IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 7, 2020, accepted August 25, 2020, date of publication August 31, 2020, date of current version September 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3020696

End-to-End Speech Endpoint Detection Utilizing Acoustic and Language Modeling Knowledge for Online Low-Latency Speech Recognition

INYOUNG HWANG AND JOON-HYUK CHANG, (Senior Member, IEEE)
Department of Electronics and Computer Engineering, Hanyang University, Seoul 04763, South Korea

Corresponding author: Joon-Hyuk Chang (jchang@hanyang.ac.kr)

This work was supported by the Institute of Information and Communications Technology Planning and Evaluation (IITP) funded by the Korea Government (MSIT), through Deep Learning Multi-Speaker Prosody and Emotion Cloning Technology Based on a High Quality End-to-End Model Using Small Amount of Data, under Grant 2020-0-00059.

ABSTRACT Speech endpoint detection (EPD) benefits from the decoder state features (DSFs) of online automatic speech recognition (ASR) system. However, the DSFs are obtained via the ASR decoding process, which can become prohibitively expensive especially in limited-resource scenarios such as the embedded devices. To address this problem, this paper proposes a language model (LM)-based end-of-utterance (EOU) predictor, which is trained to determine the framewise probabilities of the EOU token conditioned on the previous word history obtained from the 1-best decoding hypothesis of the ASR system in an end-to-end manner without an actual decoding process in the test step. Further, a novel end-to-end EPD strategy is presented to incorporate a phonetic embedding (PE)-based acoustic modeling knowledge and the proposed EOU predictor-based language modeling knowledge into an acoustic feature embedding (AFE)-based EPD approach within the recurrent neural networks (RNN)-based EPD framework. The proposed EPD algorithm is built upon the ensemble RNNs, which are independently trained for the three parts, which are the proposed LM-based EOU predictor, AFE-based EPD, and PE-based acoustic model (AM) in accordance with each target. The ensemble RNNs are concatenated at the level of the last hidden layers and then attached into the fully-connected deep neural networks (DNN)-based EPD classifier, which is trained in accordance with the ultimate EPD target. Thereafter, they are jointly retrained at the second step of the DNN training to yield the lower endpoint error. The proposed EPD framework was evaluated in terms of the endpoint accuracy and word error rate for the CHiME-3 and large-scale ASR tasks. The experimental results turn out that the proposed EPD algorithm efficiently outperforms the conventional EPD approaches.

INDEX TERMS Acoustic model (AM), end-of-turn detection, end-of-utterance (EOU) detection, feature embedding, language model (LM), online speech recognition, pause hesitation, speech endpoint detection (EPD), spoken dialogue system.

I. INTRODUCTION

Spoken dialogue systems make it possible to control contemporary devices, such as smartphones, navigation systems, and AI speakers through natural voice interaction. Usually, the interaction with such devices is user-initiated by uttering the wake-up-word. Then, an automatic speech recognition (ASR) technique is performed in an online manner until an end-of-utterance (EOU) is automatically detected by a speech endpoint detection (EPD) algorithm. The EPD is a challenging task since the utterance can be endpointed late due to the ambient noise and early due to long pause hesitation. Since an early endpoint undesirably cuts off the speech region, the performance of speech recognition is often degraded seriously; on the other hand, a late endpoint increases the response latency of the online ASR system. Consequently, degraded endpoint performance often causes the user dissatisfaction [1], [2].

The traditional EPD approaches consist of two cascaded decision processes. First, input speech is classified into

The associate editor coordinating the review of this manuscript and approving it for publication was Victor Sanchez.

speech and non-speech on a frame basis using a speech activity detection (SAD) algorithm designed with engineered features [3]–[7]. Then, the EOU is finally detected when the duration of non-speech obtained using the SAD algorithm reaches the pre-defined threshold value, i.e., 500 ms or 1 s [8]. Chung *et al.* proposed an EPD algorithm that classifies speech and non-speech states using the SAD technique based on a log-likelihood ratio (LLR) test proposed in [9], and then finds the endpoint with the online decoder designed based on a weighted finite-state transducer (wFST) [10]. Since it is difficult to optimize the LLR test-based SAD and wFST jointly, this EPD scheme was further improved by adopting the quantized LLR states as the wFST input instead of the binary speech/non-speech state [11]. The performance of these EPD structures is dramatically enhanced with the help of the SAD algorithms based on deep neural networks (DNN), which yield the state-of-the-art SAD performance via deep nonlinear hidden layers [12]–[17]. Especially, it was observed that the bottleneck features of the DNN-based acoustic model (AM), called phonetic embedding (PE), which is trained to predict senones (tied triphone states) [18], lead to improved SAD and EPD performances [19]–[21].

Another way is to directly find the EOU from the sequential input features by employing a long short-term memory (LSTM) [22], whereas the traditional EPD schemes consist of the separate SAD and online decoder. The LSTM can model the complex relations between the input feature sequence and the corresponding framewise EPD targets with the memory cell as the temporal state of the network can be successfully controlled by the input, forget, and output gates [23]–[26]. Notably, a unified architecture comprising the convolutional neural networks (CNN), LSTM, and fully-connected DNN, called CLDNN, was proposed to exploit their complementary advantages for the ASR [27] and SAD tasks [28]. However, it was observed that the capability of the convolution layer is diminished when extracting features in adverse noisy conditions [29]. To address this problem, an alternative method called grid-LSTM [30] was presented to model the time and frequency variations of input sequential features properly within separate time and frequency LSTM cells, respectively. Furthermore, a grid-LSTM DNN (GLDNN) [31] was introduced by employing the grid-LSTM in the first layer instead of the convolutional layer of the CLDNN to improve the EPD performance. However, these feature mapping-based EPD approaches often prematurely abandon the speech region due to a pause hesitation or cause a higher detection latency since they cannot adequately consider the context of input feature sequences such as phone or word alignments. In addition, the performance of the EPD approach using the LSTM can be degraded for a long utterance since the LSTM suffers from a state saturation problem due to its degradation of gate controls [32].

In addition, the EPD approaches designed to use the decoder state features (DSFs) of the ASR module as the auxiliary features have been introduced to distinguish the EOU from a short or long pause under the noisy environments. First, Ferrer *et al.* developed a prosodic feature-based EPD method that yields the EOU decision when a pause of any length is detected, where the decision statistic is determined by using the non-speech duration, prosodic feature, and language model (LM) knowledge [33]–[35]. Besides, Stuker *et al.* proposed a simple EPD approach, which is similar to the aforementioned approaches, to segment continuously recorded speech by triggering the EOU when the pause duration reaches the maximum pause threshold. Here, the pause duration can be obtained from phone alignment corresponding to the 1-best ASR decoding hypothesis [36]. However, this approach cannot be applied to the online ASR systems since the 1-best decoding hypothesis frequently changes during the online decoding process, which makes the EPD system unstable. To overcome this disadvantage, the expected pause duration is introduced as the stable feature for the online EPD task since it is obtained by interpolating the pause durations within all active hypotheses [37], [38]. Furthermore, it was observed that the word embedding (WE), which is obtained from the word LSTM [39] trained with the 1-best ASR decoding hypothesis to detect the turn-taking word, can yield the significant performance improvement of acoustic feature embedding (AFE)-based EPD without an actual decoding process, whereas the combination of the AFE, WE, and expected pause durations achieves the state-of-the-art EPD performance. Also, [40] incorporated the EOU symbol into the output within unified recurrent neural networks (RNN) transducer-based ASR system. Although various frameworks for on-device speech recognition have been proposed [41]–[43], the speech recognition accuracy is still limited due to the heavy computational cost since there is the trade-off between the word error rate (WER) and computational cost. Indeed, it is difficult to assess the advantages of the superior ASR system, which requires high complexity in limited-resource scenarios. Moreover, the WE cannot be considered as a reliable feature for the online EPD task since the context-dependent EPD approaches including the WE-based EPD suffer from the ambiguity of the turn-taking word due to the flexibility of a natural language [44].

In order to address the aforementioned disadvantages of conventional EPD approaches, this paper proposes an end-to-end EPD algorithm by incorporating both the acoustic and language modeling knowledge into the AFE-based EPD algorithm. First, the LM-based EOU predictor, which is trained to determine the framewise probabilities of the EOU token conditioned on the previous word history of the 1-best hypothesis obtained using the ASR decoding process, is presented. Once the proposed EOU-predictor is trained, it can derive the framewise probabilities of the EOU token given the input acoustic features without the actual ASR decoding process. Further, we introduce a novel EPD framework, consisting of the proposed LM-based EOU predictor, PE-based AM, and AFE-based EPD. When training the EOU predictor, the 1-best ASR decoding hypothesis with the N-gram LM is used to obtain the probabilities of the EOU token, which correspond to the training target of RNN. And, the AFE-based EPD algorithm is designed with RNN to classify the speech frame into four labels, namely speech, initial silence, final silence, and intermediate silence, on a frame-by-frame basis. Also, the AM for extracting the acoustic modeling knowledge with the use of the PE is trained with RNN by incorporating the sequential input features as the input along with senone targets. Then, the last hidden layers of the three ensemble RNNs are concatenated to train the fully-connected DNN-based classifier according to the hand-made EPD label. Finally, all the designed EPD networks are jointly retrained, thereby leading to a lower endpoint error. The proposed EPD algorithm was evaluated in terms of the early endpoint time, late endpoint time, and WER for the CHiME-3 ASR task [45], which includes various simulated and real acoustic conditions and a large-scale ASR task. Overall, the proposed EPD algorithm without the decoding process was observed to achieve a lower endpoint error, which leads to a lower WER and lower latency.

The rest of this paper is organized as follows. In the next section, we review the recently proposed EPD algorithms. In Section III, we describe the design of the proposed EPD approach. An extensive evaluation of the proposed algorithm is discussed in Section IV, and the conclusions are presented in Section V.

II. REVIEW OF PREVIOUS WORKS

This section briefly describes the conventional EPD algorithms which will be compared with the proposed EPD algorithm later.

A. EPD USING A GLDNN

The CLDNN-based architecture was previously introduced to exploit the complementary modeling advantages of the CNN, LSTM, and fully-connected DNN [27]. First, the CNN can extract the time and frequency-invariant features from sequential input features such as the Mel-frequency cepstral coefficients (MFCC) and log-Mel filterbank energies. In addition, the LSTM can model the short- and long-term temporal contexts of input features and the fully-connected DNN can model the complex relation between the features, which is represented via the CNN and LSTM, and the EPD target through multiple nonlinear hidden layers. As discussed in [29], the convolution layer for feature extraction is deteriorated in highly noisy conditions; hence, the alternative architecture called GLDNN [31] was introduced to replace the convolution layer with the grid-LSTM layer [30]. The grid-LSTM models the variations of successive features in the time and frequency axes through separate grid time LSTM (gT-LSTM) and grid frequency LSTM (gF-LSTM), respectively. Here, grid-LSTM is similar to the convolution layer in that both models are used to represent the input features over a restricted local time-frequency block and they use the shared model parameters. However, the grid-LSTM differs from the convolution layer in that it models frequency variations through a recurrent state that is passed along the frequency axis, whereas the convolution layer independently extracts the locally invariant features via the convolution and pooling operations.

The GLDNN-based EPD technique consists of the stacked grid-LSTM layers, standard LSTM layers, and fully-connected DNN layers. Once the time and frequency invariant features are extracted by the grid-LSTM layer, their short- and long-term temporal contexts are modeled by the standard LSTM layers. The EOU predictor finally classifies each frame into four distinct classes, namely speech, initial silence, final silence, and intermediate silence, to distinguish the final silence from the different silence states in the utterance. In the test step, the posterior probability of the final silence is computed and the EPD is triggered when it exceeds the given threshold value.

B. EPD BASED ON COMBINING AFE, WE, AND DSFs

The combined feature-based EPD algorithm [39] consists of three parts to detect the EOU exactly by fusing multiple features. They are an acoustic LSTM trained on the acoustic features, the word LSTM trained on the 1-best ASR decoding hypothesis, and the DSFs composed of three types of pause durations, which are described as follows. First, the acoustic LSTM trains the AFE in accordance with the framewise endpoint target. The corresponding SAD target is also trained in a multi-task fashion to distinguish the final silence from the initial silence and intermediate silence. Unlike the acoustic LSTM, the word LSTM is trained from the acoustic feature sequence to detect the turn-taking word. Hence, the word LSTM is triggered when alignments corresponding to the turn-taking word are observed instead of the final silence region, where the alignment is obtained from the 1-best ASR decoding hypothesis. To consider the decoder state, three types of expected pause durations extracted from the active ASR decoding hypotheses are utilized as the DSFs. Specifically, the DSFs consist of the best path pause duration, expected pause duration, and end pause duration, which are explained as follows. For this, Letting $\mathbf{X}_t = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t\}$ and $\mathbf{s}_t^i = \{s_1^i, s_2^i, \ldots, s_t^i\}$ be the input feature sequence until the t-th frame and the state sequence of the i-th active hypothesis until the t-th frame, respectively, the posterior probability of the i-th hypothesis is denoted by $P(\mathbf{s}_t^i|\mathbf{X}_t)$. First, the best path pause duration is determined by $L_t^{i_{\max}}$ with $i_{\max} = \operatorname{argmax}_i P(\mathbf{s}_t^i|\mathbf{X}_t)$ where L_t^i denotes the pause duration according to the i-th hypothesis. Second, the expected pause duration $\mathbb{D}(L_t)$ is obtained by interpolating the active hypotheses as follows:

$$\mathbb{D}(L_t) = \sum_{i=1}^{N_t} L_t^i P(\mathbf{s}_t^i|\mathbf{X}_t) \tag{1}$$

where N_t denotes the number of active hypotheses at the t-th frame. The expected final pause duration $\mathbb{D}_{\text{end}}(L_t^i)$ can

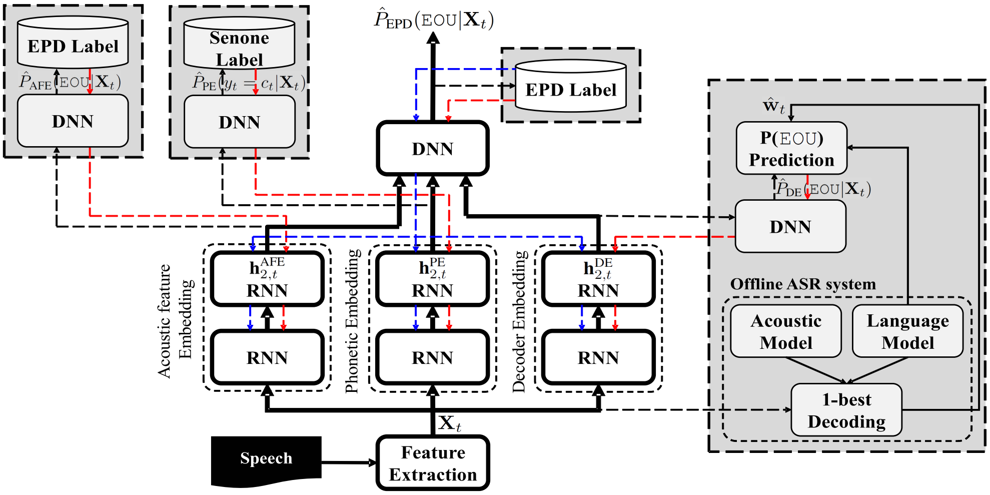

EPD Label
$\hat{P}_{\text{AFE}}(\text{EOU}|\mathbf{X}_t)$
DNN
Senone Label
$\hat{P}_{\text{PE}}(y_t = c_t|\mathbf{X}_t)$
DNN
$\hat{P}_{\text{EPD}}(\text{EOU}|\mathbf{X}_t)$
EPD Label
DNN
$\hat{\mathbf{w}}_t$
P(EOU) Prediction
$\hat{P}_{\text{DE}}(\text{EOU}|\mathbf{X}_t)$
DNN
Acoustic feature Embedding
$\mathbf{h}_{2,t}^{\text{AFE}}$ RNN
RNN
Phonetic Embedding
$\mathbf{h}_{2,t}^{\text{PE}}$ RNN
RNN
Decoder Embedding
$\mathbf{h}_{2,t}^{\text{DE}}$ RNN
RNN
Offline ASR system
Acoustic Model
Language Model
1-best Decoding
$\mathbf{X}_t$
Speech
Feature Extraction

FIGURE 1. Overall block diagram of the proposed EPD algorithm. (The black solid and dotted lines indicate the feed-forward paths for training and test stages while dotted lines are used only in the training stage, not in the test stage. The red dotted lines indicate the error back-propagation paths for separate training of each RNN in accordance with each target and DNN for the classifier. The blue dotted lines indicate the error back-propagation paths for the final joint-retraining process. The gray blocks are used in the training stage only not in the test stage.)

be determined as follows:

$$\mathbb{D}_{\text{end}}(L_t) = \sum_{i=1, s_t^i \in S_{\text{end}}}^{N_t} L_t^i P(\mathbf{s}_t^i|\mathbf{X}_t) \tag{2}$$

where S_{end} denotes the end state of the LM. At each frame, the feature vectors for the EPD are combined with the last hidden layer of both the acoustic LSTM and word LSTM along with the DSFs. The fully-connected DNN-based classifier is finally trained with the combined feature vector in accordance with the framewise endpoint target.

In the inference step, the EPD is triggered when the posterior likelihood of the endpoint exceeds a given threshold. To safeguard the lower and upper latency (pause duration) bounds, the SAD decision of the acoustic LSTM is additionally used as follows. If the pause duration obtained by the trained SAD does not reach the minimum pause duration, T_{min}, the endpoint is not triggered. Furthermore, the endpoint is enforced to be triggered if the pause duration obtained by the SAD is longer than the maximum pause duration, T_{max}.

III. PROPOSED END-TO-END EPD ALGORITHM BASED ON ENSEMBLE RNNs

As shown in Fig. 1, the novel EPD algorithm is proposed to exploit the ensemble of the AFE-based EPD, PE-based AM, and decoder embedding (DE) derived from the LM-based EOU predictor that is the main contribution of this study. The LM-based EOU predictor directly yields the framewise probability of the EOU token conditioned on the previous word history of the 1-best ASR decoding hypothesis with the N-gram LM. Accordingly, the LM-based EOU predictor, AFE-based EPD, and PE-based AM are separately trained, and then the fully-connected DNN-based EOU predictor is trained with the combined feature vector, which is composed of the last hidden layers of the three ensemble RNNs, in accordance with the framewise hand-labeled EPD targets as described in the following subsections.

A. PROPOSED LM-BASED EOU PREDICTOR

As shown in [39], the combination of the AFE and WE can yield superior EPD performance without the actual decoding process, closely matching the performance of the EPD system based on the AFE, WE, and DSFs, which can be obtained by performing the online ASR decoding process. However, in natural language processing (NLP), it is difficult to detect the turn-taking word in an online fashion due to the flexibility of the natural language. The natural language can express the user's intentions variously according to grammatical rules [44]. For instance, the user's intention tends to be expressed with the action and object information only such as "turn the lights on". Also, the user's intention is often expressed by including the specific location information by attaching an additional phrase such as "in the kitchen". In other words, from the expression "turn the lights on", it cannot be clearly identified whether "on" is the turn-taking word or not, whereas "on" is not the turn-taking word if the phrase "in the kitchen" follows the above expression. Thus, the WE, which is extracted from the word LSTM, cannot be considered as a reliable feature for the online EPD task since it is trained to detect the turn-taking word as depicted in Fig. 2(a). This figure shows the example pairs of the sentence and label from [46], which are used to train the word LSTM. It can be observed that different labels are given for the same word sequence, depending on whether the additional phrase follows or not.

(a)

Sentence 1:	SIL	Turn	the	lights	**on**	SIL			
Label:	0	0	0	0	**1**	1			
Sentence 2:	SIL	Turn	the	lights	**on**	in	the	kitchen	SIL
Label:	0	0	0	0	**0**	0	0	1	1

(b)

Sentence 1:	SIL	Turn	the	lights	**on**	SIL			
Label:	0	0	0	0	**0.065**	0.372			
Sentence 2:	SIL	Turn	the	lights	**on**	in	the	kitchen	SIL
Label:	0	0	0	0	**0.065**	0.372	0	0	0.609

FIGURE 2. Example pairs of the sentence and target used to train the embedding depending on the 1-best ASR decoding hypothesis for the EPD task: (a) word LSTM [39] and (b) the proposed LM-based EOU predictor.

In order to address these problems, this paper proposes the LM-based EOU predictor, which is similar to the word LSTM in that they are trained depending on the 1-best ASR decoding hypothesis. However, it differs from the word LSTM in that the proposed EOU predictor is trained to determine the framewise probabilities of the EOU token conditioned on the previous word history instead of binary classification for finding the turn-taking word. As depicted in Fig. 2(b), the same probabilities of the EOU token for training the EOU predictor are given without a reference to whether each word is the turn-taking word or not, where each probability of the EOU token is obtained from the 4-gram LM. After the word "on" is shown, the probability of the EOU token is 0.372 since the probability that the additional phrase is attached after the observed sentence to contain the specific information additionally is 0.628, which is obtained from the 4-gram LM. And, the probability of the EOU token is decreased to almost zero after the word in the middle of the sentence "in" is observed since $P(\texttt{EOU}|\text{lights, on, in}) \approx 0$. On the other hand, the probability of the EOU token rapidly increases after the last word in the sentence "kitchen" is detected. The method to obtain the framewise probability of the EOU token conditioned on the previous word sequence is described as follows.

The ASR technique aims to determine the most likely word sequence $\hat{\mathbf{w}}$, given the input acoustic feature sequence $\mathbf{X}$, where $\hat{\mathbf{w}}$ is expressed as follows:

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmax}}\, P(\mathbf{w}|\mathbf{X}). \tag{3}$$

Instead, the Bayes' rule represents it into the equivalent form as follows:

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmax}}\, P(\mathbf{X}|\mathbf{w})P(\mathbf{w}) \tag{4}$$

where the likelihood $P(\mathbf{X}|\mathbf{w})$ is determined by the AM usually based on the DNN and the prior probability $P(\mathbf{w})$ is obtained by the LM. Here, the LM is utilized to derive the probability of each word conditioned on the previous word history as $P(w_i|w_{<i})$. For large vocabulary continuous speech recognition (LVCSR), it is approximated by the N-gram LM according to the Markov chain rule, where the N-gram LM determines the probability of each word conditioned on the last $N-1$ words only, instead of the entire word history. However, the major drawback of the N-gram LM originates from data sparsity when trained with insufficient corpora. It can be mitigated by the combination of discounting and backing-off algorithms, called the Katz smoothing algorithm [47]. The 3-gram LM is suggested to obtain the probability of the EOU token conditioned on the word history as follows:

$$P(\texttt{EOU}|\mathbf{w}_{<i}) \approx \begin{cases} d\dfrac{C(w_{i-2}, w_{i-1}, \texttt{EOU})}{C(w_{i-2}, w_{i-1})} & \text{if } 0 < C \leq C' \\ \dfrac{C(w_{i-2}, w_{i-1}, \texttt{EOU})}{C(w_{i-2}, w_{i-1})} & \text{if } C > C' \\ \alpha(w_{i-2}, w_{i-1})P(\texttt{EOU}|w_{i-1}) & \text{otherwise} \end{cases} \tag{5}$$

where C' is a count threshold value, C is short-hand for $C(w_{i-2}, w_{i-1}, \texttt{EOU})$, d is a discount coefficient, and $\alpha(w_{i-2}, w_{i-1})$ is the normalisation constant. From (5), $P(\texttt{EOU}|w_{i-1})$ also can be alternatively obtained via the backing-off method if $C(w_i, \texttt{EOU}) = 0$ or the discounting method if $0 < C(w_i, \texttt{EOU}) \leq C'$. The 1-best ASR decoding hypothesis at t, called $\hat{\mathbf{w}}_t$, can be obtained as follows:

$$\hat{\mathbf{w}}_t = \underset{\mathbf{w}}{\operatorname{argmax}}\, P(\mathbf{X}_t|\mathbf{w})P(\mathbf{w}) \tag{6}$$

where $\mathbf{X}_t = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t\}$. The probability of the EOU token at t can be derived according to the last two words of the 1-best ASR decoding hypothesis by employing the 3-gram LM as follows:

$$P(\texttt{EOU}|\mathbf{X}_t) = \sum_{\mathbf{w}} P(\texttt{EOU}|\mathbf{w}, \mathbf{X}_t)P(\mathbf{w}|\mathbf{X}_t) \tag{7}$$

$$\approx P(\texttt{EOU}|\hat{\mathbf{w}}_t, \mathbf{X}_t) \tag{8}$$

$$\approx P(\texttt{EOU}|\hat{\mathbf{w}}_t) \tag{9}$$

$$\approx P(\texttt{EOU}|\hat{w}_{t,U-1}, \hat{w}_{t,U}) \tag{10}$$

where $\hat{w}_{t,u}$ and U denote the u-th word of $\hat{\mathbf{w}}_t$ and the number of words of $\hat{\mathbf{w}}_t$, respectively. Specifically, the probability of the EOU token given $\mathbf{X}_t$ can be obtained by marginalizing overall all the possible hypotheses at t. It can be represented by (8) with the assumption that the probability of the 1-best hypothesis dominates the probability mass of all the possible hypotheses such that $P(\hat{\mathbf{w}}_t|\mathbf{X}_t) = 1$. Furthermore, (8) can be rewritten as in (9) since it can be assumed that the probability of the EOU token is conditionally independent to $\mathbf{X}_t$. Finally, the probability of the EOU token given $\mathbf{X}_t$ can be determined according to the last two words of the 1-best hypothesis at t via the 3-gram LM approximation as in (10).

In this study, the LM-based EOU predictor is first presented to determine directly the probability of the EOU token $P(\texttt{EOU}|\mathbf{X}_t)$ in an end-to-end manner. As depicted in the upper part of Fig. 3, the framewise probabilities of the EOU token in the training stage are obtained from the 1-best ASR decoding hypothesis of each training-dataset $\hat{\mathbf{w}}_t$ with the help of the decoding module. The probability of the EOU token conditioned on the previous word history is obtained by the N-gram LM, used as the target for the training. Then, the proposed

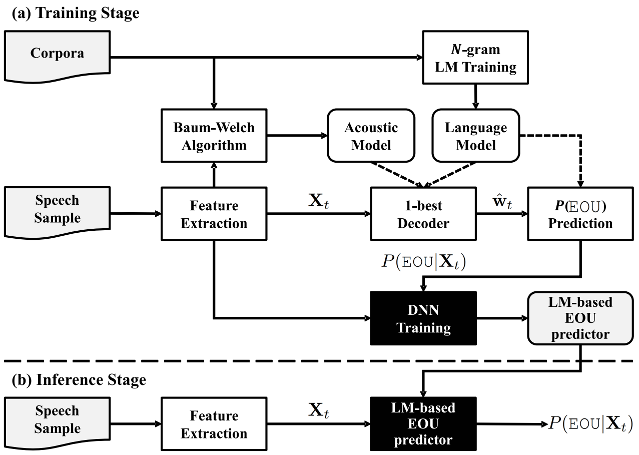


FIGURE 3. Overall pipeline of our proposed LM-based end-to-end EOU predictor.

LM-based EOU predictor using RNN is trained along with the targeted probability of the EOU token. The key idea is to train the LSTM network to minimize the mean square error (MSE) function for the LM-based EOU predictor, which is expressed as follows:

$$\mathbf{h}_{1,t}^{\mathrm{DE}} = \mathrm{RNN}(\mathbf{X}_t, \Theta_1^{\mathrm{DE}}) \tag{11}$$

$$\mathbf{h}_{2,t}^{\mathrm{DE}} = \mathrm{RNN}(\mathbf{h}_{1,t}^{\mathrm{DE}}, \Theta_2^{\mathrm{DE}}) \tag{12}$$

$$\hat{P}_{\mathrm{DE}}(\texttt{EOU}|\mathbf{X}_t) = \sigma(\mathbf{h}_{2,t}^{\mathrm{DE}}\mathbf{V}^{\mathrm{DE}} + \mathbf{b}^{\mathrm{DE}}) \tag{13}$$

where Θ_l^{DE} is the model parameter of l-th RNN layer and $\mathbf{h}_{l,t}^{\mathrm{DE}}$ denotes the hidden state of the l-th hidden layer at the t-th frame for the DE, respectively. Also, $\mathbf{V}^{\mathrm{DE}}$, $\mathbf{b}^{\mathrm{DE}}$, and σ denote the weight parameter, bias parameter, and logistic sigmoid function, respectively. Once the proposed end-to-end EOU predictor is completely trained, the framewise posterior probabilities of the EOU token are determined at the inference stage as in (11)–(13) without the actual ASR decoding process while eliminating the gray block as depicted in Fig. 1. Furthermore, they will be used as the LM knowledge for the final EPD decision.

B. AFE-BASED EPD

According to [31], the AFE-based EPD method can be used to classify each frame into four states, i.e., speech, initial silence, final silence, and intermediate silence to distinguish the final silence from the other silence states, where the high posterior probability of the final silence is likely to be the true endpoint. The AFE-based EPD is formulated as follows:

$$\mathbf{h}_{1,t}^{\mathrm{AFE}} = \mathrm{RNN}(\mathbf{X}_t, \Theta_1^{\mathrm{AFE}}) \tag{14}$$

$$\mathbf{h}_{2,t}^{\mathrm{AFE}} = \mathrm{RNN}(\mathbf{h}_{1,t}^{\mathrm{AFE}}, \Theta_2^{\mathrm{AFE}}) \tag{15}$$

$$\hat{P}_{\mathrm{AFE}}(\texttt{EOU}|\mathbf{X}_t) = \mathrm{softmax}(\mathbf{h}_{2,t}^{\mathrm{AFE}}\mathbf{V}^{\mathrm{AFE}} + \mathbf{b}^{\mathrm{AFE}}) \tag{16}$$

where Θ_l^{AFE} is the model parameter of l-th RNN layer and $\mathbf{h}_{l,t}^{\mathrm{AFE}}$ denotes the hidden state of the l-th hidden layer at the t-th frame for the AFE, respectively. Also, $\mathbf{V}^{\mathrm{AFE}}$ and $\mathbf{b}_{\mathrm{AFE}}$ denote the weight and bias parameters of the output layer, respectively. All the parameters in the LSTM for the AFE-based EPD are trained to minimize the cross-entropy (CE) error function.

C. PE-BASED AM

According to the previous studies [19], [20], [48], and [49] on the phone-aware training method using the latent feature of the DNN-based AM (called PE), the main idea can be further improved for other applications such as the speech enhancement and SAD tasks. Hence, in this study, we incorporate the PE for the EPD task to reduce the endpoint error. The PE-based ASR is derived as follows:

$$\mathbf{h}_{1,t}^{\mathrm{PE}} = \mathrm{RNN}(\mathbf{X}_t, \Theta_1^{\mathrm{PE}}) \tag{17}$$

$$\mathbf{h}_{2,t}^{\mathrm{PE}} = \mathrm{RNN}(\mathbf{h}_{1,t}^{\mathrm{PE}}, \Theta_2^{\mathrm{PE}}) \tag{18}$$

$$\hat{P}_{\mathrm{PE}}(y_t = c_t|\mathbf{X}_t) = \mathrm{softmax}(\mathbf{h}_{2,t}^{\mathrm{PE}}\mathbf{V}^{\mathrm{PE}} + \mathbf{b}^{\mathrm{PE}}) \tag{19}$$

where Θ_l^{PE} is the model parameter of l-th RNN layer and $\mathbf{h}_{l,t}^{\mathrm{PE}}$ denotes the hidden state of the l-th hidden layer at the t-th frame for the PE, respectively. Furthermore, $\mathbf{V}_{\mathrm{PE}}$ and $\mathbf{b}_{\mathrm{PE}}$ represent the weight and bias parameters of the output layer, respectively. It is expected that the LSTM better models the PE-based AM to minimize the CE error function with the framewise senone label c_t, which can be obtained by performing the forced alignment process based on the Gaussian mixture model-hidden Markov model (GMM-HMM)-based ASR system [50].

D. PROPOSED END-TO-END ENDPOINT DETECTION BASED ON ENSEMBLE RNNs

We propose the novel EPD framework that reduces the early and late endpoint times simultaneously by leveraging the AFE-based EPD algorithm with the acoustic and language modeling knowledge. As in [19]–[21], which show that the complementary advantages of multiple features can be easily combined using the DNN by injecting the features together, the last hidden layers of the PE-based AM and the proposed LM-based EOU predictor are concatenated with that of the AFE-based EPD algorithm as the acoustic modeling context and language modeling context, respectively. After the AFE-based EPD, PE-based AM, and LM-based EOU predictor are independently trained in accordance with each target, the ensemble RNNs are concatenated at the level of the last hidden layers and then fed into the DNN-based EPD classifier, which is used to classify each input frame into four states indicating the speech, initial silence, final silence, and intermediate silence, as follows:

$$\mathbf{h}_{1,t}^{\mathrm{EPD}} = \sigma([\mathbf{h}_{2,t}^{\mathrm{AFE}}, \mathbf{h}_{2,t}^{\mathrm{PE}}, \mathbf{h}_{2,t}^{\mathrm{DE}}]\mathbf{V}_1^{\mathrm{EPD}} + \mathbf{b}_1^{\mathrm{EPD}}) \tag{20}$$

$$\mathbf{h}_{2,t}^{\mathrm{EPD}} = \sigma(\mathbf{h}_{1,t}^{\mathrm{EPD}}\mathbf{V}_2^{\mathrm{EPD}} + \mathbf{b}_2^{\mathrm{EPD}}) \tag{21}$$

$$\hat{P}_{\mathrm{EPD}}(\texttt{EOU}|\mathbf{X}_t) = \mathrm{softmax}(\mathbf{h}_{2,t}^{\mathrm{EPD}}\mathbf{V}_3^{\mathrm{EPD}} + \mathbf{b}_3^{\mathrm{EPD}}) \tag{22}$$

where $\mathbf{h}_{l,t}^{\mathrm{EPD}}$ denotes the hidden state of the l-th layer at the t-th frame. In addition, $\mathbf{V}_l^{\mathrm{EPD}}$ and $\mathbf{b}_l^{\mathrm{EPD}}$ denote the weight and bias parameters at the l-th hidden layer, respectively. To build the model, the CE error function is directly applicable to the objective criterion, thus, the posterior probability of the final silence representing the speech endpoint is established. After the classifier based on the DNN is trained, all the modules including the ensemble RNNs for extracting the AFE, PE, and DE and the DNN for the classifier are dependently optimized again by the joint retraining (JRT) process, which is similar to phase 3 of [19], in accordance with the EPD label to further enhance the EPD performance, whereas they consist of differentiable parameters as shown in Fig. 1, which illustrates the feed-forward and error back-propagation paths.

In the inference stage, the probability of the EOU is computed by feeding the input acoustic feature sequence into the proposed EPD algorithm. The EOU is finally detected when $\hat{P}_{\mathrm{EPD}}(\mathtt{EOU}|\mathbf{X}_t)$ exceeds the probabilities corresponding to the speech, initial silence, and intermediate silence.

IV. EXPERIMENTS AND RESULTS

This section describes the performance evaluation of the proposed EPD approach. For the objective comparison, our approach was compared with the conventional GLDNN-based EPD [31] and the EPD based on combining the AFE, WE, and DSFs [39]. Since the DSFs-based approach in [39] and the proposed EPD approach are commonly based on the combination of the trained embeddings, such as [AFE, WE, DSFs] and [AFE, PE, DE], respectively, the performances of the sub-EPD systems based on single embedding alone and their combinations were tested also to verify the superiority of the DE for the proposed EPD algorithm. In [31] and [39], the performances of the EPD systems were evaluated using the following metrics. First, the EPD performances were assessed using the late endpoint time describing how the final EPD decision is triggered late compared with the EPD label. The late endpoint time can be considered as the response latency of the online speech recognition system since 1-best ASR decoding hypothesis can be obtained when the EPD is triggered. Besides, the EPD performances were compared using the WER since bad early endpoint errors undesirably cut off the speech region and increase the deletion error rate. In order to evaluate the performance of the EPD systems in terms of early endpoint error itself also, we reported the WER as well as the early endpoint time, which describes how the final EPD decision is prematurely triggered compared with the true EPD label. For the performance comparison, the EOU of each speech sample on the evaluation-dataset was obtained by independently performing the EPD systems, and then the EPD performances were assessed in the following. The early endpoint time was obtained by averaging the gap between the actual EOU and the moment the EPD algorithm was triggered within the speech samples for which the EPD approach was prematurely triggered. The late endpoint time was obtained by averaging the gap between the actual EOU and the moment the EPD algorithm was triggered within the speech samples for which the EPD decision was triggered late. Then, the WER was evaluated by performing the ASR decoding process from the first frame to the EOU frame determined by each EPD algorithm, while the WER was computed by the summation of the substitution, deletion, and insertion error rates [51].

The first part of the experiments used a relatively small speech dataset, namely CHiME-3, to evaluate and analyze the conventional and proposed EPD algorithms with various acoustic configurations. The second part of the experiments scaled up the size of utterances to be augmented by using the acoustic environment simulation method with the clean speech database, namely SiTEC Dict01 [52]. These experiments mainly show the effectiveness of the proposed EPD framework. Note that all the frameworks were implemented using the TensorFlow library [53].

A. CHiME-3 ASR TASK

1) DATA PREPARATION

We emphasize that the simulation must be very realistic; hence, we selected the CHiME-3 dataset [45] developed for the far-field ASR task with a multi-microphone tablet device in everyday environments, i.e., a bus, cafe, pedestrian area, and street, each of which consists of real speech data (REAL) and simulated speech data (SIMU). The real speech data consist of six-channel recordings and were sampled at 16 kHz. Twelve English speakers were asked to read the sentences from the WSJ0 corpora [54] while using the multi-microphone tablet. They were encouraged to adjust their reading positions so that the target distance continued to change over time. The simulated speech data were generated by artificially mixing the clean utterances from the WSJ0 into background recordings. The speech data consists of three datasets, including the training-, development-, and evaluation-datasets, which have 18 h of speech data (3 h REAL and 15 h SIMU) uttered by 87 speakers, 2.9 h of speech data uttered by 4 speakers, and 2.2 h of speech data uttered by 4 speakers, respectively. The development- and evaluation-datasets have a 1:1 ratio of REAL and SIMU. We used the training- and development-datasets for the training of each EPD framework and the evaluation-dataset for the performance comparison. In particular, "Beamformit", which is a weighted delay-and-sum beamforming algorithm [55] was performed to extract the speech signal of interest from background noise. Note that the beamforming algorithm was carried out using only five microphones facing the speaker, and we excluded the second microphone since it was located on the rear side of the tablet device and contained less speech.

To prepare the senone targets and $P(\mathtt{EOU}|\mathbf{X}_t)$ labels used to train the AM and the LM-based-EOU predictor, respectively, we used the baseline ASR system of the CHiME-3 task provided by the KALDI framework [56]. The baseline ASR system was prepared using the training- and development-datasets described as follows. First, the training- and development-datasets were represented with 25 ms frames of 13-dim MFCC features computed every 10 ms with the Hamming window. We obtained 1,952 types of senone labels by training the GMM-HMM-based ASR system with a 40-dim feature space maximum likelihood linear regression (fMLLR) context by speaker adaptive training (SAT), whereas the input feature was spliced with three left and three right feature frames (91-dimension). Subsequently, the DNN for the AM, which has 7 hidden layers and 2,048 hidden units with the sigmoid activation function on each hidden layer, was trained as in the following steps. First, each hidden layer of the DNN was initialized via the layerwise unsupervised learning process called pre-training by the contrastive divergence (CD) algorithm [57]. Then, the DNN was trained to minimize the CE error function, whereas the DNN input was also spliced with five left and five right fMLLR contexts (440-dimension). Finally, the DNN was trained again with the state-level minimum Bayes risk (sMBR) criteria [58]. The 3-gram LM was used for the baseline ASR system, which was developed within the 5k vocabulary and pruned by the pre-defined threshold values. After the DNN-based AM was trained, the senone labels of each dataset were prepared by performing the ASR decoding process. Furthermore, the framewise $P(\texttt{EOU}|\mathbf{X}_t)$ labels of each dataset were established according to the word alignment, which was obtained from the 1-best ASR decoding hypothesis. In addition, we made framewise reference EPD decisions on the enhanced speech data of each dataset by manually labeling each frame as speech, initial silence, final silence, and intermediate silence for every 10 ms.

2) TRAINING PROCESS FOR EACH EPD MODEL

The proposed EPD framework was constructed as follows. First, the training- and development-datasets were represented with 25 ms frames of 64-dim log-Mel filterbank energies computed every 10 ms, which were used as the input feature for the EPD task. The AFE-based EPD and PE-based AM consisted of two LSTM layers with 100-dim cells per layer and the fully-connected DNN-based classifier with the soft-max function, yielding the 4-dim and 1,952-dim output, respectively, for classifying the input frame into four types of states, which are speech, initial silence, final silence, and intermediate silence frames and senone labels, respectively. The EOU predictor-based EPD also consisted of two LSTM layers with 100-dim cells per layer and the fully-connected DNN yielding the 1-dim output layer through the sigmoid logistic function. The EOU predictor-based EPD was trained with the MSE function where the probabilities of the EOU token $P(\texttt{EOU}|\mathbf{X}_t)$ obtained using the N-gram LM were used to train the EOU predictor. After they were trained, their last hidden layers were concatenated to train the EPD classifier consisting of two 100-dim fully-connected DNN layers and the 4-dim soft-max layer. The batch size was set to 64. The learning rates for the training of the AFE-based EPD, PE-based AM, LM-based EOU predictor, and classifier were set to 0.01, 0.01, 0.001, and 0.01, respectively, for the first 10 epochs, and then decreased by 10% after each epoch. When the proposed EPD architecture was jointly retrained for further optimization, the initial learning rate was set to 0.0001, and then decreased by 10% upon each epoch.

For the EPD performance comparisons, the conventional EPD approaches were established as follows. For the GLDNN-based EPD [31], the grid-LSTM used 12-dim grid-LSTM units, where the filter size was 8 with the stride 2 (overlapped by 6). Furthermore, two LSTM layers with 64-dim cells per layer and two 100-dim fully-connected DNN layers with the 4-dim soft-max layer were cascaded. The GLDNN-based EPD was trained with the 64-dim log-Mel filterbank energies in accordance with four types of labels: speech, initial silence, final silence, and intermediate silence. The batch size and learning rate were set to 64 and 0.01, respectively. As for [39], 64-dim log-Mel filterbank energies were also used as the feature. The acoustic LSTM and word LSTM were constructed with two LSTM layers with 100-dim cells per layer and the fully-connected DNN-based classifier. The acoustic LSTM was trained in accordance with the four types of EPD labels, which are speech, initial silence, final silence, and intermediate silence unlike [39] since the post-processing for safeguarding the lower and upper pause duration bounds was not used for a reasonable comparison. The word LSTM was also trained with the binary labels, which were given depending on whether the turn-taking word or not and were obtained by performing the ASR baseline for the CHiME-3 task. After the acoustic LSTM and word LSTM were trained, the classifier consisting of the two 100-dim fully-connected DNN layers with 4-dim soft-max function was trained from the sequential features, which were composed of the last hidden layer of both LSTMs and the DSFs (three types of expected pause durations), which were obtained by performing the ASR decoding process in an online manner. The batch size and learning rate were set to 64 and 0.01, respectively, for training the acoustic LSTM, word LSTM, and classifier.

The sub-EPD systems based on the single embedding alone or their combinations were built also in order to verify the superiority of the DE for the proposed EPD. Specifically, the embeddings including the AFE, PE, WE, and DE were prepared by feeding the training-dataset into the AFE-based EPD, AM, word LSTM, and proposed LM-based EPD system and capturing the hidden states at the level of the last hidden layer, respectively. The classifiers of the sub-EPD system were separately trained in accordance with the framewise endpoint target by feeding the single embedding alone or their combinations into the EPD classifiers, while the CE error function was used. The batch size and learning rate were set to 64 and 0.01, respectively, for training each EPD classifier.

The Adam optimization algorithm [59] was commonly applied for all the training processes. Furthermore, an early stopping scheme was performed using the development-dataset to avoid the over-fitting, after 50 epochs were completed.

3) EXPERIMENTAL RESULTS

Before demonstrating our experiments, we assessed the performance of the EPD systems based on the various DEs, which were obtained by the N-gram LM with different orders, since the performance of the LM-based EOU predictor for the EPD task is highly dependent on not only the DNN architecture but also the N-gram LM used to build the targets. Table 1 shows the average early and late endpoint times obtained within the development-dataset, where DE_N denotes the EPD based on the DE trained using the N-gram LM, and the bold numbers indicate the best result among the DE-based EPD systems. The performances of the AFE-, PE, and WE-based EPD algorithms are also reported for the relative performance comparison. The DE trained with the 4-gram LM achieved lower endpoint errors compared with the others; thus, we used the 4-gram LM for training the EOU predictor.

TABLE 1. Performance comparison of the DE-based EPD algorithms, which were trained with the *N*-gram LM of different orders. All time values are in sec.

Endpoint Time	[AFE]	[PE]	[WE]	[DE_2]	[DE_3]	[DE_4]	[DE_5]
Early	2.206	2.242	2.070	2.140	1.932	**1.834**	2.050
Late	0.214	0.201	0.254	0.233	0.223	**0.219**	0.231

The proposed EPD algorithm and the conventional methods were extensively evaluated on the CHiME-3 ASR task to assess the EPD performance under the bus, cafe, pedestrian, and street scenarios for both the simulated acoustic condition and the everyday environment. Fig. 4 shows an example of the prediction result of $P(\texttt{EOU}|\mathbf{X}_t)$ and the final EPD decision according to each EPD algorithm under the REAL bus noise scenario, where this example includes the short pause regions at 2.4, 3.6, and 4.2 s and the long pause region from 4.8 to 5.0 s. As shown in Fig. 4(b), $P(\texttt{EOU}|\mathbf{X}_t)$ was observed to be high in the short and long pause regions and is likely to detect the short pause as an endpoint since the GLDNN-based EPD cannot fully consider the language modeling knowledge such as the phone or word alignments. Especially, the probability of the EOU was sufficiently high to trigger the final EPD decision prematurely in the short and long pause regions. In contrast, the final EPD decision of the proposed EPD algorithm was correctly triggered in the final silence region. Further, the late endpoint time could be reduced by the JRT process. The performance of the proposed EPD algorithm was compared with that of the conventional EPD approaches in terms of objective measures described as follows.

First, the performance of each EPD algorithm was evaluated in terms of the early endpoint time. Table 2 shows the performance comparison for the conventional and proposed EPD algorithms under the various acoustic conditions in terms of the early endpoint time where the bold numbers indicate the best result in terms of the early endpoint time. In Table 2, the [embeddings] denotes the EPD system based on the given embeddings, where the CLDNN, [AFE, WE, DSFs], and [AFE, DE, PE] with or without JRT indicate [31], [39], and the proposed EPD framework, respectively. As shown in Table 2, it was evident that the [AFE] and [PE] classifiers yielded a higher endpoint error compared with the [WE] and [DE] classifiers, which were trained based on the 1-best ASR decoding hypothesis.

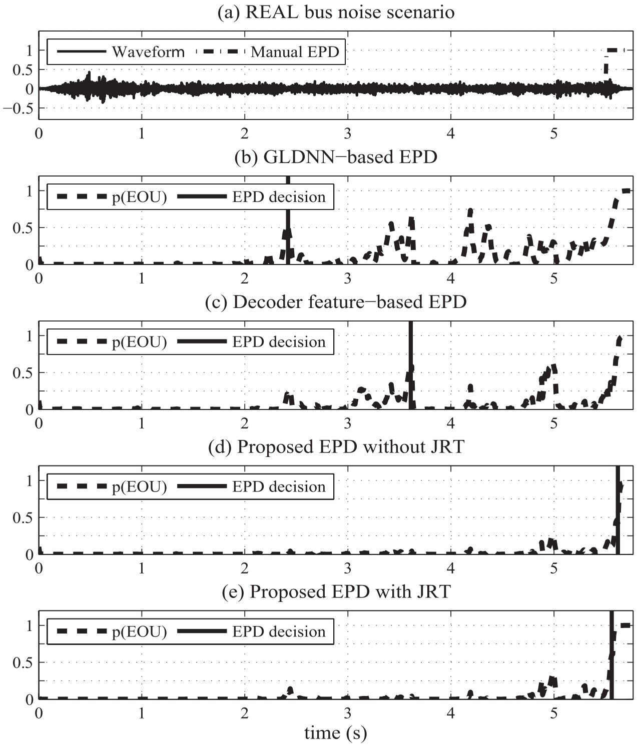


FIGURE 4. Performance evaluation of endpoint detection algorithms in the REAL bus noise condition. This example includes the short pause regions at 2.4, 3.6, and 4.2 s and the long pause region from 4.8 to 5.0 s.

These results indicate that the WE and DE are useful features for the EPD task to avoid the early endpoints since they can distinguish the EOU from the intermediate silence well compared with the AFE and PE, which were trained without considering the context of the input feature sequence. Furthermore, the [DE] classifier achieved a better EPD performance than the [WE] classifier, where the WE was trained to detect the turn-taking word and it cannot be considered reliable for the natural language, as mentioned earlier. And, the performance of the [AFE] classifier can be improved by incorporating the WE or DE as an additional input feature. Especially, the [AFE, WE] classifier showed a higher endpoint error than the [AFE, DE] classifier, which is more desirable for the EPD task regarding the natural language. The GLDNN-based EPD algorithm, which can be considered as the complex version of the [AFE] classifier, showed a lower early endpoint error than the single embedding-based EPD system. However, the EPD systems based on their combination outperformed the GLDNN-based EPD approach in terms of the early endpoint time. The additional use of the DSFs for the EPD task could enhance the EPD performance of the [AFE, WE] classifier. Furthermore, the proposed EPD algorithm, namely [AFE, PE, DE] classifier, showed a superior EPD performance compared with that of the conventional EPD algorithms under the overall acoustic conditions, and the early endpoint time of the proposed EPD algorithm was further improved by the JRT process as reported in Table 2.

TABLE 2. Performance comparison of the conventional and proposed EPD approaches for the CHiME-3 in terms of early endpoint time. All time values are in ms.

Conditions		GLDNN	[AFE]	[PE]	[WE]	[DE]	[AFE, WE]	[AFE, DE]	[AFE, WE, DSFs]	[AFE, DE, PE]	
Locations	Types									without JRT	with JRT
Bus	SIMU	-2169.34	-2843.14	-2781.73	-2341.13	-2298.80	-1985.98	-1899.23	-1845.56	-1781.32	**-1628.24**
	REAL	-2132.85	-2797.64	-2726.67	-2313.54	-2258.05	-1943.87	-1862.50	-1802.28	-1726.22	**-1592.30**
Cafe	SIMU	-1992.10	-2420.13	-2344.73	-2131.25	-2079.79	-1958.53	-1863.09	-1818.53	-1772.21	**-1690.16**
	REAL	-2052.67	-2630.44	-2557.88	-2226.05	-2148.47	-1991.55	-1910.60	-1856.02	-1801.35	**-1731.82**
Pedestrian	SIMU	-2217.11	-2644.72	-2591.52	-2371.48	-2367.01	-2084.58	-1998.99	-1926.66	-1880.62	**-1810.50**
	REAL	-2289.18	-2761.74	-2707.90	-2446.97	-2410.60	-2102.93	-2037.06	-1991.84	-1926.29	**-1847.22**
Street	SIMU	-1877.72	-2183.94	-2134.49	-2079.37	-2024.08	-1857.87	-1784.13	-1762.63	-1704.29	**-1622.86**
	REAL	-2092.79	-2561.78	-2490.26	-2223.10	-2180.59	-1975.61	-1907.52	-1876.16	-1789.81	**-1722.98**
Average		-2102.97	-2605.44	-2541.90	-2266.61	-2220.92	-1987.62	-1907.89	-1859.96	-1797.76	**-1705.76**

Moreover, the performance of each EPD algorithm was compared in terms of the late endpoint time. Table 3 shows the performance comparison for the conventional and proposed EPD algorithms under the various acoustic conditions in terms of the late endpoint time. From Table 3, it is evident that the [WE] classifier exhibited the highest endpoint time among the single embedding-based EPD architectures. Furthermore, the late endpoint time of the [AFE] classifier was reduced by using the WE or DE as an additional feature for the EPD task, while the DE can be considered as a more reliable feature for the EPD task compared with the WE. The proposed EPD framework yielded a superior EPD performance than the conventional EPD approach in terms of the late endpoint time. Moreover, the late endpoint time was further reduced by the JRT process.

The performance of each EPD algorithm was also assessed in terms of the WER by using the baseline ASR system. The EOU frame of each speech utterance was obtained by performing each EPD algorithm and then the ASR decoding was accomplished from the first frame to the EOU frame determined by each EPD algorithm. As reported in Table 4, the proposed EPD algorithm also achieved a better performance than the conventional EPD approaches, and the WERs were further improved by the JRT scheme since this scheme can enhance the early and late endpoint times. The final decision of each EPD algorithm can also be obtained based on a soft decision instead of a hard decision. The EPD decision makes the trade-off between a quick endpoint and avoiding cutting off the speech uttered by the user. More specifically, an aggressive decision threshold provides a faster response at the expense of increasing the WER, whereas a lower WER increases the late endpoint time. To show the trade-off between the WER and the late endpoint time for each EPD algorithm, the WER-median late endpoint time curve is shown in Fig. 5, which was obtained by varying the decision threshold; here, the lower curves are better. As shown in Fig. 5, the median late endpoint times of the GLDNN-based EPD approach, DSFs-based EPD approach, and the proposed EPD algorithm without and with the JRT are approximately 270, 230, 190, and 170 ms, respectively, with the same WER of approximately 20%. As shown above, the proposed EPD algorithm with the JRT process showed a better EPD performance than the conventional EPD approaches.

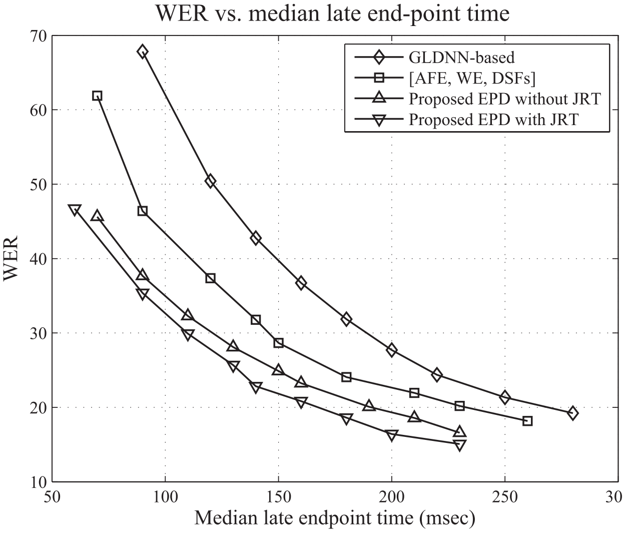


FIGURE 5. Performance evaluation of endpoint detection algorithms in terms of WER-median late endpoint time curve.

B. LARGE-SCALE ASR TASK

1) DATA PREPARATION

To assess the EPD performance of the conventional and proposed EPD approaches with large corpora, we used a large vocabulary continuous Korean speech dataset, namely DICT01, developed by the Speech Information Technology and Industry Promotion Center (SiTEC) [52]. This dataset consists of 20,833 sentences, each containing 6 to 25 words (average: 7.63 words). The speech database was recorded with 200 males and 200 females and each speaker uttered 104 or 105 sentences. The speech signal was sampled at 16 kHz where the recording conditions are described in [52]. We randomly divided the speech database into three datasets, which are the training-dataset (160 males and 160 females), development-dataset (20 males and 20 females), and evaluation-dataset (20 males and 20 females). We made the reference decision on the clean speech data of each dataset by manually labeling each frame as four types of state, which are speech, initial silence, final silence, and intermediate silence, for every 10 ms.

We constructed a noisy and reverberant speech database using an image method [60] for a comparison among the

TABLE 3. Performance comparison of the conventional and proposed EPD approaches for the CHiME-3 in terms of late endpoint time. All time values are in ms.

Conditions		GLDNN	[AFE]	[PE]	[WE]	[DE]	[AFE, WE]	[AFE, DE]	[AFE, WE, DSFs]	[AFE, DE, PE]	
Locations	Types									without JRT	with JRT
Bus	SIMU	182.48	207.16	193.77	245.34	211.66	179.74	174.27	170.12	153.21	**153.14**
	REAL	198.40	225.70	225.53	251.93	228.41	186.07	182.31	182.30	171.65	**164.40**
Cafe	SIMU	179.75	209.72	190.07	245.55	213.80	169.04	163.88	162.96	154.87	**131.64**
	REAL	213.51	247.66	238.06	292.67	266.69	212.16	208.43	201.72	193.48	**184.40**
Pedestrian	SIMU	191.27	205.43	194.77	251.14	212.80	184.56	180.66	169.33	162.23	**150.84**
	REAL	211.88	239.49	216.71	274.06	247.71	197.29	188.68	187.91	181.60	**173.37**
Street	SIMU	189.92	213.11	199.87	258.13	233.57	165.99	159.98	158.16	146.53	**132.36**
	REAL	217.82	255.56	226.53	272.67	255.82	212.72	204.60	200.42	187.99	**183.73**
Average		198.13	225.48	210.66	261.44	233.81	188.45	182.85	179.12	168.95	**159.24**

TABLE 4. Performance comparison of the conventional and proposed EPD approaches for the CHiME-3 in terms of WER (%).

Conditions		GLDNN	[AFE]	[PE]	[WE]	[DE]	[AFE, WE]	[AFE, DE]	[AFE, WE, DSFs]	[AFE, DE, PE]		Ground truth
Locations	Types									without JRT	with JRT	
Bus	SIMU	26.34	36.72	35.70	31.76	31.36	26.34	23.96	23.65	22.75	**20.26**	11.72
	REAL	38.79	42.36	39.73	37.67	35.88	38.72	35.26	34.48	30.68	**30.61**	23.77
Cafe	SIMU	32.37	39.45	38.56	37.18	35.99	29.59	29.08	28.58	27.63	**27.61**	16.91
	REAL	25.15	29.09	28.24	23.78	22.35	23.92	23.83	23.18	23.02	**20.57**	14.61
Pedestrian	SIMU	29.55	41.73	40.29	35.35	34.48	28.29	27.21	26.38	24.74	**24.16**	16.38
	REAL	25.72	29.01	25.28	23.28	22.81	25.43	25.24	24.81	24.04	**20.34**	13.08
Street	SIMU	29.75	39.75	38.04	34.30	34.15	29.71	29.40	27.63	27.40	**24.64**	18.29
	REAL	21.83	25.34	23.95	20.77	20.44	20.77	19.23	19.04	18.07	**14.70**	11.39
Average		28.69	35.43	33.72	30.51	29.68	27.85	26.65	25.97	24.79	**22.86**	15.77

TABLE 5. Performance comparison of the conventional and proposed EPD approaches for the large-scale ASR task in terms of early endpoint time. All time values are in ms.

Conditions		GLDNN	[AFE]	[PE]	[WE]	[DE]	[AFE, WE]	[AFE, DE]	[AFE, WE, DSFs]	[AFE, DE, PE]	
Locations	SNRs									without JRT	with JRT
Bus	5	-1314.29	-1719.80	-1617.80	-1509.52	-1237.07	-1171.97	-1081.64	-998.21	-965.89	**-881.07**
	10	-798.09	-897.65	-860.31	-820.76	-738.39	-709.54	-649.41	-641.81	-609.28	**-520.60**
	15	-424.56	-497.22	-493.08	-470.43	-456.56	-412.02	-402.52	-378.36	-359.49	**-292.77**
	20	-233.95	-350.71	-300.96	-290.78	-237.68	-229.79	-229.17	-219.83	-167.16	**-146.86**
Cafe	5	-1400.83	-1734.84	-1686.45	-1586.03	-1312.81	-1242.91	-1158.98	-1037.97	-1027.55	**-932.13**
	10	-800.99	-994.55	-990.90	-969.54	-773.56	-722.93	-653.75	-643.91	-600.19	**-512.65**
	15	-478.85	-585.30	-575.52	-557.05	-504.54	-450.38	-415.13	-383.50	-375.47	**-293.33**
	20	-285.34	-340.31	-321.59	-299.76	-299.43	-262.41	-239.59	-230.08	-192.01	**-151.30**
Pedestrian	5	-1408.83	-1862.43	-1775.07	-1682.40	-1338.87	-1279.14	-1163.16	-1120.06	-1056.24	**-969.42**
	10	-1009.74	-1128.45	-1087.87	-1049.79	-953.46	-901.47	-808.06	-735.61	-720.08	**-608.09**
	15	-569.59	-694.57	-677.30	-646.77	-582.92	-531.46	-452.92	-448.61	-439.64	**-334.85**
	20	-321.44	-388.92	-366.74	-357.53	-326.70	-275.38	-254.12	-252.88	-209.42	**-179.75**
Street	5	-1341.00	-1608.46	-1518.37	-1383.34	-1254.63	-1164.71	-1085.92	-1010.27	-935.70	**-872.14**
	10	-813.15	-888.76	-857.39	-832.21	-765.33	-741.45	-668.61	-654.98	-623.32	**-525.56**
	15	-470.63	-545.31	-523.09	-521.03	-497.36	-423.01	-377.17	-374.62	-354.46	**-281.32**
	20	-269.49	-323.45	-316.97	-293.82	-292.72	-256.20	-236.34	-228.51	-192.97	**-153.58**
Office	5	-1747.47	-1874.06	-1845.56	-1811.70	-1581.37	-1572.96	-1561.51	-1507.38	-1221.30	**-1165.59**
	10	-1068.02	-1246.65	-1219.12	-1178.35	-961.75	-956.15	-940.34	-878.68	-744.67	**-687.18**
	15	-598.06	-881.13	-857.70	-808.95	-678.95	-575.76	-562.30	-491.20	-441.86	**-371.12**
	20	-358.89	-623.79	-608.41	-558.08	-416.41	-330.95	-321.10	-250.78	-239.37	**-186.11**
Average		-785.66	-959.32	-925.01	-881.39	-760.53	-710.53	-663.09	-624.36	-573.80	**-503.27**

EPD approaches under the various acoustic conditions similar to real-life environments. We first simulated the reverberant environments by convolving the clean speech of training-, development-, and evaluation-datasets with the room impulse responses, which correspond to small rooms of the REVERB challenge dataset for which the reverberation time T_{60} is approximately 0.25 s [61]. Then, the bus, cafe, pedestrian, and street noises obtained from CHiME-3 [45] were artificially added to each reverberant speech dataset in a time-domain while maintaining the signal-to-noise ratio (SNR) at 5, 10, 15, and 20 dB. In addition, office noise from YouTube was artificially added to the reverberant speech of the evaluation-dataset to evaluate the performances of the conventional and proposed EPD approaches under the unseen acoustic condition. Consequently, approximately 1,342, 171, and 168 h of noisy and reverberant speech data of the training-, development-, and evaluation-datasets were prepared, respectively.

2) TRAINING PROCESS FOR EACH EPD MODEL

We constructed each EPD framework by using large corpora for the performance comparison among the conventional

TABLE 6. Performance comparison of the conventional and proposed EPD approaches for the large-scale ASR task in terms of late endpoint time. All time values are in ms.

Conditions		GLDNN	[AFE]	[PE]	[WE]	[DE]	[AFE, WE]	[AFE, DE]	[AFE, WE, DSFs]	[AFE, DE, PE]	
Locations	SNRs									without JRT	with JRT
Bus	5	249.91	283.43	272.64	314.36	283.95	264.58	245.67	231.04	217.38	**208.64**
	10	157.71	186.93	182.09	215.60	193.89	174.48	156.88	140.50	134.02	**132.68**
	15	117.54	140.98	138.73	167.92	146.68	132.13	113.04	102.94	96.08	**89.06**
	20	104.28	128.79	122.67	155.98	133.39	117.27	101.23	86.00	78.69	**76.63**
Cafe	5	256.75	290.05	283.87	321.41	298.26	273.62	253.06	239.77	226.81	**225.74**
	10	171.64	202.77	192.01	227.83	207.58	187.62	170.75	152.13	146.25	**145.59**
	15	124.41	150.04	141.69	173.09	152.52	136.38	120.99	108.67	103.10	**102.05**
	20	108.22	134.29	128.64	162.97	139.18	119.35	102.82	92.40	88.48	**84.80**
Pedestrian	5	287.71	316.22	314.45	344.87	324.04	302.10	283.75	265.07	257.11	**248.07**
	10	190.49	216.38	212.28	246.88	225.44	200.94	187.44	169.70	164.52	**162.60**
	15	134.77	163.50	157.63	185.59	164.92	149.52	134.16	124.18	112.81	**105.49**
	20	114.59	144.35	135.17	165.53	148.75	127.43	112.94	101.77	92.11	**91.56**
Street	5	231.29	267.63	255.80	292.16	270.62	245.86	227.39	210.71	206.81	**195.87**
	10	167.34	199.33	186.40	220.98	202.95	180.34	161.81	147.86	137.83	**135.00**
	15	116.97	140.55	139.70	170.10	146.84	130.98	115.35	100.49	91.10	**84.46**
	20	99.87	128.68	119.59	154.43	129.72	112.50	98.81	90.46	74.06	**73.19**
Office	5	287.73	315.15	315.05	355.67	349.15	306.91	283.57	275.84	264.23	**263.84**
	10	188.54	219.73	217.91	250.28	242.89	200.29	187.64	173.86	170.59	**162.11**
	15	155.58	194.65	183.44	224.40	214.46	178.14	154.57	147.48	137.87	**136.50**
	20	137.43	170.92	167.20	206.54	196.54	157.27	135.20	122.74	117.88	**116.25**
Average		170.14	199.72	193.35	227.83	208.59	184.89	167.35	154.18	145.89	**142.01**

TABLE 7. Performance comparison of the conventional and proposed EPD approaches for the large-scale ASR task in terms of WER (%).

Conditions		GLDNN	[AFE]	[PE]	[WE]	[DE]	[AFE, WE]	[AFE, DE]	[AFE, WE, DSFs]	[AFE, DE, PE]		Ground
Locations	SNRs									without JRT	with JRT	truth
Bus	5	20.32	23.04	22.98	20.62	18.66	18.64	17.74	17.73	16.73	**14.74**	8.53
	10	14.45	16.67	16.45	14.52	13.13	13.03	12.56	12.42	12.10	**10.81**	6.54
	15	9.66	12.13	11.80	10.37	9.79	9.63	9.24	8.76	8.65	**8.10**	5.07
	20	8.27	10.27	9.79	8.61	8.41	8.04	7.71	7.25	7.24	**6.76**	3.89
Cafe	5	22.54	25.54	24.87	22.61	20.81	20.47	20.28	19.82	19.76	**17.70**	9.03
	10	15.65	18.21	17.73	16.95	15.03	15.02	14.20	14.03	13.28	**11.94**	5.60
	15	11.71	13.61	13.34	12.38	11.35	11.14	10.46	10.45	9.63	**8.84**	4.30
	20	9.56	11.46	11.24	10.43	9.67	9.41	8.83	8.69	7.96	**7.50**	3.56
Pedestrian	5	29.24	33.04	32.48	30.07	28.05	26.91	25.77	25.53	24.53	**22.71**	12.39
	10	20.85	24.41	23.85	22.37	21.29	20.14	19.15	18.97	18.10	**16.81**	8.27
	15	15.44	18.11	17.39	16.66	16.29	15.18	14.36	14.07	13.38	**12.39**	6.50
	20	13.19	15.47	14.64	14.15	13.60	12.83	12.34	12.05	10.68	**9.92**	5.29
Street	5	24.29	28.84	27.98	25.07	23.08	22.29	21.79	20.77	18.57	**17.23**	8.28
	10	17.59	21.04	20.46	18.16	16.82	16.24	15.06	15.04	12.09	**11.29**	6.11
	15	12.99	15.60	15.07	13.68	12.33	12.15	11.39	11.35	9.16	**8.54**	4.78
	20	10.38	13.37	12.78	11.59	11.11	10.04	9.54	8.86	7.32	**6.68**	3.86
Office	5	36.43	39.72	38.11	37.19	34.59	33.91	33.44	32.15	30.87	**30.24**	19.43
	10	27.51	29.92	28.73	27.89	26.00	25.42	24.33	23.04	22.21	**21.73**	13.88
	15	19.67	23.58	22.47	21.08	18.96	16.48	16.47	15.24	14.77	**14.46**	9.91
	20	13.57	18.78	16.68	15.78	15.17	12.83	12.48	10.34	10.03	**9.78**	6.79
Average		17.67	20.64	19.94	18.51	17.21	16.49	15.86	15.33	14.35	**13.41**	7.60

and proposed EPD approaches for the large-scale ASR task. The experimental setup was similar to that of the previous CHiME-3 ASR task. First, the ASR baseline was trained to obtain the senones and framewise $P(\mathrm{EOU}|\mathbf{X}_t)$ labels, which were respectively used to train the PE and LM-based EOU predictor as follows. The SAT algorithm was carried out with the fMLLR features extracted from each utterance of the training-dataset to extract the forced alignment. After the DNN for the AM was initialized via the pre-training procedure based on the CD algorithm [57], it was trained with the CE error function and then trained again with the sMBR criteria. In each step, as in the CHiME-3 task, the development-set of large corpora was used for the early stopping scheme. The ASR decoding process was performed with the training-dataset of large corpora to prepare the senone labels for the training of the PE model. Furthermore, the framewise $P(\mathrm{EOU}|\mathbf{X}_t)$ labels of the training-dataset were prepared using the 4-gram LM and the 1-best hypothesis obtained from the built-in ASR system.

Second, the conventional GLDNN-based EPD algorithm and DSFs-based EPD algorithm were constructed with the configurations similar to the experimental setup of the CHiME-3 task. The ensemble RNNs for the proposed LM-based EOU predictor, the PE-based AM, and the AFE-based EPD modules were separately trained in accordance with the $P(\mathrm{EOU}|\mathbf{X}_t)$ label and senones, which were obtained by performing the ASR system described above and the hand-made EPD label, respectively. Subsequently, their last hidden layers were concatenated to be fed into a fully-connected DNN-based classifier, which was then trained according to the EPD label. Finally, the proposed EPD framework was jointly retrained to optimize the EPD performance further. During all the training processes, the development-dataset was used to perform the early stopping scheme after 50 epochs.

3) EXPERIMENTAL RESULTS

The performances of the EPD frameworks for the large-scale ASR task were also evaluated in terms of the early endpoint time, late endpoint time, and WER by using the evaluation-dataset of the large corpora we prepared in this study.

First, the proposed and conventional EPD algorithms were evaluated in terms of the early endpoint time under the reverberant and noisy conditions, including the bus, cafe, pedestrian, street, and office environments. Table 5 shows the early endpoint time of each EPD approach for the large-scale ASR task, where the bold numbers indicate the best result in terms of the early endpoint time. It is shown in Table 5 that the [WE] and [DE] classifiers, which were trained according to the 1-best ASR decoding hypothesis, yielded a relatively lower early endpoint time compared with the [AFE] and [PE] classifiers, which were trained without considering the context of the input sequences such as the word or phone alignments. The [WE] classifier yielded a higher early endpoint time compared with the GLDNN-based EPD method under the overall acoustic conditions. In contrast, the [DE] classifier achieved a better EPD performance than the GLDNN-based EPD method under most of the low-SNR conditions in terms of the early endpoint time. The early endpoint time of the [AFE] classifier was significantly improved by incorporating the WE or DE. From these results, it is concluded that the context-dependent embeddings such as the WE and DE can prevent the early endpoint within the short or long pause regions. While the performance of the [AFE, DE] classifier was enhanced by using the DSFs as the additional feature, the proposed EPD framework yielded a superior EPD performance which was further improved by the JRT process. Note that the proposed EPD algorithm also showed considerable performance improvement under the office noise environment as summarized in Table 5, where the office is the unseen acoustic condition; hence, it was not used in the training-step.

Second, the proposed and conventional EPD algorithms were evaluated in terms of the late endpoint time. Table 6 summarizes the late endpoint time of the EPD approaches for the large-scale ASR task, where the bold numbers indicate the best result in terms of the late endpoint time. While the [WE] classifier showed the highest late endpoint time among the EPD approaches based on the single embedding alone, the [DE] classifier achieved the late endpoint time that is relatively closer to that of the [AFE] and [PE] classifiers. The endpoint error of the [AFE] classifier was considerably improved by additionally employing the WE or DE. Note that the late endpoint time of the [AFE, WE] classifier was further improved with the help of the DSFs, which can be obtained by the online ASR decoding process with a great deal of computation and a large amount of memory. Notably, the proposed EPD scheme yielded superior EPD performance, without the actual ASR decoding process, in terms of the late endpoint, which was further improved by the JRT process.

Finally, the proposed and conventional EPD algorithms were evaluated in terms of the WER. Table 7 shows the WER, which was obtained by performing the ASR system from the first frame to the EOU frame determined by each EPD algorithm. As shown in Table 7, the proposed EPD approach yielded better performance in terms of the WER with the help of the superiority of the proposed EPD architecture, especially in terms of the early endpoint time. Overall, the proposed EPD algorithm outperformed the conventional EPD approaches under both the seen and unseen noise conditions.

V. CONCLUSION

In this paper, we proposed the speech EPD strategy for the robust online low-latency speech recognition by combining the AFE, DE, and PE to incorporate the acoustic and language modeling knowledge into the AFE-based EPD.

The first contribution of this study is to investigate the LM-based EOU predictor using the RNN to derive the framewise probabilities of EOU token given input speech without the actual decoding process to consider the decoder states which are particularly useful for the EPD task but demands a great deal of computation and a large amount of memory. Second, we present the novel EPD architecture that can be constructed by combining the last hidden states of the AE-based EPD, the PE-based AM, and LM-based EOU predictor and training the DNN-based classifier in accordance with the framewise endpoint label and be further enhanced by the JRT technique.

The superiority of the proposed EPD algorithm was assessed under the CHiME-3 and large-scale ASR tasks. According to the experimental results, the proposed EPD algorithm showed a significantly improved EPD performance in terms of both the endpoint accuracy and the WER.

REFERENCES

[1] N. G. Ward, A. G. Rivera, K. Ward, and D. G. Novick, "Root causes of lost time and user stress in a simple dialog system," in *Proc. Interspeech*, 2005, pp. 1565–1568.

[2] A. Raux, D. Bohus, B. Langner, A. W. Black, and M. Eskenazi, "Doing research on a deployed spoken dialogue system: One year of let's go! experience," in *Proc. Interspeech*, 2006, pp. 65–68.

[3] W.-H. Shin, B.-S. Lee, Y.-K. Lee, and J.-S. Lee, "Speech/non-speech classification using multiple features for robust endpoint detection," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 3, Jun. 2000, pp. 1399–1402.

[4] T. Kristjansson, S. Deligne, and P. Olsen, "Voicing features for robust speech detection," in *Proc. Interspeech*, 2005, pp. 369–372.

[5] X. Li, H. Liu, Y. Zheng, and B. Xu, "Robust speech endpoint detection based on improved adaptive band-partitioning spectral entropy," in *Proc. Int. Conf. Life Syst. Modeling Simulation (ICLSMS)*, Sep. 2007, pp. 36–45.

[6] M. Fujimoto, K. Ishizuka, and T. Nakatani, "Study of integration of statistical model-based voice activity detection and noise suppression," in *Proc. Interspeech*, 2008, pp. 2008–2011.

[7] S. O. Sadjadi and J. H. L. Hansen, "Unsupervised speech activity detection using voicing measures and perceptual spectral flux," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 197–200, Mar. 2013.

[8] R. Hariharan, J. Hakkinen, and K. Laurila, "Robust end-of-utterance detection for real-time speech recognition applications," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, May 2001, pp. 249–252.

[9] J. Sohn, N. Soo Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Process. Lett.*, vol. 6, no. 1, pp. 1–3, Jan. 1999.

[10] H. Chung, S. J. Lee, and Y. Lee, "Endpoint detection using weighted finite state transducer," in *Proc. Interspeech*, 2013, pp. 700–703.

[11] H. Chung, S. J. Lee, and Y. K. Lee, "Weighted finite state transducer-based endpoint detection using probabilistic decision logic," *ETRI J.*, vol. 36, no. 5, pp. 714–720, Oct. 2014.

[12] X.-L. Zhang and J. Wu, "Deep belief networks based voice activity detection," *IEEE Trans. Audio, Speech, Lang., Process.*, vol. 21, no. 4, pp. 697–710, Apr. 2013.

[13] X.-L. Zhang and J. Wu, "Denoising deep neural networks based voice activity detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 853–857.

[14] I. Hwang and J. H. Chang, "Voice activity detection based on statistical model employing deep neural network," in *Proc. 10th Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, Aug. 2014, pp. 582–585.

[15] I. Hwang, J. Sim, S.-H. Kim, K.-S. Song, and J.-H. Chang, "A statistical model-based voice activity detection using multiple DNNs and noise awareness," in *Proc. Interspeech*, 2015, pp. 2277–2281.

[16] I. Hwang, H.-M. Park, and J.-H. Chang, "Ensemble of deep neural networks using acoustic environment classification for statistical model-based voice activity detection," *Comput. Speech Lang.*, vol. 38, pp. 1–12, Jul. 2016.

[17] X.-L. Zhang and D. Wang, "Boosting contextual information for deep neural network based voice activity detection," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 2, pp. 252–264, Feb. 2016.

[18] M.-Y. Hwang and X. Huang, "Shared-distribution hidden Markov models for speech recognition," *IEEE Trans. Speech Audio Process.*, vol. 1, no. 4, pp. 414–420, Jan. 1993.

[19] S. Thomas, G. Saon, M. V. Segbroeck, and S. S. Narayanan, "Improvements to the IBM speech activity detection system for the DARPA RATS program, in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 2015, pp. 4500–4504.

[20] L. Ferrer, M. Graciarena, and V. Mitra, "A phonetically aware system for speech activity detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 5710–5714.

[21] R. Masumura, T. Asami, H. Masataki, R. Ishii, and R. Higashinaka, "Online end-of-turn detection from speech based on stacked time-asynchronous sequential networks," in *Proc. Interspeech*, Aug. 2017, pp. 1661–1665.

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

[23] T. Hughes and K. Mierle, "Recurrent neural networks for voice activity detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 7378–7382.

[24] 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 *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 483–487.

[25] S. Tong, H. Gu, and K. Yu, "A comparative study of robustness of deep learning approaches for VAD," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 5695–5699.

[26] M. Shannon, G. Simko, S.-Y. Chang, and C. Parada, "Improved end-of-query detection for streaming speech recognition," in *Proc. Interspeech*, Aug. 2017, pp. 1909–1913.

[27] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4580–4584.

[28] R. Zazo, T. N. Sainath, G. Simko, and C. Parada, "Feature learning with raw-waveform CLDNNs for voice activity detection," in *Proc. Interspeech*, Sep. 2016, pp. 3668–3672.

[29] T. N. Sainath and B. Li, "Modeling time-frequency patterns with LSTM vs. convolutional architectures for LVCSR tasks," in *Proc. Interspeech*, Sep. 2016, pp. 813–817.

[30] N. Kalchbrenner, I. Danihelka, and A. Graves, "Grid long short-term memory," 2015, *arXiv:1507.01526*. [Online]. Available: http://arxiv.org/abs/1507.01526

[31] S.-Y. Chang, B. Li, T. N. Sainath, G. Simko, and C. Parada, "Endpoint detection using grid long short-term memory networks for streaming speech recognition," in *Proc. Interspeech*, Aug. 2017, pp. 3812–3816.

[32] S.-Y. Chang, B. Li, G. Simko, T. N. Sainath, A. Tripathi, A. van den Oord, and O. Vinyals, "Temporal modeling using dilated convolution and gating for voice-activity-detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 5626–5630.

[33] L. Ferrer, E. Shriberg, and A. Stolcke, "Is the speaker done yet? Faster and more accurate end-of-utterance detection using prosody," in *Proc. Int. Conf. Spoken Lang. Process. (ICSLP)*, 2002, pp. 2061–2064.

[34] L. Ferrer, E. Shriberg, and A. Stolcke, "A prosody-based approach to end-of-utterance detection that does not require speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 1, Apr. 2003, pp. 608–612.

[35] A. Raux and M. Eskenazi, "Optimizing endpointing thresholds using dialogue features in a spoken dialogue system," in *Proc. 9th SIGdial Workshop Discourse Dialogue (SIGdial)*, 2008, pp. 1–10.

[36] B. Ramabhadran, O. Siohan, and A. Sethy, "The IBM 2007 speech transcription system for European parliamentary speeches," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand. (ASRU)*, 2007, pp. 2609–2612.

[37] B. Liu, B. Hoffmeister, and A. Rastrow, "Accurate endpointing with expected pause duration," in *Proc. Interspeech*, 2015, pp. 2912–2916.

[38] R. Maas, A. Rastrow, K. Goehner, G. Tiwari, S. Joseph, and B. Hoffmeister, "Domain-specific utterance end-point detection for speech recognition," in *Proc. Interspeech*, Aug. 2017, pp. 1943–1947.

[39] R. Maas, A. Rastrow, C. Ma, G. Lan, K. Goehner, G. Tiwari, S. Joseph, and B. Hoffmeister, "Combining acoustic embeddings and decoding features for end-of-utterance detection in real-time far-field speech recognition systems," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 5544–5548.

[40] S.-Y. Chang, R. Prabhavalkar, Y. He, T. N. Sainath, and G. Simko, "Joint endpointing and decoding with end-to-end models," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 5626–5630.

[41] M. K. Mustafa, T. Allen, and K. Appiah, "A comparative review of dynamic neural networks and hidden Markov model methods for mobile on-device speech recognition," *Neural Comput. Appl.*, vol. 31, no. S2, pp. 891–899, Feb. 2019.

[42] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, Q. Liang, D. Bhatia, Y. Shangguan, B. Li, G. Pundak, K. Chai Sim, T. Bagby, S.-y. Chang, K. Rao, and A. Gruenstein, "Streaming End-to-end speech recognition for mobile devices," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 6381–6385.

[43] K. Kim, K. Lee, D. Gowda, J. Park, S. Kim, S. Jin, Y.-Y. Lee, J. Yeo, D. Kim, S. Jung, J. Lee, M. Han, and C. Kim, "Attention based on-device streaming speech recognition with large speech corpus," in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop (ASRU)*, Dec. 2019, pp. 956–963.

[44] L. Shi, "A general purpose semantic parser using framenet and wordnet," M.S. thesis, Univ. North Texas, Denton, TX, USA, 2004.

[45] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand. (ASRU)*, Dec. 2015, pp. 504–511.

[46] L. Lugosch, M. Ravanelli, P. Ignoto, V. S. Tomar, and Y. Bengio, "Speech model pre-training for end-to-end spoken language understanding," in *Proc. Interspeech*, Sep. 2019, pp. 814–818.

[47] S. Katz, "Estimation of probabilities from sparse data for the language model component of a speech recognizer," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 3, pp. 400–401, Mar. 1987.

[48] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 708–812.

[49] B. Wu, M. Yu, L. Chen, M. Jin, D. Su, and D. Yu, "Improving speech enhancement with phonetic embedding features," in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop (ASRU)*, Dec. 2019, pp. 645–651.

[50] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 1, pp. 30–42, Jan. 2012.

[51] D. Klakow and J. Peters, "Testing the correlation of word error rate and perplexity," *Speech Commun.*, vol. 38, nos. 1–2, pp. 19–28, 2002.

[52] Y. Lee, B. Kim, and Y. Um, "Speech information technology & industry promotion center in Korea: Activities and directions," in *Proc. Int. Conf. Lang. Resour. Eval. (ICLRE)*, 2002, pp. 1851–1854.

[53] M. Abadi *et al.* (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. [Online]. Available: http://tensorflow.org/

[54] J. Garofalo, D. Graff, D. Paul, and D. Pallett, *CSR-I, (WSJ0) Complete*, vol. LDC93S6A. Philadelphia, PA, USA: Linguistic Data Consortium, 1993.

[55] X. Anguera, C. Wooters, and J. Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 15, no. 7, pp. 2011–2022, Sep. 2007.

[56] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand. (ASRU)*, 2011, pp. 1–4.

[57] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1711–1800, Aug. 2002.

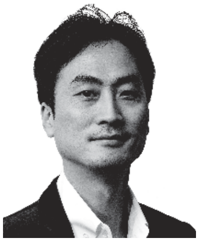
[58] B. Kingsbury, "Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 3761–3764.

[59] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[60] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoustic Soc. Amer.*, vol. 65, no. 4, p. 950, Apr. 1979.

[61] K. Kinoshita, M. Delcroix, S. Gannot, E. A. P. Habets, R. Haeb-Umbach, W. Kellermann, V. Leutnant, R. Maas, T. Nakatani, B. Raj, A. Sehr, and T. Yoshioka, "A summary of the REVERB challenge: State-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP J. Adv. Signal Process.*, vol. 2016, no. 1, pp. 1–19, Jan. 2016.

INYOUNG HWANG received the B.S. degree in electronics engineering from Suwon University, Hwaseong, South Korea, in 2013, and the Ph.D. degree in electronics and computer engineering from Hanyang University, Seoul, South Korea, in 2019. He is currently a Research Engineer of the AI Service Division, AI Technology Unit, SK Telecom, Seoul. His research interests include automatic speech recognition, voice activity detection, speech endpoint detection, and machine learning and deep learning applied to signal processing.



JOON-HYUK CHANG (Senior Member, IEEE) received the B.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 1998, and the M.S. and Ph.D. degrees in electrical engineering from Seoul National University, South Korea, in 2000 and 2004, respectively. From 2000 to 2005, he was with Netdus Corporation, Seoul, South Korea, as the CTO. From 2004 to 2005, he held a post-doctoral position at the University of California at Santa Barbara, Santa Barbara, where he was involved in adaptive signal processing and audio coding. In 2005, he joined the Korea Institute of Science and Technology, Seoul, as a Research Scientist, where he was involved in speech recognition. From 2005 to 2011, he was an Assistant Professor with the School of Electronics Engineering, Inha University, Incheon, South Korea. He is currently a Full Professor with the School of Electronics Engineering, Hanyang University, Seoul. His research interests include speech recognition, deep/machine learning, artificial intelligence (AI), speech processing, acoustic signal processing, and biomedical signal processing. He was a recipient of the IEEE/IEEK IT Young Engineer in 2011. He is currently serving on the Editorial Board of the *Digital Signal Processing* (Elsevier).