with dimension 513 was fed into VRNN. The outputs of VRNN corresponded to the spectra of demixed signals $\{\mathbf{y}_{1,t},\mathbf{y}_{2,t}\}$ with dimension 1026. Figure 4 depicts the network topology for implementation of VRNN. The feature extractors calculated $\mathbf{x}'_t$ and $\mathbf{y}'_t$ with the same dimension 250. Single layer with ReLU activation function was specified. Using $\mathbf{x}'_t$ and $\mathbf{y}'_t$ at time t and hidden state $\mathbf{h}_{t-1}$ at time $t-1$, the prior network and the inference network were constructed using ReLU acti-

vation with 150 dimensional hidden units. The outputs of two networks were calculated by linear activation and viewed as 50 dimensional mean and variance for Gaussians in prior distribution $p_{\boldsymbol{\omega}}(\mathbf{z}_t|\mathbf{x}'_t)$ and variational distribution $q_{\boldsymbol{\phi}}(\mathbf{z}_t|\mathbf{x}'_t, \mathbf{y}'_t, \mathbf{h}_{t-1})$. The latent variable $\mathbf{z}_t$ was then sampled and transformed to 150 dimensional $\mathbf{z}'_t$. The 150 dimensional hidden state $\mathbf{h}_t$ was accordingly obtained by RNN with linear activation. This hidden state was then forwarded through a hidden layer with 450 units for calculating the 513 dimensional activations $\{\mathbf{a}_{1,t}, \mathbf{a}_{2,t}\}$ for two sources. ReLU activation function was applied. An ideal ratio mask [26, 14] was implemented to find 513 dimensional masking functions $\{\mathbf{m}_{1,t}, \mathbf{m}_{2,t}\}$ for two sources by $m_{i,t,d} = \frac{|a_{i,t,d}|}{|a_{1,t,d}|+|a_{2,t,d}|}$. Finally, the demixed spectral signals of two sources $\{\hat{\mathbf{y}}_{1,t}, \hat{\mathbf{y}}_{2,t}\}$ were calculated by $\hat{\mathbf{y}}_{i,t} = \mathbf{x}_t \odot \mathbf{m}_{i,t}$ where $i \in \{1, 2\}$. In the implementation, we first trained the inference network and discriminative network by maximizing the first term in Eq. (1). After convergence, we used the trained parameters as the initialization to optimize the first and second terms of Eq. (1) to find prior network and fine-tune inference and discriminative networks.

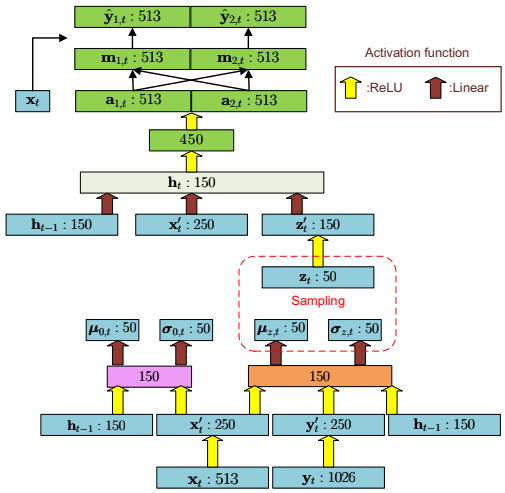

Figure 4: *Topology for implementing VRNN.*

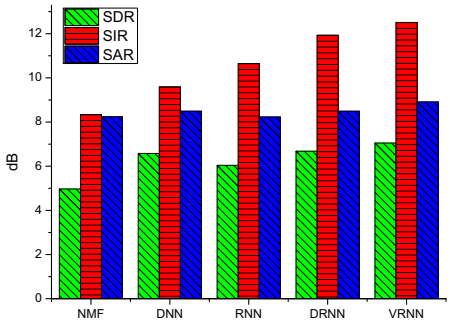

Figure 5: *Comparison of SDR, SIR and SAR by using different methods.*

4.2. Experimental results

We carried out the nonnegative matrix factorization (NMF) [5], DNN [10], RNN [12], discriminative RNN (DRNN) [14] and the proposed VRNN for comparison. The same spectral input and masking function were employed in different methods. NMF was implemented with the number of bases selected from validation data. Using DNN, we realized a topology of 513-150-150-150-1026 with three *feedforward* hidden layers where each layer has 150 units. In implementation of RNN, two *recurrent* hidden layers were constructed in a topology of 513-150-150-1026 [14]. The DRNN in [14] was implemented in a topology of 513-150-150-1026 according to a discriminative objective which measured the within-source reconstruction errors as well as the between-source discrimination information. ReLU activation was used in DNN, RNN and DRNN. The regularization parameter in DRNN and the topology of DNN, DRNN and VRNN in Figure 4 were determined by validation data. Figure 5 shows the comparison of SDR, SIR and SAR of using NMF, DNN, RNN, DRNN and VRNN.

There are some findings from this comparison. Basically, four neural network methods consistently perform better than NMF in terms of SDR and SIR. The improvement in terms of SAR is limited. The recurrent neural networks using RNN, DRNN and VRNN achieve higher SIR than feedforward neural network using DNN. DNN slightly performs better than RNN but worse than DRNN and VRNN in terms of SDR and SAR. The discriminative learning in DRNN increases SDR, SIR and SAR when compared with DNN and RNN. The interference between two sources is reduced to improve the SIR of demixed signals. Nevertheless, the proposed VRNN outperforms the other methods in terms of SDR, SIR and SAR. In terms of SIR, VRNN achieves 12.51 which are higher than 8.34 using NMF, 9.59 using DNN, 10.65 using RNN, 11.93 using DRNN. These results indicate that the stochastic modeling in RNN using variational learning can compensate the deterministic assumption in conventional RNN for monaural source separation.

5. Conclusions

We have presented a new variational recurrent neural network to tackle the deterministic assumption in conventional recurrent neural network. This model was constructed by incorporating the variational auto-encoder into recurrent neural network. The random samples of hidden variables were obtained to reconstruct the regression outputs. The encoder for recognition of hidden variables and the decoder for generation of regression outputs were guided by a variational learning algorithm which maximized the variational lower bound of logarithm of conditional likelihood. A supervised learning was carried out for speech separation with two source signals. An error backpropagation algorithm with a soft-masking function was developed to estimate model parameters for feature extraction, prior network, inference network and discriminative network. The advantage of the proposed model was illustrated through the experiments on single-channel speech separation. It was shown that higher SDR, SIR and SAR were obtained when compared with nonnegative matrix factorization and other neural network models. In the future, the stochastic learning for recurrent neural network will be extended to learning for long short-term memory and other types of recurrent models. We will also apply this general solution to either acoustic model or language model for speech recognition.

6. References

[1] S. Srinivasan and D. Wang, "Robust speech recognition by integrating speech separation and hypothesis testing," *Speech Communication*, vol. 52, no. 1, pp. 72–81, 2010.

[2] J.-T. Chien and B.-C. Chen, "A new independent component analysis for speech recognition and separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1245–1254, 2006.

[3] S. Haykin and Z. Chen, "The cocktail party problem," *Neural Computation*, vol. 17, no. 9, pp. 1875–1902, 2005.

[4] J.-T. Chien and H.-L. Hsieh, "Convex divergence ICA for blind source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 302–313, 2012.

[5] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *Proc. of Annual Conference of International Speech Communication Association*, 2007, pp. 2614–2617.

[6] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[7] A. Ozerov, P. Philippe, F. Bimbot, and R. Gribonval, "Adaptation of Bayesian models for single-channel source separation and its application to voice/music separation in popular songs," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1564–1578, 2007.

[8] F. Weninger, J. Le Roux, J. R. Hershey, and S. Watanabe, "Discriminative NMF and its application to single-channel source separation," in *Proc. of Annual Conference of the International Speech Communication Association*, 2014, pp. 865–869.

[9] 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.

[10] E. M. Grais, M. U. Sen, and H. Erdogan, "Deep neural networks for single channel source separation," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 3734–3738.

[11] F. Weninger, J. R. Hershey, J. Le Roux, and B. Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation," in *Proc. of IEEE Global Conference on Signal and Information Processing*, 2014, pp. 577–581.

[12] 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.

[13] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, 2014.

[14] 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.

[15] 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.

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

[17] 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.

[18] S. Watanabe and J.-T. Chien, *Bayesian Speech and Language Processing*. Cambridge University Press, 2015.

[19] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[20] Y. Bengio, E. Thibodeau-Laufer, G. Alain, and J. Yosinski, "Deep generative stochastic networks trainable by backprop," in *Proc. of International Conference on Machine Learning*, 2014.

[21] Y. Miao, C. Ox, L. Yu, and P. Blunsom, "Neural variational inference for text processing," in *Proc. of International Conference on Machine Learning*, 2016, pp. 1727–1736.

[22] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Advances in Neural Information Processing Systems*, 2015, pp. 2980–2988.

[23] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proc. of International Conference on Machine Learning*, 2014, pp. 1278–1286.

[24] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

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

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