Interspeech 2018
2-6 September 2018, Hyderabad

Joint Learning using Denoising Variational Autoencoders for Voice Activity Detection

Youngmoon Jung, Younggwan Kim, Yeunju Choi, Hoirin Kim

School of Electrical Engineering, KAIST, Daejeon, South Korea
{dudans,cleanthink,wkadldppdy,hoirkim}@kaist.ac.kr

Abstract

Voice activity detection (VAD) is a challenging task in very low signal-to-noise ratio (SNR) environments. To address this issue, a promising approach is to map noisy speech features to corresponding clean features and to perform VAD using the generated clean features. This can be implemented by concatenating a speech enhancement (SE) and a VAD network, whose parameters are jointly updated. In this paper, we propose denoising variational autoencoder-based (DVAE) speech enhancement in the joint learning framework. Moreover, we feed not only the enhanced feature but also the latent code from the DVAE into the VAD network. We show that the proposed joint learning approach outperforms conventional denoising autoencoder-based joint learning approach.

Index Terms: voice activity detection, speech enhancement, joint learning, joint training, denoising variational autoencoders

1. Introduction

Voice activity detection (VAD), the process of classifying a frame into speech or non-speech, is an important module in many speech applications such as speech coding, automatic speech recognition (ASR), speech enhancement (SE), speaker recognition, and speaker diarization.

Most of the early VAD approaches were based on raw acoustic features, including time domain energy, pitch, and zero-crossing rate. Another type of conventional VAD methods is a statistical model-based approach in which the distributions of speech and noise frames are modeled by Gaussian distributions in discrete Fourier transform (DFT) domain and the likelihood ratio is used to decide whether a frame is speech or non-speech [1]. Later, machine learning-based methods, such as support vector machine (SVM) and hidden Markov model (HMM) were applied for VAD. Recently, deep learning architectures, such as fully connected deep neural networks (DNNs) [2], convolutional neural networks (CNNs) [3] and Long Short-Term Memory (LSTM) recurrent neural networks [4] have achieved tremendous success in VAD, which have become popular for VAD modeling.

Despite the ongoing development over the years, VAD is still challenging in very low signal-to-noise ratio (SNR). To improve the robustness against noisy environments, we employ a joint learning method for VAD. The joint learning of a speech enhancement and a voice activity detection DNN was first introduced in [5] which shows that the joint learning approach yields better results for VAD. This approach was motivated by several previous works for noise robust speech recognition [6, 7, 8].

In this work, we extend the existing joint learning method in three ways: Firstly, we employ batch normalization [9] to reduce the internal covariate shift during training. In [10], it is already proven that the batch normalization is effective in reducing the internal covariate shift for the joint learning approach in speech recognition tasks. We show that this is also true for VAD tasks. Secondly, the parameter updates of the SE network depend not only on the SE cost function but also on the VAD cost function, which is motivated by [10]. Because of this, the front-end is able to provide enhanced features which is more suitable for the subsequent VAD task. Finally, we apply a denoising variational autoencoder (DVAE) for speech enhancement. The DVAE maps noisy features to a latent code and then reconstructs clean features by decoding the latent code. We feed not only the enhanced feature but also the latent code into the VAD network. Experimental results show that the proposed approach outperforms the conventional joint learning-based method.

The rest of this paper is organized as follows. Section 2 describes the variational autoencoder (VAE) and the proposed architecture. Section 3 introduces our joint learning approach. The experimental setup is described in Section 4. The results and analysis are provided in Section 5. We conclude this work in Section 6.

2. Model

2.1. Variational Autoencoder

The variational autoencoder (VAE) [11] is a latent variable generative model, which couples the approach of variational inference with deep learning. Here the latent variable generative model $p_\theta(\mathbf{x}|\mathbf{z})$ (also called decoder) for observed variable $\mathbf{x}$ is parametrized by a deep neural network with parameters θ. An inference model $q_\phi(\mathbf{z}|\mathbf{x})$ (also called encoder) is parametrized by a second deep neural network with parameters ϕ. A latent variable $\mathbf{z}$ is defined to embed the compressed information of the data $\mathbf{x}$, and the encoder maps a data space into its corresponding latent space. The decoder reconstructs the data from a sample point in the latent space. The parameters, θ and ϕ, are jointly learned by maximizing the variational lower bound $\mathcal{L}(\theta, \phi; \mathbf{x})$ of the log marginal likelihood with

$$\begin{aligned}\log p_\theta(x) &= D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})) + \mathcal{L}(\theta, \phi; \mathbf{x}) \\ &\geq \mathcal{L}(\theta, \phi; \mathbf{x}) \\ &= -D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) + \mathrm{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log(p_\theta(\mathbf{x}|\mathbf{z})]\end{aligned} \tag{1}$$

In the VAE framework of this paper, both the encoder and the decoder are parametrized using diagonal Gaussian distributions, which are $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \mu_\mathbf{z}, \sigma^2_\mathbf{z}\mathbf{I})$ and $p_\theta(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}; \mu_\mathbf{x}, \sigma^2_\mathbf{x}\mathbf{I})$, respectively. The prior is assumed to be an isotropic Gaussian distribution $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$ that lacks free parameters.

To yield a differentiable network after sampling, we use the reparameterization trick in which the random variable $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$ is reparametrized as a deterministic variable $\mathbf{z} = \mu_\mathbf{z} + \sigma_\mathbf{z} \odot \epsilon$, where $\odot$ denotes an element-wise product and an (auxiliary) noise variable ϵ is sampled from $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Modelling the

10.21437/Interspeech.2018-1151

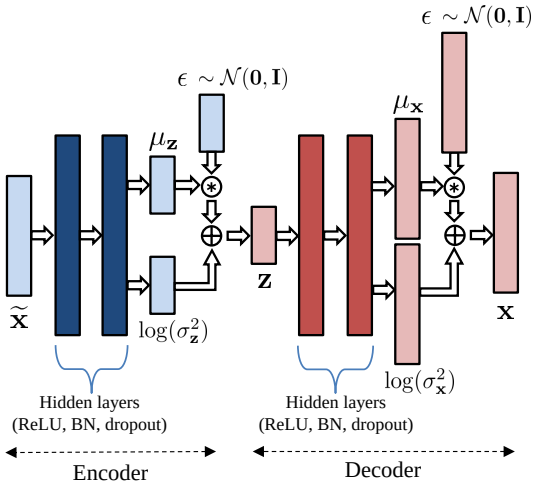

Figure 1: *The denoising variational autoencoder architecture for speech enhancement (SE-DVAE).*

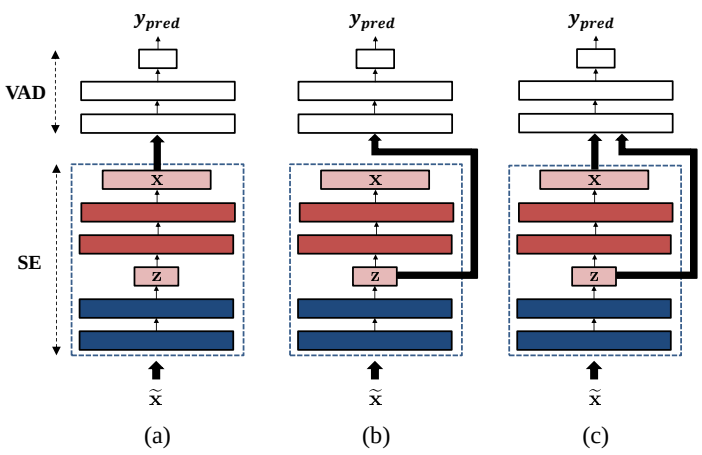

Figure 2: *Three types of proposed joint learning methods: (a) JL-DVAE-1, (b) JL-DVAE-2, and (c) JL-DVAE-3. The dotted boxes represent the SE-DVAE architecture which is shown in the Figure 1. The thick arrows indicate the input of the SE and VAD network.*

latent variable in this way allows the KL divergence in (1) to be integrated analytically, resulting in the following estimator:

$$\mathcal{L}(\theta, \phi; \mathbf{x}) \simeq \sum_{j=1}^{J} (1 + \log(\sigma^2_{z_j}) - \mu^2_{z_j} - \sigma^2_{z_j}) - \sum_{i=1}^{D} \frac{1}{2} \log(\sigma^2_{x_i}) + \frac{(x_i - \mu_{x_i})^2}{2\sigma^2_{x_i}} \quad (2)$$

where J and D are the dimensionalities of $\mathbf{z}$ and $\mathbf{x}$, respectively, and x_i is the i-th element of the vector $\mathbf{x}$. μ_{x_i} and σ_{x_i} denote the i-th element of the vector $\mu_{\mathbf{x}}$ and $\sigma_{\mathbf{x}}$. Likewise, μ_{z_j} and σ_{z_j} denote the j-th element of the vector $\mu_{\mathbf{z}}$ and $\sigma_{\mathbf{z}}$. For more detailed derivation of the above equation, please refer to the appendix in [11].

2.2. Proposed Architecture

In this work, we present a denoising variational autoencoder (DVAE) [12] framework that introduces a denoising process in training the VAE by using noisy-clean speech pairs. The training procedure is similar to how the vanilla denoising autoencoder (DAE) is trained. The input is corrupted according to some noise distribution and the model needs to learn to reconstruct the original input (e.g., by maximizing the log-probability of the clean input $\mathbf{x}$, given the corrupted input $\widetilde{\mathbf{x}}$). This procedure is akin to the regular VAE except that the input is corrupted.

In [13], the authors show the results of reconstructing the filter-bank features using VAE and AE. It is clear that VAE reconstructs better than AE. The VAE preserves the clearer harmonic structure and spectral envelope, while the AE provides more blurred results. This motivated us to apply the DVAE to speech enhancement instead of the DAE which is employed in the conventional joint learning approach.

The structure of the speech enhancement DVAE (SE-DVAE) is shown in Figure 1. The encoder takes a noisy speech feature $\widetilde{\mathbf{x}}$ as input and predicts 64-dimensional mean $\mu_{\mathbf{z}}$ and log-variance $\log(\sigma^2_{\mathbf{z}})$ that parametrize the posterior distribution $q_\phi(\mathbf{z}|\widetilde{\mathbf{x}})$. The decoder takes sampled $\mathbf{z}$ as input, and predicts the mean $\mu_{\mathbf{x}}$ and the log-variance $\log(\sigma^2_{\mathbf{x}})$ that parametrize the conditional likelihood $p_\theta(\mathbf{x}|\mathbf{z})$. As in the case of $\mathbf{z}$, the enhanced feature $\mathbf{x} \sim p_\theta(\mathbf{x}|\mathbf{z})$ is reparametrized as $\mathbf{x} = \mu_{\mathbf{x}} + \sigma_{\mathbf{x}} \odot \epsilon$ using the reparameterization trick.

The encoder and decoder DNNs both consist of two hidden layers of 2048 units. All the hidden layers use ReLU activations and no activation function is applied to Gaussian parameter layers. In order to guarantee a stable optimization of the DVAE, we put a constraint on the value of $\log(\sigma^2_{x_i})$ to be greater than a certain threshold α. This is because if $\sigma^2_{x_i}$ of Eq. (2) is close to zero, the DVAE loss (which is the negative variational lower bound) becomes close to infinity. We solve this problem by using the shifted ReLU with activation $f(x) = max(x, \alpha)$ for $\log(\sigma^2_{x_i})$. We set α to -9, which makes $\sigma^2_{x_i}$ greater than or equal to 10^{-4}. The SE-DVAE is fed with 21 consecutive frames and predicts 21 consecutive frames of enhanced features.

Batch normalization (BN) and dropout are used at every hidden layers except for the Gaussian parameter layers. As discussed in Introduction, it is known that BN has a great effect on the joint learning. When we jointly train the architecture, the output distribution of the SE network (i.e., the input distribution of the VAD network) changes significantly during the training process. This problem called internal covariate shift makes it difficult to train the entire networks. The VAD module would have to deal with an input distribution that is non-stationary and unnormalized. Thanks to BN, we are able to reduce internal covariate shift, especially at the boundary between two modules, and effectively train the whole network without pre-training.

3. Joint Learning

A joint DNN is built by concatenating an SE-DVAE and a VAD-DNN. Here, we propose three kinds of joint learning methods as shown in Figure 2 (a), (b), and (c). The input to the SE-DVAE is the noisy features $\widetilde{\mathbf{x}}$, surrounded by a context window. To reconstruct the corresponding clean features $\mathbf{x}$, the SE-DVAE is trained on parallel $\widetilde{\mathbf{x}}$ and $\mathbf{x}$ to minimize the SE loss which is the negative variational lower bound. The VAD-DNN is fed by the enhanced feature (shown in Figure 2 (a)), latent code $\mathbf{z}$ (shown in Figure 2 (b)), or both of them (shown in Figure 2 (c)) from the SE-DVAE. After that, the VAD-DNN makes a frame-wise binary speech / non-speech prediction y_{pred} and is trained

to minimize the cross entropy criterion. The input is batch normalized before feeding into the VAD-DNN. The VAD-DNN has 2 hidden layers, each of which has 2048 units with ReLU activations. Like the SE-DVAE, we apply BN and dropout to every hidden layers. The joint learning procedure can be summarized as follows:

1. Compute the loss functions at the output of the SE-DVAE and the VAD-DNN.
2. Compute the cost gradients using backpropagation.
3. Update the parameters of the SE-DVAE and the VAD-DNN.

In step 2, the VAD gradient is also back-propagated through the SE-DVAE. Therefore, the parameter updates of the SE-DVAE depend not only on the SE cost function but also on the VAD cost function, as shown below:

$$\theta_{SE} \leftarrow \theta_{SE} - \alpha_1 * [g_{SE} + \lambda g_{VAD}] \quad (3)$$

In Eq. (3), θ_{SE} are the parameters of the SE-DVAE, g_{SE} are the SE cost gradients with respect to θ_{SE}, while g_{VAD} are the VAD cost gradients with respect to θ_{SE}. Finally, λ is a hyperparameter which weights g_{VAD} and α_1 is the learning rate for θ_{SE}. Since the enhancement process is partly guided by the VAD cost function, the front-end would hopefully be able to provide the enhanced feature which is more suitable and discriminative for the subsequent VAD task. The parameter updates of the VAD-DNN only depend on the VAD cost function, as shown below:

$$\theta_{VAD} \leftarrow \theta_{VAD} - \alpha_2 * g_{VAD} \quad (4)$$

In Eq. (4), θ_{VAD} are the parameters of the VAD-DNN, g_{VAD} are the VAD cost gradients with respect to θ_{VAD}, and α_2 is the learning rate for θ_{VAD}. Notice that g_{VAD} in Eq. (4) differs from g_{VAD} in Eq. (3).

4. Experimental Setup

4.1. Datasets

We used clean utterances of the Aurora4 database [14] which contains 7138 continuous speech utterances for training and 330 utterances for testing. To construct the 35 hours training set, all the 7138 utterances of the clean training set were used. The utterances of the Aurora4 corpus are short and around 80% of which are speech; this may introduce a bias when comparing the distributions of speech and non-speech. To reduce this effect, one second of silence were inserted at the begining and the end of the utterance, which makes the ratio of speech frames around 60%. The clean speech corpus was corrupted by the public 100 noise types [1] at SNR levels varying in -5dB, 0dB, 5dB, 10dB, 15dB, 20dB. For the test data, all the 330 utterances of the clean utterances were used. They were corrupted by four unseen noises (babble, factory, destroyer-engine and F16 cockpit noise) in the NOISEX-92 noise corpus [15] at four low SNR levels : -5dB, 0dB, 5dB, 10dB. We applied Sohn VAD [1] to the clean speech corpus and the results were used as labels of the corresponding noisy corpus. This method was proved to be sufficiently reasonable to generate labels [16].

[1] web.cse.ohio-state.edu/pnl/corpus/HuNonspeech/HuCorpus.html The noise types (100) : Crowd (17), Machine (12), Alarm and siren (14), Traffic and car (3), Animal sound (9), Water sound (14), Wind (9), Bell (4), Cough (3), Clap (1), Snore (1), Click (1), Laugh (3), Yawn (2), Cry (1), Shower (1), Tooth brushing (1), Footsteps (1), Door moving (2), Phone dialing (1). In parenthesis is the number of distinct noise files.

4.2. Features

The features used are 32-dimensional multi-resolution cochleagram (MRCG) features, surrounded by a context of 10 past frames and 10 future frames. MRCG features were first proposed for a speech separation problem [17] and later employed for a VAD task [18]. For all models used in our experiments, delta and delta-delta coefficients are appended to create 96-dimensional features, which is the same setting as in [16].

4.3. Optimization

All models are trained using Adam optimizer with a mini-batch size of 512. The learning rates α_1 and α_2 start from 10^{-5} and 10^{-6} respectively. When the validation performance does not increase after one epoch, learning rates decrease to half. All the weights of the networks are initialized with Xavier initialization in [19]. We use a constant dropout rate of 0.2. The gradient weighting factor λ in Eq. (3) is set to 0.1.

5. Results

5.1. Comparison among different approaches

Table 1 lists the results of six different methods for the four unseen noises with different SNRs. The area under the ROC curve (AUC) [20] is adopted as the evaluation metric. The values in bold indicate the best results among all compared methods under each condition. Here we consider three baseline approaches (DNN, JL-DNN and JL-DAE) and the three proposed approaches (JL-DVAE-1, JL-DVAE-2 and JL-DVAE-3).

DNN denotes the conventional DNN-based VAD without joint learning scheme. It has the same network structure as our proposed VAD-DNN architecture in Section 2.2.

JL-DNN denotes the conventional denoising autoencoder-based joint learning approach which was proposed in [5]. In this method, after training the SE-DNN and VAD-DNN separately, we concatenate them and jointly train the whole network. Among the various configurations in the paper, we choose JT-DNN with 2+2 configuration (2 hidden layers for the SE-DNN and 2 hidden layers for the VAD-DNN with 2048 hidden nodes) without post-processing. We follow the same training procedure used in [5].

To compare the DVAE with the DAE, we replace the DVAE with the DAE in JL-DVAE-1 approach (which we refer to as JL-DAE). JL-DAE has the same structure as our proposed JL-DVAE-1 except for the two Gaussian parameter layers, which are replaced by fully-connceted layers of 64 and 2016 units, respectively. JL-DVAE-1 is shown in Figure 2 (a) where the VAD-DNN is fed by the enhanced feature. JL-DVAE-2 and JL-DVAE-3 are illustrated in Figure 2 (b) and (c), respectively.

As can be seen in the table, DNN shows lower performance than other five joint learning-based VADs in all noise conditions, especially at low SNRs. JL-DNN, which shows the lowest performance among all the joint learning methods, provides 1.9% relative improvement over DNN at SNR = -5 dB on average for all noises. These results indicate that the speech enhancement front-end is beneficial for the VAD task.

By comparing JL-DNN and JL-DAE, we observe that JL-DAE performs better than JL-DNN in almost all noise conditions. JL-DAE provides 0.5% relative improvement over JL-DNN at SNR = -5 dB on average for all noises. Even though both are denoising autoencoder-based joint learning approaches, their structures and learning methods are different.

JL-DVAE-1 provides 1.1% relative improvement over JL-

Table 1: *AUC (%) comparison between the conventional approaches and the proposed joint learning approaches.*

Noise	SNR	DNN	JL-DNN	JL-DAE	JL-DVAE-1	JL-DVAE-2	JL-DVAE-3
Babble	-5	85.92	87.78	88.38	89.15	88.72	**89.85**
	0	93.99	94.00	94.58	94.87	94.68	**95.10**
	5	97.12	97.14	97.46	97.39	97.22	**97.52**
	10	98.02	98.29	98.35	98.30	98.28	**98.34**
	Avg.	93.76	94.30	94.69	94.93	94.73	**95.20**
Factory	-5	77.43	81.08	82.27	83.95	83.68	**84.22**
	0	87.50	90.16	91.25	91.29	91.26	**92.75**
	5	94.95	95.19	96.26	95.96	95.68	**96.78**
	10	97.31	97.26	97.75	97.41	97.30	**97.49**
	Avg.	89.29	90.92	91.88	92.15	91.98	**92.81**
Destroyer engine	-5	92.85	93.07	93.24	94.01	93.38	**94.12**
	0	96.53	96.60	96.93	96.72	96.42	**96.81**
	5	97.37	97.96	97.85	97.57	97.50	**97.64**
	10	98.05	98.46	98.29	98.43	98.32	**98.46**
	Avg.	96.20	96.52	96.58	96.68	96.41	**96.76**
F16 cockpit	-5	90.67	91.43	91.20	92.04	91.31	**92.14**
	0	95.51	96.14	96.06	96.14	96.11	**96.20**
	5	97.20	97.69	97.73	97.62	97.60	**97.71**
	10	97.84	98.27	98.25	98.25	98.23	**98.30**
	Avg.	95.30	95.88	95.81	96.01	95.81	**96.09**

Table 2: *AUC (%) comparison of the three proposed joint learning methods on average for all noise types.*

SNR	JL-DVAE-1	JL-DVAE-2	JL-DVAE-3
-5	89.79	89.27	**90.08**
0	94.76	94.62	**95.21**
5	97.13	97.00	**97.41**
10	98.10	98.03	**98.15**
Avg.	94.95	94.73	**95.21**

Table 3: *AUC (%) comparison with and without batch normalization for JL-DVAE-3.*

SNR	$\lambda = 0$		$\lambda = 0.1$	
	no BN	with BN	no BN	with BN
-5	87.56	89.47	88.12	**90.08**
0	93.01	94.81	93.09	**95.21**
5	96.33	97.14	96.50	**97.41**
10	97.14	98.12	97.37	**98.15**
Avg.	93.51	94.89	93.75	**95.21**

DAE at SNR = -5 dB on average for all noises. These results indicate that we can achieve better performance by replacing the DAE with the DVAE for speech enhancement, especially in very low SNR conditions. As we expected in Section 2.2, we can see that the DVAE performs better than the DAE for reconstructing the clean features. To summarize, our proposed approach (JL-DVAE-1) shows higher performance than the baselines. Both JL-DVAE-2 and JL-DVAE-3 will be discussed in Section 5.2.

5.2. Comparison among the proposed approaches

Table 2 compares the results of three proposed approaches on average for all noise types. The results show that JL-DVAE-1 achieves higher performance than JL-DVAE-2 in all conditions and JL-DVAE-3 is consistently better in all conditions achieving 0.3% and 0.5% relative improvement (averaging the four SNR levels) compared to JL-DVAE-1 and JL-DVAE-2, respectively.

This implies that the enhanced feature and the latent representation $\mathbf{z}$ from the SE-DVAE complement each other and using those two features together is better than using only the enhanced feature. The learned latent representation $\mathbf{z}$ captures the factors that result in the variability of speech segments, such as the content being spoken, speaker identity, and environment. [13]. This would provide additional information to the VAD-DNN, which is useful to discriminate speech and non-speech.

5.3. Impact of batch normalization

Table 3 shows the impact of batch normalization (BN) on the joint learning. It is clear that BN is particularly helpful as explained in Section 2.2. When the gradient weighting factor λ in Eq. (3) is set to zero, JL-DVAE-3 with BN provides 1.5% relative improvement over JL-DVAE-3 without BN on average over all SNR levels. Likewise, when the gradient weighting factor λ is set to 0.1, JL-DVAE-3 with BN provides 1.6% relative improvement over JL-DVAE-3 without BN on average over all SNR levels.

5.4. Impact of the gradient weighting

The impact of the gradient weighting on the joint learning is shown in Table 3 as well. As explained in Section 3, the parameter updates of speech enhancement depend on the VAD cost function (if λ is not zero). We compare the two cases, $\lambda = 0.1$ and $\lambda = 0$. When the batch normalization is applied, JL-DVAE-3 with $\lambda = 0.1$ provides 0.3% relative improvement over JL-DVAE-3 with $\lambda = 0$ on average over all SNR levels. Likewise, when the batch normalization is not applied, JL-DVAE-3 with $\lambda = 0.1$ provides 0.3% relative improvement over JL-DVAE-3 with $\lambda = 0$ on average over all SNR levels.

6. Conclusions

This study was motivated by the result that VAD tasks are challenging in very low SNR. To overcome this problem, we employed a denoising variational autoencoder-based joint learning with batch normalization and the gradient weighting. We showed that our joint learning method performs better than the conventional denoising autoencoder-based joint learning method. As for the future work, we will focus on enabling joint learning without paired noisy-clean training data.

7. Acknowledgements

This material is based upon work supported by the Ministry of Trade, Industry and Energy (MOTIE, Korea) under Industrial Technology Innovation Program (No.10063424, Development of distant speech recognition and multi-task dialog processing technologies for in-door conversational robots).

8. References

[1] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Processing Letters*, vol. 6, no. 1, pp. 1–3, 1999.

[2] N. Ryant, M. Y. Liberman, J. Yuan, N. Ryant, M. Y. Liberman, and J. Yuan, "Speech activity detection on YouTube using deep neural networks," in *Proceedings of Interspeech 2013*, 2013, pp. 728–731.

[3] S. Thomas, S. Ganapathy, G. Saon, and H. Soltau, "Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 2519–2523.

[4] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, "Real-life voice activity detection with LSTM recurrent neural networks and an application to hollywood movies," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 483–487.

[5] Q. Wang, J. Du, X. Bao, Z. R. Wang, L. R. Dai, and C. H. Lee, "A universal VAD based on jointly trained deep neural networks," in *Proceedings of Interspeech 2015*, 2015, pp. 2282–2286.

[6] A. Narayanan and D. Wang, "Joint noise adaptive training for robust automatic speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 2504–2508.

[7] J. Du, Q. Wang, T. Gao, Y. Xu, L. Dai, and C.-h. Lee, "Robust speech recognition with speech enhanced deep neural networks," in *Proceedings of Interspeech 2014*, Sep. 2014, pp. 616–620.

[8] T. Gao, J. Du, L.-r. Dai, and C.-h. Lee, "Joint training of front-end and back-end deep neural networks for robust speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4375–4379.

[9] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of International Conference on Machine Learning (ICML 2015)*, 2015, pp. 448–456.

[10] M. Ravanelli, P. Brakel, M. Omologo, and Y. Bengio, "Batch-normalized joint training for DNN-based distant speech recognition," in *2016 IEEE Workshop on Spoken Language Technology (SLT)*, 2017, pp. 28–34.

[11] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proceedings of International Conference on Learning Representations (ICLR 2014)*, 2014.

[12] D. J. Im, S. Ahn, R. Memisevic, and Y. Bengio, "Denoising criterion for variational auto-encoding framework," in *AAAI*, 2015, pp. 2059–2065.

[13] W.-N. Hsu, Y. Zhang, and J. Glass, "Learning latent representations for speech generation and transformation," in *Proceedings of Interspeech 2017*, 2017, pp. 1273–1277.

[14] D. Pearce and J. Picone, "Aurora working group: DSR front end LVCSR evaluation AU/384/02," *Institue for Signal and Information Process, Mississippi State University, Technical Roport*, 2002.

[15] A. Varga and H. J. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech communication*, vol. 12, no. 3, pp. 247–251, 1993.

[16] X.-L. Zhang and D. Wang, "Boosting contextual information for deep neural network based voice activity detection," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 24, no. 2, pp. 252–264, 2016.

[17] J. Chen, Y. Wang, and D. Wang, "A feature study for classification-based speech separation at low signal-to-noise ratios," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 22, no. 12, pp. 1993–2002, 2014.

[18] X.-L. Zhang and D. Wang, "Boosted deep neural networks and multi-resolution cochleagram features for voice activity detection," in *Proceedings of Interspeech 2014*, 2014, pp. 1534–1538.

[19] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010, pp. 249–256.

[20] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve," *Radiology*, vol. 143, pp. 29–36, 1982.