한국음향학회지 제40권 제6호 pp. 578~586 (2021)
The Journal of the Acoustical Society of Korea Vol.40, No.6 (2021)
https://doi.org/10.7776/ASK.2021.40.6.578

pISSN : 1225-4428
eISSN : 2287-3775

변분 오토인코더와 비교사 데이터 증강을 이용한 음성인식기 준지도 학습

Semi-supervised learning of speech recognizers based on variational autoencoder and unsupervised data augmentation

조현호,[1] 강병옥,[2] 권오욱[1†]

(Hyeon Ho Jo,[1] Byung Ok Kang,[2] and Oh-Wook Kwon[1†])

[1]충북대학교 지능로봇공학과, [2]한국전자통신연구원 인공지능연구소

(Received September 28, 2021; accepted November 4, 2021)

초 록: 종단간 음성인식기의 성능향상을 위한 변분 오토인코더(Variational AutoEncoder, VAE) 및 비교사 데이터 증강(Unsupervised Data Augmentation, UDA) 기반의 준지도 학습 방법을 제안한다. 제안된 방법에서는 먼저 원래의 음성데이터를 이용하여 VAE 기반 증강모델과 베이스라인 종단간 음성인식기를 학습한다. 그 다음, 학습된 증강모델로부터 증강된 데이터를 이용하여 베이스라인 종단간 음성인식기를 다시 학습한다. 마지막으로, 학습된 증강모델 및 종단간 음성인식기를 비교사 데이터 증강 기반의 준지도 학습 방법으로 다시 학습한다. 컴퓨터 모의실험 결과, 증강모델은 기존의 종단간 음성인식기의 단어오류율(Word Error Rate, WER)을 개선하였으며, 비교사 데이터 증강 학습방법과 결합함으로써 성능을 더욱 개선하였다.
핵심용어: 종단간 음성인식, 변분 오토인코더, 데이터 증강, 준지도 학습

ABSTRACT: We propose a semi-supervised learning method based on Variational AutoEncoder (VAE) and Unsupervised Data Augmentation (UDA) to improve the performance of an end-to-end speech recognizer. In the proposed method, first, the VAE-based augmentation model and the baseline end-to-end speech recognizer are trained using the original speech data. Then, the baseline end-to-end speech recognizer is trained again using data augmented from the learned augmentation model. Finally, the learned augmentation model and end-to-end speech recognizer are re-learned using the UDA-based semi-supervised learning method. As a result of the computer simulation, the augmentation model is shown to improve the Word Error Rate (WER) of the baseline end-to-end speech recognizer, and further improve its performance by combining it with the UDA-based learning method.
Keywords: End-to-End ASR, Variational AutoEncoder (VAE), Data augmentation, Semi-supervised learning

PACS numbers: 43.72.Bs, 43.72.Ne

I. 서 론

최근 수년간 음향모델 및 언어모델 기반의 음성인식기를 하나의 모델로 학습하기 위한 종단간 음성인식(Automatic Speech Recognition, ASR) 기술이 연구되었다. 종단간 음성인식은 하나의 모델을 학습하여 입력 음성신호를 해당 텍스트로 직접 매핑한다. 종단간 음성인식은 일반적으로 인코더-디코더로 구성되며, 각 모듈을 학습하고 결합하는 과정 없이 단일 모델을 사용하기 때문에 학습과정이 간단하다. 종단간 음성인식은 DeepSpeech,[1] Listen And Spell(LAS),[2] 트랜스포머(Transformer)[3] 순서로 연구되었다. DeepSpeech, LAS

†**Corresponding author:** Oh-Wook Kwon (owkwon@cbnu.ac.kr)
Department of Intelligent Systems and Robotics, Chungbuk National University, Chungdae-ro 1, Seowon-gu, Cheongju 28644, Republic of Korea
(Tel: 82-43-261-3374, Fax: 82-43-268-2386)

는장단기 메모리(Long Short-Term Memory, LSTM) 또는 양방향 장단기 메모리(Bidirectional Long Short-Term Memory, BLSTM)를 기반으로 인코더-디코더 구조로 구성된다. 장단기 메모리 등의 회귀 구조의 단점은 모델의 계산량이 커서 많은 메모리를 필요로 하며, 학습 속도 또한 느리다. 이를 보완하기 위해서 트랜스포머가 제안되었다. 트랜스포머는 회귀 구조 없이 주의집중 및 순방향 층을 사용하여 인코더-디코더 구조를 구성한다.

종단간 음성인식의 성능을 높이기 위한 연구로는 준지도 학습이 있다. 준지도 학습은 소량의 레이블 된 데이터와 대량의 레이블 없는 데이터를 사용하여 기존 모델의 성능을 높이기 위한 방법이다. 준지도 학습은 많은 학습 데이터를 필요로 하는 종단간 음성인식의 문제점을 해결하고 성능을 높일 수 있다. 준지도 학습에 관한 연구로는 음성인식기와 음성합성기(Text-To-Speech, TTS)를 직렬로 결합하여 입력과 모델의 출력간의 손실을 최소화 하는 주기 일관성[4,5]연구가 있었다. 다음은 원본 음성과 증강된 음성 간의 분포 차이를 줄이는 일관성 손실함수를 학습에 반영한 비교사 데이터 증강(Unsupervised Data Augmentation, UDA)[6]이 제안되었다. 최근에는 자기 학습을 교사-학생[7] 메커니즘으로 학습하는 잡음 학생[8] 방법이 제안되었다.

종단간 음성인식의 성능을 높이기 위한 다른 연구로 데이터 증강 방법이 있다. 데이터 증강은 정답은 동일하지만 기존의 음성과는 다른 데이터를 생성하여 학습에 이용하는 방법이다. 속도 섭동[9]은 음성 신호의 속도를 변화시켜서 새로운 음성 신호를 생성하는 증강 방법이다. 최근에는 종단간 음성인식기의 입력으로 사용되는 특징에 직접 적용이 가능한 데이터 증강 방법들이 제안되었다. 그 대표적인 방법에는 SpecAugment,[10] SpecSwap[11] 등이 있는데, 간단한 변환 방법으로도 높은 성능 향상을 보인다. 기존의 데이터 증강 방법들은 대부분 경험에 의해서 최적의 파라미터를 찾는 과정이 필요하다는 단점이 있다.

본 연구에서는 변분 오토인코더(Variational Auto-Encoder, VAE)[12] 기반의 데이터 증강 모델을 학습하여 기존의 종단간 음성인식기의 성능을 개선하는 방법을 제안한다. 변분 오토인코더 기반의 증강모델은 모델 스스로가 증강된 데이터를 생성하기 때문에 별도의 파라미터를 찾기 위한 실험이 불필요하다. 제안 방법을 비교사 데이터 증강 기반의 준지도 학습에 적용하여 한번 더 성능을 개선시킨다.

본 논문은 다음과 같이 구성된다. 2장에서는 실험에 사용한 베이스라인 음성인식 모델 및 비교사 데이터 증강 학습에 관해서 설명한다. 3장에서는 제안한 변분 오토인코더 기반 증강모델 및 디코더 고정을 적용한 비교사 데이터 증강 학습 방법에 대해서 설명한다. 4장에서는 실험에 사용한 데이터베이스, 실험환경, 실험결과에 대해서 설명한다. 5장에서 결론을 맺는다.

II. 기존 연구

2.1 트랜스포머 음성인식

종단간 음성인식 모델은 인코더-디코더 구조의 트랜스포머를 사용한다. Fig. 1은 학습에 사용한 트랜스포머의 구조이다.[3]

인코더는 입력된 특징벡터 $\mathbf{x}$ 를 고차원의 중간 벡터 표현 $\mathbf{e}$로 변환한다.

$$\mathbf{e} = \text{Encoder}(\mathbf{x}), \tag{1}$$

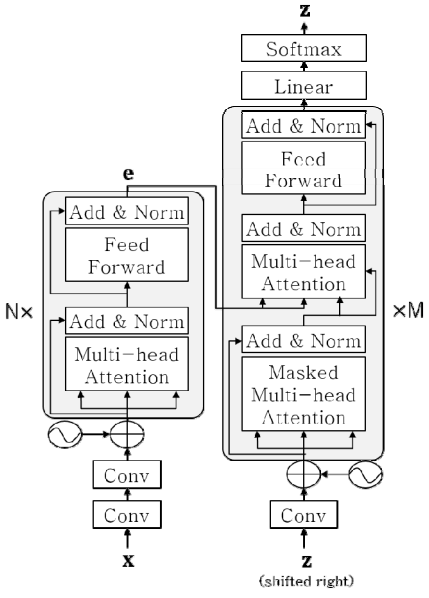

Fig. 1. Block diagram of the Transformer-based ASR model.

여기서 $\mathbf{x} = [x(1), x(2), ..., x(T)]$ 는 로그 멜-필터뱅크 특징이며, T 는 입력 음성의 프레임 개수이다. $\mathbf{e} = [e(1), e(2), ..., e(U)]$ 는 인코더의 출력 벡터이며, U 는 압축된 인코더 출력 벡터의 프레임 수다. N은 인코더 블록의 개수를 의미한다. 인코더의 출력 벡터는 디코더의 입력으로 사용된다.

디코더는 인코더의 출력벡터 $\mathbf{e}$를 입력으로 사용하여 매 시간 프레임에 대한 $\mathbf{z}$를 출력한다.

$$\mathbf{z} = \text{Decoder}(\mathbf{e}), \tag{2}$$

여기서 $\mathbf{z}$는 디코더 모델의 출력이며 각 토큰의 확률 분포를 의미한다. M은 디코더 블록의 개수를 의미한다. 트랜스포머의 인코더 및 디코더는 멀티헤드 주의집중 블록[3]을 사용한다.

2.2 비교사 데이터 증강

비교사 데이터 증강[6]은 이미지 및 텍스트 분류 문제를 위해 제안된 준지도 학습 방법이며 최근 음성인식분야에도 연구되고 있다. 비교사 데이터 증강은 레이블 없는 데이터에 데이터 증강 방법을 적용하여 증강된 데이터를 생성하고, 레이블 없는 데이터와 증강된 데이터의 모델 출력 분포 간의 차이가 최소가 되도록 모델을 학습시킨다.

비교사 데이터 증강은 2 단계의 학습 과정을 수행한다. 1 단계에서는 레이블된 데이터를 사용하여 베이스라인 모델을 학습한다. 2 단계에서는 레이블된 데이터로 기존의 교차 엔트로피 손실함수를 계산하고, 레이블 없는 데이터로 일관성 손실함수를 계산한다. 학습된 베이스라인 모델은 위의 두 손실함수를 결합한 최종 손실함수로 재학습된다. 비교사 데이터 증강은 증강된 데이터의 인식 결과가 원본 데이터의 인식 결과와 동일해지도록 학습하면서 다양한 변이에 강인한 모델을 생성한다.

Fig. 2는 음성인식을 위한 비교사 데이터 증강 학습 과정을 설명한다. 1 단계에서 레이블된 음성 데이터 $\mathbf{x}_L$과 전사 텍스트 $\mathbf{y}_L$을 사용하여 베이스라인 모델을 학습하며, 베이스라인 모델로 Fig. 1의 학습된 모델을 그대로 사용한다. 2 단계에서 베이스라인 모델은 $\mathbf{x}_L$, $\mathbf{y}_L$과 레이블 없는 음성 데이터 $\mathbf{x}_U$로 재학습된다. 교사 손실함수인 L_S는 $\mathbf{x}_L$에 대한 모델의 출력 분포인 $p_\theta(\hat{\mathbf{y}}_L \mid \mathbf{x}_L)$ 과 $\mathbf{y}_L$을 이용하여 다음과 같이 계산한다.

$$L_S = -\sum \mathbf{y}_L \log \mathbf{z}_L. \tag{3}$$

$$\mathbf{z}_L \sim p_\theta(\hat{\mathbf{y}}_L \mid \mathbf{x}_L). \tag{4}$$

$\hat{\mathbf{y}}_L$은 $\mathbf{x}_L$입력에 대한 음성인식 모델의 인식 결과를 나타내며, $\mathbf{z}_L$은 $p_\theta(\hat{\mathbf{y}}_L \mid \mathbf{x}_L)$ 로부터 얻어진 프레임 단위의 출력이다. L_S 계산에는 교차 엔트로피를 사용한다. $\mathbf{x}_U$에 데이터 증강을 적용하여 생성된 증강된 데이터 $\mathbf{x}_A$와 $\mathbf{x}_U$를 사용하여 일관성 손실함수 L_C를 다음과 같이 계산한다.

$$\begin{aligned} L_C &= D_{KL}(p_\theta(\hat{\mathbf{y}}_U \mid \mathbf{x}_U) \,\|\, p_\theta(\hat{\mathbf{y}}_A \mid \mathbf{x}_A)) \\ &= \sum \mathbf{x}_U \log \frac{\mathbf{x}_U}{\mathbf{x}_A}. \end{aligned} \tag{5}$$

$$\mathbf{z}_U \sim p_\theta(\hat{\mathbf{y}}_U \mid \mathbf{x}_U). \tag{6}$$

$$\mathbf{z}_A \sim p_\theta(\hat{\mathbf{y}}_A \mid \mathbf{x}_A). \tag{7}$$

$p_\theta(\hat{\mathbf{y}}_U \mid \mathbf{x}_U)$는 $\mathbf{x}_U$, $p_\theta(\hat{\mathbf{y}}_A \mid \mathbf{x}_A)$는 $\mathbf{x}_A$ 입력에 대한 음성인식 모델의 출력 분포를 의미하고, $\mathbf{z}_U$, $\mathbf{z}_A$는

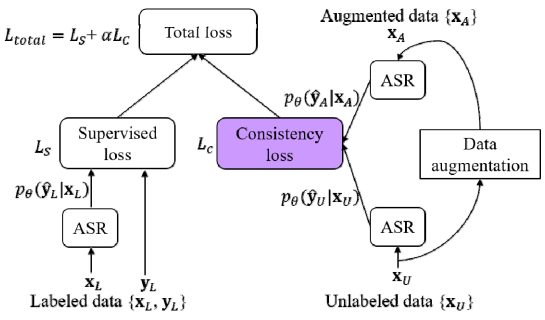

Fig. 2. (Color available online) Training objective for UDA, where ASR is a trained baseline model that predicts a distribution of $\hat{\mathbf{y}}_A$ given $\mathbf{x}_A$ and $\hat{\mathbf{y}}_U$ given $\mathbf{x}_U$.

$p_\theta(\hat{\mathbf{y}}_U \mid \mathbf{x}_U)$, $p_\theta(\hat{\mathbf{y}}_A \mid \mathbf{x}_A)$로부터 얻어진 프레임 단위의 출력이다. $\hat{\mathbf{y}}_U$는 입력 $\mathbf{x}_U$에 대한 모델의 인식 결과, $\hat{\mathbf{y}}_A$는 입력 $\mathbf{x}_A$에 대한 모델의 인식 결과, $D_{KL}(p_\theta(\hat{\mathbf{y}}_U \mid \mathbf{x}_U) \| p_\theta(\hat{\mathbf{y}}_A \mid \mathbf{x}_A))$은 두 확률 분포 $p_\theta(\hat{\mathbf{y}}_U \mid \mathbf{x}_U)$, $p_\theta(\hat{\mathbf{y}}_A \mid \mathbf{x}_A)$ 간의 KL 발산(Kullback-Leibler divergence)을 의미한다. ASR 모델은 L_S와 L_C를 결합한 최종 손실함수 L_{total}을 사용하여 재학습된다.

$$L_{total} = L_S + \alpha L_C, \tag{8}$$

여기서 α는 L_C의 결합 비율을 조절하기 위한 변수이며, 본 실험에서는 최적의 α를 찾기 위한 실험을 수행한다. 비교사 데이터 증강은 레이블 없는 데이터와 증강된 데이터의 인식 결과가 동일해지도록 학습한다. 입력 음성의 다양한 변이에 강인하게 모들을 학습시키는 효과가 있으며, 레이블 없는 데이터를 활용하여 낮은 비용으로 음성인식의 성능을 향상시킨다.

2.3 변분 오토인코더

변분 오토인코더[12]는 학습데이터의 분포와 유사한 분포를 갖는 새로운 데이터를 생성하는 오토인코더 기반의 생성모델이다. 변분 오토인코더는 변분법을 사용하는데, 변분법은 어떤 함수 $p(x)$의 극점을 찾는 문제에서 해당함수를 직접 계산하기 어려울 때, 쉽게 계산이 가능한 다른 함수 $q(x)$로 대체하여 근사적인 해를 구하는 방법이다.

변분 오토인코더의 구조는 인코더-디코더로 구성된다. 인코더는 입력 데이터 $\mathbf{x}$를 은닉 공간의 $\mathbf{z}$로 변환한다. 이때, $q_\phi(\mathbf{z} \mid \mathbf{x})$는 인코더의 출력분포이다. 디코더는 은닉 공간의 데이터 $\mathbf{z}$를 입력 공간의 데이터 $\mathbf{x}$로 변환한다. 이때, $p_\theta(\mathbf{x} \mid \mathbf{z})$는 디코더의 출력분포이다. 변분 오토인코더의 최종 목적은 원본 데이터를 복원하기 위해서 $p_\theta(\mathbf{x})$를 최대화하는 것이다.

변분 오토인코더의 학습방법은 다음과 같다. 로그우도 $\log p_\theta(\mathbf{x})$의 수식을 정리하면, KL발산은 0보다 같거나 크기 때문에 $\log p_\theta(\mathbf{x})$의 최소값은 증거하한(Evidence Lower BOund, ELBO) $L(\theta, \phi; x_i)$으로 주어진다.[12]

$$\begin{aligned}\log p_\theta(x_i) &= KL(q_\phi(z \mid x_i) \| p_\theta(z \mid x_i)) + L(\theta, \phi; x_i) \\ &\geq L(\theta, \phi; x_i).\end{aligned} \tag{9}$$

$$L(\theta, \phi; x_i) = \mathrm{E}_{q_\phi(z \mid x_i)}[\log p_\theta(x_i \mid z)] - KL(q_\phi(z \mid x_i) \| p_\theta(z)). \tag{10}$$

변분 오토인코더의 손실함수 L_A를 다음과 같이 정의하여 최소화 되도록 학습하게 되면, 증거하한 $L(\theta, \phi; x_i)$이 최대화 되고 따라서 $\log p_\theta(\mathbf{x})$가 최대화 된다.

$$\begin{aligned}L_A &= -L(\theta, \phi; x_i) \\ &= -\mathrm{E}_{q_\phi(z \mid x_i)}[\log p_\theta(x_i \mid z)] + KL(q_\phi(z \mid x_i) \| p_\theta(z)) \\ &= \frac{1}{2}\sum (x_i - x_i^{'})^2 + KL(q_\phi(z \mid x_i) \| p_\theta(z))\end{aligned}. \tag{11}$$

손실함수의 첫번째 항은 모델의 출력이 입력과 유사하도록, 두번째 항은 이상적인 샘플링을 위한 $q_\phi(\mathbf{z} \mid \mathbf{x}_i)$ 분포가 prior $p_\theta(\mathbf{z})$와 유사한 분포가 되도록 학습된다. 즉, 최대한 원본 음성과 유사하면서 이상적인 샘플링 함수의 값이 최대한 prior값과 같도록 만든다.

III. 제안 방법

3.1 변분 오토인코더 기반 증강 모델

제안 방법에서는 데이터 증강 모델로 변분 오토인코더를 사용한다. 학습은 2단계로 수행된다. 1단계에서 Fig. 3과 같이 학습데이터를 사용하여 변분 오토인코더 모델을 학습한다. 변분 오토인코더 모델은 입력 음성과 모델의 출력의 차이가 최소가 되면서 이상적인 샘플링 함수 $q_\phi(\mathbf{z} \mid \mathbf{x})$가 prior $p_\theta(\mathbf{z})$와 유사하도록 학습한다.

2단계에서는 생성된 증강 데이터 $\mathbf{x}^{'}$을 기존 데이터 $\mathbf{x}$에 추가하여 Fig. 1의 기존 음성인식 모델을 학습한다. 증강 모델로부터 생성된 데이터를 추가하여 더욱 많은 데이터로 모델을 학습하여 성능을 개선하는 효과를 기대한다.

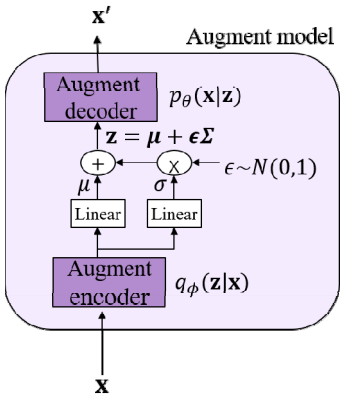

Fig. 3. (Color available online) Structural diagram of the variational autoencoder.

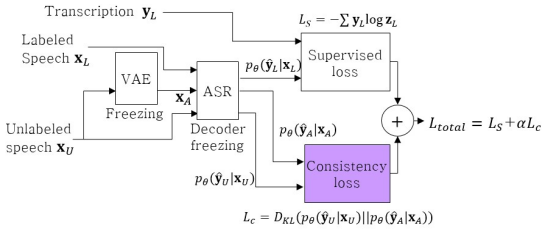

Fig. 4. (Color available online) Block diagram of the proposed UDA-based training method.

3.2 비교사 데이터 증강 학습법

Fig. 4는 학습된 데이터 증강 및 음성인식 모델을 이용한 비교사 데이터 증강 기반의 준지도 학습방법을 나타낸다. 미리 학습된 증강모델 및 음성인식 모델에 비교사 데이터 증강 학습 방법을 적용한다. 제안된 비교사 데이터 증강 학습 방법은 2.2절과 유사하지만 크게 2가지의 차이점이 있다. 데이터 증강으로 학습된 변분 오토인코더 증강모델을 고정시킨다. 이는 비교사 데이터 증강 학습 시 증강모델이 음성인식 모델과 동시에 학습되면서 잘못된 방향으로 학습되지 않도록 방지하기 위함이다.

또한, 비교사 데이터 증강 모델 학습 시에는 음성인식 모델의 디코더를 고정시킨다. 종단간 음성인식 모델에서 인코더는 음향모델, 디코더는 언어모델의 역할을 수행한다. 이미 레이블된 데이터로 학습된 음성인식 모델의 디코더를 고정시킴으로써 음향모델의 역할인 인코더를 잘 학습시키는 것에 집중하도록 한다. 증강된 데이터를 추가하여 인코더를 더 집중적으로 학습시키면 인코더에서 더 의미있는 음향정보를 추출하고 이는 음성인식 성능을 향상시킬 것으로 기대한다.

제안하는 두 가지 모델 고정기술을 바탕으로 2.2절의 비교사 데이터 증강 학습 방법에 따라서 음성인식 모델을 재학습한다.

IV. 실험결과 및 토의

4.1 데이터베이스

본 실험에서는 음성인식 실험에 많이 사용되는 WSJ[13]과 LibriSpeech[14] 데이터베이스를 사용하였다. WSJ의 학습데이터는 si84 15 h과 si284 81 h으로 구성된다. 평가 데이터는 eval93 1.1 h, 검증 데이터는 dev92 1.0 h으로 구성된다. LibriSpeech는 전체 학습 데이터는 960 h으로 train-clean-100 100 h과 train-clean-360 360 h, 그리고 train-other-500 500 h으로 구성된다. 검증 데이터는 dev-clean 5.1 h, 평가데이터는 test-clean 5.3 h으로 구성된다. UDA의 α에 따른 실험 결과를 확인하기 위해서 WSJ의 si84를 레이블된 데이터, si284를 레이블 없는 데이터로 사용하였다. 평가 및 검증에는 eval93, dev92를 사용하였다. 데이터 증강모델 실험을 위해서 LibriSpeech의 train-clean-100을 사용하였으며, 준지도 학습 실험에는 레이블된 데이터로 train-clean-100, 레이블 없는 데이터로 train-clean-360을 사용하였다. 두 실험 모두 평가 및 검증에는 test-clean, dev-clean의 2개 세트를 사용하였다.

4.2 실험

본 연구에서는 종단간 음성인식 모델의 실험환경을 제공하는 툴킷인 ESPnet[15]을 사용하였다. 음성인식 모델은 트랜스포머를 사용하며 인코더 블록 12개, 디코더 블록 6개로 구성된다. 각 멀티헤드 주의집중의 헤드는 8개, 각 층은 512 차원으로 구성된다. 배치크기는 64, 드롭아웃은 0.1, epoch 100을 사용하였다.

변분 오토인코더 모델은 음성인식의 트랜스포머 인코더 블록의 구조를 사용하여 인코더 1층, 디코더 1층을 사용하였다. 각 헤드는 4개, 각 층은 128층을 사용하였고 배치크기는 64, 드롭아웃은 0.1, epoch 50을 사용하였다.

언어모델(Language Model, LM)은 트랜스포머의 멀티-헤드 주의블록을 사용하며, 헤드 8개, 16층으로 구성된다.

4.3 실험 결과

평가 지표로 음성인식 모델의 출력 결과를 디코딩한 최종 인식 결과의 인식 단위별 인식결과인 문자오류율(Character Error Rate, CER)과 단어오류율(Word Error Rate, WER)를 측정하였다.

Table 1은 WSJ에서 α값의 변화에 따른 비교사 데이터 증강의 성능을 보여준다. WSJ의 데이터를 사용하여 최적의 α값을 찾기 위한 실험을 수행하였다. 실험 결과 α에 따라서 성능변화가 발생하는 것을 볼 수 있다. α = 5에서 비교사 데이터 증강의 성능이 가장 좋은 것을 볼 수 있었으며 베이스라인(α = 0) 대비 단어오류율이 3.8 % 감소하였다.

Fig. 5는 60 epoch 까지는 베이스라인 음성인식 모델을, 60 epoch 이후로는 비교사 데이터 증강의 학습 손실을 보여주는 그래프이다. 이때 α = 5에서 그래프를 그렸다. 그래프를 보면 비교사 데이터 증강의 추가 학습이 시작될 때 새로운 손실이 추가되면서 전체 손실 그래프가 베이스라인 대비 커지는 것을 볼 수 있다. 또한 비교사 데이터 증강 학습이 시작되면서 다시 손실이 줄어드는 것을 볼 수 있다.

α = 1에서 α = 5에서와 비슷한 경향의 그래프를 확인하였으며 전반적으로 α = 5 보다 손실값이 더 커지는 것을 확인하였다. 베이스라인에서 학습 손실의 최소값은 7.19, α = 1에서의 비교사 데이터 증강 학습에서는 6.61로 베이스라인 대비 8.1 %의 손실 감소가 있었다. α = 5에서의 비교사 데이터 증강 학습에서는 6.42로 베이스라인 대비 10.7 %의 손실 감소가 있었다. 즉, α = 5에서 α = 1 보다 손실값이 더 작으며, 학습이 잘 되었다고 볼 수 있다.

Table 2는 기존의 트랜스포머 기반 음성인식기에 제안한 증강모델과 비교사 데이터 증강 학습 방법을 순차적으로 적용하여 LibriSpeech 데이터베이스의 WER 성능을 측정한 표이다. M1은 기존의 음성인식기인 베이스라인 모델이다. M2는 학습된 증강모델을 추가하여 음성인식기를 재학습한 모델이다. M3는 학습된 증가모델 및 음성인식모델을 비교사 데이터 증강(α = 1)으로 학습한 모델이다. M4는 M3에 디코더 고정을 적용하여 비교사 데이터 증강 학습한 모델이다. M5는 학습된 증가모델 및 음성인식모델을 비교사 데이터 증강 (α = 5)로 학습한 모델이다. M6는 M5에 디코더 고정을 적용하여 비교사 데이터 증강 학습한 모델이다.

M1과 M2를 통해서 제안된 증강모델의 성능개선 효과를 확인할 수 있다. M1과 M2는 100 h의 레이블된 학습데이터를 사용하였다. M2의 결과는 베이스라인 M1 대비 상대적으로 2.9 %의 단어오류율 감소를 보였다. 변분 오토인코더로부터 생성되는 음성은 입력 음성과 유사하지만, 기존과 다른 왜곡된 음성

Table 1. WER (%) of the UDA-based training method with decoder freezing using the data augmentation model for the WSJ database.

α	0	0.1	0.2	1	5	10
CER	10.0	10.4	10.1	9.8	9.4	9.6
WER	15.7	17.1	16.7	15.7	15.1	15.3

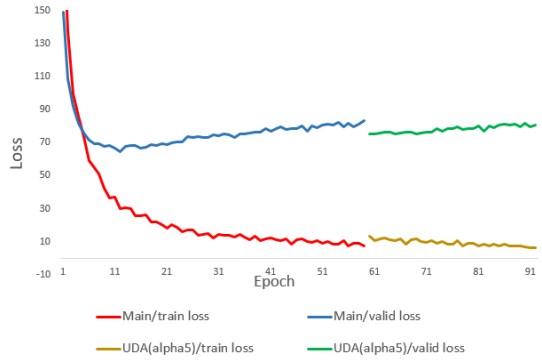

Fig. 5. (Color available online) Loss graph of the baseline model and UDA model.

Table 2. WER (%) of the UDA-based training method with decoder freezing using the data augmentation model for the LibriSpeech database.

Model	Test-clean	Dev-clean
Baseline (M1)	14.9	15.1
+VAE (M2)	14.5	14.8
+UDA (α = 1) (M3)	14.1	14.6
+Freezing (M4)	13.9	14.2
+UDA (α = 5) (M5)	14.0	14.4
+Freezing (M6)	13.9	14.1

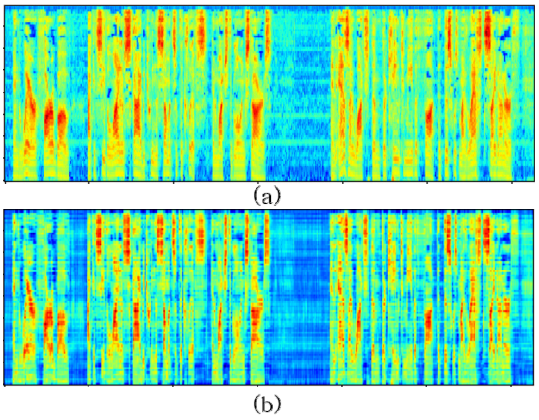

Fig. 6. (Color available online) (a) Spectrogram of the original speech data; (b) Spectrogram of the augmented speech data.

데이터가 된다. M2의 실험 결과에 의하면, 이러한 음성을 추가하여 더 많은 데이터로 학습하는 것이 성능향상에 도움이 된다는 것을 확인할 수 있다. M3와 M4는 제안된 비교사 데이터 증강 학습방법의 성능 개선 효과를 확인할 수 있다. M3와 M4는 100 h의 레이블된 데이터 및 360 h의 레이블 없는 데이터를 사용하였다. M3는 학습된 베이스라인 및 증강모델을 활용하여 비교사 데이터 증강 학습 시 성능이 개선됨을 보여준다. M3에 디코더 고정 학습을 추가한 M4는 단어오류율 13.9 %로 가장 좋은 성능을 보여준다. 이는 베이스라인 대비 상대적으로 6.7 %의 단어오류율 감소이다. 제안한 방법인 M2를 적용하고, 또다시 M4를 적용하여 제안한 방법이 음성인식 성능을 개선함을 볼 수 있다. 이는 디코더를 고정하는 학습방법이 음향모델의 역할을 하는 인코더를 집중적으로 학습하여 음향특성 더 잘 추출하게 학습된다는 기대에 부합한다. 마지막으로 M5, M6는 Table 1의 결과에서 찾은 최적의 $\alpha = 5$에서 비교사 데이터 증강 실험을 수행한 결과이다. Table 1에서와 마찬가지로 $\alpha = 5$에서 M3, M4보다 성능이 향상됨을 볼 수 있다.

Fig. 6은 실제 학습된 모델로부터 얻은 모델의 출력을 사용하여 얻은 것으로서, Fig. 6(a)는 원본 음성의 스펙트로그램, Fig. 6(b)는 변분 오토인코더 증강모델의 출력으로 얻어진 증강된 음성의 스펙트로그램이다.

Fig. 6을 보면 증강된 데이터인 Fig. 6(b)가 원본 음성인 Fig. 6(a)보다 낮은 주파수 영역에서 대체적으로

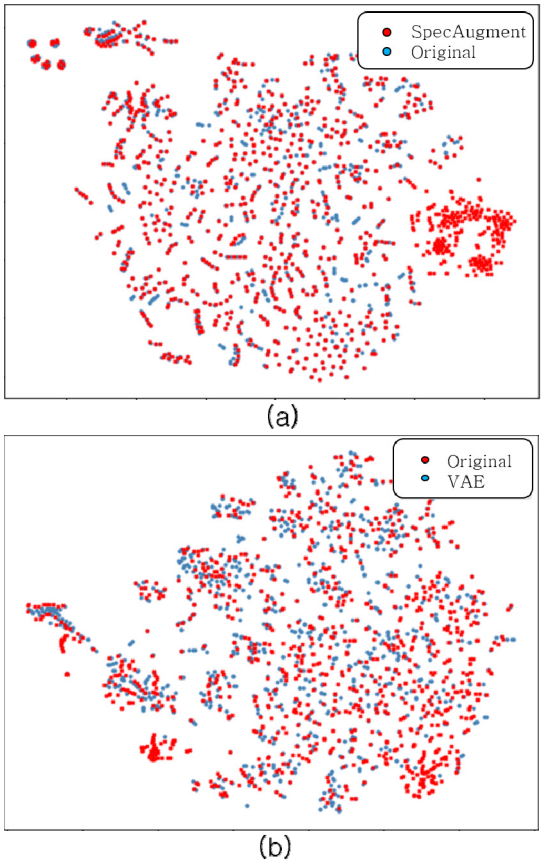

Fig. 7. (Color available online) t-SNE scatter plot. (a)The original speech data and the speech data augmented by the proposed method, (b) the original speech data and the speech data augmented by the SpecAugment method.

에너지가 낮음을 볼 수 있다. 또한 전반적으로 증강된 데이터가 원본 음성보다 에너지가 낮아진 것을 볼 수 있다. 이는 변분 오토인코더 기반 증강모델이 기존의 음성의 특성을 조금씩 왜곡시키는 데이터 증강 기술에 부합함을 의미한다. 위의 실험을 통해서 왜곡된 데이터를 학습에 활용하면 성능 향상 가능성이 있음을 확인하였다.

Fig. 7은 원본 음성과 데이터 증강된 음성간의 데이터 분포를 그린 그림이다. Fig. 7(a)는 원본 음성과 기존 방법인 SpecAugment를 적용한 음성의 분포도를 나타내며, Fig. 7(b)는 Fig. 6(a)와 Fig. 6(b)에 주어진 원본 음성과 증강된 음성의 분포도를 나타낸다. 분포도를 그리기 위해서 t-SNE[16]를 사용하여 각 데이터의 차원을 축소하였다. Fig. 7(a)와 같이 SpecAugment는 마스크를 씌운 만큼 원본 데이터에서 벗어나서

뭉쳐지는 현상이 발생하는 반면, Fig. 7(b)와 같이 변분 오토인코더로부터 증강된 데이터는 원본 음성에서 크게 멀어지지 않고 그 주변에 머물며 음성을 왜곡시킨다. Fig. 7(b)의 결과를 보면 증강된 음성이 전반적으로 원본 음성의 주변에 머물고 있음을 볼 수 있다. 이는 원본 음성에 최대한 가깝게 분포하여 그 특성을 유지하면서 기존과는 조금씩 다른 데이터를 생성하는 Fig. 6의 결과에 부합한다.

V. 결 론

본 연구에서는 변분 오토인코더 기반 데이터 증강 모델과 비교사 데이터 증강에 디코더 고정을 추가하여 학습하는 방법을 제안하였다. 증강모델 자체에서 더 많은 음성데이터를 생성시키기 위해서 생성모델인 변분 오토인코더를 사용하였으며, 음성인식 학습 시 음향특성을 잘 추출하기 위해서 디코더를 고정시켜 인코더가 잘 학습되도록 하였다. 최종적으로 학습된 변분 오토인코더와 음성인식 디코더 고정을 추가한 비교사 데이터 증강 학습을 적용한 경우에 가장 높은 성능을 보였다.

향후 연구로서, 최근에 제안된 잡음 학생 방법[8]과 변분 오토인코더 증강모델을 결합하거나, 변분 오토인코더 증강모델로부터 생성되는 데이터의 다양성을 증가시키기 위한 연구가 필요하다.

감사의 글

이 논문은 2019년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(2019-0-00004, 준지도학습형 언어지능 원천기술 및 이에 기반한 외국인 지원용 한국어 튜터링 서비스 개발).

References

1. F. Seide, G. Li, and D. Yu, “Conversational speech transcription using context-dependent deep neural networks,” Proc. INTERSPEECH, 437-440 (2011).
2. W. Chan, N. Jaitly, Q. Le, and O. Vinyals, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition,” Proc. ICASSP. 4960-4964 (2016).
3. A. Vaswami, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” Proc. NIPS. 5998-6008 (2017).
4. T. Hori, R. Astudillo, T. Hayashi, Y. Zhang, S. Watanabe, and J. L. Roux, “Cycle-consistency training for end-to-end speech recognition,” Proc. ICASSP. 6271-6275 (2019).
5. M.-K. Baskar, S. Watanabe, R. Astudillo, T. Hori, L. Burget, and J. Cernocky, “Semi-supervised sequence-to-sequence ASR using unpaired speech and text,” Proc. ICASSP. 3790-3794 (2019).
6. Q. Xie, Z. Dai, E. Hovy, M. T. Luong, and Q. V. Le, “Unsupervised data augmentation for consistency training,” arXiv:1904.12848 (2019).
7. J. Li, M. L. Seltzer, X. Wang, R. Zhao, and Y. Gong, “Large-scale domain adaptation via teacher-student learning,” Proc. INTERSPEECH, 2386-2390 (2017).
8. Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, “Self-training with noisy student improves ImageNet classification,” Proc. CVPR. 10687-10698 (2020).
9. N. Jaitly and G. E. Hinton, “Vocal tract length perturbation (VTLP) improves speech recognition,” Proc. ICML. 625-660 (2013).
10. D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A simple data augmentation method for automatic speech recognition,” Proc. INTERSPEECH, 2613-2617 (2019).
11. X. Song, Z. Wu, Y. Huang, D. Su, and H. Meng, “SpecSwap: A simple data augmentation method for end-to-end speech recognition,” Proc. INTERSPEECH, 581-585 (2020).
12. D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” Proc. ICLR. 1-14 (2014).
13. D. B. Paul and J. M. Baker, “The design for the Wall Street Journal-based CSR corpus,” Proc. ACL. 357-362 (1992).
14. V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “LibriSpeech: An ASR corpus based on public domain audio books,” Proc. ICASSP. 5206-5210 (2015).
15. S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, “ESPnet: End-to-end speech processing toolkit,” Proc. INTERSPEECH, 2207-2211 (2018).
16. L. V. D. Maaten and G. Hinton, “Visualizing data using t-SNE,” J. Mach. Learn. Res. **9**, 2579-2605 (2008).

저자 약력

▸ 조 현 호 (Hyeon Ho Jo)

2017년 2월 : 충북대학교 전자공학부 학사
2019년 2월 : 충북대학교 제어로봇공학과 석사
2019년 3월 ~ 현재 : 충북대학교 지능로봇공학과 박사과정

▸ 강 병 옥 (Byung Ok Kang)

1997년 2월 : 포항공과대학교 전자전기공학과 학사
1999년 2월 : 포항공과대학교 전자전기공공학과 석사
2017년 2월 : 충북대학교 전기·전자·정보·컴퓨터학부 박사
1999년 ~ 2001년 : 삼성전자 무선사업부 선임연구원
2001년 ~ 현재 : 한국전자통신연구원 책임연구원

▸ 권 오 욱 (Oh-Wook Kwon)

1986년 2월 : 서울대학교 전자공학과 학학사
1988년 2월 : 한국과학기술원 전기및전자공학과 석사
1997년 2월 : 한국과학기술원 전기및전자공학과 박사
1988년 3월 ~ 2000년 4월 : 한국전자통신연구원 책임연구원
2000년 5월 ~ 2001년 3월 : 한국과학기술원 연구교수
2001년 3월 ~ 2003년 8월 : UCSD 박사후 연구원
2003년 9월 ~ 현재 : 충북대학교 지능로봇공학과 교수