

Integration of WFST Language Model in Pre-trained Korean E2E ASR Model

Junseok Oh¹, Eunsoo Cho², and Ji-Hwan Kim^{1*}

¹ Department of Computer Science and Engineering, Sogang University
35 Baekbeom-ro, Mapo-gu, Seoul, 04107, Republic of Korea
[e-mail: {ohjs,kimjihwan}@sogang.ac.kr]

² Speech Recognition Laboratory, SELVAS AI
20F, 19, Gasan digital 1-ro, Geumcheon-gu, Seoul, 08594, Republic of Korea
[e-mail: victoria.e.cho@selvas.com]

*Corresponding author: Ji-Hwan Kim

Abstract

In this paper, we present a method that integrates a Grammar Transducer as an external language model to enhance the accuracy of the pre-trained Korean End-to-end (E2E) Automatic Speech Recognition (ASR) model. The E2E ASR model utilizes the Connectionist Temporal Classification (CTC) loss function to derive hypothesis sentences from input audio. However, this method reveals a limitation inherent in the CTC approach, as it fails to capture language information from transcript data directly. To overcome this limitation, we propose a fusion approach that combines a clause-level n-gram language model, transformed into a Weighted Finite-State Transducer (WFST), with the E2E ASR model. This approach enhances the model's accuracy and allows for domain adaptation using just additional text data, avoiding the need for further intensive training of the extensive pre-trained ASR model. This is particularly advantageous for Korean, characterized as a low-resource language, which confronts a significant challenge due to limited resources of speech data and available ASR models. Initially, we validate the efficacy of training the n-gram model at the clause-level by contrasting its inference accuracy with that of the E2E ASR model when merged with language models trained on smaller lexical units. We then demonstrate that our approach achieves enhanced domain adaptation accuracy compared to Shallow Fusion, a previously devised method for merging an external language model with an E2E ASR model without necessitating additional training.

Keywords: End-to-end Automatic Speech Recognition, Weighted Finite-State Transducer, Connectionist Temporal Classification, Shallow Fusion, External Language Model.

A preliminary version of this paper appeared in APIC-IST 2023, June 25-28, Fukuoka, Japan. The initial version focused on combining Grammar Transducer and Lexicon Transducer in the Korean E2E-WFST ASR system. This expanded version includes additional experiments and validation with various other WFSTs, providing a more comprehensive analysis and supporting. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-00621, Development of artificial intelligence technology that provides dialog-based multi-modal explainability)

1. Introduction

In this paper, we propose the use of a Grammar Transducer, a language model in the form of a Weighted Finite-State Transducer (WFST) [6] to improve accuracy in Korean End-to-end (E2E) Automatic Speech Recognition (ASR) without further ASR model training. So far, E2E ASR models have shown high accuracy, given that sufficient training speech data was used to train these models. Recent research has predominantly focused on leveraging convolutional neural networks (CNN) to enhance the performance of acoustic models. This approach has been instrumental in advancing the field, demonstrating significant improvements in model accuracy and efficiency [1][3][23]. For example, Conformer CTC [1] showed a Word Error Rate (WER) of 2.7 % in the test clean subset of Librispeech [2] data. Citrinet [3] showed a WER of 2.53 % in the test clean subset of Librispeech. As Librispeech is an English audiobook dataset, it was proved that E2E models show high accuracy in speech data.

However, due to the structure of E2E ASR models, they lack the ability to train language information from given transcripts from training data. They cannot train grammar, the order in which words should be output in a sentence, spelling, or information of subwords that compose each clause or vocabulary. The E2E model can only learn which token suits each input audio feature in the given time frame. This is due to conditional independence assumed between labels in the output layer of the model while calculating the output probability of each output token per time frame. This will be further explained in Related Works. Various Fusion methods were proposed to fuse pre-trained E2E ASR models with language models to provide spelling information and grammar [19]. In this paper, we propose a fusion of clause-based language models converted into WFST with E2E ASR models to improve the accuracy of the pre-trained ASR model. With the proposed method, one could not only boost the accuracy of the Korean E2E model without further training the entire ASR model.

2. Related Work

2.1 DNN-HMM-based Automatic Speech Recognition

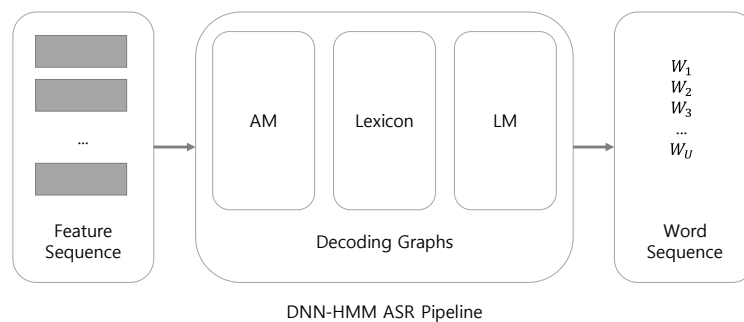


Fig. 1. DNN-HMM Based Automatic Speech Recognition Framework

Before the E2E ASR Model became the trend of Automatic Speech Recognition, the Hidden Markov Model (HMM) [21] based Acoustic Model (AM) was primarily used to train speech data [20]. HMM-based ASR framework consists of an acoustic model, lexicon, and language model (LM), as shown in Fig. 1. The Acoustic model receives acoustic features extracted from speech data as input and output phonemes. The Lexicon Transducer, whose component includes lexicon information, provides phoneme information for each word. It receives a series

of phonemes as input and output words. The grammar Transducer then receives words as input and outputs the next probable word according to the text data on which it was trained. (1) describes this process, where $P(O | V)$ denotes the acoustic model, $P(V | W)$ denotes the pronunciation model, and $P(W)$ denotes the language model."

$$\hat{W} = \underset{W}{\operatorname{argmax}} P(W | O) \approx \underset{W}{\operatorname{argmax}} \left\{ \sum_{V \in R(W)} p(O | V) P(V | W) P(W) \right\} \quad (1)$$

The system compiles the separately trained models in the form of WFST, and each component provides language information to the overall framework through the basic properties of WFST. Suppose the output of one transducer matches the other. In that case, the transducers can be compositioned into a single transducer to form a hierarchically connected Search Graph that can output the most probable sentence for given speech input. Each component's input and output are denoted in **Table 1**. Such a system enables the integration of external information, lexicon, and grammar within the system. This enables the enhancement of the ASR model with a larger language model or lexicon.

Table 1. Components of DNN-HMM-Based Automatic Speech Recognition Framework

Component	Input	Output
Acoustic Model	Speech	Phonemes
Lexicon	Phonemes	Word
Language Model	Word	Words

2.2 End-to-end based Automatic Speech Recognition

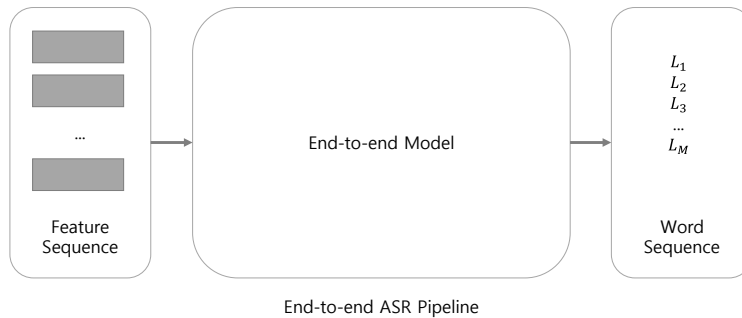


Fig. 2. End-to-end Model-Based Automatic Speech Recognition Framework

Despite the emergence of deep neural networks with high accuracy, it was inapplicable to sequence data such as speech. However, the development of Connectionist Temporal Classification [4] made it possible to apply DNN to ASR. It became possible to train ASR models using only transcription and matching speech to decode input feature vectors into the most probable output sentence. In other words, E2E models were now applicable to Automatic Speech Recognition.

$$\hat{W} = \underset{W}{\operatorname{argmax}} P(W | O) \approx \underset{L}{\operatorname{argmax}} P(L | O) \quad (2)$$

Speech recognition aims to output word sequences with the highest probability for input speech. End-to-end speech recognition models directly map input acoustic feature vectors and output word sequences. The structure of End-to-end models, denoted as (2), interprets the sequence of the most probable word sequence to input speech as the most probable sequence of Connectionist Temporal Classification (CTC) output labels. O is a sequence of acoustic features and set L , which is the output label sequence. It is also shown in Fig. 2. The E2E model, which learns using only data given as input and output without language information, is optimized for the final evaluation index in the learning process. The final evaluation index is usually word error rate (WER) or character error rate (CER).

2.2 Connectionist Temporal Classification

CTC represents the initial E2E technology that gained widespread adoption in ASR [24][25][26]. CTC [4] is the method that allows the process of labeling sequences for input sequences without segmentation. Segmentation provides information on labels aligned to corresponding time frames. CTC labels sequence data without segmentation information in neural network models through Path Probability and Path Aggregation processes.

2.3 Path Probability Calculation

First, the Path Probability Calculation process is the output probability calculation process of one path π that outputs for the input feature O . In the process of labeling the outputs of neural networks, CTC-based networks use softmax output layers. This output layer consists of output tokens, including all labels and blank labels in L . In the model, the model obtains the probability of all possible label sequences for the input. This probability is the multiplication of all conditional probability in each label output for a given input speech. A label sequence, or path, can be denoted as π in (3) [5].

$$p(\pi | O) = \prod_{t=1}^T y_{\pi_t}^t, \forall \pi \in L^T \quad (3)$$

2.4 Path Aggregation

Path Aggregation is a process in which the model outputs probability for each possible output L' . An output is derived using function \mathcal{B} from (4) [5], which uses dynamic programming to remove repeated output tokens as well as blank tokens to create final output from path π . Thus, the output of $\mathcal{B}(a - ab -)$ and $\mathcal{B}(-aa - -abb)$ are both aab .

$$\mathcal{B}(a - ab -) = \mathcal{B}(-aa - -abb) = aab \quad (4)$$

The output probability for each possible output L' is the sum of all output probabilities for path π whose result of \mathcal{B} is L' , as denoted in (5) [5].

$$p(L' | O) = \sum_{\pi \in \mathcal{B}^{-1}(L')} p(\pi | O) \quad (5)$$

2.5 Weighted Finite-State Transducer

Weighted Finite-State Transducer (WFST) [6] is a finite automaton composed of states and transitions that connect two states and are given an input/output symbol, weight each. Given an input symbol in this graph, WFST maps a set of all possible output symbol sequences. The

weights given to the edges of the graph represent the transition probability. The reason for using WFST is that this form of finite automata can be combined and optimized with WFST operations. The composition combines a WFST a with another WFST b if the output of a matches the input of b to create a single WFST that can receive input symbols in WFST a and convert them to WFST b 's output symbols. WFST Optimization operations can be used to build an efficient map of all possible paths by reducing the search space. This can be done by excluding overlapping or meaningless paths. Determinization is a process in which multiple transitions for an input symbol are combined to create a single transition. In Epsilon Removal, we remove ϵ -transition, which receives and the outputs symbol. In short, Optimization deletes unnecessary paths from WFST.

2.6 Existing Fusion Methods of Pretrained End-to-end Automatic Speech Recognition Model and Language Model

In this subsection, we describe the existing fusion method, which does not require fine-tuning or additional training for the model to adapt to a new domain or use more training data to reflect new information [7].

2.6.1 Shallow Fusion

Shallow Fusion [8][9] is a commonly used Fusion method in external knowledge integration.

$$\log P(y_t) = \log P_{AM}(y_t) + \beta \log P_{LM}(y_t) \quad (6)$$

In the Shallow Fusion, the output score of the end-to-end model used as AM and the output score of the LM are added for each decoding step [27]. The decoding score $\log P(yt)$ is calculated by the (6), where $\log P(yt)$ is the final log output probability of label yt . $\log P_{LM}(yt)$ and $\log P_{AM}(yt)$ are output probabilities of external LM and E2E ASR model, respectively. β is an adjustable parameter applied to external LM scores.

2.7 Other Works including Language Model Fusion with Weighted Finite-State Transducer

EESSEN toolkit [10] suggests a combination of the RNN-based ASR model and WFST Search Graph to output the most probable sequence of words in correspondence to the given input speech using the language model and lexicon along with the ASR model. This toolkit was also an effort to infuse language information in inference using the E2E ASR model. Lexicon information and grammar are both encoded as individual WFST and then composed as a Search Graph through a Composition Algorithm. The search Graph of EESSEN consists of a Token Transducer, Lexicon Transducer, and Grammar Transducer. Token Transducer receives a series of phonemes or graphemes from the ASR model and removes redundant tokens from the sequence using self-loops. It works as a path aggregation process of CTC Decoding. Lexicon Transducer transforms the input of phonemes or graphemes and outputs words whose spelling or pronunciation is composed of given phonemes or graphemes. The Grammar Transducer is a language model that receives words from the Lexicon Transducer and outputs the most probable word in the next decoding step. The above transducers are combined to form a single Search Graph, and the WFST form of the graph allows both greedy decoding and beam search decoding.

2.8 Conformer with Lexicon Transducer for Performance Improvement in Korean End-to-end Speech Recognition

While there has been no case of fusing a Grammar Transducer in the Korean ASR E2E model so far, there has been a case in which the fusion of a Lexicon Transducer to a Korean Conformer was suggested in [11]. The specific methodology of this suggestion is detailed in (7).

$$S = \text{shortestpath}(U \circ L) \quad (7)$$

This model extracted the Utterance Transducer from the Conformer train on phoneme-based transcripts and corresponding speech data. Utterance Transducer contains all possible paths of phonemes and blank tokens, which compose the output layer of the Conformer. Utterance Transducer is combined with the Lexicon Transducer and Grammar Transducer, which are trained separately. They are transformed into a single Transducer through the Composition and Optimization algorithm. Then, it derives the shortest path from this Search Graph.

3. Model Architecture

The overall architecture of the proposed method can be denoted as (8). E2E ASR Model, which outputs label sequence, can be denoted as $P(O|L)$. Lexicon Transducer, which provides token information for each clause in Korean language model is denoted as $P(L|W)$. Clause-based Grammar Transducer is denoted as $P(W)$. Decoding process is denoted as argmax_W , calculation of most probable sequence of words or clauses in Search Graph compiled with WFST operation. The decoding result is denoted as \hat{W} .

$$\hat{W} \approx \text{argmax}_W P(W)P(W | L)P(L | O) \quad (8)$$

3.1 Components of Search Graph

This paragraph describes each component of the Search Graph. The search Graph for each input data consists of an Utterance Transducer, a Tokens Transducer, a Lexicon Transducer, and a Grammar Transducer.

3.1.1 Utterance Transducer

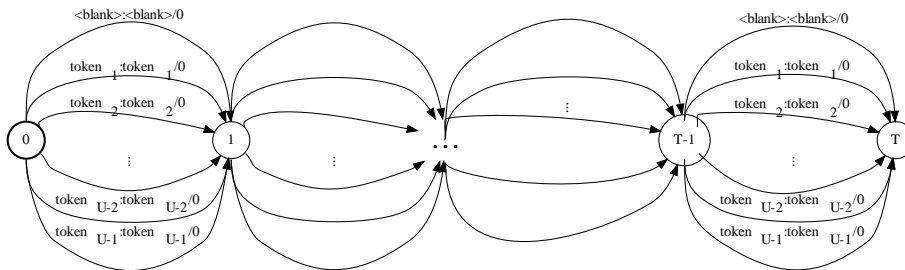


Fig. 3. Topology of Utterance Transducer (U)

Utterance Transducer [22] is the inference result of the pretrained E2E ASR model. It is shown in Fig. 3. For input speech with duration D , extracted Utterance Transducer consists of D states, and edges connecting all time step d and $d - 1$ states. All edges between states represent the output probability of each CTC output label estimated by the E2E ASR model

for each time step. They are given as weights for each transition whose inputs are the corresponding labels. The main difference between the model from [11] is the fact that the Utterance Transducer outputs a sequence of subwords instead of a sequence of phonemes. This is due to the fact that most off-the-shelf Korean E2E ASR models have subwords as output tokens, and this work focused efficiently on producing accurate results without re-training the E2E model.

3.1.2 Token Transducer

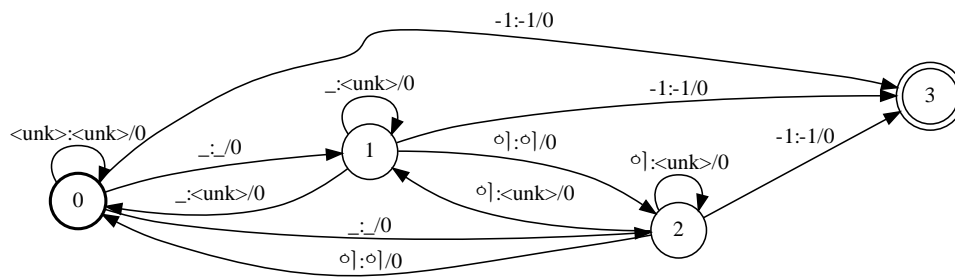


Fig. 4. Topology of Token Transducer (T)

Because Utterance Transducer paths include redundant tokens as well as blank tokens, we use Token Transducer suggested in [10] as function \mathcal{B} from (4). As such, its input symbols set consists of all tokens that make up the E2E model output layer. Transitions that link state to state also include self-loops for each state to remove repeated tokens in the path. The character-level topology of the tokens transducer that is composed of Korean subwords can be seen in Fig. 4.

3.1.3 Lexicon Transducer

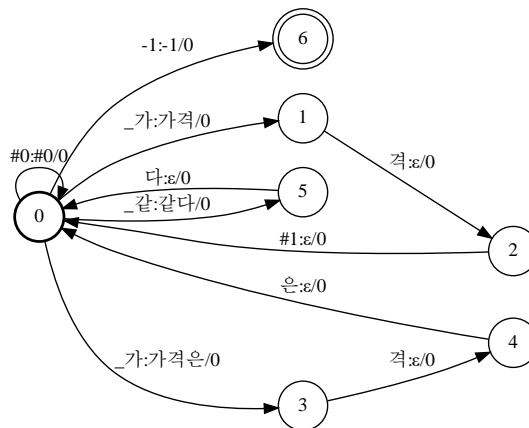


Fig. 5. Topology of Lexicon Transducer (L)

Lexicon Transducer provides lexicon information. It differs from English Lexicon Transducer because Korean is an agglutinative language. Thus, in this architecture, the Lexicon Transducer is a transducer containing token composition information that constitutes all clauses or subwords. Input is tokens from the E2E model, and output is a clause. Fig. 5 shows

an example of such a Lexicon Transducer.

3.1.4 Grammar Transducer

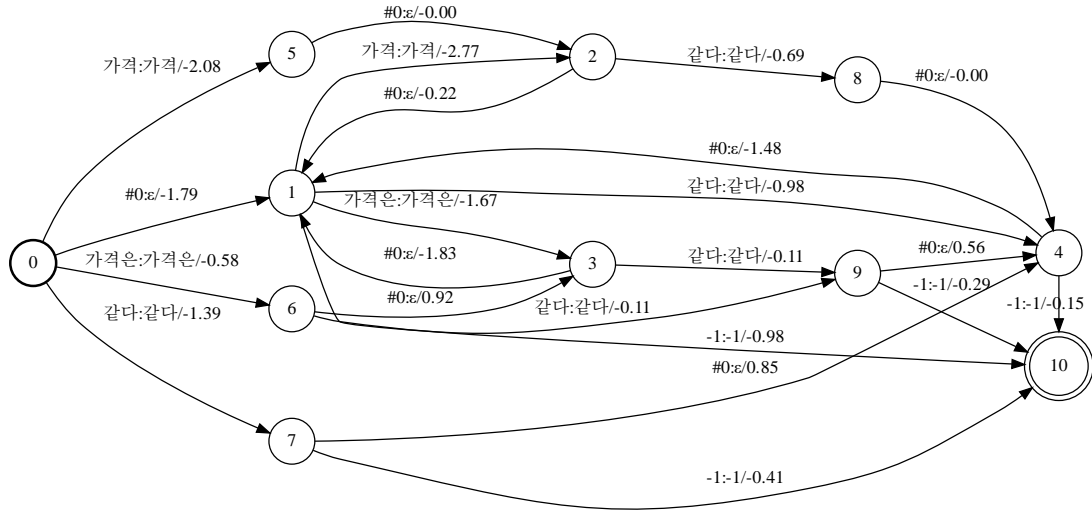


Fig. 6. Topology of Grammar Transducer (G)

Grammar Transducer is a component of the search graph that provides grammar information learned from its training corpus. The language model is an n -gram model converted into WFST. For example, in **Fig. 6**, which is a tri-gram model converted into WFST, the next word is output depending on the word pair from the last decoding step.

3.2 Implementation of Search Graph

$$S = U \circ (T \circ \text{rmeps}(\text{det}(L \circ G))) \quad (9)$$

The Search Graph is denoted as S in (9). It is built through a series of WFST operations to hierarchically combine independently trained models that provide different information. In the structure proposed in this paper, Lexicon Transducer, which receives a token and outputs a phrase, is combined with Grammar Transducer using Composition, which receives a token sequence and outputs a phrase sequence to complete a sentence. It is denoted as LG . Then, Optimization operations Determinization (det) and Epsilon Removal (rmeps) are used to increase efficiency of search in Search Graph S .

3.3 Decoding Process

While there is no defined definition for lattice [12], in this paper, lattice can be viewed as a subset of possible paths in Search Graphs S . The final output of Search Graph is the path with the highest output probability for given input or best path. In this case, the Search Graph was formed from a combination of **Fig. 4**, **Fig. 5**, and **Fig. 6**. The best path that outputs the final inference result was identical to the spoken sentence. It is shown in **Fig. 7**.

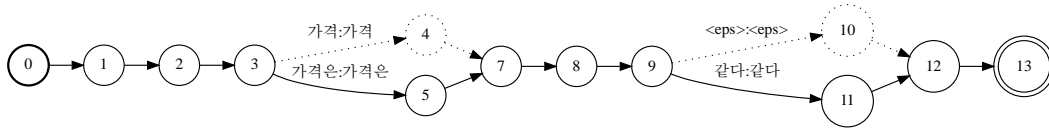


Fig. 7. Best Path Extracted from the Lattice Generated by the Proposed Method

4. Experiments

4.1 Test Dataset

The datasets that are publicly accessible and widely used in Korean speech recognition experiments include Zeroth Korean [13], Korean Loanword Dataset [15], and Ksponspeech [14]. However, Ksponspeech was excluded from the experiments due to the prevalence of fillers and similar data in its labels. In the experiments, the Zeroth Korean and Korean Loanword Dataset are used. A description of each dataset is as follows.

Zeroth Korean Zeroth Korean [13] is 51.6 hours of Korean voice data, which was created by recording 137 volunteers reading 3,000 sentences, in total consisting of 22,263 utterances. Sentences in the transcript were collected from the news media. All data were recorded sentence by sentence. It includes a variety of terminologies, pronouns, and numbers. Such data was used to observe if the fusion of the Grammar Transducer improved the performance of the ASR model in case it contained vocabularies or expressions unlikely to appear in training data.

Korean Loanword Dataset Korean Loanword Dataset [15] is approximately 3,000 hours of recorded colloquial utterances that include loanwords. Loanwords spanned from common nouns to terminology in various fields. This data was selected as a test set to further observe the performance of the proposed method in specific domains or speech that contained uncommon vocabularies. The original test set is about 374.50 hours in total. In this paper, we randomly selected a total of 10-hour duration of data to create a sample test set. The average length of the sample data is 4.90 seconds, and the standard deviation is 1.37.

4.2 Search Graph Components

In order to create Lexicon and Grammar Transducers for each test set, we used transcriptions of the training set of each dataset.

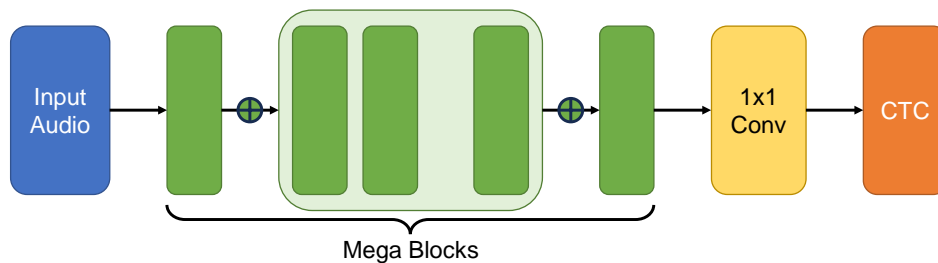


Fig. 8. Citrinet Architecture

End-to-end Automatic Speech Recognition Model Two different off-the-shelf Korean E2E ASR models were used in the experiment. RIVA Citrinet ASR Korean [16] has been trained on 3,500 hours of undisclosed multidomain Korean speech data. The architectural details of this model are presented in Fig. 8. Its output layer comprises tokens extracted from the SentencePiece [17] tokenizer with a vocabulary size of 1,024. the RIVA Conformer ASR [18] is trained on the same speech dataset with a tokenizer of the equivalent size. The Conformer model integrates SpecAugment [22] for enhanced speech data augmentation, as detailed in Fig. 9. Both models are readily accessible online.

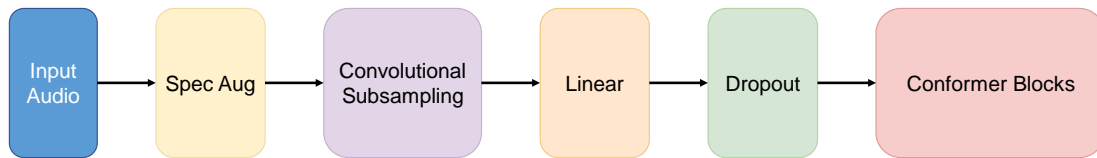


Fig. 9. Conformer Architecture

Lexicon Transducer The Lexicon Transducer was created by extracting all clauses used in training subset transcriptions and using the Google SentencePiece tokenizer from the E2E ASR model to provide lexicon information for each clause.

Grammar Transducer The Grammar Transducer was also trained to transcribe the training set. In this experiment, we used a clause-level corpus, separated by space in Korean sentences, to train a tri-gram external language model for each dataset.

4.3 Result

Comparison of Proposed Method with Baseline Model

The proposed method demonstrated significant improvements in accuracy for Korean ASR as evidenced by the results in Table 2. Specifically, for the Zeroth Korean test set, the baseline Citrinet model had a CER of 4.95%, which the proposed method reduced dramatically to 0.66% (a relatively 86.67% improvement, equivalent to a 4.29%p improvement). Similarly, for the Korean Loanword Dataset, the Citrinet model's CER of 7.31% was reduced to 1.28% by the proposed method (a relatively 82.49% improvement or a 6.03 decrease).

The proposed method also showed notable enhancements compared to the conformer model. The Conformer model's CER of 5.39% for the Zeroth Korean dataset was reduced to 0.78% (an 85.53% improvement, or a 4.61%p improvement), and for the Korean Loanword Dataset, the CER was reduced from 9.53% to 2.42% (a relatively 74.61% improvement, or a 7.11%p improvement). These substantial reductions in CER highlight the effectiveness of the proposed method over the existing ASR models.

Table 2. Comparison of RIVA Citrinet ASR Korean Baseline Model and Proposed Method

ASR Test Set	LM Training Dataset	CER (%)			
		Citrinet	Proposed	Conformer	Proposed
Zeroth Korean	Zeroth Korean	4.95	0.66	5.39	0.78
Korean Loanword Dataset	Korean Loanword Training Dataset	7.31	1.28	9.53	2.42

Comparison of Proposed Method and End-to-end Model Fused with Token Level Language Model By comparing the results of the proposed method using Grammar Transducer based on the clause-level language model and subword-level language model, we establish the necessity of using the clause-level language model. It is necessary to explore if it is possible to build a more efficient language model with the same text corpus using a smaller language unit. Using a tokenizer from the Baseline Model, we created a token-level corpus for each dataset. Then, we compared the performance of this token level Grammar Transducer and the level Grammar Transducer. The results are shown in [Table 2](#). The proposed method supersedes the results using a token-level language model. As Korean is an agglutinative language, the clause, rather than tokens, retains meanings and grammar information. Also, the number of cases in possible Korean syllables being 11,172 has made it more difficult for language models based on subwords to contain grammar information or meaning.

Table 3. Comparison of RIVA Citrinet ASR Korean Baseline Model and Proposed Method

ASR Test Set	LM Training Dataset	CER (%)			
		Citrinet w/ Shallow Fusion	Proposed	Conformer w/ Shallow Fusion	Proposed
Zeroth Korean	Zeroth Korean	2.01	0.66	2.84	0.78
Korean Loanword Dataset	Korean Loanword Training Dataset	2.40	1.28	3.82	2.42

Comparison of Proposed Method with Shallow Fusion The proposed method significantly improved accuracy in Korean ASR, as shown in [Table 3](#). In the Zeroth Korean test set, the baseline Citrinet model with Shallow Fusion achieved a CER of 2.01%, which was notably reduced to 0.66% by our method (a relative improvement of 67.16%, or a decrease of 1.35%p). For the Korean Loanword Dataset, the Citrinet model with Shallow Fusion had a CER of 2.40%, improved to 1.28% by our method (a relative improvement of 46.67%, or a decrease of 1.12%p). The proposed method also demonstrated significant advancements compared to the Conformer model incorporating Shallow Fusion. The Conformer model's CER of 2.84% for the Zeroth Korean dataset decreased to 0.78% with our method (a relative improvement of 72.54%, or a decrease of 2.06%p), and for the Korean Loanword Dataset, the CER decreased from 3.82% to 2.42% (a relative improvement of 36.65%, or a decrease of 1.40%p). These notable improvements underscore the superior performance of our method over Shallow Fusion for structured datasets, particularly in both the Zeroth Korean and Korean Loanword datasets.

5. Conclusion

Performance improvements were evident with the proposed method, Fusion with Grammar Transducer, and E2E ASR model. In news domains or loanwords containing pronouns and specialized terminology, clause-level Grammar Transducer Fusion enhanced performance, as indicated by evaluation results. However, while it exhibited improved performance in clearly segmented sentences, it encountered difficulties in inferring spontaneous speech. Although the Grammar Transducer can enhance End-to-end model performance, its effectiveness is more pronounced when utterances conform to grammar. Our proposed method introduces a novel focus on clause-level language modeling, aiming to advance the decoding of complex speech patterns in automatic speech recognition.

Acknowledgement

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00621, Development of artificial intelligence technology that provides dialog-based multi-modal explainability)

References

- [1] A. Gulati et al., “Conformer: Convolution-augmented Transformer for Speech Recognition,” in *Proc. of the Interspeech 2020*, pp. 5036–5040, 2020. [Article \(CrossRef Link\)](#).
- [2] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An ASR corpus based on public domain audio books,” in *Proc. of 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, Apr. 2015. [Article \(CrossRef Link\)](#).
- [3] S. Majumdar, J. Balam, O. Hrinchuk, V. Lavrukhin, V. Noroozi, and B. Ginsburg, “CitriNet: Closing the Gap between Non-Autoregressive and Autoregressive End-to-End Models for Automatic Speech Recognition,” *arXiv*, Apr. 04, 2021. [Article \(CrossRef Link\)](#).
- [4] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proc. of the 23rd International Conference on Machine Learning, in ICML '06*, New York, NY, USA: Association for Computing Machinery, pp. 369–376, Jun. 2006. [Article \(CrossRef Link\)](#).
- [5] D. Wang, X. Wang, and S. Lv, “An Overview of End-to-End Automatic Speech Recognition,” *Symmetry*, vol. 11, no. 8, Art. no. 8, Aug. 2019. [Article \(CrossRef Link\)](#).
- [6] M. Mohri, F. Pereira, and M. Riley, “Weighted finite-state transducers in speech recognition,” *Computer Speech & Language*, vol. 16, no. 1, pp. 69–88, Jan. 2002. [Article \(CrossRef Link\)](#).
- [7] A. Graves and N. Jaitly, “Towards End-To-End Speech Recognition with Recurrent Neural Networks,” in *Proc. of the 31st International Conference on Machine Learning*, PMLR, pp. 1764–1772, Jun. 2014.
- [8] J. Chorowski and N. Jaitly, “Towards Better Decoding and Language Model Integration in Sequence to Sequence Models,” in *Proc. of the Interspeech 2017*, pp. 523–527, 2017. [Article \(CrossRef Link\)](#).
- [9] A. Kannan, Y. Wu, P. Nguyen, T. N. Sainath, Z. Chen, and R. Prabhavalkar, “An Analysis of Incorporating an External Language Model into a Sequence-to-Sequence Model,” in *Proc. of 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5828, Apr. 2018. [Article \(CrossRef Link\)](#).
- [10] Y. Miao, M. Gowayyed, and F. Metze, “EESSEN: End-to-end speech recognition using deep RNN models and WFST-based decoding,” in *Proc. of 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 167–174, Feb. 2015. [Article \(CrossRef Link\)](#).
- [11] H. Son, “Convolution Augmented Transformer with Lexicon Transducer for Performance Improvement in Korean End-to-end Speech Recognition,” M.S. thesis, Dept. of Computer Science and Engineering, Sogang Univ., 2021.
- [12] D. Povey et al., “Generating exact lattices in the WFST framework,” in *Proc. of 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4213–4216, Mar. 2012. [Article \(CrossRef Link\)](#).
- [13] L. Jo and W. Lee, “Zeroth-Korean,” [Online]. Available: <https://www.openslr.org/40/>.
- [14] J.-U. Bang et al., “KsponSpeech: Korean Spontaneous Speech Corpus for Automatic Speech Recognition,” *Applied Sciences*, vol. 10, no. 19, Jan. 2020. [Article \(CrossRef Link\)](#).
- [15] NHN dquest Inc., “Korean Loanword Dataset,” [Online]. Available: <https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=131>.
- [16] NVIDIA, “RIVA CitriNet ASR Korean,” [Online]. Available: https://catalog.ngc.nvidia.com/orgs/nvidia/teams/riva/models/spechtotext_ko_kr_citriNet

- [17] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing," in *Proc. of the 2018 Conference on EMNLP, Belgium*, pp. 66–71, Jan. 2018. [Article \(CrossRef Link\)](#).
- [18] NVIDIA, "RIVA Conformer ASR Korean," [Online]. Available: https://catalog.ngc.nvidia.com/orgs/nvidia/teams/tao/models/speechoctext_ko_kr_conformer
- [19] Z. Wang, Y. Zhao, L. Wu, X. Bi, Z. Dawa, and Q. Ji, "Cross-Language Transfer Learning-based Lhasa-Tibetan Speech Recognition," *CMC*, vol. 73, no. 1, pp. 629–639, 2022. [Article \(CrossRef Link\)](#).
- [20] G. Hinton et al., "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Jan. 2012. [Article \(CrossRef Link\)](#).
- [21] L. Rabiner and B. Juang, "An introduction to hidden Markov models," *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, Jan. 1986. [Article \(CrossRef Link\)](#).
- [22] D. Povey et al., "Generating exact lattices in the WFST framework," in *Proc. of 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4213–4216, Mar. 2012. [Article \(CrossRef Link\)](#).
- [23] W.-T. Sung, H.-W. Kang, and S.-J. Hsiao, "Speech Recognition via CTC-CNN Model," *CMC*, vol. 76, no. 3, pp. 3833–3858, 2023. [Article \(CrossRef Link\)](#).
- [24] H. Soltau, H. Liao, and H. Sak, "Neural Speech Recognizer: Acoustic-to-Word LSTM Model for Large Vocabulary Speech Recognition," *arXiv*, Oct. 31, 2016. [Article \(CrossRef Link\)](#)
- [25] G. Zweig, C. Yu, J. Droppo, and A. Stolcke, "Advances in all-neural speech recognition," in *Proc. of 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4805–4809, Mar. 2017. [Article \(CrossRef Link\)](#).
- [26] A. Zeyer, E. Beck, R. Schlüter, and H. Ney, "CTC in the Context of Generalized Full-Sum HMM Training," in *Proc. of the Interspeech 2017*, pp. 944–948, 2017. [Article \(CrossRef Link\)](#).
- [27] J. Li et al., "Developing RNN-T Models Surpassing High-Performance Hybrid Models with Customization Capability," in *Proc. of the Interspeech 2020*, pp. 3590–3594, 2020. [Article \(CrossRef Link\)](#).



Junseok Oh received his B.E. degree in Computer Science and Engineering from Sogang University, Republic of Korea, in 2017. He also received his M.E. degree in Computer Science and Engineering from Sogang University in 2019. He is currently pursuing a Ph.D. degree in Computer Science and Engineering at Sogang University. His research interests include speech recognition and audio content information.



Eunsoo Cho received her B.A. degree in Chinese Culture, B.A.Sc in Art and Technology, and B.E. in Convergence Software Course from Sogang University in 2021. She also received her M.E. degree in Computer Science and Engineering from Sogang University in 2023. Her main research interest is speech recognition.



Ji-Hwan Kim received the B.E. and M.E. degrees in Computer Science from KAIST (Korea Advanced Institute of Science and Technology) in 1996 and 1998, respectively, and a Ph.D. degree in Engineering from the University of Cambridge in 2001. From 2001 to 2007, he was a chief research engineer and a senior research engineer at LG Electronics Institute of Technology, where he was engaged in the development of speech recognizers for mobile devices. In 2004, he was a visiting scientist at MIT Media Lab. Since 2007, he has been a faculty member in the Department of Computer Science and Engineering and the Department of Artificial Intelligence at Sogang University. Currently, he is a full professor. His research interests include spoken multimedia content search, speech recognition for embedded systems, and dialogue understanding.