JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 18, No. 1, pp. 93-99, Feb. 2017

CTC를 이용한 LSTM RNN 기반 한국어 음성인식 시스템

이 동 현 · 임 민 규 · 박 호 성 · 김 지 환*
서강대학교 컴퓨터공학과

LSTM RNN-based Korean Speech Recognition System Using CTC

Donghyun Lee · Minkyu Lim · Hosung Park · Ji-Hwan Kim*
Department of Computer Science and Engineering, Sogang University, Seoul 04107, Korea

[요 약]

Long Short Term Memory (LSTM) Recurrent Neural Network (RNN)를 이용한 hybrid 방법은 음성 인식률을 크게 향상시켰다. Hybrid 방법에 기반한 음향모델을 학습하기 위해서는 Gaussian Mixture Model (GMM)-Hidden Markov Model (HMM)로부터 forced align된 HMM state sequence가 필요하다. 그러나, GMM-HMM을 학습하기 위해서 많은 연산 시간이 요구되고 있다. 본 논문에서는 학습 속도를 향상하기 위해, LSTM RNN 기반 한국어 음성인식을 위한 end-to-end 방법을 제안한다. 이를 구현하기 위해, Connectionist Temporal Classification (CTC) 알고리즘을 제안한다. 제안하는 방법은 기존의 방법과 비슷한 인식률을 보였지만, 학습 속도는 1.27 배 더 빨라진 성능을 보였다.

[Abstract]

A hybrid approach using Long Short Term Memory (LSTM) Recurrent Neural Network (RNN) has showed great improvement in speech recognition accuracy. For training acoustic model based on hybrid approach, it requires forced alignment of HMM state sequence from Gaussian Mixture Model (GMM)-Hidden Markov Model (HMM). However, high computation time for training GMM-HMM is required. This paper proposes an end-to-end approach for LSTM RNN-based Korean speech recognition to improve learning speed. A Connectionist Temporal Classification (CTC) algorithm is proposed to implement this approach. The proposed method showed almost equal performance in recognition rate, while the learning speed is 1.27 times faster.

Key word : Connectionist temporal classification, Long short term memory, Recurrent neural network, Acoustic model, Speech recognition

색인어 : Connectionist temporal classification, Long short term memory, Recurrent neural network, 음향모델, 음성인식

http://dx.doi.org/10.9728/dcs.2017.18.1.93

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-CommercialLicense(http://creativecommons.org/licenses/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 19 January 2017; **Revised** 31 January 2017
Accepted 25 February 2017

***Corresponding Author; Ji-Hwan Kim**

Tel: +82-2-705-8924
E-mail: kimjihwan@sogang.ac.kr

Ⅰ. 서 론

음성인식 기술은 키보드나 마우스를 사용하지 않고, 사람의 음성을 이용하여 스마트폰같은 기기 및 다양한 서비스를 제어할 수 있는 human-computer interaction 기술이다[1]. 음성인식 기술이 적용된 대표적인 사례는 Apple의 Siri, 삼성의 S Voice, LG의 Q Voice와 같은 지능형 개인 비서 시스템 (IPA; intelligent personal assistant) 이다[2]. 이외에도, Google, 네이버, 다음은 모바일 검색에 음성인식 기술을 도입한 검색 서비스도 제공하고 있다[3].

이러한 음성인식 기술은 사람의 음성으로부터 단어 sequence를 추정하는 것이 목표이며, 통계모델인 음향모델과 언어모델을 이용해서 단어 sequence를 추정한다[4].

$$\hat{W} = \arg\max_W P(W|O) = \arg\max_W \frac{P(O|W)P(W)}{P(O)} \approx \arg\max_W P(O|W)P(W) \quad (1)$$

식 (1)은 주어진 단어 sequence 를 사람이 발성했을 때, 그에 대한 음향학적 벡터 정보 O로부터 통계적으로 가장 비슷한 단어 sequence W를 계산하기 위하여 베이지안 이론을 적용한 수식이다. 수식에서 $P(O)$는 음향학적 벡터 정보에 대한 확률로써, W와 독립 관계이므로, 유도된 수식에서는 제외되었다. 따라서 식 (1)의 마지막 수식으로부터, 모든 가능한 경우의 $P(W)$와 $P(O|W)$의 곱을 계산하고, 가장 높은 값을 반환하는 W를 음성인식의 결과로 반환한다. 이 때, $P(W)$와 $P(O|W)$는 통계모델로부터 추정하며, 각각 언어모델과 음향모델이라 한다.

언어모델은 식 (1)에서 추정된 단어 sequence W 내의 각 단어들 간 문법에 대한 정보를 제공한다. 언어모델은 텍스트로 구성된 학습 자료로부터 단어들의 빈도수 정보를 이용해, N-gram에 기반하여 단어들의 관계를 표현한다.

음향모델은 사람이 발성한 음성 특징 정보를 모델링하고, 이에 대한 정보를 제공한다. 음향모델은 음성 정보 및 이에 대한 스크립트로 구성된 학습 자료로부터, HMM (hidden markov model)의 각 state에서 발생되는 출력 확률 (observation probability)을 GMM (gaussian mixture model)로 표현하는 GMM-HMM 모델을 이용한다[5]-[6].

언어모델과 음향모델을 생성하기 위한 기존의 방법들은 최근에 딥러닝 (deep learning)이 적용되면서, 더 높은 성능을 보이고 있다. 언어모델은 Mikolov가 딥러닝의 일종인 RNN (recurrent neural network)에 기반한 언어모델을 N-gram rescoring에 적용하여 기존의 언어모델보다 향상된 성능을 보였다[7].

음향모델은 딥러닝의 일종인 DNN (deep neural newtork)을 이용하여 Hinton 교수가 제안한 DNN-HMM 모델이 GMM-HMM 모델보다 높은 성능을 보였으며, RNN의 hidden node를 LSTM (long short term memory) 구조로 구성한, LSTM RNN 기반 음향모델이 제안되어 DNN-HMM 모델의 성능을 향상시켰다[8]-[10].

하지만, HMM의 출력확률을 DNN이나 LSTM RNN으로 모델링하는 hybrid 방법은 감독학습 (supervised learning)으로 학습되므로, GMM-HMM 기반 음향모델로부터 forced alignment 된 각 음성 특징 정보에 대한 HMM state sequence 정보가 요구된다. 이는 1) 대부분의 음향모델 학습 자료가 음성 정보 및 이에 대한 단어 단위의 스크립트만 제공되기 때문에, DNN-HMM 모델 학습 이전에 GMM-HMM 기반 음향모델이 학습되어야 하는 문제점, 2) GMM-HMM 기반 음향모델을 학습하고, DNN-HMM 기반 음향모델을 학습하기 때문에, 학습 시간이 많이 소요된다는 문제점, 3) 인간이 아닌, GMM-HMM 기반 음향모델로부터 통계적으로 forced alignment된 HMM state sequence 정보를 얻으므로, 잘못된 alignment 정보가 제공된다는 문제점이 발생한다.

이러한 문제점을 해결하기 위하여, Graves는 end-to-end (또는 sequence-to-sequence) 방법의 일종인 CTC (connectionist temporal classification)를 이용한 음향모델 학습 방법을 제안하였다[11]-[13]. CTC는 주어진 딥러닝 모델의 output layer에서 output node의 objective function의 일종으로, 각 output node는 인식하려는 언어에서 정의된 phoneme 또는 character를 반영하며, forward-backward 알고리즘을 이용하여 phoneme sequence 또는 character sequence를 추정한다.

하지만 CTC를 이용한 한국어 음향모델은 아직까지 연구가 활발하게 진행되지 않고 있다. 본 논문에서는 LSTM 및 CTC 알고리즘에 대해서 분석하고, Kaldi 음성인식 오픈소스 툴킷을 이용하여 CTC를 이용한 LSTM RNN 기반 한국어 음향모델을 학습한다. 학습된 모델은 인식률 및 학습속도로 DNN-HMM기반 음향모델과 성능 비교를 수행한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 통해, LSTM 구조 및 CTC 알고리즘에 대해서 분석한다. 3장에서는 영어와 한국어에 대해서, CTC를 이용한 LSTM RNN 기반 음향모델을 DNN-HMM 기반 음향모델과 성능 비교 평가를 수행하고, 4장에서는 결론을 기술한다.

Ⅱ. 관련 연구

본 장에서는 LSTM 구조 및 CTC 알고리즘에 대해서 분석한다. 2.1절에서는 LSTM 구조, 2.2절에서는 CTC 알고리즘에 대해서 분석한다.

2-1 LSTM 구조에 대한 분석

LSTM은 (그림 1)과 같이, hidden node 구조 중 하나로, 1개 이상의 memory cell과 input gate, output gate, forget gate로 구성되어 있다[14]-[16].

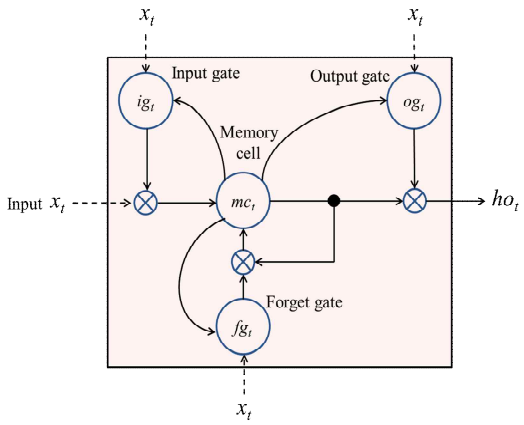

그림 1. LSTM 기반 hidden node 구조
Fig. 1. Architecture of LSTM-based hidden node

기존의 DNN 기반 모델에서는 error back-propagation을 수행할 때, 아래의 hidden layer로 내려올수록, error rate가 0으로 수렴하는 vanishing 문제가 발생한다. 또한, RNN에서 error back-propagation을 수행할 때, long context를 갖는 입력 자료에서 time t가 증가함에 따라, time (t-1)의 hidden layer의 error rate가 time t의 hidden layer의 error rate에 계속 반영되어, error rate가 계속 0으로 수렴한다.

이러한 DNN과 RNN 모델에서의 vanishing 문제를 해결하기 위하여, hidden node에 LSTM 구조를 적용하여, LSTM의 memory cell에 의해, long context에서도 error rate가 0으로 수렴하지 않았다.

$$ig_t = \sigma(W_{x,ig}x_t + W_{ho,ig}ho_{t-1} + W_{mc,ig}mc_{t-1} + b_{ig}) \quad (2)$$
$$og_t = \sigma(W_{x,og}x_t + W_{ho,og}ho_{t-1} + W_{mc,og}mc_{t-1} + b_{og}) \quad (3)$$
$$fg_t = \sigma(W_{x,fg}x_t + W_{ho,fg}ho_{t-1} + W_{mc,fg}mc_{t-1} + b_{fg}) \quad (4)$$
$$mc_t = fg_t mc_{t-1} + ig_t(W_{x,mc}x_t + W_{ho,mc}ho_{t-1} + b_{mc}) \quad (5)$$
$$ho_t = og_t \tanh(mc_t) \quad (6)$$

식 (2)는 (그림 1)에서 input gate ig_t에 대한 수식이며, $W_{x,ig}$는 입력 벡터 x_t와 input gate 간의 weight matrix, $W_{ho,ig}$는 time (t-1)에서의 hidden node와 input gate 간의 weight matrix, $W_{mc,ig}$는 time (t-1)에서의 memory cell과 input gate 간의 weight matrix, ho_{t-1}은 time (t-1)에서의 hidden node의 출력값, mc_{t-1}은 time (t-1)에서의 memory cell의 출력값, b_{ig}는 input gate의 bias 값이다. 식 (3)은 (그림 1)에서 output gate og_t에 대한 수식이며, $W_{x,og}$는 입력 벡터 x_t와 output gate 간의 weight matrix, $W_{ho,og}$는 time (t-1)에서의 hidden node와 output gate 간의 weight matrix, $W_{mc,og}$는 time (t-1)에서의 memory cell과 output gate 간의 weight matrix, b_{og}는 output gate의 bias 값이다. LSTM 구조에서 input gate와 output gate는 error back-propagation을 수행할 때, weight의 error rate를 증가 또는 감소시키는 역할을 수행하여, DNN 모델의 vanishing 문제를 해결하였다.

식 (4)는 (그림 1)에서 forget gate fg_t에 대한 수식이며, $W_{x,fg}$는 입력 벡터 x_t와 forget gate 간의 weight matrix, $W_{ho,fg}$는 time (t-1)에서의 hidden node와 forget gate 간의 weight matrix, $W_{mc,fg}$는 time (t-1)에서의 memory cell과 forget gate 간의 weight matrix, b_{fg}는 forget gate의 bias 값이다. Forget gate는 time (t-1)까지 memory cell에 저장되어 있던 정보가 time t의 error signal과 연관이 없다면, memory cell의 값을 0으로 reset하는 역할을 수행한다. 이는 RNN 모델의 vanishing 문제를 해결하였다.

식 (5)는 (그림 1)에서 memory cell mc_t에 대한 수식이며, $W_{x,mc}$는 입력 벡터 x_t와 memory cell 간의 weight matrix, $W_{ho,mc}$는 time (t-1)에서의 hidden node와 memory cell 간의 weight matrix, b_{mc}는 memory cell의 bias 값이다. Memory cell은 long context를 갖는 입력 자료에 대해서, error back-propagation을 수행할 때, 시간에 관계없이 동일한 error 값을 유지할 수 있는 역할을 수행한다. 식 (6)은 (그림 1)에서 LSTM 기반 hidden node의 time t에서의 최종 출력값이다.

[17]에서 LSTM RNN 기반 음향모델은 TIMIT 자료에 대해 Phoneme Error Rate (PER) 기준 DNN 기반 음향모델보다 2.1% 더 낮은 PER을 보이면서, LSTM RNN 기반 음향모델의 성능이 더 우수하다는 것을 보였다.

이러한 LSTM 구조를 이용하는 딥러닝 모델은 기존의 모델보다 3~4배 이상의 weight를 학습해야 하므로, 많은 양의 학습 자료가 필요하며, 그만큼 많은 학습 시간이 소요된다는 문제점이 있다.

2-2 CTC 알고리즘에 대한 분석

기존의 hybrid 방법은 이용한 음향모델은 감독학습을 이용했기 때문에, 정답 정보를 제공하기 위하여, 각 음성 특징 정보에 대해서 forced alignment된 HMM state sequence 정보가 요구된다. 이는 1) GMM-HMM 기반 음향모델이 DNN-HMM 모델 학습 이전에 학습해야만 하는 문제점, 2) DNN-HMM 기반 음향모델 학습 이전에 GMM-HMM 기반 음향모델을 학습해서 학습 시간이 많이 소요되는 문제점, 3) 통계적으로 forced alignment된 정보를 얻으므로, 잘못된 alignment 정보가 제공된다는 문제점이 발생한다.

Graves는 hybrid 방법의 문제점을 해결하기 위해, GMM-HMM 모델을 학습하지 않고 딥러닝 모델 상에서 각 음성 특징 정보로부터 phoneme 또는 character를 얻기 위한 end-to-end 방법의 CTC (connectionist temporal classification)를 이용한 음향모델 학습 방법을 제안하였다. CTC는 주어진 딥러닝 모델의 output layer에서 output node의 objective function의 일종으로, 각 output node는 인식하려는 언어에서 정의된 phoneme 또는 character를 반영한다[18]. CTC를 이용한 딥러닝 모델을 학습할 때는 단어 단위 또는 phoneme 단위의 정답 스크

립트가 제시되므로, GMM-HMM 모델을 통해 얻은 HMM state sequence가 요구되지 않는다.

CTC를 이용하는 모델은 인식하려는 언어의 학습 자료로부터 K개의 라벨이 주어졌을 때, output layer가 (K+1)개의 output node로 구성된다. 여기서 1개의 output node가 추가된 이유는 공백 라벨 $Ø$를 반영하기 위함이다. 공백 라벨은 주어진 음성 특징 벡터가 K개의 라벨 중 어느 라벨도 연관되지 않았을 때, 의미가 없는 라벨로써 출력하기 위해 사용된다. 이러한 모델 구조상에서 CTC는 자동으로 각 음성 특징 정보와 phoneme 또는 character같은 라벨이 서로 쌍을 이루면서 학습된다. 즉, 식 (7)과 같이, 주어진 음성 특징 정보 sequence X로부터, 정답 라벨 sequence L^*와 가장 비슷한 L을 찾는 것을 목표로 학습이 진행된다.

$$L^* = \arg max_L P(L|X) \tag{7}$$

이 때, $P(L|X)$는 식 (8)과 동일하다.

$$\sum_{\theta \in E^{-1}(L)} P(\theta|X) = \sum_{\theta \in E^{-1}(L)} \prod_{t=1}^{T} opt_\theta^t \tag{8}$$

식 (8)에서 Θ는 라벨을 구성하는 phoneme 또는 character 집합에 공백 라벨을 추가한 집합 C로부터 만들어낼 수 있는 모든 라벨 sequence이다. E는 Θ에서 중복되는 라벨 sequence 및 공백 라벨을 제거하는 함수이다. 중복되는 라벨을 제거하는 규칙은 1) 공백 라벨을 제외한 모든 라벨에 대해서, 중복되는 라벨을 제거하고, 2) 공백 라벨을 제거하는 방식으로 진행된다. 예를 들어, ØaaØbaØbbØ 라는 라벨 sequence가 함수 E의 입력으로 주어졌을 때, E(ØaaØbaØbbØ) = abab 로 변환된다. opt_θ^t는 Θ의 time t에 해당하는 라벨의 출력 확률 값이다. 이러한 CTC의 목표를 위해, time t에서 CTC가 적용된 output layer의 loss function LF는 식 (9)와 같다.

$$LF(X,y) = -\ln P(y|X) = -\ln \sum_{s=1}^{|os^*|} \alpha(t,s)\beta(t,s) \tag{9}$$

식 (9)에서 y는 time t까지 생성될 수 있는 output sequence os^*에 대해서 함수 E를 취한 L의 sub-label sequence이다. s는 output layer를 구성하는 output node의 index 정보이다. α는 forward variable로, L의 s/2 만큼의 prefix와 대응되는 time 0부터 time t까지의 모든 sequence에 대한 확률들의 합이다. β는 backward variable로, α를 통해서 구한 path가 있을 때, time (t+1)부터 $|L|$까지의 모든 sequence에 대한 확률들의 합이다. α와 β를 계산하기 위하여 forward-backward 알고리즘을 사용한다. (그림 2)는 길이 T만큼의 음성 특징 벡터 sequence X가 주어졌을 때, forward-backward 알고리즘에 대한 그림이다.

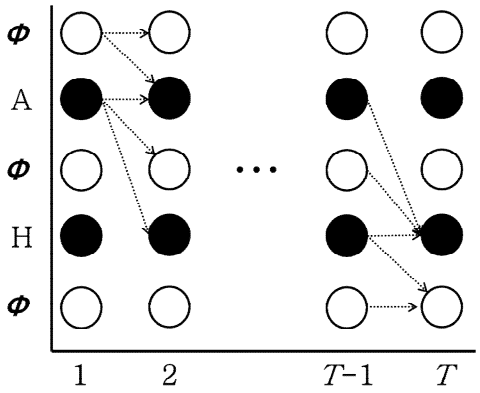

그림 2. CTC의 forward-backward 알고리즘
Fig. 2. Forward-Backward Algorithm for CTC

(그림 2)에서 흰색 원은 공백 라벨을 의미하며, 검은색 원은 공백 라벨을 제외한 모든 라벨들 중 하나를 의미한다. 공백 라벨은 학습 과정에서 정답 라벨 sequence의 양 끝의 silence 또는 short-pause와 대응된다. Forward-backward 알고리즘에서 처음에 선택될 수 있는 라벨은 공백 라벨 또는 공백 라벨을 제외한 모든 라벨들 중 1개의 라벨이다. 이후 time t (t≥1)에서 선택될 수 있는 라벨은 1) time (t-1)에서 선택된 라벨이 공백 라벨인 경우, 공백 라벨이 다시 선택되거나, 공백 라벨을 제외한 모든 라벨들 중 1개가 선택되며, 2) time (t-1)에서 선택된 라벨이 공백 라벨이 아닌 다른 라벨들 중 1개인 경우, 동일한 라벨이 다시 선택되거나, 공백 라벨이 선택되거나, time (t-1)에서 선택된 라벨 및 공백 라벨을 제외한 다른 라벨들 중 1개가 선택된다.

이러한 forward-backward 알고리즘을 이용하여 error back-propagation을 통해 딥러닝 모델 학습을 수행할 때, time t에서 output layer의 loss function gradient $\frac{\triangle L(X,y)}{\triangle act_i^t}$에 대한 수식은 식 (10)과 같다.

$$opt_i^t - \frac{1}{P(y|X)} \sum_{s \in A(y,i)} \alpha(t,s)\beta(t,s) \tag{10}$$

식 (10)에서 opt_i^t는 time t에서 output layer의 i번째 output node (라벨 i)에 activation function이 적용되기 이전의 변수이며, $A(y, i)$는 라벨 i가 y에서 나타날 수 있는 index의 집합이다. 이 수식을 통해, CTC를 학습하기 위해서는 forward-backward 알고리즘에서 사용되는 α와 β도 함께 학습되어야 함을 보이고 있다.

Ⅲ. 실험 및 평가

본 장에서는 CTC를 이용한 LSTM RNN 기반 음향모델과 DNN-HMM 기반 음향모델에 대해, 한국어 학습자료 상에서 성

능 비교 실험을 진행하고, 이에 대한 평가를 수행한다. 3.1절에서는 실험 환경에 대해서 기술하고, 3.2절에서는 한국어 학습 자료상에서의 성능 비교 평가의 결과에 대해서 기술한다.

3-1 실험 환경

음향모델을 학습하기 위하여, Kaldi 음성인식 오픈소스 툴킷을 사용하였다. Kaldi 툴킷은 C++ 기반의 툴킷으로, 다양한 딥러닝 기반 음향모델 학습 및 WFST (weighted finite state transducer) 기반 디코딩을 지원하고 있다. 언어모델을 학습하기 위하여, SRILM 오픈소스 툴킷을 사용하였다. SRILM 툴킷은 *N*-gram 기반 언어모델 학습뿐만 아니라, 다양한 언어모델 관련 학습 알고리즘을 지원한다.

3.2절에서 성능 평가를 수행할 한국어 학습 자료는 ETRI와 SiTEC 및 자체 수집한 총 740시간 분량의 한국어 코퍼스이다. ETRI와 SiTEC의 한국어 코퍼스는 뉴스 또는 책을 낭독한 낭독체 문장이며, 자체 수집한 코퍼스는 실생활에서 발성한 구어체 문장이다. 테스트 자료는 IPA 도메인 (날씨, 위치, 환율 등)으로 제한하여 자체 수집한 약 2시간 분량의 한국어 대화 발성 코퍼스를 사용하였다.

모든 음성 코퍼스 자료는 16비트 resolution, 16kHz의 sampling rate, mono 채널로 녹음되어 있다. 본 녹음자료로부터, 25ms의 frame, 10ms의 frame shift를 수행하면서, 매 frame 마다 MFCC 음성 특징 정보를 추출하였다. 추출된 MFCC는 40차 MFCC로, 40개의 cepstrum과 40개의 triangular mel-frequency bin을 사용하였다.

3-2 한국어 학습 자료를 이용한 성능 비교 평가

3.1절에서 기술한 한국어 자료를 사용하여, end-to-end 방법의 CTC를 이용한 LSTM RNN 기반 음향모델 및 hybrid 방법에 기반한 DNN-HMM 음향모델을 학습하였다. LSTM RNN 및 DNN-HMM 기반 음향모델의 topology는 <표 1>과 같다.

<표 1>에서 LSTM RNN 모델은 bi-directional RNN이며, forward RNN과 backward RNN의 hidden layer는 각각 4개로, 총 8개의 hidden layer를 사용했다고 기술하였다. Weight parameter는 두 모델 모두 8.5M개로, weight 개수가 성능에 영향을 주지 않는다고 고려한다.

구축된 topology 상에서 한국어 테스트 자료를 이용하여 테스트를 진행했고, 이에 대한 결과는 <표 2>와 같다. 한국어의 경우, 영어와 달리, 띄어쓰기에 따른 성능 변화가 발생하므로, 띄어쓰기를 모두 제거한 상태에서 음절이 1개라도 다르면, 오류가 발생했다고 보는, Sentence Error Rate (SER)을 비교 척도로 사용하였다.

표 1. LSTM RNN 및 DNN-HMM 기반 음향모델의 Topology
Table. 1. Topology for Acoustic Models based on LSTM RNN and DNN-HMM

Topology \ Acoustic Model	LSTM RNN	DNN-HMM
No. of Hidden Layers	8 (Forward: 4, Backward: 4)	4
No. of Hidden Nodes per Hidden Layer	320	2,700 (500 nodes for projections layers)
No. of Weight Parameters	8.5M	

표 2. 음향모델 성능과 학습 속도에 대한 비교
Table. 2. Comparison of Acoustic Model Performance and Learning Speed

Acoustic Model	Sentence Error Rate (SER, %)	Learning Speed (Hour)
LSTM RNN	28.42	196
DNN-HMM	22.83	249 (GMM-HMM: 68)

CTC를 이용한 LSTM RNN 기반 음향모델은 DNN-HMM 기반 음향모델과 비슷한 SER 수준을 보였다. 하지만, 학습 속도에서 CTC를 이용한 LSTM RNN 기반 음향모델이 GMM-HMM 모델 학습 시간 68시간까지 포함한 DNN-HMM 기반 음향모델보다 1.27배 더 빠른 학습 속도를 보였음을 확인하였다.

Ⅳ. 결 론

본 논문에서는 LSTM 구조 및 CTC 알고리즘에 대해서 분석하였으며, 한국어 코퍼스 상에서 CTC를 이용한 LSTM RNN 기반 음향모델의 성능 평가 결과를 보였다. LSTM 구조는 기존의 DNN 및 RNN에서 발생했던 vanishing 문제를 해결하였지만, 구조의 복잡성에 의하여, 학습 시간이 많이 소요된다는 문제점을 보였다. CTC 알고리즘은 기존의 hybrid 방법의 문제점을 해결했고, 기존의 DNN-HMM 기반 음향모델과 비슷한 성능을 보였지만, GMM-HMM 모델 학습을 배제하면서, 학습 속도가 더 향상된 것을 보였다. 추후, CTC를 이용한 LSTM RNN 모델의 디코딩 구조를 개선하여, 디코딩 속도를 개선하는 연구를 수행하고자 한다.

참고문헌

[1] A. Acero et al., “Live search for mobile: web services by voice on the cellphone,” in *Proceeding of the Interspeech*,

Brisbane, Australia, pp. 5256-5259, 2008.

[2] J. Jiang et al., Automatic online evaluation of intelligent assistants, in *Opportunities and Challenges for Next-Generation Applied Intelligence*, Berlin, Germany: Springer, pp. 285-290, 2009.

[3] S. Kim and J. Ahn, "Speech Recognition System in Car Noise Environment," *The Journal of Digital Contents Society*, Vol. 10, No. 1, pp. 121-127, Mar. 2009.

[4] L. Rabiner and B. Juang, *Fundamentals of Speech Recognition*, 1st ed. Englewood Cliffs, NJ: Prentice Hall, 1993.

[5] D. Su, X. Wu, and L. Xu, "GMM-HMM acoustic model training by a two level procedure with gaussian components determined by automatic model selection," in *Proceeding of the International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP*), Dallas: TX, pp. 4890-4893, 2010.

[6] H. Hermansky, D. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," in *Proceeding of the International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP*), Istanbul, Turkey, pp. 1635-1638, 2000.

[7] T. Mikolov and G. Zweig, Context dependent recurrent neural network language model, Microsoft Research, Redmond: WA, Technical Report MSR-TR-2012-92, 2012.

[8] G. Hinton et al., "Deep Neural Networks for Acoustic Modeling in Speech Recognition," *The IEEE Signal Processing Magazine*, Vol. 29, No. 6, pp. 82-97, Oct. 2012.

[9] L. Deng, G. Hinton, and B. Kingsbury, "New types of deep neural network learning for speech recognition for speech recognition and related applications: An overview," in *Proceeding of the International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP*), Vancouver, Canada, pp. 8599-8603, May. 2013.

[10] A. Graves et al., "Hybrid speech recognition with deep bidirectional LSTM," in *Proceeding of the IEEE Automatic Speech Recognition and Understanding Workshop*, Olomouc, Czech Republic, pp. 273-278, 2013.

[11] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceeding of the 31st International Conference on Machine Learning*, Beijing, China, pp. 1764-1772, 2014.

[12] A. Graves, Supervised sequence labelling with recurrent neural networks, Ph.D. dissertation, Technische Universitat Munchen, Munchen, Germany, 2008.

[13] A. Graves et. al, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceeding of the 23rd International Conference on Machine Learning*, Pittsburgh: PA, pp. 369-376, 2006.

[14] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, Vol. 9, No. 8, pp. 1735-1780, Nov. 1997.

[15] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," arXiv:1402.1128, pp. 1-5, Feb. 2014.

[16] M. Liwicki, A. Graves, H. Bunke and J. Schmidhuber, "A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks," in *Proceeding of the 9th International Conference on Document Analysis and Recognition*, Curitiba, Brazil, pp. 367-371, 2017.

[17] Y. Miao et al., "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *Proceeding of the IEEE Automatic Speech Recognition and Understanding Workshop*, Scottsdale: AZ, pp. 167-174, 2015.

[18] Y. Rao, A. Senior, and H. Sak, "Flat start training of CD-CTC-sMBR LSTM RNN acoustic models," in *Proceeding of the International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP*), Shanghai, China, pp. 5405-5409, 2016.

이 동 현 (Donghyun Lee)

2013년 : 서강대학교 (학사)

2009년~2013년: 서강대학교 컴퓨터공학과 학사
2013년~현 재: 서강대학교 컴퓨터공학과 석박사 통합과정
※관심분야 : Speech Recognition using Deep Learning, Artificial Intelligence, Multimedia Content Search

임 민 규 (Minkyu Lim)

2008년 : 서강대학교 (학사)
2010년 : 서강대학교 대학원 (공학석사)

2002년~2008년: 서강대학교 기계/컴퓨터공학 학사
2008년~2010년: 서강대학교 컴퓨터공학 석사
2012년~2015년: 휴맥스/아이큐브
2010년~현 재: 서강대학교 컴퓨터공학과 박사과정
※관심분야 : 음성인식, 오디오 컨텐츠 분석

박 호 성 (Hosung Park)

2016년 : 한동대학교 (학사)

2009년~2016년: 한동대학교 컴퓨터공학과 학사
2016년~현 재: 서강대학교 컴퓨터공학과 석사과정
※관심분야 : 음성인식

김 지 환 (Ji-Hwan Kim)

1996년 : KAIST (학사)
1998년 : KAIST (석사)
2001년 : University of Cambridge (박사)

2001년~2007년: LG 전자 책임연구원
20017년~현 재: 서강대학교 컴퓨터공학과 교수
※관심분야 : Spoken Multimedia Content Search, Speech Recognition using Cloud Computing and Dialogue Understanding