LARGE CONTEXT END-TO-END AUTOMATIC SPEECH RECOGNITION VIA EXTENSION OF HIERARCHICAL RECURRENT ENCODER-DECODER MODELS

Ryo Masumura, Tomohiro Tanaka, Takafumi Moriya, Yusuke Shinohara, Takanobu Oba, Yushi Aono

NTT Media Intelligence Laboratories, NTT Corporation, Japan

masumura.ryo@lab.ntt.co.jp

ABSTRACT

This paper describes a novel end-to-end automatic speech recognition (ASR) method that takes into consideration long-range sequential context information beyond utterance boundaries. In spontaneous ASR tasks such as those for discourses and conversations, the input speech often comprises a series of utterances. Accordingly, the relationships between the utterances should be leveraged for transcribing the individual utterances. While most previous end-to-end ASR methods only focus on utterance-level ASR that handles single utterances independently, the proposed method (which we call "large-context end-to-end ASR") can explicitly utilize relationships between a current target utterance and all preceding utterances. The method is modeled by combining an attention-based encoder-decoder model, which is one of the most representative end-to-end ASR models, with hierarchical recurrent encoder-decoder models, which are effective language models for capturing long-range sequential contexts beyond the utterance boundaries. Experiments on Japanese discourse speech tasks demonstrate the proposed method yields significant ASR performance improvements compared with the conventional utterance-level end-to-end ASR system.

***Index Terms*—** End-to-end automatic speech recognition, attention based encoder-decoder, hierarchical recurrent encoder-decoder

1. INTRODUCTION

In the automatic speech recognition (ASR) field, end-to-end ASR methods that directly model a generative probability of a text given an input speech have attracted much attention. While classical ASR methods have introduced three component models, i.e., an acoustic model, a language model, and a pronunciation model, the end-to-end ASR methods only use a single model that integrates them. In fact, in the classical ASR methods, it is difficult to optimize the overall system since each component models were independently trained. On the other hand, the end-to-end ASR methods can learn the overall system in one step.

There are several modeling methods for performing end-to-end ASR. One of the main ones is connectionist temporal classification, in which a blank token is leveraged for handling differences in the length of input acoustic features and output tokens [1–5]. Another is attention based encoder-decoder models, which are language models conditioned on input speech. In this method, an attention mechanism is utilized for automatically determining which acoustic features should be used to predict the next token [6–11]. Also, recurrent neural network (RNN) transducers and recurrent neural aligners have been developed for use in online decoding [12, 13].

However, previous end-to-end ASR methods have mainly focused on utterance-level ASR in which each utterance is independently transcribed. Therefore, they can not capture relationships between utterances even when discourse speech and conversation speech, which comprise a series of utterances, have to be transcribed. In language modeling, it has been reported that long-range linguistic context information beyond utterance boundaries is effective for improving perplexity and ASR performance [14–17]. Therefore, improvements in end-to-end ASR systems can also be expected by explicitly capturing long-range sequential contexts beyond utterance boundaries.

In this paper, we propose a large-context end-to-end ASR method that is suitable for transcribing a series of utterances. Our idea is to combine attention-based encoder-decoder models with hierarchical recurrent encoder-decoder models, which are language models that effectively capture long-range sequential contexts beyond utterance boundaries [18–20]. These two models can be naturally integrated since both are language models conditioned on different contexts. The proposed method makes it possible to utilize not only a target utterance's speech information but also all preceding transcribed text information for transcribing a target utterance. The method also achieves effective ASR decoding of a series of utterances by repeatedly feeding the transcribed text of an utterance just before a target utterance and acoustic features of the target utterance.

The method is closely related to context-dependent utterance-level end-to-end ASR methods. Various auxiliary features such as speaker information or language information have been utilized for enhancing utterance-level end-to-end ASR methods [21–23]. The large-context end-to-end ASR method can be regarded as an utterance-level end-to-end ASR method that utilizes all transcribed texts as auxiliary features in an end-to-end manner. Long-range contexts beyond utterance boundaries have also been recently utilized in neural conversation models [18–20] and neural machine translation models [24–26] that are similar generative models to the end-to-end ASR models. Actually, the large-context end-to-end ASR method is inspired by them. To the best of our knowledge, however, our work constitutes the initial study on end-to-end ASR methods that can handle long-range contexts beyond the utterance boundaries.

In experiments on discourse speech tasks using a corpus of spontaneous Japanese, we demonstrated the proposed method yields significant ASR performance improvements compared with the utterance-level end-to-end ASR system.

2. UTTERANCE-LEVEL END-TO-END AUTOMATIC SPEECH RECOGNITION

This section briefly describes utterance-level end-to-end ASR using attention-based encoder-decoder modeling [6–11]. It models a generative probability of a text $\boldsymbol{W} = \{w_1, \cdots, w_N\}$ given speech $\boldsymbol{X} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_M\}$, where w_n is the n-th token in the text and $\boldsymbol{x}_m$

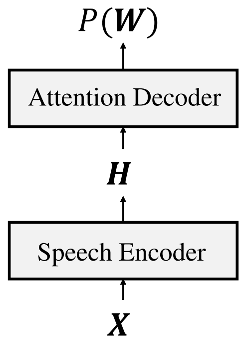

Fig. 1. Network structure of utterance-level end-to-end ASR system.

is the m-th acoustic feature in the speech. N is the number of tokens in the text and M is the number of acoustic features in the speech. In attention-based encoder-decoder modeling, the generative probability of $\boldsymbol{W}$ is defined as

$$P(\boldsymbol{W}|\boldsymbol{X},\boldsymbol{\Theta}_{\mathtt{e2e}})=\prod_{n=1}^{N}P(w_n|w_1,\cdots,w_{n-1},\boldsymbol{X},\boldsymbol{\Theta}_{\mathtt{e2e}}), \quad (1)$$

where $\boldsymbol{\Theta}_{\mathtt{e2e}}$ represents the model parameter sets. $P(w_n|w_1,\cdots,w_{n-1},\boldsymbol{X},\boldsymbol{\Theta}_{\mathtt{e2e}})$ can be computed using a speech encoder and an attention decoder, both of which are composed of neural networks.

2.1. Network Structure

Fig. 1 shows the network structure of the utterance-level end-to-end ASR system. The speech encoder converts acoustic features into the hidden representations $\boldsymbol{H}$. These are defined as

$$\boldsymbol{H}=\mathtt{SpeechEnc}(\boldsymbol{X};\boldsymbol{\Theta}_{\mathtt{e2e}}), \quad (2)$$

where $\mathtt{SpeechEnc}()$ is a function of the speech encoder, which is usually modeled by bidirectional RNNs.

The attention decoder computes the generative probability of a token from preceding tokens and the hidden representations of the speech using an attention mechanism. The predicted probabilities of the n-th token w_n are calculated as

$$\begin{aligned}&P(w_n|w_1,\cdots,w_{n-1},\boldsymbol{X},\boldsymbol{\Theta}_{\mathtt{e2e}})\\&\quad=\mathtt{AttenDec}(w_1,\cdots,w_{n-1},\boldsymbol{H};\boldsymbol{\Theta}_{\mathtt{e2e}}),\end{aligned} \quad (3)$$

where $\mathtt{AttenDec}()$ is a function of the attention decoder, which is usually modeled by unidirectional RNNs and an attention mechanism.

2.2. Training

In utterance-level end-to-end ASR, a model parameter set can be optimized from the utterance-level training data set $\mathcal{D}_{\mathtt{e2e}}=\{(\boldsymbol{X}^1,\boldsymbol{W}^1),\cdots,(\boldsymbol{X}^T,\boldsymbol{W}^T)\}$, where T is the number of utterances in the training data set. The parameter sets are optimized by

$$\hat{\boldsymbol{\Theta}}_{\mathtt{e2e}}=\underset{\boldsymbol{\Theta}_{\mathtt{e2e}}}{\operatorname{argmin}}-\sum_{t=1}^{T}\sum_{n=1}^{N^t}\log P(w_n^t|w_1^t,\cdots,w_{n-1}^t,\boldsymbol{X}^t,\boldsymbol{\Theta}_{\mathtt{e2e}}), \quad (4)$$

where w_n^t is the n-th token for the t-th utterance and $\boldsymbol{X}^t$ is the acoustic features in the t-th utterance. N^t is the number of tokens in the t-th utterance.

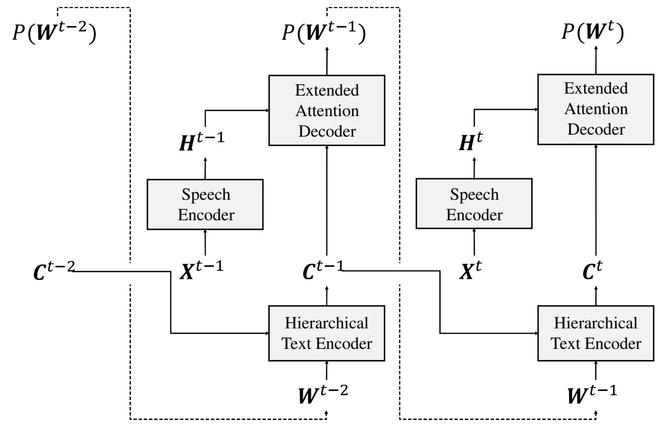

Fig. 2. Network structure of large-context end-to-end ASR system.

3. LARGE CONTEXT END-TO-END AUTOMATIC SPEECH RECOGNITION

This section details a large context end-to-end ASR system composed of attention-based encoder-decoders integrated with hierarchical recurrent encoder-decoders. The large context end-to-end ASR can effectively handle a series of utterances, i.e., conversation-level data or discourse-level data, while utterance-level end-to-end ASR handles each utterance independently. The proposed method models a generative probability of a sequence of utterance-level texts $\mathcal{W}=\{\boldsymbol{W}^t,\cdots,\boldsymbol{W}^T\}$ given a sequence of utterance-level speech $\mathcal{X}=\{\boldsymbol{X}^1,\cdots,\boldsymbol{X}^T\}$, where $\boldsymbol{W}^t=\{w_1^t,\cdots,w_{N^t}^t\}$ is the t-th utterance-level text composed of tokens and $\boldsymbol{X}^t=\{\boldsymbol{x}_1^t,\cdots,\boldsymbol{x}_{M^t}^t\}$ is the t-th utterance-level speech composed of acoustic features. T is the number of utterances in a series of utterances, N^t is the number of tokens in the t-th text and M^t is the number of acoustic features in the t-th utterance. The generative probability of $\mathcal{W}$ is defined as

$$\begin{aligned}P(\mathcal{W}|\mathcal{X},\boldsymbol{\Theta}_{\mathtt{1e2e}})&=\prod_{t=1}^{T}P(\boldsymbol{W}^t|\boldsymbol{W}^1,\cdots,\boldsymbol{W}^{t-1},\boldsymbol{X}^t,\boldsymbol{\Theta}_{\mathtt{1e2e}})\\&=\prod_{t=1}^{T}\prod_{n=1}^{N^t}P(w_n^t|w_1^t,\cdots,w_{n-1}^t,\\&\qquad\boldsymbol{W}^1,\cdots,\boldsymbol{W}^{t-1},\boldsymbol{X}^t,\boldsymbol{\Theta}_{\mathtt{1e2e}}),\end{aligned} \quad (5)$$

where $\boldsymbol{\Theta}_{\mathtt{1e2e}}$ is the model parameter set. $P(w_n^t|w_1^t,\cdots,w_{n-1}^t,\boldsymbol{W}^1,\cdots,\boldsymbol{W}^{t-1},\boldsymbol{X}^t,\boldsymbol{\Theta}_{\mathtt{1e2e}})$ can be computed using a hierarchical text encoder, a speech encoder, and an extended attention decoder.

3.1. Network Structure

Fig. 2 shows the network structure of the large-context end-to-end ASR system. The hierarchical text encoder converts all preceding texts into a continuous vector. The t-th continuous vector $\boldsymbol{C}^t$ is defined as

$$\begin{aligned}\boldsymbol{C}^t&=\mathtt{HierarchicalTextEnc}(\boldsymbol{W}^1,\cdots,\boldsymbol{W}^{t-1};\boldsymbol{\Theta}_{\mathtt{1e2e}}),\\&=\mathtt{HierarchicalTextEnc}(\boldsymbol{W}^{t-1},\boldsymbol{C}^{t-1};\boldsymbol{\Theta}_{\mathtt{1e2e}})\end{aligned} \quad (6)$$

where $\mathtt{HierarchicalTextEnc}()$ is a function of the hierarchical text encoder. The speech encoder converts an utterance into continuous vectors. The t-th speech continuous vectors $\boldsymbol{H}^t$ is defined as

$$\boldsymbol{H}^t = \texttt{SpeechEnc}(\boldsymbol{X}^t; \boldsymbol{\Theta}_{\texttt{le2e}}), \tag{7}$$

where $\texttt{SpeechEnc}()$ is a function of the speech encoder. The extended attention decoder computes the generative probability of a token from preceding tokens in a target utterance, the continuous vector of all preceding texts, and hidden vectors of the target speech. The generative probability of w_n^t is calculated as

$$\begin{aligned} &P(w_n^t | w_1^t, \cdots, w_{n-1}^t, \boldsymbol{W}^1, \cdots, \boldsymbol{W}^{t-1}, \boldsymbol{X}^t, \boldsymbol{\Theta}_{\texttt{le2e}}) \\ &= \texttt{ExtAttenDec}(w_1^t, \cdots, w_{n-1}^t, \boldsymbol{C}^t, \boldsymbol{H}^t; \boldsymbol{\Theta}_{\texttt{le2e}}), \end{aligned} \tag{8}$$

where $\texttt{ExtAttenDec}()$ is a function of the extended attention decoder.

3.2. Implementation

Hierarchical text encoder: The hierarchical text encoder is constructed from a token-level encoder and an utterance-level encoder. In the token-level encoder, each token is converted into a continuous vector as

$$\boldsymbol{w}_n^{t-1} = \texttt{Embed}(w_n^{t-1}; \boldsymbol{\theta}_{\texttt{w}}), \tag{9}$$

where $\texttt{Embed}()$ is a function to convert a token into a continuous vector and $\boldsymbol{\theta}_{\texttt{w}}$ is a trainable parameter. In the token-level encoder, all tokens in each text are embedded into a continuous vector as

$$\begin{aligned} \boldsymbol{u}_n^{t-1} &= \texttt{Recurrent}(\boldsymbol{w}_1^{t-1}, \cdots, \boldsymbol{w}_n^{t-1}; \boldsymbol{\theta}_{\texttt{u}}) \\ &= \texttt{Recurrent}(\boldsymbol{w}_n^{t-1}, \boldsymbol{u}_{n-1}^{t-1}; \boldsymbol{\theta}_{\texttt{u}}), \end{aligned} \tag{10}$$

where $\texttt{Recurrent}()$ is a function based on unidirectional RNNs and $\boldsymbol{\theta}_{\texttt{u}}$ is a trainable parameter. Therefore, the entire information of a single text can be embedded into $\boldsymbol{u}_{N^{t-1}}^{t-1}$, which is expressed as

$$\begin{aligned} \boldsymbol{U}^{t-1} &= \texttt{Recurrent}(\boldsymbol{W}^{t-1}; \boldsymbol{\theta}_{\texttt{u}}) \\ &= \boldsymbol{u}_{N^{t-1}}^{t-1}. \end{aligned} \tag{11}$$

In addition, in order to capture multiple preceding texts, continuous vectors extracted from individual preceding texts are embedded into a continuous vector using the utterance-level decoder. A continuous vector that embeds all information from an initial text into the $t-1$-th text is defined as

$$\begin{aligned} \boldsymbol{C}^t &= \texttt{Recurrent}(\boldsymbol{U}^1, \cdots, \boldsymbol{U}^{t-1}; \boldsymbol{\theta}_{\texttt{c}}) \\ &= \texttt{Recurrent}(\boldsymbol{U}^{t-1}, \boldsymbol{C}^{t-1}; \boldsymbol{\theta}_{\texttt{c}}), \end{aligned} \tag{12}$$

where $\boldsymbol{\theta}_{\texttt{c}}$ is the trainable parameter.

Speech encoder: In a speech encoder, utterance-level acoustic features are converted into hidden vector sequences. The t-th hidden vector sequence $\boldsymbol{H}^t = \{\boldsymbol{h}_1^t, \cdots, \boldsymbol{h}_{K^t}^t\}$ is produced by

$$\boldsymbol{h}_k^t = \texttt{BiRecurrent}(\boldsymbol{x}_1^t, \cdots, \boldsymbol{x}_{M^t}^t, k; \boldsymbol{\theta}_{\texttt{h}}), \tag{13}$$

where $\texttt{BiRecurrent}()$ is the bidirectional RNNs and $\boldsymbol{\theta}_{\texttt{h}}$ is the trainable parameter. K^t is the length of the subsampled acoustic features in the t-th utterance.

Extended attention decoder: In an extended attention decoder, which corresponds to a conditional generative model, the history of both preceding tokens in the current utterance and all preceding utterances is first summarized as a continuous vector. The continuous vector that summarizes from the initial token in the initial utterance to the n-th token in the t-th utterance is defined as

$$\begin{aligned} \boldsymbol{v}_n^t &= \texttt{Recurrent}(\boldsymbol{z}_1^t, \cdots, \boldsymbol{z}_n^t; \boldsymbol{\theta}_{\texttt{v}}) \\ &= \texttt{Recurrent}(\boldsymbol{z}_n^t, \boldsymbol{v}_{n-1}^t; \boldsymbol{\theta}_{\texttt{v}}), \end{aligned} \tag{14}$$

$$\boldsymbol{z}_n^t = [{\boldsymbol{w}_n^t}^\top, {\boldsymbol{C}^t}^\top]^\top, \tag{15}$$

where $\boldsymbol{\theta}_{\texttt{v}}$ is the model parameter. The continuous vector is used for summarizing hidden speech vectors as a continuous vector. The t-th continuous vector in the t-th utterance is calculated as

$$\boldsymbol{d}_n^t = \sum_{k=1}^{K^t} \frac{\exp \texttt{Atten}(\boldsymbol{h}_k^t, \boldsymbol{v}_n^t; \boldsymbol{\theta}_{\texttt{d}})}{\sum_{k'=1}^{K^t} \exp \texttt{Atten}(\boldsymbol{h}_{k'}^t, \boldsymbol{v}_n^t; \boldsymbol{\theta}_{\texttt{d}})} \boldsymbol{h}_k^t, \tag{16}$$

where $\texttt{Atten}()$ is the function for computing attention weights and $\boldsymbol{\theta}_{\texttt{d}}$ is the trainable parameter. A context vector for estimating the t-th token in the t-th utterance is produced by

$$\boldsymbol{s}_n^t = \texttt{NonLinear}([{\boldsymbol{v}_n^t}^\top, {\boldsymbol{d}_n^t}^\top, {\boldsymbol{C}^t}^\top]^\top; \boldsymbol{\theta}_{\texttt{s}}), \tag{17}$$

where $\texttt{NonLinear}()$ is a non-linear transformational function and $\boldsymbol{\theta}_{\texttt{s}}$ is the trainable parameter. Predicted probabilities of the n-th token in the t-th utterance are produced by

$$\begin{aligned} &P(w_n^t | w_1^t, \cdots, w_{n-1}^t, \\ &\quad \boldsymbol{W}^1, \cdots, \boldsymbol{W}^{t-1}, \boldsymbol{X}^t, \boldsymbol{\Theta}) = \texttt{SOFTMAX}(\boldsymbol{s}_n^t; \boldsymbol{\theta}_{\texttt{o}}), \end{aligned} \tag{18}$$

where $\texttt{SOFTMAX}()$ is a softmax transformational function and $\boldsymbol{\theta}_{\texttt{o}}$ is the trainable parameter.

3.3. Training

In the large context end-to-end ASR, a model parameter set that includes all trainable parameters can be summarized as

$$\boldsymbol{\Theta}_{\texttt{le2e}} = \{\boldsymbol{\theta}_{\texttt{w}}, \boldsymbol{\theta}_{\texttt{u}}, \boldsymbol{\theta}_{\texttt{c}}, \boldsymbol{\theta}_{\texttt{h}}, \boldsymbol{\theta}_{\texttt{v}}, \boldsymbol{\theta}_{\texttt{d}}, \boldsymbol{\theta}_{\texttt{s}}, \boldsymbol{\theta}_{\texttt{o}}\}. \tag{19}$$

The model parameter set can be optimized from training data set $\mathcal{D}^{\texttt{le2e}} = \{(\mathcal{X}^1, \mathcal{W}^1), \cdots, (\mathcal{X}^D, \mathcal{W}^D)\}$, where D is the number of conversation-level or discourse-level data in the training data set. The d-th data element is represented as $\mathcal{X}^d = \{\boldsymbol{X}^{1,d}, \cdots, \boldsymbol{X}^{T^d,d}\}$ and $\mathcal{W}^d = \{\boldsymbol{W}^{1,d}, \cdots, \boldsymbol{W}^{T^d,d}\}$, where $\boldsymbol{W}^{t,d} = \{w_1^{t,d}, \cdots, w_{N^{t,d}}^{t,d}\}$. The model parameter set is optimized by

$$\begin{aligned} \hat{\boldsymbol{\Theta}}_{\texttt{le2e}} = \mathop{\mathrm{argmin}}_{\boldsymbol{\Theta}_{\texttt{le2e}}} - \sum_{d=1}^{D} \sum_{t=1}^{T^d} \sum_{n=1}^{N^{t,d}} \log P(w_n^{t,d} | w_1^t, \cdots, w_{n-1}^{t,d}, \\ \boldsymbol{W}^{1,d}, \cdots, \boldsymbol{W}^{t-1,d}, \boldsymbol{X}^{t,d}, \boldsymbol{\Theta}_{\texttt{le2e}}). \end{aligned} \tag{20}$$

Actually, most of the trainable parameters are with the same as those in utterance-level end-to-end ASR. In order to efficiently optimize these parameters, those optimized in the utterance-level end-to-end ASR system should be used as the initial parameters in the large-context end-to-end ASR systems.

3.4. ASR Decoding

ASR decoding of a sequence of utterance-level texts from a sequence of utterance-level acoustic features using the large context end-to-end ASR is achieved by recursively conducting utterance-level decoding. The ASR decoding problem for the t-th utterance is defined as

$$\hat{\boldsymbol{W}}^t = \mathop{\mathrm{argmax}}_{\boldsymbol{W}^t} P(\boldsymbol{W}^t | \hat{\boldsymbol{W}}^1, \cdots, \hat{\boldsymbol{W}}^{t-1}, \boldsymbol{X}^t, \boldsymbol{\Theta}_{\texttt{le2e}}), \tag{21}$$

Table 1. Experimental data sets.

	Data size (Hours)	Number of discourses	Number of utterances	Number of characters
Train	512.6	3,181	413,240	13,349,780
Valid	4.8	33	4,166	122,097
Test 1	1.8	10	1,272	48,064
Test 2	1.9	10	1,292	47,970
Test 3	1.3	10	1,385	32,089

where $\hat{\boldsymbol{W}}^{t-1}$ is ASR output of the $t-1$-th utterance. Therefore, $\hat{\boldsymbol{W}}^{t}$ is recursively used for decoding the text of the $t+1$-th utterance. Thus, the computation cost of ASR decoding using the large-context end-to-end ASR is almost comparable to that using utterance-level end-to-end ASR.

4. EXPERIMENTS

In experiments, we used the Corpus of Spontaneous Japanese (CSJ) [27]. We divided the CSJ into a training set (Train), a validation set (Valid), and three test sets (Test 1, 2, and 3). The validation set was used for optimizing several hyper parameters. Each discourse-level speech was segmented into utterances in accordance with previous work [28]. This paper used characters as the tokens. Details of the data sets are shown in Table 1.

4.1. Setups

For evaluation purposes, we constructed an utterance-level end-to-end ASR system and the large-context end-to-end ASR system. In addition, we constructed both systems without introducing a speech encoder. Note that the utterance-level end-to-end ASR system without a speech encoder is regarded as an RNN-based language model [29, 30] and the large-context end-to-end ASR system without a speech encoder is regarded as a discourse-context language model based on a hierarchical recurrent encoder-decoder [17].

In the hierarchical text encoder, a 1-layer unidirectional long short-term memory RNN (LSTM-RNN) with 512 units was introduced into both the token-level encoder and the utterance-level encoder. In the speech encoder, we used 40 log mel-scale filterbank coefficients appended with delta and acceleration coefficients as acoustic features; the frame shift was 10 ms. We stacked 7 consecutive acoustic features as the input of the speech encoder where we formed them on every 30 ms for subsampling. We used a sigmoid non-linear layer at the bottom layer and a stacked 4-layer bidirectional LSTM-RNN with 512 units. In the attention decoder and the extended attention decoder, a unidirectional LSTM-RNN with 512 units was introduced. For the attention mechanism, we used global attention [31]. The output unit size, which corresponds to the number of characters in the training set, was set to 3,084. For training these models, we used mini-batch stochastic gradient descent with gradient norm clipping 1.0. In each LSTM-RNN, we used variational dropout where its rate was set to 0.2 for the speech encoder and 0.4 for the hierarchical text encoder, the attention decoder and the extended attention decoder. Initial parameters in the utterance-level end-to-end ASR were randomly initialized. Optimized parameters in the utterance-level end-to-end ASR system were partly used for the initial parameters in the large-context end-to-end ASR systems. For the mini-batch training, we truncated each lecture to 30 utterances. Mini-batch size was set to 2. For ASR decoding using both the utterance-level and the large-context end-to-end ASR, we used a beam search algorithm in which the beam size was set to 20.

Table 2. Character-level perplexity results.

	Speech encoder	Test 1	Test 2	Test 3
Utterance-level ASR	w/o	12.48	14.13	14.75
Large-context ASR	w/o	11.62	12.95	13.26
Utterance-level ASR	w	1.35	1.28	1.32
Large-context ASR	w	**1.31**	**1.25**	**1.28**

Table 2. Character error rate results (%).

	Preceding utterances	Test 1	Test 2	Test 3
Utterance-level ASR	-	11.5	8.8	10.8
Large-context ASR	Hypotheses	**10.7**	**8.1**	**10.0**
Large-context ASR	Oracle texts	10.6	8.0	9.8

4.2. Results

First, we evaluated whether or not the long-range contexts can improve performance in correctly predicting transcriptions using character-level perplexity, which is a measurement of language models. Table 2 shows the character-level perplexity results obtained with utterance-level end-to-end ASR and large-context end-to-end ASR, both with and without a speech encoder. The results show that the large-context end-to-end ASR without the speech encoder outperformed the utterance-level end-to-end ASR without the speech encoder. This indicates that large-context linguistic information improves performance in correctly predicting transcriptions. The large-context end-to-end ASR with the speech encoder also outperformed the utterance-level end-to-end ASR with the speech encoder. This confirms that the long-range contexts are also effective in improving the end-to-end ASR performance. Next we evaluated ASR performance in terms of character error rate. Table 3 shows the experimental results obtained for both utterance-level end-to-end ASR and large context end-to-end ASR. We also evaluated the large-context ASR using oracle texts of preceding utterances to reveal whether or not recognition errors of the preceding utterances affect the ASR performance. The results show that the large-context end-to-end ASR yielded significant ASR performance improvements compared with the utterance-level end-to-end ASR. This confirms that the long-range contexts were an effective way to improve ASR performance. Actually, a slight performance improvement was obtained by using the oracle texts of the preceding utterances. This indicates that the large-context end-to-end ASR was slightly affected by recognition errors of the preceding utterances.

5. CONCLUSIONS

This paper proposed large-context end-to-end automatic speech recognition (ASR) methods that can consider long-range sequential context information beyond utterance boundaries in an end-to-end manner. The proposed method is modeled by combining attention-based encoder-decoder models with hierarchical recurrent encoder-decoder models. This achieves to utilize not only a target utterance's speech information but also all preceding transcribed text information for estimating a generative probability of a target utterance's text. Experimental results showed the proposed method is effective in improving ASR performance of a series of utterances compared with conventional utterance-level end-to-end ASR methods.

6. REFERENCES

[1] Hasim Sak, Andrew Senior, Kanishka Rao, and Francoise Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1468–1472, 2015.

[2] Geoffrey Zweig, Chengzhu Yu, Jasha Droppo, and Andreas Stolcke, "Advances in all-neural speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4805–4809, 2017.

[3] Kartik Audhkhasi, Bhuvana Ramabhadran, George Saon, Michael Picheny, and David Nahamoo, "Direct acoustics-to-word models for English conversational speech recognition," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 959–963, 2017.

[4] Hagen Soltau, Hank Liao, and Hasim Sak, "Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 3707–3711, 2017.

[5] Kartik Audhkhasi, Brian Kingsbury, Bhuvana Ramabhadran, George Saon, and Michael Picheny, "Building competitive direct acoustics-to-word models for English conversational speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4759–4763, 2018.

[6] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio, "End-to-end attention-based large vocabulary speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4945–4949, 2015.

[7] Liang Lu, Xingxing Zhang, Kyunghyun Cho, and Steve Renals, "A study of the recurrent neural network encoder-decoder for large vocabulary speech recognition," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 3249–3253, 2015.

[8] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4960–4964, 2016.

[9] Liang Lu, Xingxing Zhang, and Steve Renals, "On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5060–5064, 2016.

[10] Chung-Cheng Chiu, Tara N. Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J. Weiss, Kanishka Rao, Ekaterina Gonina, Navdeep Jaitly, Bo Li, Jan Chorowski, and Michiel Bacchiani, "State-of-the-art speech recognition with sequence-to-sequence models," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4774–4778, 2018.

[11] Albert Zeyer, Kazuki Irie, Ralf Schluter, and Hermann Ney, "Improved training of end-to-end attention models for speech recognition," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 7–11, 2018.

[12] Kanishka Rao, Hasim Sak, and Rohit Prabhavalkar, "Exploring architectures, data and units for streaming end-to-end speech recognition with RNN-transducer," *In Proc. Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 193–199, 2017.

[13] Hasim Sak, Matt Shannon, Kanishka Rao, and Francoise Beaufays, "Recurrent neural aligner: An encoder-decoder neural network model for sequence to sequence mapping," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1298–1302, 2017.

[14] Rui Lin, Shujie Liu, Muyun Yang, Mu Li, Ming Zhou, and Sheng Li, "Hierarchical recurrent neural network for document modeling," *In Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 899–907, 2015.

[15] Tian Wang and Kyunghyun Cho, "Larger-context language modelling with recurrent neural network," *In Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1319–1329, 2016.

[16] Bing Liu and Ian Lane, "Dialogue context language modeling with recurrent neural networks," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5715–5719, 2017.

[17] Ryo Masumura, Tomohiro Tanaka, Atsushi Ando, Hirokazu Masataki, and Yushi Aono, "Role play dialogue aware language models based on conditional hierarchical recurrent encoder-decoder," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1259–1263, 2018.

[18] Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie, "A hierarchical recurrent encoder-decoder for generative context-aware query suggestion," *In Proc. ACM International on Conference on Information and Knowledge Management (CIKM)*, pp. 553–562, 2015.

[19] Iulian V. Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," *In Proc. AAAI Conference on Artificial Intelligence (AAAI)*, pp. 3776–3783, 2016.

[20] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio, "A hierarchical latent variable encoder-decoder model for generating dialogues," *In Proc. AAAI Conference on Artificial Intelligence (AAAI)*, pp. 3295–3301, 2017.

[21] Marc Delcroix, Shinji Watanabe, Atsunori Ogawa, Shigeki Karita, and Tomohiro Nakatani, "Auxiliary feature based adaptation of end-to-end asr systems," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2444–2448, 2018.

[22] Shinji Watanabe, Takaaki Hori, and John R. Hershey, "Language independent end-to-end architecture for joint language identification and speech recognition," *In Proc. Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 265–271, 2017.

[23] Bo Li, Tara N. Sainath, Khe Chai Sim, Michiel Bacchiani, Eugene Weinstein, Patrick Nguyen, Zhifeng Chen, Yonghui Wu, and Kanishka Rao, "Multi-dialect speech recognition with a single sequence-to-sequence model," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4749–4753, 2018.

[24] Sebastien Jean, Stanislas Lauly, Orhan Firat, and Kyunghyun Cho, "Does neural machine translation benefit from larger context?," *arXiv preprint arXiv:1704.05135*, 2017.

[25] Longyue Wang, Zhaopeng Tu, Andy Way, and Qun Liu, "Exploiting cross-sentence context for neural machine translation," *In Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2826–2831, 2017.

[26] Sameen Maruf and Gholamreza Haffari, "Document context neural machine translation with memory networks," *In Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1275–1284, 2018.

[27] Kikuo Maekawa, Hanae Koiso, Sadaoki Furui, and Hitoshi Isahara, "Spontaneous speech corpus of Japanese," *In Proc. International Conference on Language Resources and Evaluation (LREC)*, pp. 947–952, 2000.

[28] Takaaki Hori, Shinji Watanabe, Yu Zhang, and William Chan, "Advances in joint CTC-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 949–953, 2017.

[29] Tomas Mikolov, Martin Karafiat, Lukas Burget, Jan Cernocky, and Sanjeev Khudanpur, "Recurrent neural network based language model," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1045–1048, 2010.

[30] Martin Sundermeyer, Hermann Ney, and Ralf Schluter, "From feedforward to recurrent LSTM neural networks for language models," *IEEE/ACM Transactions of Audio, Speech and Language processing*, vol. 23, no. 3, pp. 517–529, 2015.

[31] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning, "Effective approaches to attention-based neural machine translation," *In Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1412–1421, 2015.