End-to-end Speech Translation with Transcoding by Multi-task Learning for Distant Language Pairs

Takatomo Kano, *Student Member, IEEE,* Sakriani Sakti, *Member, IEEE,* and Satoshi Nakamura, *Fellow, IEEE*

***Abstract*—Directly translating spoken utterances from a source language to a target language is challenging because it requires a fundamental transformation in both linguistic and para/non-linguistic features. Traditional speech-to-speech translation approaches concatenate automatic speech recognition (ASR), text-to-text machine translation (MT), and text-to-speech synthesizer (TTS) by text information. The current state-of-the-art models for ASR, MT, and TTS have mainly been built using deep neural networks, in particular, an attention-based encoder-decoder neural network with an attention mechanism. Recently, several works have constructed end-to-end direct speech-to-text translation by combining ASR and MT into a single model. However, the usefulness of these models has only been investigated on language pairs of similar syntax and word order (e.g., English-French or English-Spanish). For syntactically distant language pairs (e.g., English-Japanese), speech translation requires distant word reordering. Furthermore, parallel texts with corresponding speech utterances that are suitable for training end-to-end speech translation are generally unavailable. Collecting such corpora is usually time-consuming and expensive. This paper proposes the first attempt to build an end-to-end direct speech-to-text translation system on syntactically distant language pairs that suffer from long-distance reordering. We train the model on English (subject-verb-object (SVO) word order) and Japanese (SOV word order) language pairs. To guide the attention-based encoder-decoder model on this difficult problem, we construct end-to-end speech translation with transcoding and utilize curriculum learning (CL) strategies that gradually train the network for end-to-end speech translation tasks by adapting the decoder or encoder parts. We use TTS for data augmentation to generate corresponding speech utterances from the existing parallel text data. Our experiment results show that the proposed approach provides significant improvements compared with conventional cascade models and the direct speech translation approach that uses a single model without transcoding and CL strategies.**

***Index Terms*—End-to-end speech-to-text translation, automatic speech recognition, machine translation, multi-task learning.**

AS globalization continues to expand, language barriers remain notorious obstacles to free communication. Spoken language translation is one innovative technology that enables people to communicate among speakers of different languages. However, translating spoken language remains a very complicated task that involves recognizing and automatically translating speech in real time.

A traditional approach in speech-to-speech translation systems constructs ASR, MT, and TTS systems that are trained and tuned independently. Given speech input, ASR processes and transforms the speech into text in the source language, which MT then transforms into corresponding text in the target language. Finally, TTS converts the target language text into speech utterances [1]. The basic unit for information sharing among these components is the "text representation" of words. Even though significant progress has been made and various commercial speech translation systems have been introduced, this approach still suffers from several significant limitations.

One drawback is that over half of the world's languages are actually only spoken and have no written form. Thus, constructing speech translation that heavily relies on information sharing of the text representation of words is difficult. Another problem is that speech acoustics generally involve both linguistic and paralinguistic information (i.e., rhythm, emphasis, or emotion). Unfortunately, since such paralinguistic information is not a factor in written communication, much cannot be expressed in text. Consequently, the text output by ASR has lost all paralinguistic information; only the linguistic parts are translated by MT. Some studies have proposed the inclusion of additional components that just handle paralinguistic translation, but this step introduces more complexity and delay [2–4]. We need an architecture that can handle both linguistic and acoustic feature contents at once and generate a translation to other languages.

Deep learning has shown much promise in many tasks. An attention-based encoder-decoder neural network is a powerful model for ASR, MT, and TTS [5–7]. Several recent works have extended the task and constructed an end-to-end, direct speech-to-text translation system that combines ASR and MT tasks in a single model. Duong et al. introduced the first study that considered speech-to-text translation with deep-neural networks [8] and proposed alignment and translation reranking directly from source-language speech with target text translations. However, their work was only based on Spanish-English language pairs with similar syntax and word order, and the results failed to outperform the traditional cascade approach based only on a statistical word level MT (MOSES) [9]. Their proposed attention-based model achieved a BLEU score of 14.6%, where the MOSES baselines out-

Manuscript received June 06, 2019; revised October 10, 2019, February 30, 2019 and March 6, 2020; accepted March 22, 2020. Date of publication April 5, 2020; This work was supported by the JSPS KAKENHI Grant Numbers JP17H06101 and JP17K00237.

T. Kano is with the Division of Information Science, Nara Institute of Science and Technology, Ikoma 630-0192, Japan (e-mail: kano.takatomo.km04@is.naist.jp).

S. Sakti is with the Graduate School of Science and Technology, Nara Institute of Science and Technology, Ikoma 630-0192, Japan, and also with the RIKEN Center for Advanced Intelligence Project (ssakti@is.naist.jp).

S. Nakamura is with the Data Science Center and Graduate School of Science and Technology, Nara Institute of Science and Technology, Ikoma 630-0192, Japan, and also with the RIKEN Center for Advanced Intelligence Project (s-nakamura@is.naist.jp).

performed their proposed method on the BLEU scores in a range between 18.2 and 20.2. Bérard et al. built a full-fledged, end-to-end attention-based speech-to-text translation system [10]. It was the first work that used speech generated by TTS as a corpus for speech translation (ST) tasks to simplify the expansion of training data. However, this work was only compared with statistical text-based MT systems and performed on a small French-English synthetic corpus. These two languages share similar word order (SVO-SVO). For such languages, only local movements are sufficient for translation. Bérard et al. [11] further investigated speech-to-text translation with various units as character-, subword-, and word-based speech translation with a beam search, a greedy search, and an ensemble search that combined them and successfully improved the translation accuracy. Their studies concentrated on French-English translation using both synthesis and natural speech data. Weiss et al. focused on Spanish-English speech-to-text translation and proposed sharing the parameters of an ASR encoder to an ST encoder. Their study revealed that an encoder could transform speech into a consistent interlingual subword unit representation, which the respective decoders assembled into phrases in either language [12]. Bansal et al. then performed speech translation with natural speech from multi-speakers and used unsupervised term discovery on cluster repeated patterns in the audio to create pseudo-text instead of performing ASR [13]. Ultimately, all of these previous researches just focused on syntactically similar language pairs.

In this work, we propose the first attempt to build an end-to-end attention-based speech translation system on syntactically distant language pairs that suffer from long-distance reordering phenomena. We trained our attention-based encoder-decoder model on English-Japanese language pairs with SVO versus SOV word order and utilized TTS for data augmentation to generate the corresponding speech utterances from the existing parallel text data. To guide the attention-based encoder-decoder model to learn this difficult problem, we proposed transcoding based on a curriculum learning (CL) strategy. Unlike a conventional CL strategy that starts with easy data and gradually emphasizes difficult data examples, we formalized CL strategies that start the training with an end-to-end encoder-decoder for ASR or MT tasks and gradually trained the network for end-to-end speech translation tasks by adapting the decoder or encoder parts.

I. Related Works

Curriculum learning, which is a learning paradigm, was inspired by the learning processes of humans and animals that start by grasping easy aspects and gradually increasing to more difficult ones. Although the application of such training strategies to machine learning has been discussed between machine learning and cognitive science researchers as far back as Elman et al. [14], CL's first formulation in the context of machine learning was introduced by Bengio et al. (2009) [15].

Using CL might help avoid bad local minimums, hasten training convergence, and improve generalization. These advantages have been empirically demonstrated in various tasks, including shape recognition [15], object classification [16], and language modeling [17]. However, most studies focus on how to organize the sequence of the learning data examples in the context of single-task learning. Bengio et al. [15] proposed CL for multiple tasks. Again, all of these tasks still belong to the same type of problem (object classification) and share identical input and output spaces.

In speech translation tasks, the translation difficulty depends on the relationship between source and target sentences, such as word re-ordering, alignment, insertion, and deletion. Even if both the source and target are clean speech and short sentences, some cases are difficult to translate. For example, some sentences might include uncommon names or words with multiple definitions. These translation difficulties for ST systems are found after training.

In contrast to most previous CL studies,
(1) instead of utilizing the CL strategy for simple recognition/classification problems we use it for an attention-based encoder-decoder neural network learning problems in speech translation tasks;
(2) we train the model step by step from the easy task to the complicated task changing the model structures. We start training of an end-to-end encoder-decoder for ASR and MT task. Then we gradually extend to the ST task by respectively adapting the decoder or encoder parts; (3) in the original CL learning, the input and output spaces are kept the same even the training data become difficult. However in our CL learning, the model input and output space will be changed as we go to a more difficult task during the training steps.

II. Basic Attention-based Speech Translation

A. Attention-based encoder-decoder using RNN

We built an end-to-end speech translation system on a standard attention-based encoder-decoder neural network architecture using an RNN [6,18] that consists of encoder, decoder, and attention modules. Given input sequence $\mathbf{x} = [x_1, x_2, ..., x_N]$ with length N, the encoder produces a sequence of vector representation $h^{enc} = (h_1^{enc}, h_2^{enc}, ..., h_N^{enc})$. We used the following bidirectional recurrent neural network (BiRNN) with long short-term memory units (Bi-LSTM) [19] and a gated recurrent unit (GRU) [20] that consists of forward and backward recurrent neural networks (RNNs) [20] comprised of forward and backward recurrent neural networks (RNNs):

$$\begin{aligned} h_n^f &= LSTM(x_n) \\ h_n^b &= LSTM(x_{N-n}) \\ h_n^{enc} &= [h_n^f, h_n^b]. \end{aligned} \tag{1}$$

Here h^f denotes the forward RNN hidden states and h^b denotes the backward RNN hidden states. Thus, for each input x_n, we obtain h_n^{enc} by concatenating forward h_n^f and backward h_n^b. The decoder predicts target sequence $\mathbf{y} = [y_1, y_2, ..., y_T]$ with length T by estimating conditional probability $p(\mathbf{y}|\mathbf{x})$. We use a uni-directional GRU. Conditional probability $p(\mathbf{y}|\mathbf{x})$ is estimated based on the entire sequence of the previous output:

$$p(y_t|y_1, y_2, ..., y_{t-1}, x) = \textit{softmax}(W_y \tilde{o}_t^{dec}). \tag{2}$$

Decoder output vector o_t^{dec} is computed by applying linear layer W_o to context information c_t and current hidden state h_t^{dec}:

$$\begin{aligned} h_t^{dec} &= GRU(emb_{t-1}^{y}) \\ c_t &= attention(h^{enc}, h_t^{dec}) \\ o_t^{dec} &= W_o[c_t; h_t^{dec}]. \end{aligned} \tag{3}$$

Here c_t is the context information of the input sequence when generating the current output at time t, estimated by the attention module over encoder hidden states h_n^{enc}:

$$c_t = \sum_{n=1}^{N} a_t(n) * h_n^{enc}, \tag{4}$$

where variable-length alignment vector a_t is computed whose size equals the length of input sequence x:

$$\begin{aligned} a_t(n) &= align(h_n^{enc}, h_t^{dec}) \\ &= softmax(dot(h_n^{enc}, h_t^{dec})). \end{aligned} \tag{5}$$

This step helps the decoder find relevant information on the encoder side based on the current decoder hidden states. There are several variations for calculating align(h_n^{enc}, h_t^{dec}); we simply use the general attention between the encoder and decoder hidden states [21].

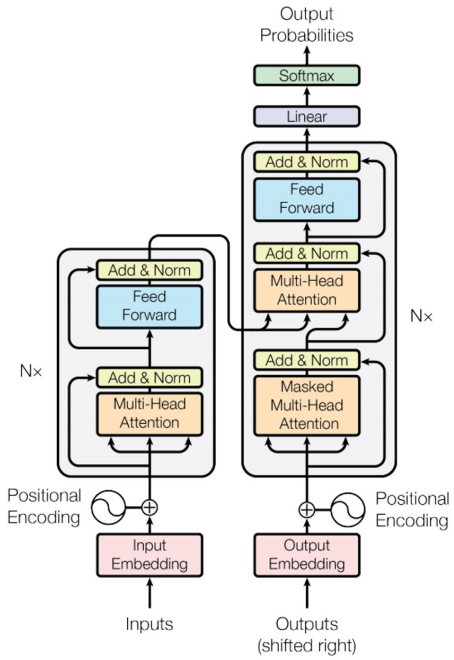

Fig. 1. Overview of Transformer architecture [22]

.

B. Attention-based encoder-decoder using Transformer

In performing RNN, since each step calculation needs to weight the previous step process, the model cannot compute the sequence data in parallel. Ashish et al. proposed an attention-based encoder-decoder transaction model without a recurrent mechanism called Transformer [22]. The encoder maps an input sequence of symbol representations $\mathbf{x} = [x_1, ..., x_N]$ to a sequence of continuous representations $\boldsymbol{h} = [h_1, ..., h_N]$ using a stacked feed-forward neural network (FNN). Given $\boldsymbol{h}$, the decoder generates output sequence $\boldsymbol{y} = [y_1, ..., y_T]$ of the symbols one element at a time. Transformer follows this overall architecture using a stacked, self-attention, point-wise FNN for both the encoder and decoder. The encoder is composed of a stack of multiple layers, each of which has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a position-wise FNN. Transformer has a residual connection around each of the two sub-layers, followed by layer normalization [23,24]. The decoder is also composed of multiple layers like the encoder. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the encoder stack's output. The attention function resembles mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. Fig. 1 illustrates the overall architecture of the Transformers. Basically, the Transformer model offers two benefits: (1) it models sequential data without recurrent connections. Therefore it does not need to wait to process previous results and can process the whole sequence at once; (2) self-attention provides an opportunity for injecting the global context of the whole sequence into each input frame to directly build long-range dependencies. When performing forward and backward processing for the current input and output, Transformer only makes a calculation graph path for relative states that are attended by a self-attention mechanism. But the RNN-based encoder-decoder model makes a calculation graph path for all the previous states because it models the global context with recurrent structures.

C. Basic attention-based speech translation

In this study, we apply this basic architecture to various tasks.

- ASR system:
  Input sequence $\mathbf{x} = [x_1, ..., x_N]$ is the input speech sequence of the source language, and target sequence $\mathbf{y} = [y_1, ..., y_T]$ is the predicted text of the corresponding transcription (Fig. 2). The RNN-based ASR system has multi-layer perceptron (MLP) with a rectified linear unit (ReLu) activation function as an input layer. The encoder part consists of a Bi-LSTM at the 1st layer and a unidirectional GRU at the 2nd layer. Both networks only use an even number of input vector sequences to reduce the memory and calculation time. For the decoder part, we use Luong's design decoder [21] with a GRU decoder. The Transformer-based ASR system has an MLP and a convolution network as an input layer. The encoder part consists of three fully connected (FC) layers and a self-attention function, and the decoder part

consists of six FC layers and a self-attention function, which are identical as the original Transformer encoder and decoder layers [22].

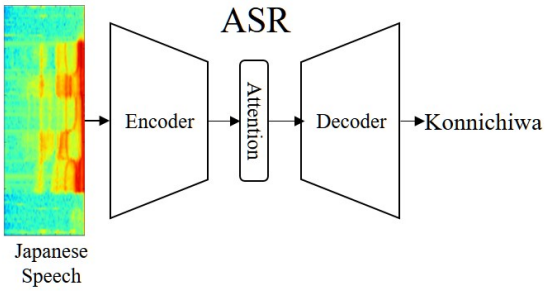

Fig. 2. ASR system

- Neural Machine Translation (NMT) system:
  Input sequence $\mathbf{x} = [x_1, ..., x_N]$ is the text sequence of the source language, and target sequence $\mathbf{y} = [y_1, ..., y_T]$ is the predicted text sequence in the target language (Fig. 3). The text can be represented by words, subwords, or character sequences. The RNN-based NMT encoder part consists of two Bi-LSTM layers. For the decoder part, we used Luong's design decoder [21] with a GRU decoder. The Transformer-based NMT encoder consists of three FC layers and a self-attention function, and the decoder part consists of six FC layers and a self-attention function, which are identical to the original Transformer encoder and decoder layers [22].

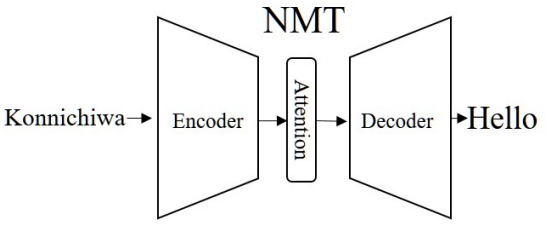

Fig. 3. NMT system

- Speech translation (ST) system:
  - Direct ST system
    Input sequence $\mathbf{x} = [x_1, ..., x_N]$ is the input speech sequence of the source language, and target sequence $\mathbf{y} = [y_1, ..., y_T]$ is the predicted corresponding text sequences (words, subwords, or character sequences) in the target language (Fig. 4). As a baseline, we trained the attention-based encoder-decoder neural network from an initial state. Each RNN and Transformer-based direct ST has identical architecture as each ASR model.

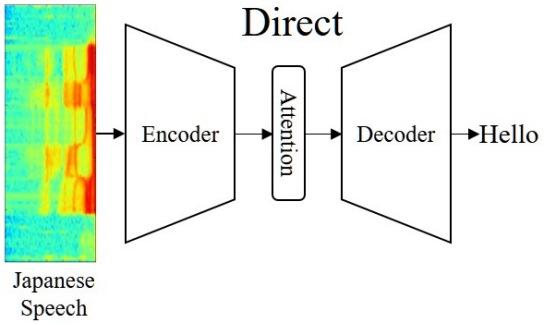

Fig. 4. Direct ST system

  - Cascade ST system:
    The Cascade ST system combines two systems: ASR and NMT (Fig. 5). The overall system has input sequence $\mathbf{x} = [x_1, ..., x_N]$, which is the input speech sequence of the source language and the target language sequence $\mathbf{y} = [y_1, ..., y_T]$. We did not perform joint training or adaptation for this model.

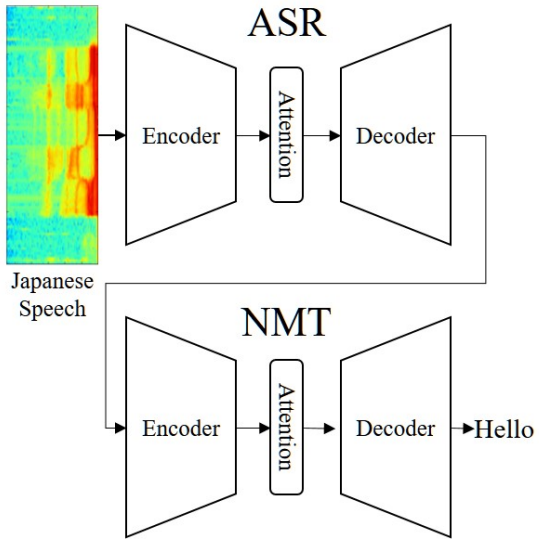

Fig. 5. Cascade ST system

  - ASRenc-NMTdec ST system:
    Input sequence $\mathbf{x} = [x_1, ..., x_N]$ is the input speech sequence of the source language, and target sequence $\mathbf{y} = [y_1, ..., y_T]$ is the predicted corresponding text (words, subwords, or character sequences) in the target language. Here the attention-based encoder-decoder neural network uses the pre-trained ASR encoder and NMT decoder parts, following previous works [11,12] (Fig. 6).

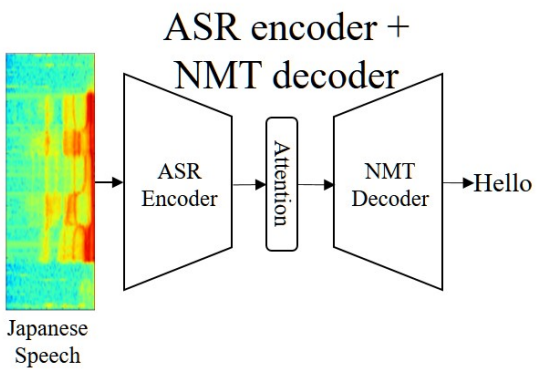

Fig. 6. ASRenc-NMTdec ST system

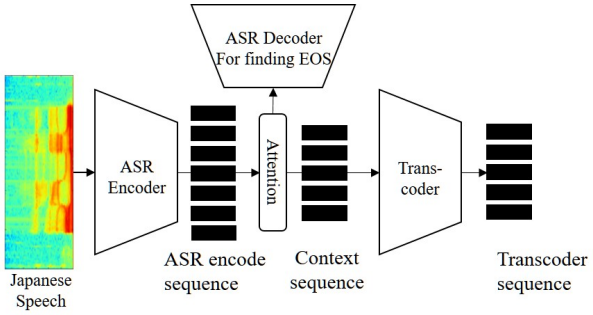

Fig. 7. Transcoding process

– Proposed CL-Transcoder ST system:
Input sequence $\mathbf{x} = [x_1, ..., x_N]$ is the input speech sequence of the source language, and target sequence $\mathbf{y} = [y_1, ..., y_T]$ is the predicted corresponding word sequence in the target language. For the ST system encoding and decoding parts, we pre-trained the ASR encoder and NMT decoder networks. To combine them smoothly to improve performance, we used the transcoder to transfer the context vector from the ASR encoder to the NMT decoder (Fig. 1). The transcoder architecture follows the NMT encoder architecture. If we build an RNN-based ST system, the transcoder should be a two-layer Bi-LSTM. If we build a Transformer-based ST system, the transcoder should be Transformer.

For the RNN-based ST and ASR encoder, since the input is speech features (continuous space), we also did downsampling during the encoding process to save memory resources [10]. The encoder's first layer uses the entire input sequence. The second layer only uses every other index, the third only uses every fourth index, and so forth. The decoder maps input embeddings to a hidden space using a unidirectional RNN network and applies attention to get a context vector. Finally, we concatenate the context and decoder hidden states and map this super vector to the target language dictionary space using a FC network [21]. Our end-to-end ST models also skip odd index states, as shown in Fig. 8. However, we do not apply this downsampling to the Transformer-based model, because

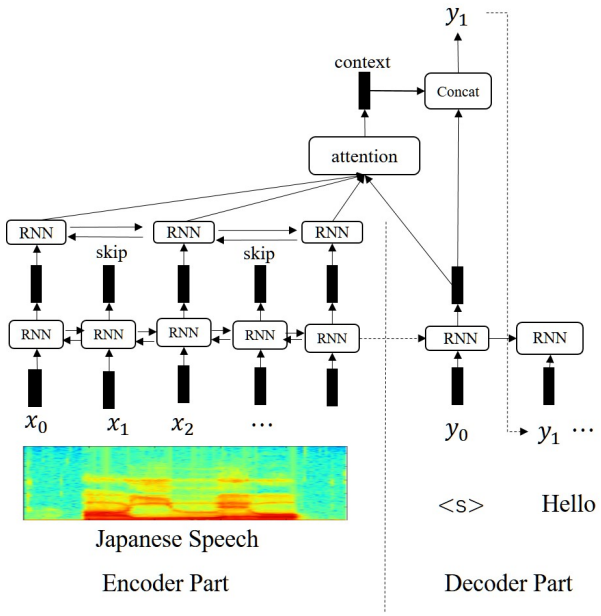

Fig. 8. End-to-end speech translation skipping odd index states

Transformer uses self-attention to model long-term context information.

III. Attention-based Speech Translation by Multi-task Learning with Transcoding

Utilizing attention-based encoder-decoder architecture for constructing a Direct ST task is difficult because the model needs to solve two complex problems:
(1) learning how to process a long speech sequence and mapping it to the corresponding words, similar to the issues focused on in the ASR field [5];
(2) learning how to make proper alignment rules between the source and target languages, similar to the issues discussed in the NMT field [6,25]. In our proposed method, the attention-based encoder-decoder neural network is not trained directly for speech translation tasks using parallel data. Inspired by the original CL ideas, we proposed a new CL training strategy. Instead of increasingly adding difficult data examples, as in the conventional CL, we train the model step by step from the easy task to the complicated task changing the model structures. We start training of an attention-based encoder-decoder for speech recognition (speech-to-text on the same source language) and text-based machine translation (text-to-text on the source and target languages) task. Then we gradually extend to the end-to-end speech translation (speech-to-text or speech-to-speech on the source and target languages) task. We describe each training phase's input and target sequence with their structures in Figs. 9-11.

The encoder part received the speech features and converted them by an MLP layer and output encoder state sequence $H^{\text{enc}} = [h_0^{\text{enc}}, ..., h_n^{\text{enc}}]$. We did the ASR decoding process to get ASR decoder outputs h_m^{dec} and used the MLP attention mechanism for the RNN-based model and the Multi-head attention mechanism for the Transformer-based model. The

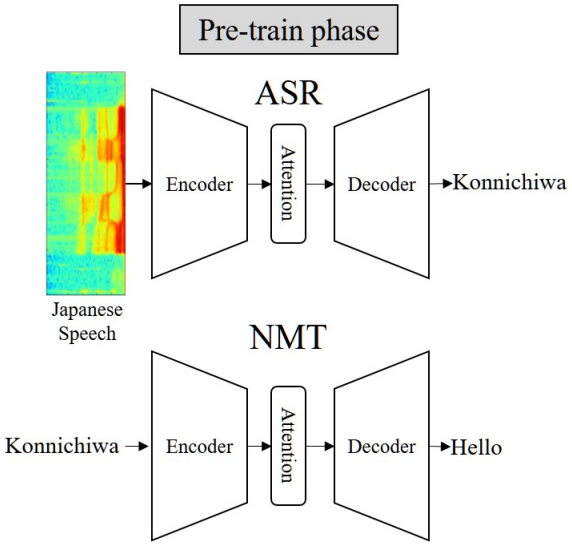

Fig. 9. Proposed: pre-training phase

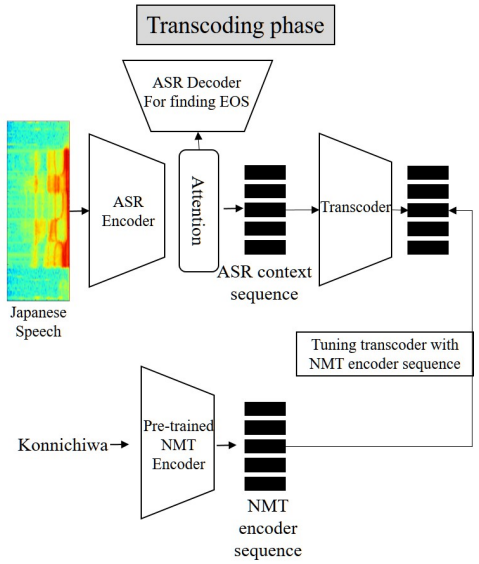

Fig. 10. Proposed: training transcoding phase

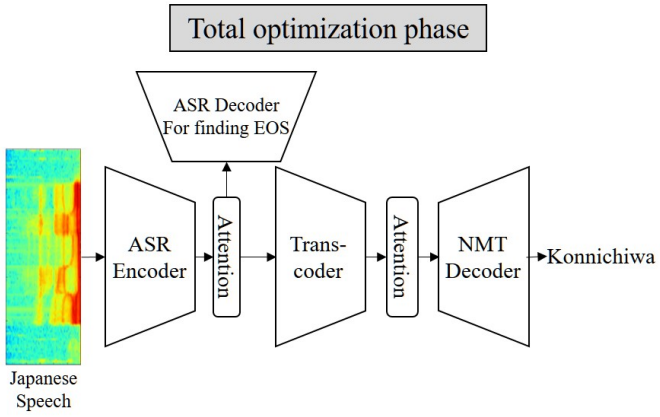

Fig. 11. Proposed: total optimization phase

ASR model provides a context vector sequence $C^{\text{ASR}} = [c_0^{\text{ASR}}, ..., c_m^{\text{ASR}}]$, where m represents the length of a source-language text (words, subwords, or character sequences):

$$\begin{aligned} h_m^{dec} &= Decoder^{ASR}(emb_{m-1}^{y}) \\ a_m &= attention(h^{enc}, h_m^{dec}) \\ c_m &= \sum_{n=1}^{N} a_t(n) * h_n^{enc}. \end{aligned} \tag{6}$$

The $Decoder^{ASR}$ is a pre-trained ASR decoder, where y denotes the source language-text (words, subwords, or character sequences) and emb_{m-1}^{y} denotes the previous ASR decoder embedding vector. a_m is an attention score vector at decoding step m. C^{ASR} denotes a hidden representation of the source-language acoustic information. We input source text embedding X^{emb} into the pre-trained NMT encoder and generated output as linguistic hidden states. The transcoding process maps these acoustic hidden states to the linguistic hidden states of the source language. This process improves the attention result between the source and target sequences in the NMT decoder. The transcoding receives ASR context vector sequence C^{ASR} and generates transcoder output H^{TRC}. The transcoder has identical architecture as the pre-trained NMT encoder. We copied the pre-trained NMT encoder (except the embedding layer) and initialized the parameters. We then used NMT encoder hidden states H^{NMT} as a target to optimize the transcoder:

$$\begin{aligned} H^{\text{TRC}} &= \textit{Transcoder}(C^{\text{ASR}}), \\ H^{\text{NMT}} &= \textit{NMTEncoder}(X^{\text{emb}}). \end{aligned} \tag{7}$$

Here X^{emb} is a source-sentence embedding. The number of H^{TRC} and H^{NMT} states equals the source text length. During this transcoding process training, we froze the NMT encoder parameters and only updated the ASR encoder and transcoder parameters. We thoroughly optimized the transcoder to minimize the smooth L1 loss between H^{TRC} and H^{NMT}:

$$loss(H^{\text{TRC}}, H^{\text{NMT}}) = \begin{cases} 0.5 * (h_m^{\text{TRC}} - h_m^{\text{NMT}})^2, \\ \quad if |h_m^{\text{TRC}} - h_m^{\text{NMT}}| < 1, \\ |h_m^{\text{TRC}} - h_m^{\text{NMT}}| - 0.5, \\ \quad otherwise. \end{cases} \tag{8}$$

The model can learn a difficult problem on a small dataset using CL. End-to-end speech translation is difficult. Solving this problem requires the preparation of a deeper neural network and larger amounts of data compared to regular text NMT tasks. Since preparing parallel speech data is very expensive, we start training a model on a simple task and proceed to a more difficult task. In this way, the difficulty of the problems gradually increases with each training phase. In the end, the model can perform end-to-end speech translation using only a small initial training dataset. We first performed end-to-end speech translation with a small linguistic distant-language-pair dataset to confirm the CL benefits and compared the translation performance of several model architectures. We also performed speech translations with various language-pair large datasets to evaluate our proposed model and the baseline model. Finally, we used TED natural speech and performed end-to-end speech translation to confirm the effectiveness of our proposed approach.

IV. Experimental Set-Up and Results

A. Experiments on BTEC data

1) Experimental set-up for BTEC: First, we conducted our experiments using a basic travel expression corpus (BTEC) [26,27]. The BTEC English-Japanese parallel corpus consists of training (480k) and test (20k) utterances. The BTEC English-French and Japanese-Korean corpus consist of training (160k) and test (500) utterances. We only used utterances that exceed 4-words. Since the corresponding speech utterances for this text corpus are unavailable, we used the Google text-to-speech synthesis[1] to generate a speech corpus of the source language. To investigate the performance of our proposed system in natural speech, we also utilized the BTEC corpus that consists of 190k utterances of natural English speech. Since it only has an 8k speech-to-text parallel data of English-French and English-Japanese, we used natural and generated speech to train the ASR and ST systems and tested it on BTEC natural speech data.

Throughout this experiment, we describe the benefits and potential of our proposal's results. First, we demonstrate the BTEC translation task with the RNN-based model on the generated speech. Then we performed end-to-end translation tasks on natural speech with the Transformer model [22,28], which is a state-of-the-art sequential model. We changed all the RNN networks to the FC layer and the self-attention function and applied our proposed method to confirm how it works with natural speech translation tasks. We segmented the speech utterances into multiple frames with a 50-ms window and 12-ms steps and extracted 80-dimension Mel-spectrogram features using LibROSA[2]. We further used these data to build an attention-based ASR, an NMT system, the baseline Direct ST system, and our proposed ST system. The hyperparameter settings of these models are displayed in Tables III-VIII.

TABLE I
DATA SETTING OF BTEC GENERATED SPEECH GENERATED BY GOOGLE TTS

BTEC generated speech.			
Language pairs	En-Ja	En-Fr	Ja-Ko
Paired speech	480k	160k	160k

TABLE II
DATA SETTING OF BTEC NATURAL SPEECH

BTEC natural speech	
Language pairs	En-Ja
Unpaired speech	190k
Paired speech	8k

[1]Google TTS: https://pypi.python.org/pypi/gTTS
[2]LibROSA: https://librosa.github.io/librosa/

For each system, we prepared characters, subwords [29], and words as translation sequences. At the evaluation steps, our final goal is to increase the translation accuracy, which is the word level. Therefore, we combined characters or subwords into words for evaluation.

Next we summarize the network parameters. For all systems, we used the same learning rate and adopted Adam [30] in all of the models (Table VII).

We applied the attention-based encoder-decoder architecture described in Section II to train the ASR, NMT, and Direct ST systems. We also constructed a Cascade ST system and an ASRenc-NMTdec ST system, as described in Figs. 5-6. For our proposed models, we applied our proposed CL-based training strategy to the attention-based encoder-decoder architecture described in Section III.

Baseline Cascade ST:
: A conventional speech-to-text translation model that cascades ASR and NMT systems (Fig. 5).

Baseline Direct ST:
: A direct end-to-end speech translation model that uses a single attention-based neural network (Fig. 4).

Baseline ASRenc-NMTdec ST:
: An end-to-end speech translation model that uses a pre-trained ASR encoder and an NMT decoder (Fig. 6).

Proposed CL-Transcoder ST:
: Our proposed direct end-to-end speech translation model trained with transcoder and CL strategies (Fig. 11).

To confirm our assumptions and the behavior of the proposed method, we extracted a small dataset from the original dataset. It was only 45k utterances for training and 500 utterances for testing, and we also limited the length of the input speech to less than 500 frames to save memory resources. Our ASR system achieved an 8% word error rate (WER). For translation quality, we compared the BLEU+1 scores of each model's performance. We chose BLEU+1 because a BTEC corpus consists of many short utterances and BLEU+1 is a more suitable objective evaluation method than BLEU [31] scores for short translations [32].

2) Experimental results on BTEC: First, we show how our proposed method works during training with a small amount of data. In this experiment, we limited the training data to only 45 k of generated speech utterances. We report the validation set softmax cross-entropy of each model in Fig. 12. From this figure, we conclude that the direct speech translation model encounters difficulties in the training process. This leads us to suspect that we require more training data. On the other hand, our proposed model and the pre-trained ASR encoder and NMT decoder concatenation model reduced the validation loss even with fewer training data. Note also that the ASRenc-NMTdec ST model's first epoch validation loss is as high as that of the direct translation model. Furthermore, our transcoding method begins and converges with better validation loss than the ASRenc-NMTdec ST model.

Table IX shows the translation results of the baseline and proposed systems with the BLEU+1 scores. We also include

TABLE III
ASR SETTINGS

ASR system	
Input units	80
Downsampling ratio	0.25
MLP hidden units	256
Encoder RNN layers	LSTM, GRU
LSTM and GRU hidden units	256,512
Encoder dropout ratio	0, 0.3
Attention	General
Decoder layer depth	GRU
Decoder dropout ratio	0.8
Embed size	128
Embed dropout ratio	0.5

TABLE IV
NMT SETTINGS

NMT system	
Encoder layers	LSTM, GRU
LSTM and GRU hidden units	256, 512
Encoder dropout ratio	01, 0.3
Attention	General
Decoder layer depth	GRU
Decoder dropout ratio	0.3
Embed size	128
Embed dropout ratio	0.5

TABLE V
DIRECT ST SETTINGS

ST system	
Input units	80
Downsampling ratio	0.25
MLP hidden units	256
Encoder layers	LSTM, GRU
LSTM and GRU hidden units	256, 512
Encoder dropout ratio	0, 0.3
Attention	General
Decoder layer	GRU
Decoder dropout ratio	0.3
Embed size	128
Embed dropout ratio	0.5

TABLE VI
PROPOSED ST SETTINGS

ST system	
Input units	80
Downsampling ratio	0.25
MLP hidden units	256
Encoder layers	LSTM, GRU
LSTM and GRU hidden units	256, 512
Transcoder layers	LSTM, GRU
LSTM and GRU hidden units	256, 512
Decoder layer	GRU
Decoder dropout ratio	0.3
Embed size	128
Embed dropout ratio	0.5

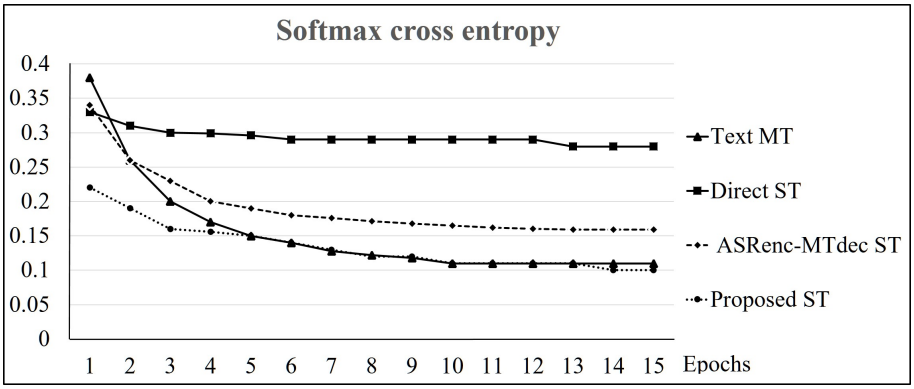

Fig. 12. Small dataset of English-to-Japanese training steps with softmax cross-entropy loss

TABLE VII
OPTIMIZER SETTINGS

Optimization	
Initial learning rate	0.001
Optimizing method	Adam [30]

TABLE VIII
VOCABULARY SIZE OF EACH LANGUAGE AND SEGMENT ON BTEC

Vocabulary size			
Language	Word	Subword	Character
English	27011	2918	29
Japanese	32794	2899	2691
Korean	55092	2902	1422
French	14802	2789	31

TABLE IX
TRANSLATION RESULTS (BLEU+1) OF ENGLISH-TO-JAPANESE LANGUAGE PAIRS WITH A SMALL DATASET

Model	BLEU+1
Baseline Cascade ST	28.6
Baseline Direct ST	14.0
Baseline ASRenc-NMTdec ST	28.2
Proposed CL-Transcoder ST	34.3
Text-based NMT	33.2

TABLE X
ASR WORD ERROR RATE ON SMALL AMOUNT OF BTEC DATA

Language	Characters	Subwords	Words
Ja	14.3	7.1	6.9
En	10.1	6.0	5.9

text-to-text translation results (text-based NMT). The baseline Direct ST system with a single attention module failed to translate the English speech to Japanese text. Learning such syntactically distant languages as Japanese and English is difficult when the training data are limited. However, the performance greatly improved when we applied pre-trained ASR and NMT parameters to the encoder and decoder in the ASRenc-NMTdec ST system, which achieved identical performance as the baseline Cascade ST systems. Our proposed CL-Transcoder ST system achieved the best performance. It can be stably trained and successfully outperformed all the baseline systems with a significant BLEU+1 score margin. The proposed method's performance even surpassed that of the text-based NMT. We constructed our proposed system with [ASRenc+att]+[Transcoder]+[att+MTdec]. From a text-based MT system viewpoint, the combination of the second and third parts resembles MT, and the additional components in the first part, which introduced more noise to the MT system's input, might function as a denoising encoder-decoder that prevents overfitting.

Next we evaluated with a complete dataset and further investigated the performance of the systems in various units (character, subword, and word units) and various language pairs. We first calculated the WER for each ASR system on a small BTEC dataset (Table X). Our ASR achieved a satisfactory performance below 10% WER. We achieved a higher performance on ASR because we used speech generated from TTS to train and evaluate the models. A single speaker generated TTS speech, and the speaking style is very stable.

We then evaluated the translation quality for each system and show the results in Tables XI-XIII. Tables XI and XII demonstrate that the performances of the baseline Cascade ST and Direct ST approaches are similar on subword and word translation on syntactically similar language pairs. However, similar to the phenomena with the small dataset, the baseline Direct ST did not perform well for syntactically distant language pairs. In such language pairs, richer architecture and a better training strategy are necessary.

In contrast, our proposed models outperformed both baseline systems on syntactically distant language pairs in the character-, subword-, and word-based systems (Table XIII). Even on similar language pairs, our proposed approach successfully improved the end-to-end speech translation quality in the subword and word units.

These experiments show that our proposed system has the potential to outperform the Cascade ST model. However, these results are based on generated speech data; therefore, we also performed a BTEC translation task with natural speech with a state-of-the-art sequential model Transformer for the ASR

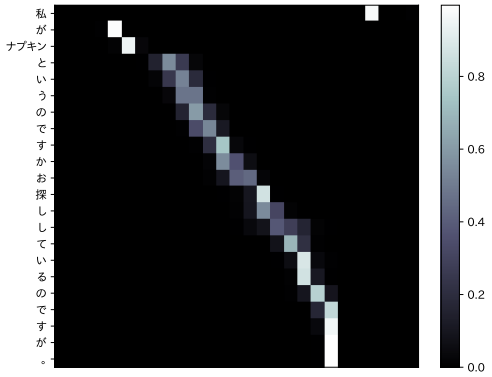

Fig. 13. Japanese ASR attention

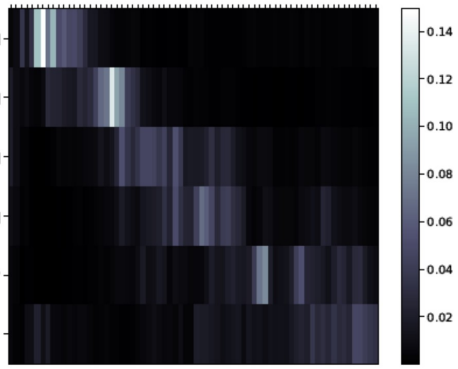

Fig. 14. Japanese-to-Korean direct translation attention

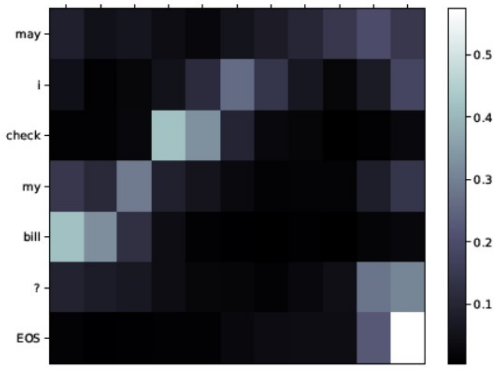

Fig. 15. Japanese-to-English cascade translation attention

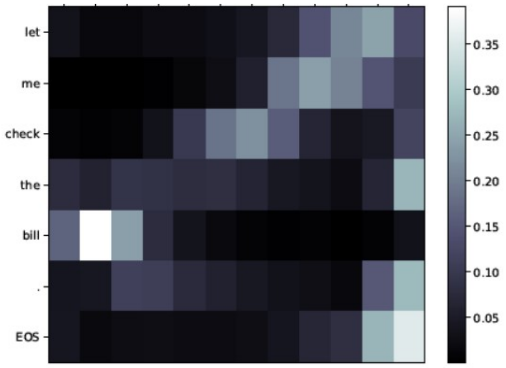

Fig. 16. Proposed Japanese-to-English translation attention compared with cascade translation

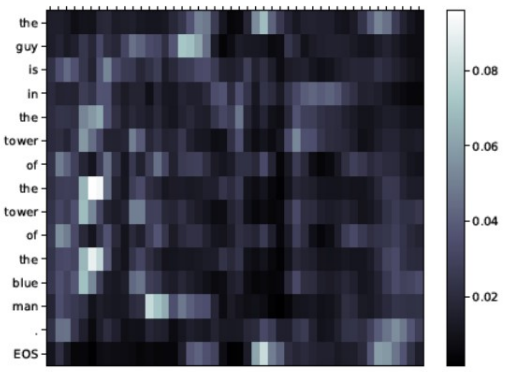

Fig. 17. Japanese-to-English direct translation attention

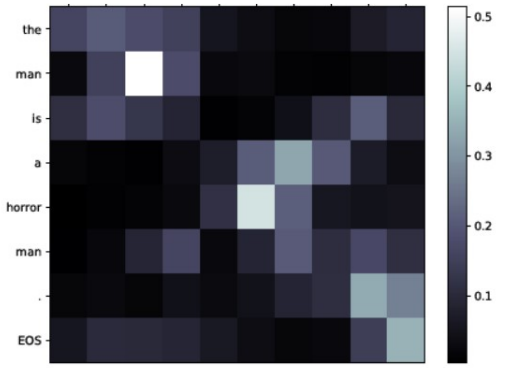

Fig. 18. Proposed Japanese-to-English translation attention compared with direct translation

TABLE XI
BLEU+1 SCORE OF BASELINE CASCADE ST SYSTEM

Language pair	Characters	Subwords	Words
Ja to En	25.5	30.5	32.6
Ja to Ko	31.0	40.1	41.9
En to Fr	34.8	39.9	39.7
En to Ja	29.2	32.7	33.1

TABLE XII
BLEU+1 SCORE OF BASELINE DIRECT ST MODEL

Language pair	Characters	Subwords	Words
Ja to En	17.3	18.7	19.3
Ja to Ko	29.6	39.4	39.0
En to Fr	22.3	35.2	36.8
En to Ja	20.4	27.0	22.3

and NMT models. In this experiment, we used a character as a basic unit for ASR model. We prepared a BTEC English natural speech dataset for English-French and English-Japanese translation. We summarize our Transformer parameters in

TABLE XIII
BLEU+1 SCORE OF PROPOSED CL-TRANSCODER ST MODEL

Language pair	Characters	Subwords	Words
Ja to En	30.5	36.2	37.4
Ja to Ko	30.1	42.8	43.0
En to Fr	33.0	41.4	42.7
En to Ja	30.9	37.0	38.6

Table XIX. We used the same Transformer model for ASR and NMT, but ASR has a prenet module (FC and Convolution network) instead of a source embedding layer [28]. We added 20% Gaussian noise to the decoder embedding vectors to increase the beam-search performance on the test set. The transcoder model has three FC layers and a self-attention function, which is identical to the Transformer encoder layers. First, we trained the ASR model using the BTEC natural and generated speech. ASR achieved a 6% WER on the BTEC natural speech test set (500 utterances). The BTEC text NMT systems were given English character sequences as input and output subword target sequences. We show our ASR model's performance in Table XIV and present the text NMT, Cascade ST, and Proposed ST BLEU scores in Table XV. The results for the RNN-based model in natural speech are slightly worse compared to the performance with generated speech. But, if we use the Transformer instead of RNN that is trained using both natural and generated speech, we achieved a high ASR performance of 6% WER. Also, the Transformer framework improved the text translation performance when compared to RNN-based model. Based on these results, we used the Transformer architecture as the standard architecture of our proposed model in later sections.

TABLE XIV
ASR WORD ERROR RATE OF BTEC NATURAL SPEECH

Model	Test speech data	
	Generated BTEC	Natural BTEC
RNN ASR	8 %	9 %
Transformer ASR	1 %	6 %

TABLE XV
BLEU SCORES OF BTEC NATURAL SPEECH TRANSLATION

Model	En to Fr	En to Ja
Direct ST	36.2	28.2
ASR and NMT Cascade ST	40.3	35.0
End-to-end Proposed ST	43.8	40.0

B. Experiments on TED Talk data

1) Experimental set-up for TED Talk: We also performed experiments on the TED corpus[3], which consisted of 270k English sentences. All of these sentences have a corresponding French translation, but only 210k have German translations.

[3]TED talks: https://www.ted.com/talks

TABLE XVI
DATA SETTING OF TED GENERATED SPEECH CREATED BY GOOGLE TTS SYSTEM

TED generated speech		
Language pairs	En-Fr	En-De
Paired speech	270k	210k

TABLE XVII
DATA SETTING OF TED NATURAL SPEECH

TED natural speech		
Language pairs	En-Fr	En-De
Unpaired speech	236k	236k
Paired speech	8k	178k

TABLE XVIII
ASR PRENET SETTINGS FOR TED TALK

ASR prenet functions	
Input FC units	80
Output FC units	512
hline Convolution layers	3
Convolution input units	512
Convolution kernel size	5
Convolution dropout ratio	0.2
Batch normalize	True
Post FC units input and output units	512

TABLE XIX
TRANSFORMER SETTINGS FOR TED TALK

Transformer parameters	
Encoder layers	3
Decoder layers	6
Input and hidden units	512
Transformer FC units	1024
Multi-head number	8
Dropout ratio	0.2

TABLE XX
TEXT EMBEDDING SETTINGS FOR TED TALK

Text embedding layers	
English character embedding input units	32
English subword embedding input units	31011
German subword embedding input units	33124
French subword embedding input units	31092
Embedding output units	512
Position encoding	True
Decoder noising rate	0.2

TABLE XXI
OPTIMIZER SETTINGS

Optimization	
Optimizing method	Adam [30]
Warm-up steps	4000
Initial learning rate	0.001

We used these text datasets to train each English-French and English-German MT model. For the speech-to-text model, we used the TEDLIUM English 58k natural speech. Only 6k English utterances overlap between the TEDLIUM 58k natural speech and the TED talk parallel text dataset. We made a 6k-utterance English-French and English-German natural speech-to-text corpus using these overlapping data. To add more data, we used Google TTS to generate another 270k English speech utterances from a parallel text corpus. Finally, we also utilized the IWSLT2018 English-German speech-to-text TED dataset, which consists of 2k talk waveform data. Based on the provided IWSLT2018 alignment information, we segmented those waveforms and got 178k English-German parallel utterances. Therefore, we trained our English ASR and transcoder with the TEDLIUM 58k utterances of natural speech, the IWSLT2018 178k utterances of natural speech, and 270k generated speech utterances. The English-French ST model was trained with a 270k generated speech-to-text corpus and a 6k natural speech-to-text corpus, and the English-German ST model was trained with a 210k generated speech-to-text corpus, a 6k natural speech-to-text corpus, and the IWSLT2018 178k utterances of the natural speech-to-text corpus. For evaluation, we used the IWSLT2018 "dev2010" dataset for a validation set as well as a "tst2015" and "tst2018" datasets for test sets.

2) Experimental results on TED Talk: We also trained the ASR model using TED natural speech, BTEC natural speech, and TED generated speech. ASR achieved a 18% WER on the TED natural speech test set shown in Table XXII. The TED natural speech included lots of noise, and therefore the generated speech and the BTEC natural speech did not improve the TED natural test speech ASR performance. We also trained the NMT model using only the TED corpus. We present the text NMT, Cascade ST, and Proposed ST BLEU scores in Table XXIII. For comparison in the same condition, the TED text NMT systems were given English character sequences as input and output subword target sequences. We trained the text NMT for each language pair and chose the best performance setting's output segments to train the ST model.

TABLE XXII
ASR WORD ERROR RATE OF TED NATURAL SPEECH

Model	tst2015	tst2018
Transformer ASR	18 %	19 %

TABLE XXIII
BLEU SCORES OF TED NATURAL SPEECH TRANSLATION

	En to Fr	En to De	
Model	tst2015	tst2015	tst2018
Text-based NMT	29.8	25.1	25.3
Cascade ST	16.3	13.1	12.7
Proposed ST	17.1	13.8	13.0

The results of this experiment are displayed in Table XXIII. The performances of Cascade ST and Proposed ST models were affected by ASR errors. Although Transformer can give better performance than that of RNN, the Transformer still has some weaknesses. If the Transformer NMT got incorrect inputs (i.e., ASR errors), then NMT may output very short sequences (e.g., "so" or "and"). However, our proposed method has a potential to recover the ASR errors in the translation process. Therefore, our proposed method outperformed Cascade ST in natural speech translation tasks.

C. Discussion

To provide more detail, we visualized a sample attention matrix of alignment weights from the ASR and all translation methods. Each pixel shows the alignment weight of the j-th source sequence (in the horizontal axis) to the i-th target sequence (in the vertical axis) in gray-scale (0: black indicates the lowest weight, and 1: white indicates the highest weight). The attention weights of the ASR system are shown in Fig. 13. The values are high along the diagonal of the matrix, which illustrates the monotonous (left-to-right) natural alignment between speech and text. Fig. 14 shows the attention weights of the baseline Direct ST for similar language pairs of Japanese and Korean. It also has a diagonal line, which indicates that the alignment of words between Japanese and Korean is mostly monotonic. However, the attention scores are not as high as in the ASR system since there are many possible ways to translate from one language to another. Therefore, instead of having a significant weight in a single pixel, it has weak weights in several pixels that represent several possible candidates.

Next we investigated the attention weights for the ST task with distant language pairs. First, we compared the attention weights of the baseline Cascade ST in Fig. 15 with the proposed CL-Transcoder ST in Fig. 16 for Sentence 1 of Table XXIV. In our proposed method, we have two attention modules. The first aligns the encoder's hidden states of speech features with the transcoder input sequence, and the second aligns the transcoder output sequence with the decoder. In this figure, we display only the second. The first attention module has a monotonic shape that resembles ASR attention (Fig. 13). In the baseline Cascade ST, the NMT model input is the sampled output from the ASR model hypotheses, which is a single character, subword, or word unit. Unfortunately, when the ASR hypotheses are wrong, the NMT is unable to fix such errors. On the other hand, our proposed CL-Transcoder ST model solves this problem by directly mapping the speech context vector into a latent translation state using the transcoder. In this way, the speech context vector has more latent information compared with the sampled character or word. In other words, instead of having the 1-best ASR hypothesis, it had an n-best hypothesis. As seen from Figs. 15-16, our proposed method produces better attention and results, specifically on the top part of attention weights "may I" versus "let me" due to the ability to recover some ASR errors. Thus, our proposed CL-Transcoder ST with transcoder successfully solved the error propagation problem in traditional model cascade architecture.

We also compared the attention weights of the baseline Direct ST in Fig. 17 with the attention weights of the proposed

TABLE XXIV
TRANSLATION RESULTS FOR FIGS. 14-17

Speech translation results of sentence 1		
Recognition (Ja)	Reference	**seisan syo wo kakunin sa se te kuda sa i .**
	ASR Result	**seisan syo wo kakunin** *shi ma su ka* **.**
Translation (Ja to En)	Reference	**let me check your account .**
	Cascade ST	*may i* **check** *my bill ?*
	Proposed ST	**let me check** *the bill .*
Translation results of sentence 2		
Recognition (Ja)	Reference	**ano otoko ha do ahoh de su .**
	ASR Result	**ano otoko ha** ** **ahoh de su .**
Translation (Ja to En)	Reference	**that man is an asshole .**
	Direct ST	*the guy* **is** *in the tower of the tower of the blue man* **.**
	Proposed ST	*the* **man is** *a horror man* **.**

CL-Transcoder ST from the second attention module in Fig. 18 for Sentence 2 of Table XXIV. As seen from Fig. 17, there is no clear association between the speech input frames and their corresponding words. This indicates that direct translation for distant language pairs using standard attention-based end-to-end architecture is difficult. In contrast, our proposed architecture addresses the problem with two attention modules and one transcoder. The first attention handles the mapping between the source speech encoder and the hidden representation of the transcoder input, and the transcoder maps the underlying representation of the input speech context vector into the latent representation of the text transcription of the transcoder output. Therefore, the task of the second attention is only to map between the transcoder output state to the target text translation of the decoder. In this way, the attention module generated a better attention map and result (Fig. 18) for Sentence 2 in Table XXIV.

Our most significant contribution is that we directly transferred the ASR hidden vector to the translation part. Since the cascade models transfer the transcript results, when an ASR error occurs, and attention spreads widely, the attention module cannot find useful information to recover the ASR error. In our proposed models, we processed the BiRNNs or Transformer for the ASR context vectors in the transcoding part, and the transcoder hidden states are the attention keys, such as in the hidden states of machine translation encoders. Therefore, when an ASR error occurs and the attention scores spread widely, the attention module can gather information before and after the hidden sequences to recover the error. Another contribution is simplifying the task of the attention module. Constructing a direct speech translation in a single model means that we combine both tasks and perform many-to-many mapping tasks over long sequences (i.e., mapping from various speech utterances of identical contents to multiple possible translations). For language pairs with similar syntax and word order, the alignment between the source and target sequences is almost monotonous (left-to-right), but the model can still handle the task. In contrast, with syntactically distant language pairs, speech translation requires distant word order. Therefore the attention module suffers when it handles a long input sequence and a distant alignment between the source and target sequence states. Such a condition is computationally expensive and often produces misalignment. Therefore, we propose a framework with two attention modules and one transcoder instead of just a single attention module. The first attention module task works on the alignment between a vector representation sequence of the speech utterance to the corresponding text sequence in the source language and produces context information (similar in complexity to an ASR attention module). The second attention module task works on the alignment between a vector representation of the text sequence in the source language to the target language (similar in complexity to an NMT attention module). Finally, the transcoder task connects both attention modules and converts the ASR context information to text vector representation in the source language. In this way, we can perform direct speech-to-text translation while keeping the complexity almost similar to the cascade ASR-NMT systems.

Our final contribution is providing the possibility to perform direct speech translation without large-size parallel speech data. One main issue in developing a direct ST system is the need for parallel speech (speech-text or speech-speech translation) data. Many studies have addressed this problem by utilizing TTS to generate speech data. In this paper, we also use 160k parallel text and the corresponding speech generated by TTS for pre-training ASR and NMT as well as for the total optimization of the direct ST system. This was done to have a fair comparison with the baseline Cascade ST and Direct ST. Our proposed architecture, however, allows us to perform without extensive size parallel speech data. For example, we can pre-train ASR with natural and synthesized data that do not have the corresponding translation and pre-train NMT with standard parallel text that lacks relevant speech data. We perform total optimization with only a small amount of parallel natural speech data. Furthermore, our proposed method utilizes pre-trained ASR and NMT models and transfers and optimizes pre-trained models in a hidden state sequence as a transcoding process. The optimization process no longer depends on the text data of the source language.

Our second experiment on the BTEC and TED dataset shows that our proposed method also works on difficult natural speech translation tasks. The ASR and transcoder model can use natural and generated speech corpus in training. The transcoding process maps the ASR context sequence to the

NMT encoder sequence; even then, however, ASR occurs, and the context vector includes much noise. However, the transcoding process can recover the noise. Therefore, our proposed method outperforms the standard cascade ASR + NMT model.

If the source language lacks a written form or a sufficient amount of data, we can first pre-train the ASR and NMT models using similar languages or other languages from the same family that have written forms. We can then perform total optimization with the required language. Further investigation is needed about the effectiveness of our proposed architecture to extend the application to various languages (even for those without a written form) as well as possibly extending the TTS part and performing a complete speech-to-speech translation.

V. Conclusions

We presented the construction of end-to-end speech translation for distant language pairs with transcoding based on CL strategies that gradually trained the network for end-to-end speech translation tasks by adapting decoder or encoder parts.

Our experimental results demonstrated that the translation quality outperformed the Cascaded ST, standard Direct ST, and ASRenc-NMTdec ST systems and revealed that our proposed model effectively decreased loss even using direct complex problems. These results still rely on synthetic data because we need to prepare the same data for all the baseline and proposed systems, and a huge parallel speech corpus is not available. Our proposed architecture, however, can effectively use monolingual speech and parallel text. Our transcoding process may also be useful for joint TTS schemes to achieve speech-to-speech and/or paralinguistic translations. In the future, we will investigate the effectiveness of our proposed method using natural speech data and paralinguistic information and expand the speech-to-text translation task to a speech-to-speech translation task.

References

[1] S. Nakamura, K. Markov, H. Nakaiwa, G. Kikui, H. Kawai, T. Jitsuhiro, J. Zhang, H. Yamamoto, E. Sumita, and S. Yamamoto, "The ATR multilingual speech-to-speech translation system," *IEEE Trans. Audio, Speech & Language Processing*, vol. 14, no. 2, pp. 365–376, 2006. [Online]. Available: https://doi.org/10.1109/TSA.2005.860774

[2] T. Kano, S. Takamichi, S. Sakti, G. Neubig, T. Toda, and S. Nakamura, "Generalizing continuous-space translation of paralinguistic information," in *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25-29, 2013*, F. Bimbot, C. Cerisara, C. Fougeron, G. Gravier, L. Lamel, F. Pellegrino, and P. Perrier, Eds. ISCA, 2013, pp. 2614–2618. [Online]. Available: http://www.isca-speech.org/archive/interspeech_2013/i13_2614.html

[3] Q. T. Do, S. Sakti, G. Neubig, and S. Nakamura, "Transferring emphasis in speech translation using hard-attentional neural network models," in *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016*, N. Morgan, Ed. ISCA, 2016, pp. 2533–2537. [Online]. Available: https://doi.org/10.21437/Interspeech.2016-898

[4] P. D. Agüero, J. Adell, and A. Bonafonte, "Prosody generation for speech-to-speech translation," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing, ICASSP 2006, Toulouse, France, May 14-19, 2006*. IEEE, 2006, pp. 557–560. [Online]. Available: https://doi.org/10.1109/ICASSP.2006.1660081

[5] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., 2015, pp. 577–585. [Online]. Available: http://papers.nips.cc/paper/5847-attention-based-models-for-speech-recognition

[6] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1409.0473

[7] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. V. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, F. Lacerda, Ed. ISCA, 2017, pp. 4006–4010. [Online]. Available: http://www.isca-speech.org/archive/Interspeech_2017/abstracts/1452.html

[8] L. Duong, A. Anastasopoulos, D. Chiang, S. Bird, and T. Cohn, "An attentional model for speech translation without transcription," in *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, K. Knight, A. Nenkova, and O. Rambow, Eds. The Association for Computational Linguistics, 2016, pp. 949–959. [Online]. Available: https://doi.org/10.18653/v1/n16-1109

[9] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open source toolkit for statistical machine translation," in *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*, J. A. Carroll, A. van den Bosch, and A. Zaenen, Eds. The Association for Computational Linguistics, 2007, pp. 177–180. [Online]. Available: https://www.aclweb.org/anthology/P07-2045/

[10] A. Berard, O. Pietquin, C. Servan, and L. Besacier, "Listen and translate: A proof of concept for end-to-end speech-to-text translation," *CoRR*, vol. abs/1612.01744, 2016. [Online]. Available: http://arxiv.org/abs/1612.01744

[11] A. Berard, L. Besacier, A. C. Kocabiyikoglu, and O. Pietquin, "End-to-end automatic speech translation of audiobooks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*. IEEE, 2018, pp. 6224–6228. [Online]. Available: https://doi.org/10.1109/ICASSP.2018.8461690

[12] R. J. Weiss, J. Chorowski, N. Jaitly, Y. Wu, and Z. Chen, "Sequence-to-sequence models can directly translate foreign speech," in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, F. Lacerda, Ed. ISCA, 2017, pp. 2625–2629. [Online]. Available: http://www.isca-speech.org/archive/Interspeech_2017/abstracts/0503.html

[13] S. Bansal, H. Kamper, A. Lopez, and S. Goldwater, "Towards speech-to-text translation without speech recognition," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, M. Lapata, P. Blunsom, and A. Koller, Eds. Association for Computational Linguistics, 2017, pp. 474–479. [Online]. Available: https://doi.org/10.18653/v1/e17-2076

[14] S. Nolfi, D. Parisi, and J. L. Elman, "Learning and evolution in neural networks," *Adaptive Behaviour*, vol. 3, no. 1, pp. 5–28, 1994. [Online]. Available: https://doi.org/10.1177/105971239400300102

[15] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, ser. ACM International Conference Proceeding Series, A. P. Danyluk, L. Bottou, and M. L. Littman, Eds., vol. 382. ACM, 2009, pp. 41–48. [Online]. Available: https://doi.org/10.1145/1553374.1553380

[16] C. Gong, D. Tao, S. J. Maybank, W. Liu, G. Kang, and J. Yang, "Multi-modal curriculum learning for semi-supervised image classification," *IEEE Trans. Image Processing*, vol. 25, no. 7, pp. 3249–3260, 2016. [Online]. Available: https://doi.org/10.1109/TIP.2016.2563981

[17] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language*

Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015, L. Màrquez, C. Callison-Burch, J. Su, D. Pighin, and Y. Marton, Eds. The Association for Computational Linguistics, 2015, pp. 379–389. [Online]. Available: https://doi.org/10.18653/v1/d15-1044

[18] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., 2014, pp. 3104–3112. [Online]. Available: http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[20] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014. [Online]. Available: http://arxiv.org/abs/1412.3555

[21] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, L. Màrquez, C. Callison-Burch, J. Su, D. Pighin, and Y. Marton, Eds. The Association for Computational Linguistics, 2015, pp. 1412–1421. [Online]. Available: https://doi.org/10.18653/v1/d15-1166

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 5998–6008. [Online]. Available: http://papers.nips.cc/paper/7181-attention-is-all-you-need

[23] L. J. Ba, R. Kiros, and G. E. Hinton, "Layer normalization," *CoRR*, vol. abs/1607.06450, 2016. [Online]. Available: http://arxiv.org/abs/1607.06450

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 770–778. [Online]. Available: https://doi.org/10.1109/CVPR.2016.90

[25] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, HLT-NAACL 2003, Edmonton, Canada, May 27 - June 1, 2003*, M. A. Hearst and M. Ostendorf, Eds. The Association for Computational Linguistics, 2003. [Online]. Available: https://www.aclweb.org/anthology/N03-1017/

[26] G. Kikui, E. Sumita, T. Takezawa, and S. Yamamoto, "Creating corpora for speech-to-speech translation," in *8th European Conference on Speech Communication and Technology, EUROSPEECH 2003 - INTERSPEECH 2003, Geneva, Switzerland, September 1-4, 2003*. ISCA, 2003. [Online]. Available: http://www.isca-speech.org/archive/eurospeech_2003/e03_0381.html

[27] G. Kikui, S. Yamamoto, T. Takezawa, and E. Sumita, "Comparative study on corpora for speech translation," *IEEE Trans. Audio, Speech & Language Processing*, vol. 14, no. 5, pp. 1674–1682, 2006. [Online]. Available: https://doi.org/10.1109/TASL.2006.878262

[28] L. Dong, S. Xu, and B. Xu, "Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*. IEEE, 2018, pp. 5884–5888. [Online]. Available: https://doi.org/10.1109/ICASSP.2018.8462506

[29] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016. [Online]. Available: https://doi.org/10.18653/v1/p16-1162

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[31] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*. ACL, 2002, pp. 311–318. [Online]. Available: https://www.aclweb.org/anthology/P02-1040/

[32] P. Nakov, F. Guzmán, and S. Vogel, "Optimizing for sentence-level BLEU+1 yields short translations," in *COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, 8-15 December 2012, Mumbai, India*, M. Kay and C. Boitet, Eds. Indian Institute of Technology Bombay, 2012, pp. 1979–1994. [Online]. Available: https://www.aclweb.org/anthology/C12-1121/

[33] L. Màrquez, C. Callison-Burch, J. Su, D. Pighin, and Y. Marton, Eds., *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*. The Association for Computational Linguistics, 2015. [Online]. Available: https://www.aclweb.org/anthology/volumes/D15-1/

Takatomo Kano received his B.E. from Doshisha University, Kyoto, Japan, in 2011, and his M.S. from the Graduate School of Information Science, NAIST, Nara, Japan in 2013. He is currently in the doctoral course at NAIST, Japan. He interested in speech and natural language processing, with an end-to-end speech translation. He is a student member of ISCA, and ASJ.

Sakriani Sakti received her B.E. degree in Informatics (cum laude) from Bandung Institute of Technology, Indonesia, in 1999. In 2000, she received DAAD-Siemens Program Asia 21st Century Award to study in Communication Technology, University of Ulm, Germany, and received her MSc degree in 2002. During her thesis work, she worked with Speech Understanding Department, DaimlerChrysler Research Center, Ulm, Germany. Between 2003-2009, she worked as a researcher at ATR SLC Labs, Japan, and during 2006-2011, she worked as an expert researcher at NICT SLC Groups, Japan. While working with ATR-NICT, Japan, she continued her study (2005-2008) with Dialog Systems Group University of Ulm, Germany, and received her PhD degree in 2008. She actively involved in collaboration activities such as Asian Pacific Telecommunity Project (2003-2007), A-STAR and U-STAR (2006-2011).

Satoshi Nakamura is Professor of Graduate School of Information Science, Nara Institute of Science and Technology, Japan, Honorarprofessor of Karlsruhe Institute of Technology, Germany, and ATR Fellow. He received his B.S. from Kyoto Institute of Technology in 1981 and Ph.D. from Kyoto University in 1992. He was Associate Professor of Graduate School of Information Science at Nara Institute of Science and Technology in 1994-2000. He was Director of ATR Spoken Language Communication Research Laboratories in 2000-2008 and Vice president of ATR in 2007-2008. He was Director General of Keihanna Research Laboratories and the Executive Director of Knowledge Creating Communication Research Center, National Institute of Information and Communications Technology, Japan in 2009-2010. He is currently Director of Augmented Human Communication laboratory and a full professor of Graduate School of Information Science at Nara Institute of Science and Technology. He is interested in modeling and systems of speech-to-speech translation and speech recognition. He is one of the leaders of speech-to-speech translation research and has been serving for various speech-to-speech translation research projects in the world including C-STAR, IWSLT and A-STAR. He received Yamashita Research Award, Kiyasu Award from the Information Processing Society of Japan, Telecom System Award, AAMT Nagao Award, Docomo Mobile Science Award in 2007, ASJ Award for Distinguished Achievements in Acoustics. He received the Commendation for Science and Technology by the Minister of Education, Science and Technology, and the Commendation for Science and Technology by the Minister of Internal Affair and Communications. He also received LREC Antonio Zampoli Award 2012. He has been Elected Board Member of International Speech Communication Association, ISCA, since June 2011, IEEE Signal Processing Magazine Editorial Board Member since April 2012, IEEE SPS Speech and Language Technical Committee Member since 2013, and IEEE Fellow since 2016.