INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

End-to-End Automatic Speech Recognition with a Reconstruction Criterion Using Speech-to-Text and Text-to-Speech Encoder-Decoders

Ryo Masumura, Hiroshi Sato, Tomohiro Tanaka, Takafumi Moriya, Yusuke Ijima, Takanobu Oba

NTT Media Intelligence Laboratories, NTT Corporation, Japan

ryou.masumura.ba@hco.ntt.co.jp

Abstract

In this paper, we present a novel end-to-end automatic speech recognition (ASR) method that considers whether an input speech can be reconstructed from a generated text or not. A speech-to-text encoder-decoder model is one of the most powerful end-to-end ASR methods since it does not make any conditional independence assumptions. However, encoder-decoder models often suffer from a problem that is caused from a gap between the teacher forcing in a training phase and the free running in a testing phase. In fact, there is no guarantee that texts can be generated correctly when some generation errors occur in conditioning contexts. In order to mitigate this problem, our proposed method utilizes not only a generation probability of the text computed from a speech-to-text encoder-decoder but also a reconstruction probability of the speech computed from a text-to-speech encoder-decoder on the basis of a maximum mutual information criterion. We can expect that considering the reconstruction criterion can impose a constraint against generation errors. In addition, in order to compute the reconstruction probability, we introduce a mixture density network into the text-to-speech encoder-decoder. Our experiments on Japanese lecture ASR tasks demonstrate that considering the reconstruction criterion can yield ASR performance improvements.

Index Terms: end-to-end speech recognition, speech-to-text encoder-decoder, text-to-speech encoder-decoder, reconstruction criterion, maximum mutual criterion

1. Introduction

In the automatic speech recognition (ASR) field, there has been growing interest in achieving end-to-end ASR systems that directly convert an input speech into a text. While traditional ASR systems have been built from noisy channel formulation using several component models (i.e., an acoustic model, a language model, and a pronunciation model), the end-to-end ASR systems can learn the overall conversion in one step without any intermediate processing.

For achieving the end-to-end ASR systems, several modeling methods including connectionist temporal classification [1, 2], recurrent neural aligner [3], recurrent neural network (RNN) transducer [4], and encoder-decoder [5–8] have been examined in recent studies. Among them, we mainly focus on an encoder-decoder that is an auto-regressive generative model conditioned on an input speech. A main strength of the encoder-decoder is to be expected to achieve total optimization of overall ASR systems since it does not make any conditional independence assumptions. In addition, the encoder-decoder can automatically align the input speech with the text by introducing an attention mechanism.

While the encoder-decoder can provide powerful ASR performance, it often suffers from the exposure bias problem [9] that is caused from a gap between a training phase and a testing phase. In the training phase, the encoder-decoder is trained with teacher forcing in which ground-truth texts are used as conditioning contexts. On the other hand, in the testing phase, texts are generated via free running in which generated tokens are recursively fed as an input at the next steps. Therefore, the conditioning contexts encountered during the training phase do not match those encountered at the testing phase. In other words, there is no guarantee that texts are generated correctly when some generation errors are compounded in the conditioning contexts. The main technique to mitigate the exposure bias problem is to incorporate noisy conditioning contexts into the training phase [10–12]. However, it cannot completely take away the gap between the training phase and the testing phase.

In order to robustly generate texts while suffering from the exposure bias problem, our key idea is to consider whether an input speech can be reconstructed from a generated text or not. It can be considered that it is difficult to generate an original speech from a text with some generation errors while it is comparatively easy to generate the original speech from a ground-truth text. We can expect that considering the reconstruction criterion can impose a constraint against the generation errors that occur due to the exposure bias problem.

In this paper, we propose a novel end-to-end ASR method that utilizes both a generation probability of a text computed from a speech-to-text encoder-decoder and a reconstruction probability of a speech computed from a text-to-speech encoder-decoder. To the best of our knowledge, this paper is the first study that considers the reconstruction criterion into end-to-end ASR although a similar idea was introduced in response generation, machine translation and video caption generation methods [13–15]. In the proposed method, the generation probability and the reconstruction probability are simultaneously taken into consideration on a basis of a maximum mutual information (MMI) criterion. In previous studies, the MMI criterion was introduced into discriminative training of acoustic models [16, 17]. On the other hand, we directly use the MMI between the input speech and the output text for a scoring function for ASR (see Section 3.1). In order to compute the reconstruction probability while considering speech variability, this paper introduces a mixture density network that outputs parameters of a Gaussian mixture model [18, 19] while typical speech-to-text encoder-decoders were modeled as a regression model [20, 21]. We can expect that the mixture density network is suitable for one-to-many mapping problems, i.e., text-to-speech mapping. In our experiments using Japanese lecture ASR tasks, we demonstrate that considering a reconstruction criterion can yield ASR performance improvements.

2. Related Work

The proposed method is closely related to a criterion for the end-to-end ASR. Although most end-to-end ASR systems are

http://dx.doi.org/10.21437/Interspeech.2019-2111

trained with a maximum likelihood criterion, they are not optimal for ASR since a sequence-level evaluation metric for ASR is word error rate (WER) or character error rate (CER). In order to minimize the expected WER or CER, minimum word error rate training [22] and minimum risk training using policy gradient or reinforce learning [23–25] were proposed. In addition, a heuristic criterion that uses log-linear interpolation of a generation probability computed from an end-to-end ASR model and that computed from an external language model (called shallow fusion) was also examined [26, 27]. In contrast to these approaches, the proposed method uses an MMI criterion between an input speech and a generated text for end-to-end ASR.

The proposed method is also related to methods that improve end-to-end ASR using text-to-speech [28–30]. In the previous studies, text-to-speech models were used for data augmentation so as to leverage unlabeled texts for improving the end-to-end ASR based on speech chain modeling [28] and back-translation [29]. Unlike them, the proposed method uses the text-to-speech encoder-decoder to impose a constraint against generation errors. In addition, we introduce a mixture density network into the text-to-speech encoder-decoder while previous studies introduced a regression model that cannot compute a generation probability of a speech.

3. End-to-End Automatic Speech Recognition with a Reconstruction Criterion

This section details an end-to-end automatic speech recognition (ASR) method that considers whether an input speech can be reconstructed from a generated text or not.

3.1. Definition

The end-to-end ASR is a problem that directly converts a speech $\boldsymbol{X} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_M\}$ into a text $\boldsymbol{W} = \{w_1, \cdots, w_N\}$ where w_n is the n-th token in the text and $\boldsymbol{x}_m$ is the m-th acoustic feature in the input speech. N is the number of tokens in the text and M is the number of acoustic features in the speech. We introduce a maximum mutual information (MMI) between the input speech and the output text as a scoring function of the end-to-end ASR while a maximum likelihood criterion is usually used as the objective function for typical end-to-end ASR systems. In this case, the end-to-end ASR is formulated as

$$\begin{aligned}\hat{\boldsymbol{W}} &= \operatorname*{argmax}_{\boldsymbol{W}} \log \frac{P(\boldsymbol{X}, \boldsymbol{W})}{P(\boldsymbol{X})P(\boldsymbol{W})^{\lambda}} \\ &= \operatorname*{argmax}_{\boldsymbol{W}} \log P(\boldsymbol{W}|\boldsymbol{X}) - \lambda \log P(\boldsymbol{W}) \\ &= \operatorname*{argmax}_{\boldsymbol{W}} \log P(\boldsymbol{W}|\boldsymbol{X}) - \lambda\{\log P(\boldsymbol{W}|\boldsymbol{X}) \\ &\qquad + \log P(\boldsymbol{X}) - \log P(\boldsymbol{X}|\boldsymbol{W})\} \\ &= \operatorname*{argmax}_{\boldsymbol{W}} (1-\lambda) \log P(\boldsymbol{W}|\boldsymbol{X}) + \lambda \log P(\boldsymbol{X}|\boldsymbol{W}),\end{aligned} \tag{1}$$

where $P(\boldsymbol{W}|\boldsymbol{X})$ is a generation probability of the output text that can be computed from a speech-to-text encoder-decoder and $P(\boldsymbol{X}|\boldsymbol{W})$ is a generation (reconstruction) probability of the input speech that can be computed from a text-to-speech encoder-decoder. λ is a hyper parameter that controls influence of the reconstruction probability. In fact, this formulation exactly matches the maximum likelihood criterion when λ is set to 0. Figure 1 shows a relationship between the speech-to-text encoder and the text-to-speech encoder-decoder.

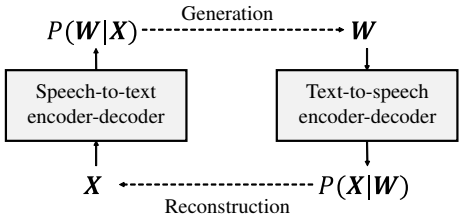

Figure 1: *Relationship between a speech-to-text encoder and a text-to-speech encoder-decoder.*

3.2. Speech-to-Text Encoder-Decoder

A speech-to-text encoder-decoder defines a generation probability of a text $\boldsymbol{W}$ given a speech $\boldsymbol{X}$. The generation probability is defined as

$$P(\boldsymbol{W}|\boldsymbol{X}; \boldsymbol{\Theta}_{\mathtt{stt}}) = \prod_{n=1}^{N} P(w_n|w_1, \cdots, w_{n-1}, \boldsymbol{X}; \boldsymbol{\Theta}_{\mathtt{stt}}), \tag{2}$$

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

Speech encoder: In a speech encoder, the acoustic features are converted into a hidden vector sequence. The hidden vector sequence $\boldsymbol{H} = \{\boldsymbol{h}_1, \cdots, \boldsymbol{h}_M\}$ is produced by

$$\boldsymbol{H} = \mathtt{BiRecurrent}(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_M; \boldsymbol{\theta}^{\mathtt{h}}_{\mathtt{stt}}), \tag{3}$$

where $\mathtt{BiRecurrent}()$ is the bidirectional RNNs and $\boldsymbol{\theta}^{\mathtt{h}}_{\mathtt{stt}}$ is the trainable parameter.

Text decoder: In a text decoder, which corresponds to an auto-regressive generative model, each token in conditioning contexts is first converted into a continuous vector as

$$\boldsymbol{w}_{n-1} = \mathtt{Embed}(w_{n-1}; \boldsymbol{\theta}^{\mathtt{w}}_{\mathtt{stt}}), \tag{4}$$

where $\mathtt{Embed}()$ is a function to convert a token into a continuous vector and $\boldsymbol{\theta}^{\mathtt{w}}_{\mathtt{stt}}$ is a trainable parameter. The continuous vector that summarizes from the initial token to the $n-1$-th token is defined as

$$\begin{aligned}\boldsymbol{v}_n &= \mathtt{Recurrent}(\boldsymbol{w}_1, \cdots, \boldsymbol{w}_{n-1}; \boldsymbol{\theta}^{\mathtt{v}}_{\mathtt{stt}}) \\ &= \mathtt{Recurrent}(\boldsymbol{w}_{n-1}, \boldsymbol{v}_{n-1}; \boldsymbol{\theta}^{\mathtt{v}}_{\mathtt{stt}}),\end{aligned} \tag{5}$$

where $\boldsymbol{\theta}^{\mathtt{v}}_{\mathtt{stt}}$ is the model parameter. The continuous vector is used for summarizing hidden speech vectors as a continuous vector. The continuous vector for generating the n-th token is calculated as

$$\boldsymbol{d}_n = \sum_{m=1}^{M} \frac{\exp \mathtt{Attend}(\boldsymbol{h}_m, \boldsymbol{v}_n; \boldsymbol{\theta}^{\mathtt{d}}_{\mathtt{stt}})}{\sum_{m'=1}^{M} \exp \mathtt{Attend}(\boldsymbol{h}_{m'}, \boldsymbol{v}_n; \boldsymbol{\theta}^{\mathtt{d}}_{\mathtt{stt}})} \boldsymbol{h}_m, \tag{6}$$

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

$$\boldsymbol{s}_n = \mathtt{NonLinear}([\boldsymbol{v}_n{}^{\top}, \boldsymbol{d}_n{}^{\top}]^{\top}; \boldsymbol{\theta}^{\mathtt{s}}_{\mathtt{stt}}), \tag{7}$$

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

$$P(w_n|w_1, \cdots, w_{n-1}, \boldsymbol{X}, \boldsymbol{\Theta}) = \mathtt{SOFTMAX}(\boldsymbol{s}_n; \boldsymbol{\theta}^{\mathtt{y}}_{\mathtt{stt}}), \tag{8}$$

where $\mathtt{SOFTMAX}()$ is a softmax transformational function and $\boldsymbol{\theta}^{\mathtt{y}}_{\mathtt{stt}}$ is the trainable parameter.

3.3. Text-to-Speech Encoder-Decoder

A text-to-speech encoder-decoder defines a generation probability of a speech $\boldsymbol{X}$ given a text $\boldsymbol{W}$. The generation probability is defined as

$$P(\boldsymbol{X}|\boldsymbol{W};\boldsymbol{\Theta}_{\texttt{tts}}) = \prod_{m=1}^{M} P(\boldsymbol{x}_m|\boldsymbol{x}_1,\cdots,\boldsymbol{x}_{m-1},\boldsymbol{W};\boldsymbol{\Theta}_{\texttt{tts}}), \tag{9}$$

where $\boldsymbol{\Theta}_{\texttt{tts}}$ represents the model parameter sets. In order to flexibly capture variability of speech, $P(\boldsymbol{x}_m|\boldsymbol{x}_1,\cdots,\boldsymbol{x}_{m-1},\boldsymbol{W};\boldsymbol{\Theta}_{\texttt{tts}})$ is modeled as a mixture density network that estimates parameters of a Gaussian mixture model (GMM). It is defined as

$$P(\boldsymbol{x}_m|\boldsymbol{x}_1,\cdots,\boldsymbol{x}_{m-1},\boldsymbol{W};\boldsymbol{\Theta}_{\texttt{tts}}) = \sum_{j=1}^{J} \alpha_{m,j}\mathcal{N}(\boldsymbol{x}_m;\boldsymbol{\mu}_{m,j},\boldsymbol{\sigma}_{m,j}^2), \tag{10}$$

where $\mathcal{N}()$ represents a Gaussian distribution, $\boldsymbol{\mu}_{m,j}$ is the j-th mean vector, $\boldsymbol{\sigma}_{m,j}^2$ is the j-th diagonal variance vector, and $\alpha_{m,j}$ is the j-th mixture weight for generating the m-th acoustic feature $\boldsymbol{x}_m$. J represents the number of mixtures of the GMM. The parameters are estimated using a text encoder and a speech decoder, both of which are composed of neural networks.

Text encoder: In a text encoder, each token is individually converted into a continuous vector. The continuous vector for the n-th token is represented as

$$\boldsymbol{w}_n = \texttt{Embed}(w_n;\boldsymbol{\theta}^{\texttt{w}}_{\texttt{tts}}), \tag{11}$$

where $\boldsymbol{\theta}^{\texttt{w}}_{\texttt{tts}}$ is the trainable parameter. In addition, the continuous vectors for the all tokens are converted into hidden representations using bidirectional RNN. The hidden vector sequence $\boldsymbol{C} = \{\boldsymbol{c}_1,\cdots,\boldsymbol{c}_N\}$ is produced by

$$\boldsymbol{C} = \texttt{BiRecurrent}(\boldsymbol{w}_1,\cdots,\boldsymbol{w}_N;\boldsymbol{\theta}^{\texttt{c}}_{\texttt{tts}}), \tag{12}$$

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

Speech decoder: In a speech decoder, which corresponds to an auto-regressive generative model, each acoustic feature in conditioning contexts is first converted into a continuous vector by

$$\boldsymbol{u}_{m-1} = \texttt{NonLinear}(\boldsymbol{x}_{m-1};\boldsymbol{\theta}^{\texttt{u}}_{\texttt{tts}}), \tag{13}$$

where $\boldsymbol{\theta}^{\texttt{u}}_{\texttt{tts}}$ is the trainable parameter. Next, we summarize acoustic features from $\boldsymbol{x}_1$ to $\boldsymbol{x}_{m-1}$ as a continuous vector. The continuous vector is defined as

$$\begin{aligned}\boldsymbol{z}_m &= \texttt{Recurrent}(\boldsymbol{u}_1,\cdots,\boldsymbol{u}_{m-1};\boldsymbol{\theta}^{\texttt{z}}_{\texttt{tts}}) \\ &= \texttt{Recurrent}(\boldsymbol{u}_{m-1},\boldsymbol{z}_{m-1};\boldsymbol{\theta}^{\texttt{z}}_{\texttt{tts}}),\end{aligned} \tag{14}$$

where $\boldsymbol{\theta}^{\texttt{z}}_{\texttt{tts}}$ is the model parameter. The continuous vector is used for summarizing hidden text vectors as a continuous vector. The m-th continuous vector is calculated as

$$\boldsymbol{r}_m = \sum_{n=1}^{N} \frac{\exp \texttt{Attend}(\boldsymbol{c}_n,\boldsymbol{z}_m;\boldsymbol{\theta}^{\texttt{r}}_{\texttt{tts}})}{\sum_{n'=1}^{N}\exp \texttt{Attend}(\boldsymbol{c}_{n'},\boldsymbol{z}_m;\boldsymbol{\theta}^{\texttt{r}}_{\texttt{tts}})}\boldsymbol{c}_n, \tag{15}$$

where $\boldsymbol{\theta}^{\texttt{r}}_{\texttt{tts}}$ is the trainable parameter. A context vector for estimating the m-th acoustic feature is produced by

$$\boldsymbol{q}_m = \texttt{NonLinear}([\boldsymbol{z}_m{}^\top,\boldsymbol{r}_m{}^\top]^\top;\boldsymbol{\theta}^{\texttt{q}}_{\texttt{tts}}), \tag{16}$$

where $\boldsymbol{\theta}^{\texttt{q}}_{\texttt{tts}}$ is the trainable parameter. In an output layer, model parameters of the GMM for generating $\boldsymbol{x}_m$ are estimated by

$$\begin{aligned}&[\boldsymbol{o}^{\mu}_{m,1}{}^\top,\cdots,\boldsymbol{o}^{\mu}_{m,J}{}^\top, \\ &\quad \boldsymbol{o}^{\sigma}_{m,1}{}^\top,\cdots,\boldsymbol{o}^{\sigma}_{m,J}{}^\top,o^{\alpha}_{m,1},\cdots,o^{\alpha}_{m,J}]^\top \\ &\qquad = \texttt{Linear}(\boldsymbol{q}_m;\boldsymbol{\theta}^{\texttt{o}}_{\texttt{tts}}),\end{aligned} \tag{17}$$

$$\boldsymbol{\mu}_{m,j} = \boldsymbol{o}^{\mu}_{m,j}, \tag{18}$$

$$\boldsymbol{\sigma}_{m,j} = \exp(\boldsymbol{o}^{\sigma}_{m,j}), \tag{19}$$

$$\alpha_{m,j} = \frac{\exp(o^{\alpha}_{m,j})}{\sum_{j'=1}^{J}\exp(o^{\alpha}_{m,j'})}, \tag{20}$$

where $\texttt{Linear}()$ is a linear transformational function and $\boldsymbol{\theta}^{\texttt{o}}_{\texttt{tts}}$ is the trainable parameter.

3.4. Training

Both the speech-to-text and the text-to-speech encoder-decoders are trained from the utterance-level training data set $\mathcal{D} = \{(\boldsymbol{X}_1,\boldsymbol{W}_1),\cdots,(\boldsymbol{X}_T,\boldsymbol{W}_T)\}$. The model parameter sets for the speech-to-text encoder-decoder and the text-to-speech encoder-decoder are summarized as

$$\boldsymbol{\Theta}_{\texttt{stt}} = \{\boldsymbol{\theta}^{\texttt{h}}_{\texttt{stt}},\boldsymbol{\theta}^{\texttt{w}}_{\texttt{stt}},\boldsymbol{\theta}^{\texttt{v}}_{\texttt{stt}},\boldsymbol{\theta}^{\texttt{d}}_{\texttt{stt}},\boldsymbol{\theta}^{\texttt{s}}_{\texttt{stt}},\boldsymbol{\theta}^{\texttt{y}}_{\texttt{stt}}\}, \tag{21}$$

$$\boldsymbol{\Theta}_{\texttt{tts}} = \{\boldsymbol{\theta}^{\texttt{w}}_{\texttt{tts}},\boldsymbol{\theta}^{\texttt{c}}_{\texttt{tts}},\boldsymbol{\theta}^{\texttt{u}}_{\texttt{tts}},\boldsymbol{\theta}^{\texttt{z}}_{\texttt{tts}},\boldsymbol{\theta}^{\texttt{r}}_{\texttt{tts}},\boldsymbol{\theta}^{\texttt{q}}_{\texttt{tts}},\boldsymbol{\theta}^{\texttt{o}}_{\texttt{tts}}\}. \tag{22}$$

Since both parameters are completely independent, the parameter sets are individually optimized by

$$\hat{\boldsymbol{\Theta}}_{\texttt{stt}} = \mathop{\text{argmin}}_{\boldsymbol{\Theta}_{\texttt{stt}}} -\sum_{t=1}^{T}\log P(\boldsymbol{W}_t|\boldsymbol{X}_t;\boldsymbol{\Theta}_{\texttt{stt}}), \tag{23}$$

$$\hat{\boldsymbol{\Theta}}_{\texttt{tts}} = \mathop{\text{argmin}}_{\boldsymbol{\Theta}_{\texttt{tts}}} -\sum_{t=1}^{T}\log P(\boldsymbol{X}_t|\boldsymbol{W}_t;\boldsymbol{\Theta}_{\texttt{tts}}). \tag{24}$$

These optimizations are conducted using mini-batch stochastic gradient descent.

3.5. Testing

In the testing phase, the proposed method first generates n-best lists using the speech-to-text encoder-decoder and then rescores them using the text-to-speech encoder-decoder since the reconstruction probability cannot be calculated until the text has been generated. We denote an n-best hypotheses computed using a beam-search decoding for the input utterance $\boldsymbol{X}$ as $\boldsymbol{\Omega}(\boldsymbol{X},L)$ where L represents the number of hypotheses. In this case, the end-to-end ASR problem is defined as

$$\begin{aligned}\hat{\boldsymbol{W}} = \mathop{\text{argmax}}_{\boldsymbol{W}\in\boldsymbol{\Omega}(\boldsymbol{X},L)}\ &(1-\lambda)\log P(\boldsymbol{W}|\boldsymbol{X};\boldsymbol{\Theta}_{\texttt{stt}}) \\ &+\lambda\log P(\boldsymbol{X}|\boldsymbol{W};\boldsymbol{\Theta}_{\texttt{tts}}),\end{aligned} \tag{25}$$

where $P(\boldsymbol{W}|\boldsymbol{X};\boldsymbol{\Theta}_{\texttt{stt}})$ is computed in a first decoding step and $P(\boldsymbol{X}|\boldsymbol{W};\boldsymbol{\Theta}_{\texttt{tts}})$ is computed from a second re-ranking step.

4. Experiments

In experiments, we used the Corpus of Spontaneous Japanese (CSJ) [31]. 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 lecture was segmented into utterances. This paper used characters as the tokens. Details of the data sets are shown in Table 1.

4.1. Setups

For experiments, we constructed a speech-to-text encoder-decoder and four text-to-speech encoder-decoders.

Speech-to-text encoder-decoder: In the speech encoder, we used 40 dimensional 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 text-decoder, a token embedding layer with 512 units and a unidirectional LSTM-RNN with 512 units were stacked. For the attention mechanism, we used global attention [32]. The output unit size, which corresponds to the number of characters in the training set, was set to 3,084. For training them, we used a mini-batch stochastic gradient descent with gradient norm clipping 1.0. In each LSTM-RNN, we used variational dropout [33] where its rate was set to 0.2 for the speech encoder and 0.4 for the text decoder. The validation data sets were used for early stopping.

Text-to-speech encoder-decoder: In the text encoder, we stacked a token embedding layer with 512 units and 3-layer bidirectional LSTM-RNN with 512 units. In the speech decoder, 3 consecutive acoustic features were handled as one representation. We stacked a sigmoid non-linear layer and a unidirectional LSTM-RNN with 512 units. For the attention mechanism, we also used the global attention. The output unit size was different with respect to the number of mixtures in the mixture density network. For example, the output unit size was 1,442 (720 for mean vectors, 720 for diagonal variance vectors, 2 for mixture weights) when the number of mixtures was set to 2. For training them, we used a mini-batch Adadelta optimizer 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 0.6 for the speech decoder. In addition, we used dropout where its rate was set to 0.6 for the input acoustic features in the speech decoder. The validation data sets were used for early stopping.

In ASR evaluations, we generated 50-best hypotheses using the constructed speech-to-text encoder-decoder with a beam-search decoding and rescored them using the constructed speech-to-text encoder-decoder and also any one of the constructed text-to-speech encoder-decoders. Other hyper parameters were tuned using the validation sets.

4.2. Results

Experimental results in terms of character error rate are shown in Table 2. J represents the number of mixtures for the mixture density network in Eq. (10), and λ represents a hyper parameter that controls the influence of a reconstruction probability in Eq. (1). "Baseline" represents results in which a generation probability computed from the speech-to-text encoder-decoder was only used by setting λ as 0.0. "Proposed" represents results that considered not only the generation probability computed from the speech-to-text encoder but also the reconstruction probability from the text-to-speech encoder. Note that the reconstruction probability was only used when λ was set to 1.0. "Oracle" represents upper bound results in which texts with the best WER were selected from the 50-best hypotheses.

The experimental results show that the proposed methods, which simultaneously use the generation probability and the reconstruction probability, outperformed the baseline method. This suggests that the reconstruction criterion is effective for improving the end-to-end ASR performance. On the other hand, the proposed methods, which only used the reconstruction probability by setting λ as 1.0, were inferior to the baseline method. This is because the generation probabilities of texts were not taken into consideration at all. In addition, small performance improvements were attained by increasing the number of mixtures in the mixture density network. This indicates that considering the speech variability is effective for robustly computing the reconstruction probability. The highest results were achieved by setting the number of mixtures to 4. Actually, insertion and deletion errors were decreased by considering the reconstruction criterion while substitution errors were increased since the proposed method cannot take account of homonyms. These results verified that considering the reconstruction criterion can efficiently impose a constraint against generation errors in the speech-to-text encoder-decoder.

Table 1: *Experimental data sets.*

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

Table 2: *Character error rate results.*

	J	λ	Test 1	Test 2	Test 3
Baseline	-	0.0	11.5	8.8	10.8
Proposed	1	0.3	11.1	8.4	10.3
Proposed	1	1.0	19.3	17.1	19.7
Proposed	2	0.3	11.0	8.4	10.2
Proposed	4	0.3	**10.9**	**8.3**	**10.1**
Proposed	8	0.3	11.1	8.4	10.3
Oracle	-	-	5.8	4.7	4.7

5. Conclusions

In this paper, we have proposed an end-to-end automatic speech recognition (ASR) method that considers whether an input speech can be reconstructed from a generated text or not. In the proposed method, we used not only a generation probability of an output text computed from a speech-to-text encoder-decoder but also a reconstruction probability of an input speech computed from a text-to-speech encoder-decoder as a scoring function on a basis of a maximum mutual information criterion. A main advantage of the reconstruction criterion is that it allows us to impose a constraint against generation errors that occur in the speech-to-text encoder. In experiments on Japanese lecture ASR tasks, we demonstrated that the reconstruction criterion can yield ASR performance improvements. In addition, we verified that it is effective to take speech variability into consideration in the text-to-speech encoder-decoder.

6. References

[1] G. Zweig, C. Yu, J. Droppo, and A. Stolcke, "Advances in all-neural speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4805–4809, 2017.

[2] K. Audhkhasi, B. Ramabhadran, G. Saon, M. Picheny, and D. 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.

[3] H. Sak, M. Shannon, K. Rao, and F. 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.

[4] K. Rao, H. Sak, and R. 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.

[5] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. 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.

[6] L. Lu, X. Zhang, K. Cho, and S. 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.

[7] W. Chan, N. Jaitly, Q. Le, and O. 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.

[8] L. Lu, X. Zhang, and S. 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.

[9] M. A. Ranzato, S. Chopra, M. Auli, and W. Zaremba, "Sequence level training with recurrent neural networks," *arXiv preprint arXiv:1511.06732*, 2015.

[10] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural network," *In Proc. Advances in Neural Information Processing Systems (NIPS)*, pp. 1171–1179, 2015.

[11] Z. Xie, S. I. Wang, J. Li, D. Levy, A. Nie, D. Jurafsky, and A. Y. Ng, "Data noising as smoothing in neural network language models," *In Proc. International Conference on Learning Representations (ICLR)*, 2017.

[12] K. Goyal, C. Dyer, and T. Berg-Kirkpatrick, "Differentiable scheduled sampling for credit assignment," *In Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 366–371, 2017.

[13] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A diversity-promoting objective function for neural conversation models," *In Proc. Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 110–119, 2016.

[14] Z. Tu, Y. Liu, L. Shang, X. Liu, and H. Li, "Neural machine translation with reconstruction," *In Proc. AAAI Conference on Artificial Intelligence (AAAI)*, pp. 3097–3103, 2017.

[15] B. Wang, L. Ma, W. Zhang, and W. Liu, "Reconstruction network for video captioning," *In Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7622–7631, 2018.

[16] L. R. Bahi, P. F. Brown, P. V. de Souza, and R. L. Mercer, "Maximum mutual information estimation of hidden markov model parameters for speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 49–52, 1986.

[17] P. C. Woodland and D. Povey, "Large scale discriminative training of hidden markov models for speech recognition," *Computer Speech and Language*, vol. 16, pp. 25–47, 2002.

[18] H. Zen and A. Senior, "Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 3872–3876, 2014.

[19] T. Capes, P. Coles, A. Conkie, L. Golipour, A. Hadjitarkhani, Q. Hu, N. Huddleston, M. Hunt, J. Li, M. Neeracher, K. Prahallad, T. Raitio, R. Rasipuram, G. Townsend, B. Williamson, D. Winarsky, Z. Wu, and H. Zhan, "Siri on-device deep learning-guided unit selection text-to-speech system," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 4011–4015, 2017.

[20] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 4006–4010, 2017.

[21] J. Shen, R. Pang, R. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. J. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4779–4783, 2018.

[22] R. Prabhavalkar, T. N. Sainath, Y. Wu, P. Nguyen, Z. Chen, C.-C. Chiu, and A. Kannan, "Minimum word error rate training for attention-based sequence-to-sequence models," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4839–4843, 2018.

[23] S. Shen, Y. Cheng, Z. He, W. He, H. Wu, M. Sun, and Y. Liu, "Minimum risk training for neural machine translation," *In Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1683–1692, 2016.

[24] S. Karita, A. Ogawa, M. Delcroix, and T. Nakatani, "Sequence training of encoder-decoder model using policy gradient for end-to-end speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5839–5843, 2018.

[25] A. Tjandra, S. Sakti, and S. Nakamura, "Sequence-to-sequence asr optimization via reinforcement learning," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5829–5833, 2018.

[26] J. Chorowski and N. Jaitly, "Towards better decoding and language model integration in sequence to sequence models," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 523–527, 2017.

[27] T. Hori, S. Watanabe, Y. Zhang, and W. 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.

[28] A. Tjandra, S. Sakti, and S. Nakamura, "Listening while speaking: Speech chain by deep learning," *In Proc. Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 301–308, 2017.

[29] T. Hayashi, S. Watanabe, Y. Zhang, T. Toda, T. Hori, R. Astudillo, and K. Takeda, "Back-translation-style data augmentation for end-to-end ASR," *In Proc. Spoken Language Technology Workshop (SLT)*, pp. 426–432, 2018.

[30] M. Mimura, S. Ueno, H. Inaguma, S. Sakai, and T. Kawahara, "Leveraging sequence-to-sequence speech synthesis for enhancing acoustic-to-word speech recognition," *In Proc. Spoken Language Technology Workshop (SLT)*, pp. 477–484, 2018.

[31] K. Maekawa, H. Koiso, S. Furui, and H. Isahara, "Spontaneous speech corpus of Japanese," *In Proc. International Conference on Language Resources and Evaluation (LREC)*, pp. 947–952, 2000.

[32] M.-T. Luong, H. Pham, and C. 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.

[33] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," *In Proc. International Conference on Neural Information Processing System (NIPS)*, pp. 1027–1035, 2016.