Deep-Learning Based Automatic Spontaneous Speech Assessment in a Data-Driven Approach for the 2017 SLaTE CALL Shared Challenge

Yoo Rhee Oh, Hyung-Bae Jeon, Hwa Jeon Song, Byung Ok Kang, Yun-Kyung Lee, Jeon-Gue Park, and Yun-Keun Lee

Speech Intelligence Research Group,
Electronics and Telecommunications Research Institute, South Korea

yroh@etri.re.kr, hbjeon@etri.re.kr, songhj@etri.re.kr, bokang@etri.re.kr, yunklee@etri.re.kr, jgp@etri.re.kr, yklee@etri.re.kr

Abstract

This paper presents a deep-learning based assessment method of a spoken computer-assisted language learning (CALL) for a non-native child speaker, which is performed in a data-driven approach rather than in a rule-based approach. Especially, we focus on the spoken CALL assessment of the 2017 SLaTE challenge. To this end, the proposed method consists of four main steps: speech recognition, meaning feature extraction, grammar feature extraction, and deep-learning based assessment. At first, speech recognition is performed on an input speech using three automatic speech recognition (ASR) systems. Second, twenty-seven meaning features are extracted from the recognized texts via the three ASRs using language models (LMs), sentence-embedding models, and word-embedding models. Third, twenty-two grammar features are extracted from the recognized text via one ASR system using linear-order LMs and hierarchical-order LMs. Fourth, the extracted forty-nine features are fed into a full-connected deep neural network (DNN) based model for the classification of acceptance or rejection. Finally, an assessment is performed by comparing the probability of a output unit of the DNN-based classifier with a predefined threshold. For the experiments of a spoken CALL assessment, we use English spoken utterances by Swiss German teenagers. It is shown from the experiments that the D score is 4.37 for the spoken CALL assessment system employing the proposed method.

Index Terms: Spoken CALL assessment, DNN based classifier, sentence-embedding, word-embedding, sequence-to-sequence, dependency parse tree

1. Introduction

There are considerable researches on a computer-assisted language learning (CALL) based on speech recognition. Moreover, many researches are related to the pronunciation assessment of an imitated speech such as SRI's EduSpeak [1]. On the other hand, some researches are related to various kinds of assessments (grammar, semantic, vocabulary, etc) of a speech with the increased freedom of speaking, such as CALL-SLT [2] and GenieTutor [3]. Among them, this paper focuses on a spoken CALL assessment of the CALL-SLT system, as a participant of the 2017 SLaTE CALL shared challenge [4]. To this end, we propose a deep-learning based spoken CALL assessment method, which is performed in a data-driven approach while the baseline method supported by the organizer of the challenge is performed in a rule-based approach.

2. The 2017 SLaTE CALL shared challenge

The challenge aims to assess the utterance obtained from a CALL system in terms of the meaning and the grammar for a given prompt. And, the corpus consists of 5,222 utterances and 996 utterances for a training data and a test data, respectively. In addition, the annotation of each utterance contains a prompt, a transcription, a meaning evaluation result, and an overall evaluation result.

A baseline assessment method first performs speech recognition on an utterance and then determines the utterance to be acceptable if the recognized text exactly matches one of the reference texts corresponding to the given prompt [5]. Thus, the organizer provides the recognized text ($Text_{Nuance}$) from a Nuance ASR (ASR_{Nuance}) and the recognized text $Text_{Kaldi}$ from a Kaldi ASR (ASR_{Kaldi}) for each speech data. Also, it provides the reference texts corresponding to each prompt (Texts_{Ref}). When evaluating the test data using a metric of D, the baseline method achieves 2.35, 1.69, and 4.51 for three CALL assessment systems: (a) a CALL system ($CALL^{Nuance}_{baseline}$) using $Text_{Nuance}$, (b) a CALL system ($CALL^{Kaldi}_{baseline}$) using $Text_{Kaldi}$, and (c) a CALL system ($CALL^{transcription}_{baseline}$) using the transcription texts. In addition, the word error rates (WERs) of ASR_{Nuance} and ASR_{Kaldi} are 33.1% and 25.1%.

The detailed description of the challenge is explained in [4].

3. Proposed deep-learning based spoken CALL assessment in a data-driven approach

Fig. 1 shows the proposed deep-learning based spoken CALL assessment method. In other words, the speech processing component first performs speech recognition for an input speech using three ASRs such as ASR_{Nuance}, ASR_{Kaldi}, and an ASR ($ASR_{SLaTE2017}$) developed for the challenge. And then, three recognized texts, $Text_{Nuance}$, $Text_{Kaldi}$, and $Text_{SLaTE2017}$, are obtained from ASR_{Nuance}, ASR_{Kaldi}, and $ASR_{SLaTE2017}$, respectively.

Next, the text processing component performs a meaning feature extraction, a grammar feature extraction, and a deep-learning based assessment using $Text_{Nuance}$, $Text_{Kaldi}$, and $Text_{SLaTE2017}$, a given prompt, and Texts_{Ref}. That is, the meaning feature extraction step generates nine meaning features from each of $Text_{Nuance}$, $Text_{Kaldi}$, and $Text_{SLaTE2017}$ and then concatenates the three sets of the nine meaning features. Moreover, the grammar feature extraction step generates twenty-two grammar features from $Text_{SLaTE2017}$; we only

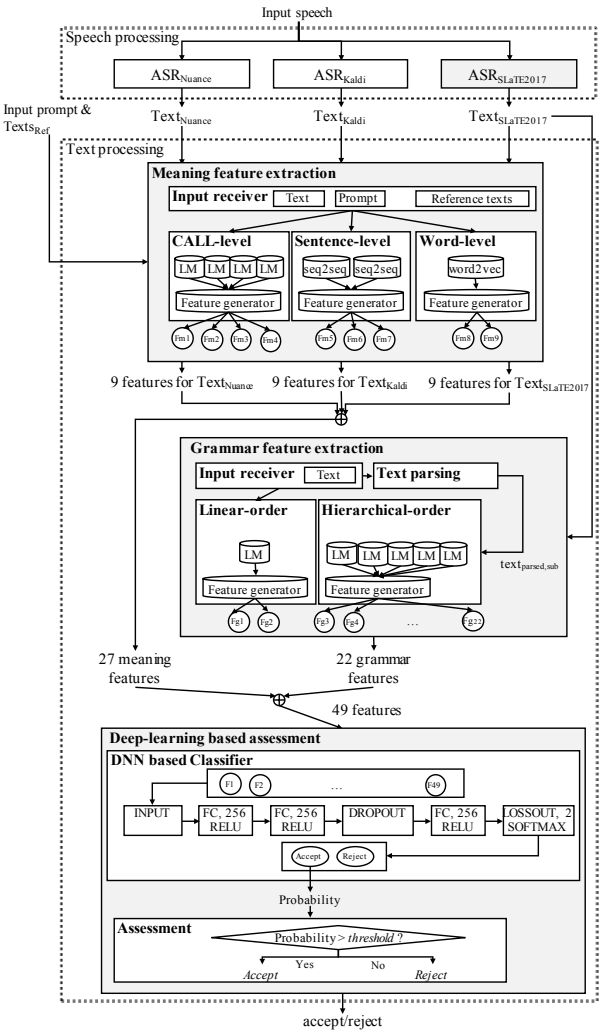


Figure 1: *Main procedure of the proposed deep-learning based spontaneous spoken CALL assessment method in a data-driven approach for the challenge.*

use $Text_{SLaTE2017}$ having the best ASR performance since ASR errors could degrade the performance of a grammar error detection. Finally, the deep-learning based assessment step determines whether to accept or reject the input speech using the forty-nine features extracted in the previous steps.

3.1. An ASR system for the challenge

An original ASR system ($ASR^{original}_{SLaTE2017}$) is a common-domain American English ASR system for Korean speakers. In addition, the characteristics of $ASR^{original}_{SLaTE2017}$ are a sampling rate of 16 kHz, the target language of American English, the speaker's mother tongue of Korean, and the target LM domain of a common-domain. Moreover, the acoustic models (AMs) of $ASR^{original}_{SLaTE2017}$ [6, 7] are configured as deep neural network hidden Markov models (DNN-HMMs) and the LM of $ASR^{original}_{SLaTE2017}$ is configured as a 3-gram LM that is trained with common-domain texts.

In order to provide a better text to the text processing component of the proposed method, we adjust $ASR^{original}_{SLaTE2017}$ to $ASR_{SLaTE2017}$ by reflecting the characteristics of the speech data of the challenge, such as a sampling rate of 8 kHz, the target language of British English, and the speaker's mother tongue of German. That is, the AMs are adapted using the training data of the challenge and the LM ($M_{ASR,3gram}$) is interpolated with a challenge-domain 3-gram LM that is trained with Texts$_{Ref}$. In addition, a pronunciation model is adapted by adding English pronunciation variants that are frequently mispronounced by the German people.

When evaluating the test data, the WER is 14.9% for $ASR_{SLaTE2017}$ and the D score is 2.86 for the spoken CALL system ($CALL^{SLaTE2017}_{baseline}$) employing the baseline assessment method using the recognized text from $ASR_{SLaTE2017}$. Moreover, Table 1 summarizes the performance comparisons of the four CALL systems such as $CALL^{Nuance}_{baseline}$, $CALL^{Kaldi}_{baseline}$, $CALL^{transcription}_{baseline}$, and $CALL^{SLaTE2017}_{baseline}$.

Table 1: *Performance comparison based on the D and WER (%) for $CALL^{Nuance}_{baseline}$, $CALL^{Kaldi}_{baseline}$, $CALL^{transcription}_{baseline}$, and $CALL^{SLaTE2017}_{baseline}$.*

System	D	WER (%) of an ASR system
$CALL^{Nuance}_{baseline}$	2.35	33.1
$CALL^{Kaldi}_{baseline}$	1.69	25.1
$CALL^{transcription}_{baseline}$	4.51	-
$CALL^{SLaTE2017}_{baseline}$	2.86	14.9

Table 2: *Summary of the nine meaning features of the proposed method.*

Feature	Type	Model	Metric
Fm1	CALL-level	$M_{domain,3gram}$	ppl1
Fm2		$M_{domain,5gram}$	
Fm3		$M_{common,3gram}$	
Fm4		$M_{domain,lstm}$	
Fm5	Sentence-level	$M_{seq2seq,fwd}$	cos_{1_best}
Fm6			cos_{100_best}
Fm7		$M_{seq2seq,bwd}$	cos_{1_best}
Fm8	Word-level	$M_{word2vec}$	DTW
Fm9			DTW/keyword weighting

3.2. Deep-learning based meaning features

As summarized in Table 2, nine meaning features are extracted from an input text and the features are categorized into three levels: CALL-level, sentence-level, and word-level, depending on the scope of the meaning to be analyzed. The CALL-level features are extracted using four LMs to determine whether the meaning of the input text corresponds to the learning scope of a CALL system. The sentence-level features are extracted using two sentence-embedding models to determine whether the meaning of the input text includes the meaning of the prompt. The word-level features are extracted using a one word-embedding model to determine whether the meanings of words contained in the input text are related to the meanings of words contained in Texts$_{Ref}$.

3.2.1. CALL-level meaning features based on LMs

The assumption is that, if the meaning of an input text is within the language learning scope of a CALL system, then the probability of an input text would be larger when using a CALL-domain LM than when using a common-domain LM. Three CALL-domain LMs using challenge-related texts and a common-domain LM using general-domain texts are generated as follows:

- $M_{domain,3gram}$: CALL-domain, 3-gram LM
- $M_{domain,5gram}$: CALL-domain, 5-gram LM
- $M_{domain,lstm}$: CALL-domain, a two-layer long short-term memory (LSTM) recurrent neural network (RNN) with 200 hidden units
- $M_{common,3gram}$: common-domain, 3-gram LM

When an input text is entered, the average log-probability per word excluding word boundaries ($ppl1$, [8]) is calculated using each CALL-level LM. In particular, predefined penalties under certain conditions are applied in the calculation of the log-probability of a word. As a result, we obtain four CALL-level meaning features ($Fm1$, $Fm2$, $Fm3$, and $Fm4$) using $M_{domain,3gram}$, $M_{domain,5gram}$, $M_{common,3gram}$, and $M_{domain,lstm}$.

3.2.2. Sentence-level meaning features based on sentence-embedding models

For sentence-level meaning feature extraction, we adopt a sentence-embedding approach that is increasingly utilized in various research such as natural language processing, machine translation, and so on [9]. We generate two sequence-to-sequence (seq2seq) models using challenge-related texts as follows:

- $M_{seq2seq,fwd}$: a forward LSTM-RNN based encoder-decoder model with 200 hidden units
- $M_{seq2seq,bwd}$: a backward LSTM-RNN based encoder-decoder model with 200 hidden units

When an input text and its prompt are entered, the reference sentences ($\text{Texts}_{Ref,prompt}$) corresponding to the prompt are first obtained from Texts_{Ref}. Then, a cosine similarity is calculated between the two vectors of the hidden state values of the encoder of a sentence-level model for the input text and each of $\text{Texts}_{Ref,prompt}$. Next, we obtain the maximum cosine similarities and the average of up to 100 values, hereafter referred to as cos_{1_best} and cos_{100_best}, respectively. As a result, we obtain two sentence-level meaning features, $Fm5$ and $Fm6$, by selecting cos_{1_best} and cos_{100_best} using $M_{seq2seq,fwd}$. In addition, we obtain one feature, $Fm7$, by selecting cos_{1_best} using $M_{seq2seq,bwd}$.

3.2.3. Word-level meaning features based on word-embedding models

Similar to Section 3.2.2, we adopt a word-embedding approach for a word-level meaning feature extraction. We generate one word2vec model [10] using the challenge-related texts as follows:

- $M_{word2vec}$: a word2vec model

When an input text and its prompt are entered, the reference sentences ($\text{Texts}_{Ref,prompt}$) corresponding to the prompt are first obtained from Texts_{Ref}. Then, we measure the similarity distances between the input text and each $\text{Texts}_{Ref,prompt}$ using the word-level model and obtain the minimum of the similarity distances. The similarity distance between two texts is calculated by performing a dynamic time warping (DTW), wherein the distance between two words is the output of the word2vec model. As a result, we obtain one word-level meaning feature, $Fm8$, by selecting the minimum similarity distance using $M_{word2vec}$.

In order to capture the use of the keywords of the prompt, we also obtain one additional feature, $Fm9$, by selecting keywords that are most likely present in $\text{Texts}_{Ref,prompt}$ and calculating the penalized similarity distance with the modified distance value of each keyword.

Table 3: *Summary of the twenty-two grammar features of the proposed method.*

Feature	Type	Model	Text	Metric
Fg1	Hierarchical-order	$M_{formal.6gram}$	Parsed	PPL1/max
Fg2				PPL1/avg.
Fg3				PPL/max
Fg4				PPL/avg.
Fg5		$M_{informal.3gram}$		PPL1/min
Fg6				PPL1/avg.
Fg7				PPL/min
Fg8				PPL/avg.
Fg9	Linear-order	$M_{ASR,3gram}$	Raw	PPL1/avg.
Fg10				PPL/avg.
Fg11	Hierarchical-order	$M_{ASR,3gram}$	Parsed	PPL1/max
Fg12				PPL1/avg.
Fg13				PPL/max
Fg14				PPL/avg.
Fg15		$M_{formal.3gram}$		PPL1/max
Fg16				PPL1/avg.
Fg17				PPL/max
Fg18				PPL/avg.
Fg19		$M_{formal.3gram.ext}$		PPL1/max
Fg20				PPL1/avg.
Fg21				PPL/max
Fg22				PPL/avg.

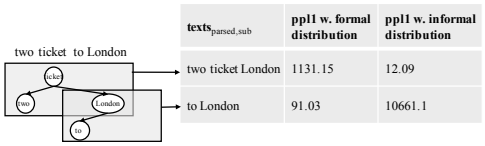
3.3. Parsed-text based grammar features

We first detect awkward word sequences by examining the linear word sequence of an input text. However, this is not sufficient to detect grammartical errors because the order of sentence constituents is not considered. In order to consider the order of sentence constituents, the input text is first parsed into a dependency parse tree and then each 1-depth subtree is converted into a word sequence, which is referred to as '$\text{text}_{parsed,sub}$'. Next, we identify incorrect grammar by examining the word sequence of $\text{texts}_{parsed,sub}$.

As summarized in Table 3, we extract two grammar features from an input text for a grammar check based on the linear word sequence. We also extract twenty grammar features from the $\text{texts}_{parsed,sub}$ of an input text to conduct a grammar check based on the hierarchical word sequence.

3.3.1. Linear-order grammar features from an input text

For a grammar check that detects awkward word sequences in an input text, the word sequence of the input text is compared with the general distribution of word sequences obtained from



$\text{texts}_{\text{parsed,sub}}$	ppl1 w. formal distribution	ppl1 w. informal distribution
two ticket London	1131.15	12.09
to London	91.03	10661.1

Figure 2: *Example of the $\text{texts}_{parsed,sub}$ of the text, 'two ticket to London', and the comparison of $ppl1$ using the formal distribution of $M_{formal.3gram}$ and the informal distribution of $M_{informal.3gram}$.*

a large amount of texts. The following model is chosen as the general distribution of word sequences.

- $M_{ASR,3gram}$: the 3-gram LM for $ASR_{SLaTE2017}$

When an input text is entered, $ppl1$ is calculated using $M_{ASR,3gram}$. Moreover, the average log-probability per word including word boundaries (*ppl*, [8]) is calculated using $M_{ASR,3gram}$. As a result, two linear-order grammar features of $Fg9$ and $Fg10$ are obtained by calculating $ppl1$ and *ppl* of the input text using $M_{ASR,3gram}$.

3.3.2. Hierarchical-order grammar features from the parsed texts of an input text

For a grammar check that detects awkward word sequences based on the sentence constituents of an input text, each parsed text, $\text{text}_{parsed,sub}$, of the input text is compared with a formal distribution of parsed texts and an informal distribution of parsed texts, respectively. The formal distribution is generated using the $\text{texts}_{parsed,sub}$ of the formal texts from three kinds of resources: the texts of English text books, Texts_{Ref}, and the ones of GenieTutor CALL system [3]. The informal distribution is generated using the $\text{texts}_{parsed,sub}$ of the informal texts. The informal texts are obtained by performing three steps: generating texts with grammar errors from the formal texts, obtaining $_{parsed,sub}$ of texts with grammar errors, and excluding $\text{texts}_{parsed,sub}$ that are overlapped with $\text{texts}_{parsed,sub}$ of the formal texts. To this end, three correct distribution LMs, one incorrect distribution LM, and $M_{ASR,3gram}$ are used as follows:

- $M_{formal.3gram}$: formal, 3-gram, parsed text
- $M_{formal.6gram}$: formal, 6-gram, parsed text
- $M_{formal.3gram.ext}$: formal, 3-gram, parsed text, expanded reference responses with the synonym words
- $M_{informal.3gram}$: informal, 3-gram, parsed text
- $M_{ASR,3gram}$: a 3-gram LM for $ASR_{SLaTE2017}$, raw text

In fact, $M_{ASR,3gram}$ is chosen since the text obtained from a phrase-level grammar chunk can be a part of the linear word sequence of an input text. Fig. 2 shows an example of the $\text{texts}_{parsed,sub}$ of the text, 'two ticket to London' and the comparison of the $ppl1$ of the $\text{texts}_{parsed,sub}$ using the formal and informal distribution of $M_{formal.3gram}$ and $M_{informal.3gram}$, respectively.

When an input text is entered, the text is decomposed into $\text{texts}_{parsed,sub}$ using a Stanford parser [11]. Next, $ppl1$ is computed using an LM for each $\text{text}_{parsed,sub}$ and then the maximum, minimum, and average values of the set of $ppl1$ are obtained. In addition, we similarly obtain the maximum, minimum, and average *ppl* values for the input text. Thus, we obtain 16 hierarchical-order grammar features by selecting the maximum and average values of $ppl1$ and *ppl* for each of the three formal distribution models and $M_{ASR,3gram}$. We also obtain four features by selecting the minimum and average values of $ppl1$ and *ppl* for the informal distribution model.

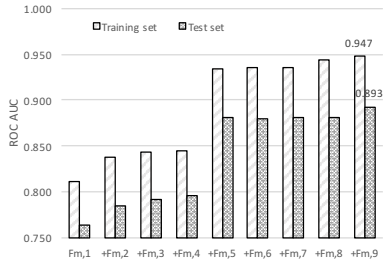


Figure 3: *Performance comparisons based on the ROC AUC of the proposed method by increasing each feature by one for the 9 meaning features*

3.4. Deep-learning based classifier for the spoken CALL assessment

In order to assess an input utterance using various kinds of meaning and grammar features, we adopt a deep-learning based approach that is a state-of-art method in many applications. To this end, we empirically configure a six-layer DNN of one input layer, four hidden layers, and one output layer. The input layer consists of 49 linear units for the 27 meaning features and 22 grammar features; the output layer is a softmax layer with two units that correspond to the final targets, accept or reject. Each of the first, second, and fourth hidden layers is a fully-connected (FC) layer that contains 256 units with rectified linear unit (ReLU) activation, while the third hidden layer is a dropout layer that prevents an overfitting.

When the 49 meaning and grammar features are entered, we calculate the output unit probabilities of the 6-layer DNN based classifier. And then, the input data is decided as $accept$ when the probability of the accept-labeled output unit is greater than a predefined threshold. The predefined threshold is selected by maximizing the D value throughout the training data in the constraints of the minimum values of Fr and Cr as 0.04 [1]. In particular, the constraints are empirically selected in order to compensate for the fact that the D value becomes too larger when the Fr value is getting close to zero.

4. Experiments on the CALL assessment

We first evaluated the proposed method using a transcription text in order to eliminate the effect of ASR errors in Section 4.1. Next, we compared the performances of the proposed method using recognized texts from three ASR systems in Section 4.2. Especially, we additionally used a common binary classification performance metric, a receiver operating characteristic area under curve (ROC AUC).

4.1. Performance on the proposed method using a transcribed text

First, we validated the nine meaning features of the proposed method using transcription texts by entering the features one by one into the proposed method. In other words, we extracted $Fm1$ from each transcription, trained a DNN-based classifier which input layer has one unit corresponding to $Fm1$, and evaluated the performance based on ROC AUC. Next, we extracted $Fm1$ and $Fm2$ from each transcription, trained a DNN-based classifier which input layer has two units corresponding to $Fm1$

[1] The descriptions of the D, Fr, and Cr are explained in [4].

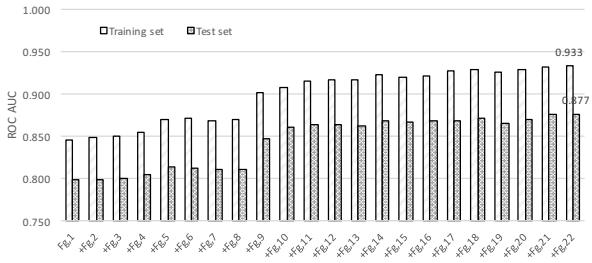


Figure 4: *Performance comparisons based on the ROC AUC of the proposed method by increasing each feature by one for the 22 grammar features*

and $Fm2$, and evaluated the performance. This process was performed until the nine meaning features were used. It was shown from the Fig. 3 that the performance of a CALL assessment system was generally improved each time the feature was increased by one for the meaning feature.

Second, we validated the 22 grammar features of the proposed method using transcription texts in the same way to the verification of the meaning features. It was shown from the Fig. 4 that the performance was generally improved each time the feature was increased by one for the meaning feature.

Third, we evaluated the proposed method using both the meaning and grammar features extracted from transcription texts. When evaluating the test set, the ROC AUC score was 0.921 for a CALL assessment system ($CALL^{transcription}_{proposed}$) employing the proposed method using both meaning and grammar features, whereas the ROC AUC scores were 0.893 and 0.877 for a CALL system employing the proposed method using meaning features and a CALL system employing the proposed method using grammar features, respectively.

From the experiments on the proposed method using transcription texts, we concluded that each meaning features were positively correlated for the proposed method. Similarly, each grammar features were positively correlated for the proposed method. In addition, the performance of the proposed method was considerably improved when combining the meaning features and the grammar features.

4.2. Performance on the proposed method using the recognized texts from three ASR systems

Fig. 5 shows the ROC AUCs of the four CALL assessment systems: (a) a CALL assessment system ($CALL^{Nuance}_{proposed}$) employing the proposed method using $Text_{Nuance}$, (b) a CALL assessment system ($CALL^{Kaldi}_{proposed}$) employing the proposed method using $Text_{Kaldi}$, (c) a CALL assessment system ($CALL^{SLaTE2017}_{proposed}$) employing the proposed method using $Text_{SLaTE2017}$, and (d) a CALL assessment system ($CALL^{Nuance,Kaldi,SLaTE2017}_{proposed}$) employing the proposed method using $Text_{Nuance}$, $Text_{Kaldi}$, and $Text_{SLaTE2017}$. First, the ROC AUC of $CALL^{SLaTE2017}_{proposed}$ was 0.81 whereas the ROC AUCs of $CALL^{Nuance}_{proposed}$ and $CALL^{Kaldi}_{proposed}$ were 0.75 and 0.77, respectively. Therefore, it was notible from Table 1 and Figure 5 that the performance of the proposed method could be improved according to the performance improvement of an ASR system. Second, the ROC AUC of $CALL^{Nuance,Kaldi,SLaTE2017}_{proposed}$ was measured as 0.83 whereas the ROC AUC of $CALL^{SLaTE2017}_{proposed}$ was 0.81. From the performance comparison, it was noticed that the performance of the

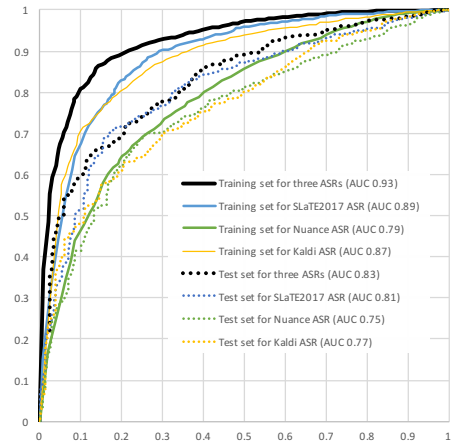


Figure 5: *Performance comparisons based on the ROC AUCs of $CALL^{Nuance}_{proposed}$, $CALL^{Kaldi}_{proposed}$, $CALL^{SLaTE2017}_{proposed}$, and $CALL^{Nuance,Kaldi,SLaTE2017}_{proposed}$ when evaluating the training set and test set, respectively.*

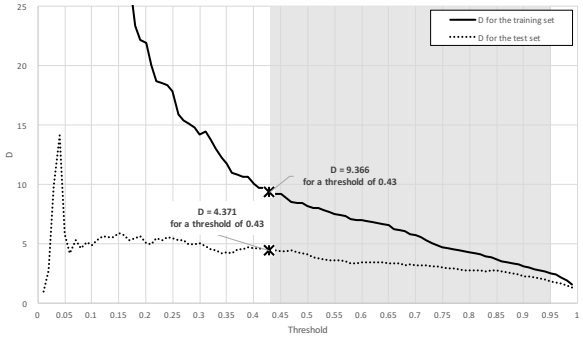


Figure 6: *An example of the optimal threshold selection of $CALL^{Nuance,Kaldi,SLaTE2017}_{proposed}$, where the straight and dotted lines represent the D scores according to thresholds when evaluating the training data and the test data, respectively, and the shaded area presents the corresponding range to the constraints.*

proposed method could be improved by combining the recognized texts from various kinds of ASR systems.

Moreover, we evaluated the assessment performance of the proposed method using the metric D. To this end, we first selected an optimal threshold to maximize the D score using the training set. Especially, we used the constraints of the minimum values of Fr and Cr of 0.04 as described in Section 3.4. After that, the D score of the test set was calculated using the selected optimal threshold. Fig. 6 shows an example of the optimal threshold selection for $CALL^{Nuance,Kaldi,SLaTE2017}_{proposed}$ where the straight and dotted lines presented the D scores for the training data and the test data when a threshold was ranged from 0 to 1. In addition, the shaded area represented the range corresponding to the constraints of the minimum values of Fr and Cr of 0.04. It could be seen from the figure that the optimal threshold was selected as 0.43 and the corresponding D score was obtained as 4.371 when evaluating the test data.

Finally, we obtained the D scores for the five CALL assessment system, $CALL^{transcription}_{proposed}$,

Table 4: *Performance comparison based on the metric D of the baseline method and the proposed method using a transcription, $Text_{Nuance}$, $Text_{Kaldi}$, and $Text_{SLaTE2017}$*

	Baseline method	Proposed-method
Transcription	4.51	5.38
$Text_{Nuance}$	2.35	2.56
$Text_{Kaldi}$	1.69	2.28
$Text_{SLaTE2017}$	2.85	3.98
$Text_{Nuance,Kaldi,SLaTE2017}$ (submitted)	-	4.37
$Text_{Nuance,Kaldi,SLaTE2017}$ (fixed)	-	4.49

$CALL^{Nuance}_{proposed}$, $CALL^{Kaldi}_{proposed}$, $CALL^{SLaTE2017}_{proposed}$, and $CALL^{Nuance,Kaldi,SLaTE2017}_{proposed}$, as shown in Table 4. In addition, the last row of the table presented the D score of a fixed version of the proposed method. It was noted from the table that the performances of the proposed method were improved for all CALL assessment systems when compared to the baseline method. Moreover, the performance of the proposed method was considerably improved when combining the recognized texts from three ASR systems.

5. Conclusion and discussion

This paper proposed the deep-learning based spoken CALL assessment method for the 2017 SLaTE CALL shared challenge. Especially, we focused on the generation of the spoken CALL assessment system with minimal manual efforts, by using a data-driven approach rather than using a rule-based approach. Moreover, we tried to adopt deep-learning method, which was known as the state-of-the-art machine learning method.

The proposed method consisted of the speech processing component and the text processing component. In other words, when an input speech was entered, speech recognition was performed using each of ASR_{Nuance}, ASR_{Kaldi}, and $ASR_{SLaTE2017}$ in the speech processing component. After that, the three texts recognized from three ASRs were passed into the text processing component. In the text processing component, the twenty-seven meaning features were generated by extracting nine feature from each of the recognized texts and by concatenating them. Moreover, the nine meaning features consisted of four CALL-level features using LMs, three sentence-level features using sentence-embedding models, and two word-level features using a word-embedding model. Next, the twenty-two grammar features were extracted from $Text_{SLaTE2017}$; among them, two features were for a linear-order text and twenty features were for a hierarchical-order text. Finally, the forty-nine features were fed into the fully-connected 6-layer neural network for the classification of acceptance. And then, an assessment was performed by comparing the output probability of the DNN based classifier with a predefined threshold. We first validated the designed meaning and grammar features of the proposed method and then compared the performances of the several CALL assessment systems, $CALL^{transcription}_{proposed}$, $CALL^{Nuance}_{proposed}$, $CALL^{Kaldi}_{proposed}$, $CALL^{SLaTE2017}_{proposed}$, and $CALL^{Nuance,Kaldi,SLaTE2017}_{proposed}$. It was shown from the spoken CALL assessment experiments that the D score was 4.37 for the proposed method whereas the D score was 2.35 or 1.69 for the baseline method. In addition, after modifying some mistakes, we obtained the D score as 4.49 using the proposed method.

In short, the proposed method considerably improved the performance of the spoken CALL assessment system by adopting several deep-learning based methods when compared to the baseline method. Moreover, the proposed method reduced the manual efforts by using a data-driven approach. However, there are several future works of the proposed method such as the investigation of the significant features among the proposed 27 meaning features and 22 grammar features, the development of the hybrid method combining the proposed method and the baseline method, etc.

6. Acknowledgements

This work was partially supported by the ICT R&D program of MSIP/IITP. [R0126-15-1117, Core technology development of the spontaneous speech dialogue processing for the language learning] and by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government. [17ZS1800, Development of self-improving and human-augmenting cognitive computing technology]

7. References

[1] H. Franco, H. Bratt, R. Rossier, V. R. Gadde, E. Shriberg, V. Abrash, and K. Precoda, "EduSpeak a speech recognition and pronunciation scoring toolkit for computer-aided language learning applications," *Language Testing*, vol. 27, no. 3, pp. 401–418, Jul. 2010.

[2] E. Rayner, N. Tsourakis, C. Baur, P. Bouillon, and J. Gerlach, "Call-slt: A spoken call system based on grammar and speech recognition," *Linguistic Issues in Language Technology*, vol. 10, no. 2, 2014.

[3] O.-W. Kwon, K.-Y. Lee, Y.-H. Roh, J.-X. Huang, S.-K. Choi, Y. K. Kim, H.-B. Jeon, Y. R. Oh, Y.-K. Lee, B. O. Kang, E. Chung, J. G. Park, and Y. Lee, "Genietutor: A computer-assisted second-language learning system based on spoken language understanding." in *Natural Language Dialog Systems and Intelligent Assistants*. Springer, 2015, pp. 257–262.

[4] C. Baur, J. Gerlach, M. Rayner, M. Russell, and H. Strik, "A shared task for spoken call?" in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. Paris, France: European Language Resources Association (ELRA), may 2016.

[5] M. Rayner, C. Baur, C. Chua, and N. Tsourakis, "Supervised learning of response grammars in a spoken call system," in *in Proc. SLaTE*, 2015, pp. 83–88.

[6] B. O. Kang and O. Kwon, "Combining multiple acoustic models in GMM spaces for robust speech recognition," *IEICE Transactions*, vol. 99-D, no. 3, pp. 724–730, 2016.

[7] S. J. Lee, B. O. Kang, H. Chung, and J. G. Park, "A useful feature-engineering approach for a lvcsr system based on cd-dnn-hmm algorithm." in *EUSIPCO*. IEEE, 2015, pp. 1421–1425.

[8] A. Stolcke, "Srilm - an extensible language modeling toolkit," 2002, pp. 901–904.

[9] H. Palangi, L. Deng, Y. Shen, J. Gao, X. He, J. Chen, X. Song, and R. K. Ward, "Deep sentence embedding using the long short term memory network: Analysis and application to information retrieval," *CoRR*, vol. abs/1502.06922, 2015.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013.

[11] M.-C. de Marneffe and C. D. Manning, "The stanford typed dependencies representation," in *Coling 2008: Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation*, ser. CrossParser '08. Stroudsburg, PA, USA: Association for Computational Linguistics, 2008, pp. 1–8.