PAPER

Neural Incremental Speech Recognition Toward Real-time Machine Speech Translation

Sashi NOVITASARI[†], *Nonmember*, Sakriani SAKTI[†,††a)], *and* Satoshi NAKAMURA[†,††], *Members*

SUMMARY Real-time machine speech translation systems mimic human interpreters and translate incoming speech from a source language to the target language in real-time. Such systems can be achieved by performing low-latency processing in ASR (automatic speech recognition) module before passing the output to MT (machine translation) and TTS (text-to-speech synthesis) modules. Although several studies recently proposed sequence mechanisms for neural incremental ASR (ISR), these frameworks have a more complicated training mechanism than the standard attention-based ASR because they have to decide the incremental step and learn the alignment between speech and text. In this paper, we propose attention-transfer ISR (AT-ISR) that learns the knowledge from attention-based non-incremental ASR for a low delay end-to-end speech recognition. ISR comes with a trade-off between delay and performance, so we investigate how to reduce AT-ISR delay without a significant performance drop. Our experiment shows that AT-ISR achieves a comparable performance to the non-incremental ASR when the incremental recognition begins after the speech utterance reaches 25% of the complete utterance length. Additional experiments to investigate the effect of ISR on translation tasks are also performed. The focus is to find the optimum granularity of the output unit. The results reveal that our end-to-end subword-level ISR resulted in the best translation quality with the lowest WER and the lowest uncovered-word rate.

key words: *attention transfer, incremental speech recognition, real-time speech translation*

1. Introduction

As globalization rapidly expands, language barriers continue to be the most notorious restriction on free communication among different language speakers. In some situations, the problems can be solved by human interpreters. Their services are needed especially for direct human-to-human communications, where the participants do not speak in the same language. An example of such a situation is real-time lecture translation for audiences from various nationalities where the interpretation is done simultaneously to the lecturer's speech so the audience can follow it. Professional interpretation services, however, are expensive because speech interpretation is a complex skill that takes years to master. The availability of language pairs also remains scarce.

Speech-to-speech translation (S2ST) technology [1], in other words recognizing speech and translating it into another language, is one innovative technology that can provide support in many everyday situations. S2ST systems commonly consist of three components: automatic speech recognition (ASR) system, machine translation (MT) system, and text-to-speech synthesis (TTS) system. In conventional modular S2ST systems, MT starts the translation from the source language into the target language after receiving a complete sentence from the ASR [2], and TTS begins its synthesis after receiving a complete sentence from the MT system [3]. Recent studies also focus on end-to-end S2ST systems [4–6], where all processes are done by a single model. Both kinds of systems, however, suffer from a long translation processing delay since the length of the complete sentences in some talks can be long, complicated, or poorly structured. Consequently, such systems are not practical in situations where the delivery delay of the translation result to the user is critical. A solution to this problem is a real-time S2ST system that can mimic human interpreters, who generally recognize and translate the speech based on partial information with minimum delay.

Real-time S2ST systems require a low-latency ASR as the foremost component. Several studies recently proposed sequence mechanisms for incremental speech recognition (ISR) that transcribe the speech within a low delay [7–13]. For low delay recognition, ISR needs to decide the incremental steps to extract the transcription information from a short part of the speech. For this reason, the training mechanisms and frameworks of neural ISR systems are more complex than standard non-incremental neural ASRs that do not need to consider the speech segment boundaries. Among the existing ISR frameworks, neural transducer has the most similar neural network structure to the standard neural ASR [7]. This framework performs end-to-end incremental speech recognition through fix-sized speech segments recognition by learning the alignment between speech and text segments. The construction of it requires several alignment computations and updates through the training process, which cause the framework to be more complicated than the standard attention-based ASR.

In this work, we propose attention-transfer ISR (AT-ISR) for low delay speech recognition. AT-ISR employs the original architecture of a standard attention-based ASR to do the incremental recognition with the shorter sequences. It learns the incremental step from the standard non-incremental ASR, therefore, we consider the non-incremental ASR as a teacher model and ISR as a student model. Attention transfer allows AT-ISR to mimic the alignments that are produced by teacher ASR's attention component. AT-ISR construction only uses a standard non-incremental model, from which the AT-ISR parameter can be initialized, allowing a simple mechanism in the model

[†] The author is with the Augmented Human Communication Lab, Nara Institute of Science and Technology, Japan

[††] The author is with the RIKEN, Center for Advanced Intelligence Project AIP, Japan

a) E-mail: ssakti@is.naist.jp

DOI: 10.1587/trans.E0.??.1

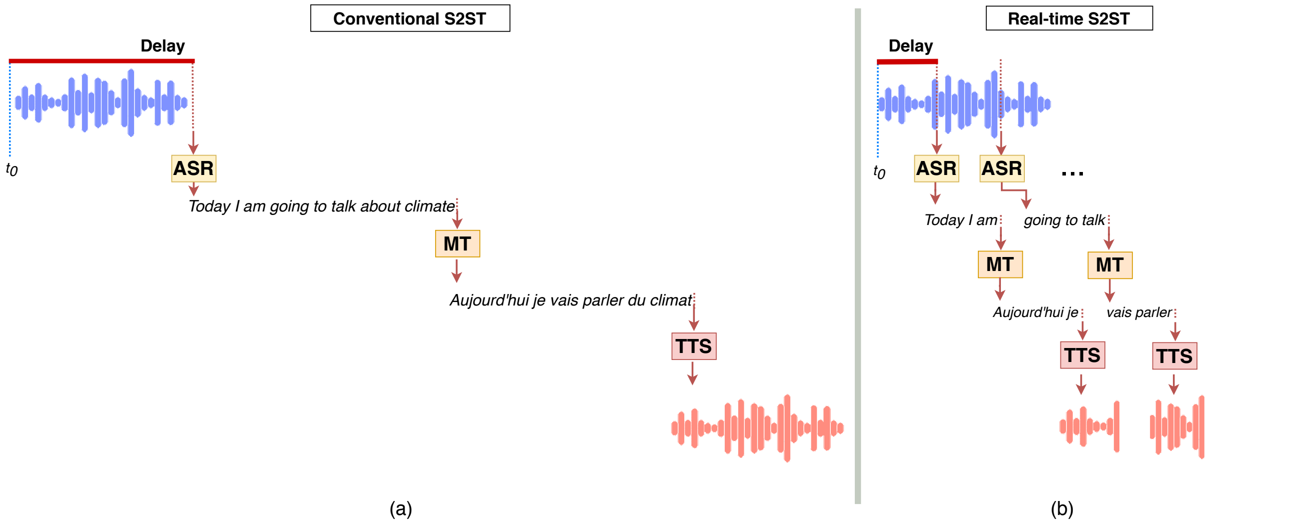

Fig. 1 Modular S2ST system: conventional framework (a) and real-time framework (b).

construction and the incremental recognition process.

Toward real-time speech translation, we performed additional experiments and explored two factors in ISR that might affect the translation: ISR delay and ISR output unit. End-to-end neural MT (NMT) systems are generally designed to process subwords, whereas the basic end-to-end neural ASR is generally trained to model speech-to-character. Therefore, we investigate the interdependency of both components' construction in terms of ISR output and NMT input unit parity. As developing incremental NMT is not part of this study, we evaluate the ISR using a standard NMT as a downstream task to see how the ISR performance and error affect the translation. The languages involved in the experiment are English, the source language, and French, the target language.

2. Overview of Speech-to-speech Translation System

2.1 Components

Modular S2ST system consists of three main, interconnected components: ASR, MT, and TTS. An illustration of modular S2ST systems can be seen in Fig. 1. The following are the details of each component.

2.1.1 Automatic Speech Recognition (ASR)

An ASR system converts the speech signal into a corresponding transcription. The conversion is done by generating a sequence of text unit $\mathbf{Y}^{(src)}$ from source speech features $\mathbf{X}^{(src)}$ extracted from the speech signal. Text generation is done by satisfying the following condition:

$$\hat{\mathbf{Y}}^{(src)} = \underset{y^{(src)}}{\operatorname{argmax}}\, P(\mathbf{Y}^{(src)}|\mathbf{X}^{(src)}). \tag{1}$$

Attention-based end-to-end ASR, which consists of an encoder-decoder with an attention mechanism, predicts character or subword sequence from a speech features sequence by modeling the conditional probability in Eq. (1) directly. In the remaining parts of this paper, we simplify the notations in Eq. (1) by denoting $\mathbf{X}^{(src)}$ as $\mathbf{X}$ and denoting $\mathbf{Y}^{(src)}$ as $\mathbf{Y}$.

2.1.2 Machine Translation (MT)

An MT system translates transcription in a certain language into the target language. In the S2ST system, it translates ASR output $\mathbf{Y}^{(src)}$ into text in target language $\mathbf{Y}^{(tgt)}$. The translation is done by satisfying the following condition:

$$\hat{\mathbf{Y}}^{(tgt)} = \underset{y^{(tgt)}}{\operatorname{argmax}}\, P(\mathbf{Y}^{(tgt)}|\mathbf{Y}^{(src)}). \tag{2}$$

We utilized attention-based NMT system for our experiment. Both NMT input and output are represented as a sequence of subwords in the corresponding language.

2.1.3 Text-to-Speech Synthesis (TTS)

In an S2ST system, TTS synthesizes a speech from a transcription that is given by MT. This model takes hypothesis text $\mathbf{Y}^{(tgt)}$ from MT and produces a sequence of speech features $\mathbf{X}^{(tgt)}$. The resulting speech is uttered in the target language with the same meaning as the source speech.

$$\hat{\mathbf{X}}^{(tgt)} = \underset{x^{(tgt)}}{\operatorname{argmax}}\, P(\mathbf{X}^{(tgt)}|\mathbf{Y}^{(tgt)}). \tag{3}$$

In this paper, we did not involve TTS in the experiment because we aimed to focus only on the ASR system and its connection to MT system.

2.2 Real-time Speech Translation

A real-time S2ST system is illustrated in Fig. 1(b). It consists of the same components as the conventional S2ST system in Fig. 1(a). The difference between both systems lies in the starting condition of each component's process. In the conventional S2ST system, each component has to wait for the complete result from the previous component. The translated speech can be only heard after the source speech is finished. On the other side, the real-time system does not limit each component to wait for the complete result from the other component. It just waits for a part of input rather than a complete input and works on the fly.

The performance of speech translation task involves output delivery speed and accuracy. Output delivery speed corresponds to the delay or time lag that occurs during the speech translation task. Delay is a time difference between the source speech start time and the initial time when the system produces the output [14]. Time delay in the conventional system (Fig. 1(a)) equals the total length of the source speech and the delay in the real-time system (Fig. 1(b)) equals the size of the first-recognized speech segment and is shorter than the conventional system. The actual delay also includes computational delay.

In many situations, a short speech translation delay is more preferable than a long delay. A short delay can relax the listener and facilitate indirect communication between the source speaker and the listener [15]. In real-time speech translation by human from English speech, the delay generally ranges from two to six seconds [16, 17], or roughly about four to twelve words [18, 19]. A short delay is also beneficial for human translators because it does not burden their short-term memory heavily.

Although it costs a long waiting time, a longer delay implies that we can get more information about the speech content, so the understanding of it for the translation can be improved [20]. For human interpreters, however, a long delay might also cause the translation quality to decrease because it can burden their working memory. In contrast to humans, memory load is not a vital issue in speech interpretation by a machine. However, a machine cannot understand speech utterance well like a human does unless it is trained using a large amount and variety of data. In this work, we not only construct ISR but also investigate how the delay of ISR system affects speech recognition and translation quality.

3. Sequence-to-sequence ASR Framework

A neural sequence-to-sequence (seq2seq) ASR consists of encoder and decoder components with an attention mechanism [21, 22]. It directly models $P(\mathbf{Y}|\mathbf{X})$ in Eq. (1) given a speech utterance feature sequence $\mathbf{X} = [x_1, ..., x_S]$ with length S and corresponding transcription $\mathbf{Y} = [y_1, ..., y_T]$ with length T. The encoder in the network transforms input sequence $\mathbf{X}$ into hidden states $\mathbf{h}^e$. The decoder then predicts target sequence probability p_{y_t}, given previous output $\mathbf{Y}_{<t}$, current context information c_t, and current decoder hidden states h_t^d. Context information c_t is produced by attention modules [23] at time t with the following formula:

$$c_t = \sum_{s=1}^{S} a_t(s) * h_s^e, \tag{4}$$

$$a_t(s) = \frac{exp(Score(h_s^e, h_t^d))}{\sum_{s=1}^{S} exp(Score(h_s^e, h_t^d))}. \tag{5}$$

The scoring for the context can be done using one of the following scoring functions [24]:

$$\text{Score}(h_s^e, h_t^d) = \begin{cases} \langle h_s^e, h_t^d \rangle, & \text{dot product,} \\ h_s^{e\top} W_s h_t^d, & \text{bilinear,} \\ V_s^\top \tanh(W_s[h_s^e, h_t^d]), & \text{MLP.} \end{cases} \tag{6}$$

The model loss function is formulated:

$$Loss_{ASR}(y, p_y) = -\frac{1}{T} \sum_{t=1}^{T} \sum_{c=1}^{C} \mathbb{1}(y_t = c) * \log p_{y_t}[c], \tag{7}$$

where C is the number of output classes.

4. Proposed Attention-Transfer ISR (AT-ISR)

We applied seq2seq ASR architecture to our ISR. Incremental recognition was done by applying attention transfer during the training phase and performing segment-based recognition. The details are explained in the following subsections.

4.1 Recognition Method

AT-ISR predicts the transcription $\mathbf{Y}$ with length T from a speech utterance $\mathbf{X}$ with length S through N recognition steps, where each step performs a short-segment-based recognition. In each recognition step $n = [1, ..., N]$, the model takes the n-th speech segment from $\mathbf{X}$, which segment consists of w frames, denoted as $\mathbf{X}_n$, to predict the n-th text segment of $\mathbf{Y}$ that consists of k_n tokens, denoted as $\mathbf{Y}_n$. The following is the mechanism of each model component for each recognition step n.

4.1.1 Encoding

In each step n, ISR encodes $\mathbf{X}_n = [x_{(n-1)w+1}, ..., x_{(n-1)w+w}]$, which is the n-th speech segment from $\mathbf{X}$, inside the input window with a fixed-length of w frames, where $w < S$. The input delay or the waiting time for the encoder to start the encoding is calculated as:

$$delay = w \cdot feat_{shift} + (feat_{win} - feat_{shift}), \tag{8}$$

where $feat_{win}$ and $feat_{shift}$ are the speech feature window length and window shift length. In the experiment, we allowed the encoder to look at several frames ahead of the main input frames. The look-ahead frames are regarded as the contextual input to enrich the information of the main input.

4.1.2 Decoding

After the encoding in step n finishes, the decoder predicts $\mathbf{Y}_n = [y_{n,1}, ..., y_{n,k_n}]$, which is the n-th text segment of $\mathbf{Y}$ with a length of k_n, where $0 \leq k_n < T$. $\mathbf{Y}_n$ aligns with $\mathbf{X}_n$. If the encoder input also includes the contextual frames, $\mathbf{Y}_n$ only aligns with the main input speech. In the text segment, y_{n,k_n} is a segment delimiter that is learned by the ISR during training. We define the text segment delimiter as an end-of-block symbol denoted as *</m>*. In the actual transcription, the actual last token in $\mathbf{Y}_n$ is y_{n,k_n-1}. During the decoding in step $n+1$, y_{n+1,k_1} is a token next to y_{n,k_n-1} in the actual transcription.

The decoding process in step n starts by taking y_{n-1,k_n-1} as the decoder's first token input, which is the actual last

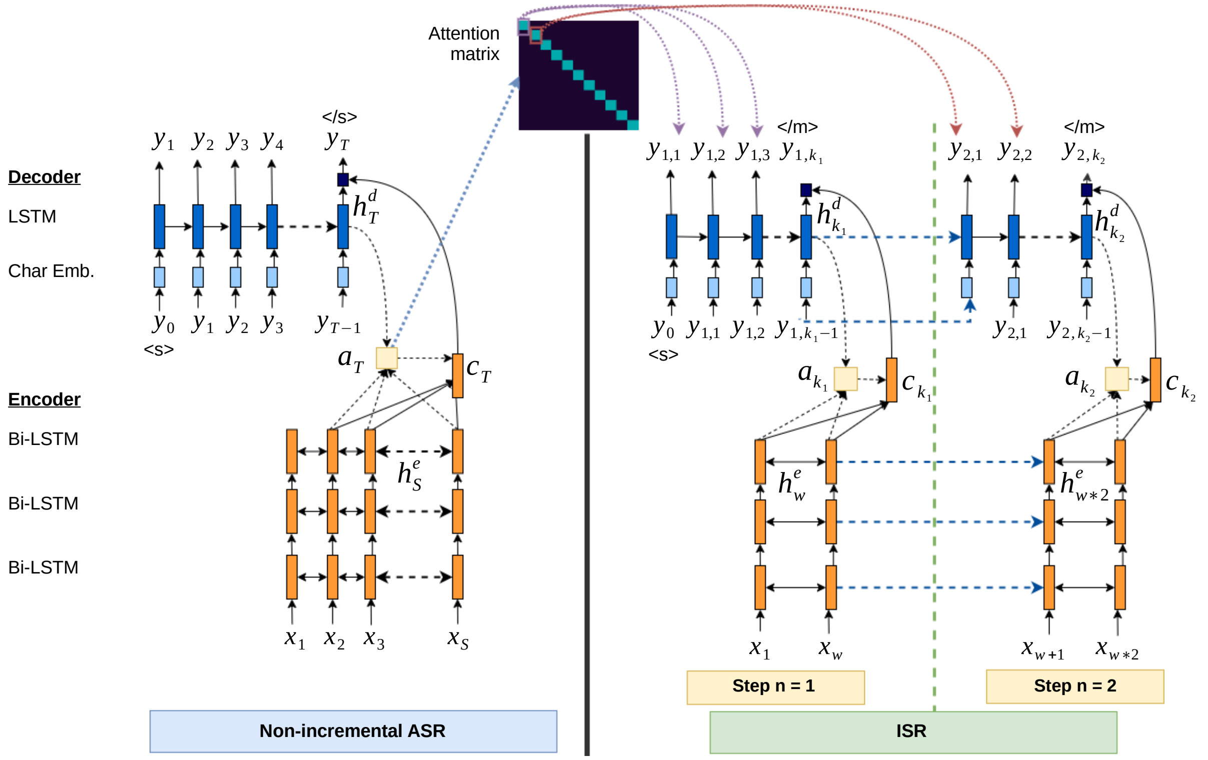

Fig. 2 Attention-transfer ISR training.

token from the previous step. When the decoder predicts </m> token, it marks that the text prediction from $\mathbf{X}_n$ has finished. The prediction of $\mathbf{Y}_n$ is done by attending only to $\mathbf{X}_n$. To avoid additional computation delay, we applied greedy decoding in our experiment.

The recognition in step $n + 1$ starts by shifting the input window w frames while keeping the model states from the previous step. If the encoder is set to take contextual input frames, the window shift equals the length of the main input frames. These incremental recognition processes are repeated until the step reaches the N-th speech segment or until an end-of-sentence token is predicted.

4.2 Training

We applied attention information transfer from a non-incremental ASR to train an AT-ISR, which mechanism is illustrated in Fig. 2. To enable short-segment-based speech recognition, AT-ISR is trained using $\mathbf{Y}_n$ that is followed by an end-of-block </m> as the target of $\mathbf{X}_n$. The $\mathbf{X}_n$ and $\mathbf{Y}_n$ pairs are decided based on alignments from the attention component of the non-incremental ASR, which acts as a teacher, during a teacher-forcing text generation. Here the alignments are generated once without using another system.

In the alignment, output token y_t at time t is aligned to s-th input frame x_s, which correspond to encoder state h^e_s. Speech frame index s which y_t aligns to (l_t) is chosen by following the monotonic condition:

$$l_t = \underset{l_{t-1} \leq s \leq l_{t+1}}{\text{argmax}} \; Score(h^e_s, h^d_t). \tag{9}$$

In the training data based on the obtained alignments, each transcription segment $\mathbf{Y}_n$ consists of the output tokens, where each token is sequentially aligned to one of the speech frames in $\mathbf{X}_n$. If states downsampling [25] is applied in the encoder, encoder state h^e_s will correspond to multiple speech frames, depending on the downsampling rate. The AT-ISR incremental unit or delay can be controlled by combining consecutive alignment units during the training. The shortest or basic incremental unit equal the number of speech frames that an encoder state represents in the attention alignment.

The transfer of attention information aims to make AT-ISR mimic the alignments by a non-incremental ASR. AT-ISR applies identical architecture as the non-incremental ASR. By priorly adding the special tokens in the non-incremental ASR output vocabulary, AT-ISR also can be initialized with the non-incremental ASR parameters. Attention transfer mechanism is only applied during AT-ISR training, therefore, AT-ISR inference is done without involving the teacher model.

4.3 Output Unit

In this work, we considered two types of ISR output representation units based on token granularity: characters and subwords. We did not consider whole-word units as our ISR output because word vocabulary is large so it is impractical when the ISR is utilized in the translation system. The word-level ISR also could not cope with the out-of-vocabulary condition.

4.3.1 Characters

Fig. 3(a) illustrates character-level ISR. It models an end-to-end relation between acoustic features and character se-

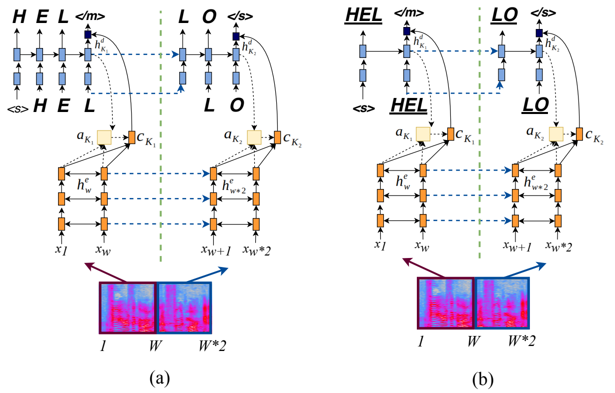

Fig. 3 End-to-end character level ISR (a) and subword-level ISR (b).

quences. Character unit is one of the basic token units that is commonly used in end-to-end ASR [21, 26, 27]. Since character-level representation enhances the ASR generality, it can prevent overfitting and out-of-vocabulary conditions [27].

In our experiment, character-level ISR's output vocabulary only included alphabet tokens and special tokens required for ISR. A special token that symbolizes whitespace was placed between character sequences that belong to different words so they could be segmented back into a word sequence.

4.3.2 Subwords

A subword-level ISR predicts a sequence of subwords, as shown in Fig. 3(b). Subword is a sequence of characters that is tokenized from a word. We can consider a word as a combination of one or several subwords. In terms of token granularity level, subwords have a coarser granularity than characters, but finer than words. The subwords that we discuss in this work are subwords that are generally used in the machine translation system. This kind of subword unit is also used in the recent end-to-end ASR systems [28, 29]. The utilization of subword unit in ASR is generally done to avoid out-of-vocabulary conditions, similar to character-level ASR, and also to keep a longer context of a word. Subword sequences of different words are separated with a whitespace token.

In this work, we constructed a subword vocabulary by training a word-to-subword segmentation model using a byte-pair-encoding (BPE) algorithm [30], which is implemented in the SentencePiece tokenizer toolkit [31]. Here the segmentation model is trained based on text sentences that consist of word tokens. The BPE algorithm first trains the model by initializing the subword vocabulary with a list of unique characters and converting each word from the training data into a character sequence. In the subsequent processes, the algorithm iteratively replaces the most frequent token pair in the training data with a new token, which merged from that token pair, and adds it to the vocabulary. The segmentation model construction is done using only text sentences, without depending on language and phonemes. In inference, given a word, the segmentation model converts the word into subwords by representing it as a character sequence and applying the merge operation learned by the model.

5. Experiment Settings

5.1 Dataset

We utilized the Wall Street Journal (*WSJ*) dataset [32] to train our basic non-incremental ASR and proposed ISR, and compared those systems with other speech recognition systems that used different frameworks. The *WSJ* dataset consists of multi-speaker speech utterances recorded by reading English news passages. We used the *SI-284* set as the training set, *dev93* as the development set, and *eval92* as the evaluation set. The *SI-284* set consists of 81 hours of speech. All models that were trained with the *WSJ* set were character-level models.

We utilized our proposed ISR in a speech translation task. Automatic real-time lecture translation is a challenging task that requires a real-time speech translation system. In this work, since we focus on ISR for lecture translation tasks with a less-restricted content domain, we used corpora that were collected from TED talks to create our ISR system and an NMT system for modular speech translation.

Data from TED talks consist of lecture speech and transcription that were presented at TED talks. The lectures covered various topic domains that were spoken by speakers with various speaking styles. Following this condition, and also since the speech originated from actual talks, the transcription and translation texts were written in a spoken language style, which is slightly different from a written language style. These conditions lead to ISR and NMT systems with highly diversified training examples in a matching language style.

We trained the ISR model using the *TED-LIUM release 1* dataset [33]. *TED-LIUM release 1* corpus consists of 118 hours of English speech data that were recorded from TED talks. This dataset was split into training, development, and evaluation sets based on the Kaldi recipe [34]. The acoustic features for the ISR input consisted of 80 dimensions of Mel-spectrogram with a 0.05 seconds window ($feat_{win}$) and 0.0125 seconds shift ($feat_{shift}$).

The NMT model was trained using English-French translation dataset from the *IWSLT 2017* shared task [35]. This dataset consists of English speech transcription and French translation texts from TED talks. We used the in-domain *IWSLT 2017* training set to train the model and *dev2010* as the development data. The translation evaluation was done based on *tst2010* set for the translation from the correct text and ISR text.

To minimize the dissimilarity between the ISR and NMT training materials, we removed the punctuation and normalized the numbers in the NMT training texts. The Unicode symbols in the English texts were also normalized into basic Latin alphabet letters due to *TED-LIUM release 1* text conditions that did not contain punctuation, numbers, and Unicode letters. The *TED-LIUM release 1* transcriptions contain speech fillers, unlike the NMT dataset. Therefore, we removed the fillers in the ISR output before passing it to NMT.

The NMT model in our experiment applied subword units as the input and output representation. Each input and output vocabulary consisted of 16,000 subwords. All the subword vocabularies were constructed using the BPE algorithm in the SentencePiece tokenizer based on the cleaned NMT training data in their respective languages. The English subword tokenizer was also utilized to tokenize the training texts for the subword-level ISR model.

5.2 Model Configuration

The following are the descriptions of the model configurations. In this work, we did not utilize external language model in any neural non-incremental ASR, ISR, and NMT models.

5.2.1 ISR

We used attention-based network with an encoder and a decoder [21–23] to construct our non-incremental ASR and ISR, and applied the similar structure to the character- and the subword-level models.

The encoder consisted of a feed-forward layer (512 units) followed by three stacked bidirectional long-short term memory neural network (BiLSTM) layers (256 units). The encoder applied state downsampling with a downsampling rate that equal eight states into one state. Consequently, the shortest alignment unit in the attention transfer for an output token was eight speech feature frames (0.14 seconds). In this work, we define eight speech frames as one block of speech features.

The decoder side consisted of an embedding layer (256 units), an LSTM layer (512 units), and a softmax layer. The embedding layer and the softmax layer sizes were configured according to the model output unit. In the attention component, we applied an attention mechanism with MLP-scoring based on multi-scale alignment and contextual history [36].

We evaluated the AT-ISR by comparing it to the teacher non-incremental ASR as the topline and also to the baseline ISR. The baseline ISR was an ISR that applied the same architecture as the topline and AT-ISR, but the incremental steps were taught based on the alignments generated with forced-alignment procedure [34, 37] by hidden Markov model and Gaussian mixture model (HMM-GMM) ASR [38], which is the standard alignment generation method that applied in neural transducer based on the HMM-GMM alignments. The baseline subword- and character-level alignments were obtained by aligning all the tokens (in their respective unit) of a word into speech segments where the word ends [8].

We also compared our AT-ISR to a neural ISR without an attention mechanism. The structure consisted of unidirectional LSTM network layers with a connectionist temporal classification (CTC) training objective and the optimal incremental output was determined through beam-searching with depth-pruning [39].

5.2.2 NMT

The NMT model was constructed by applying an encoder-decoder structure with an attention mechanism. The NMT encoder consists of an embedding layer (256 units), a feed-forward layer (512 units), and two BiLSTM layers (256 units). The decoder side consists of an embedding layer (512 units), two LSTM layers (512 units), and a softmax layer.

Table 1 Character-level speech recognition performance on *WSJ*: Average full speech utterance duration was 7.88 sec. (↓ = lower is better; *m* = main input frame block; *la* = look-ahead input frame block; 1 block = 8 frames = 0.14 sec.)

Model	CER ↓ (%)	WER ↓ (%)	UCR ↓ (%)
Topline: Non-incremental ASR			
CTC [40]	8.97	-	-
Att Enc-Dec Content [40]	11.08	-	-
Att Enc-Dec Location [40]	8.17	-	-
Joint CTC+Att (MTL) [40]	7.36	18.20	-
Att Enc-Dec (ours; AT-ISR teacher)	**6.26**	**16.49**	**7.69**
Baseline ISR: Att Enc-Dec ISR + HMM-GMM alignment			
delay = 0.24 sec (1 *m* + 1 *la*)	20.15	49.75	25.10
delay = 0.54 sec (1 *m* + 4 *la*)	11.95	30.77	16.93
Proposed ISR: AT-ISR			
delay = 0.24 sec (1 *m* + 1 *la*)	18.37	43.59	20.14
delay = 0.54 sec (1 *m* + 4 *la*)	**7.52**	**20.06**	**11.10**
Other existing ISR			
LSTM + CTC [39]	10.96	38.37	-

6. Experiments Result and Discussion

6.1 ISR Performance in Error Rates

We first evaluated our non-incremental ASR and ISR systems on basic character-level speech recognition task on *WSJ* dataset, which results is shown in Table 1. The average length of the full-utterance speech in this experiment was 7.88 seconds. The performance scores here are reported as the character error rate (CER). For our proposed AT-ISR and the baseline ISR, the reported delays are the input delay that corresponds to the size of the input window for an incremental step. Here the ISR computational delay was below 0.05 seconds; our non-incremental ASR computational delay averaged 0.3 seconds. Our non-incremental ASR, which is a standard encoder-decoder network with an attention mechanism, achieved the best performance.

By using our non-incremental ASR, we taught two AT-ISR models for incremental speech recognition with the input window size of 0.24 seconds and 0.54 seconds each. In each kind of input window, the main speech input segment was set to a block of speech frames, which consists of eight frames, to see the ISR performance with the basic incremental unit that implies the shortest delay that it could made. However, based on our exploration, ISR without a contextual input could not perform a reliable recognition [41]. So in addition to the main input, we used contextual input, which was look-ahead speech segment, to improve the ISR performance. We used look-ahead segment with a size of one or four speech frame blocks to keep the recognition delay low with a reliable performance[†]. Our experiment results show that AT-ISR resulted in a better performance than the baseline model and

[†] Parts of this work have been presented in [41]. The work here provides a more comprehensive and systematic description of the method, additional experiments related to translation task, and deeper analyses of the experiment results. We also updated our results on *WSJ* with our recent scores.

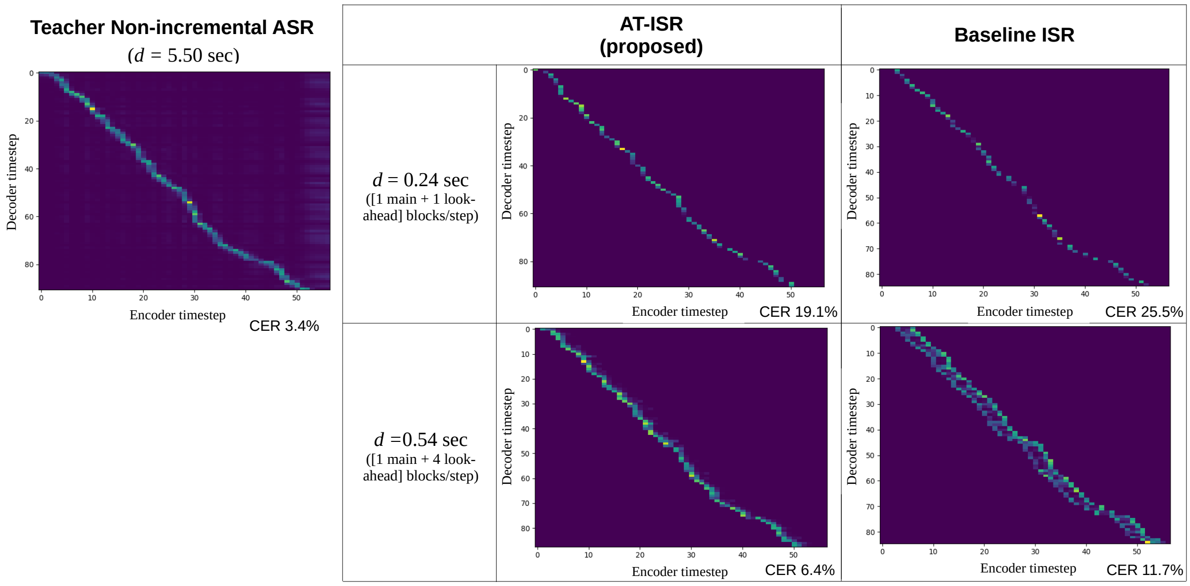

Fig. 4 Examples of attention matrix generated by a non-incremental ASR and ISRs during inference on a 5.50 sec-long speech from *WSJ eval92* set. In all ISR attention matrices presented, attention scores to speech segments that were not in the range of the input window were filled with zero. (d = delay; 1 block = 8 frames = 0.14 sec.)

a close CER to the teacher model.

We further examined the attention sequence of AT-ISR to see how it mimicked the teacher's attention alignments. Fig. 4 shows the attention matrices generated during the inference by teacher non-incremental ASR, AT-ISR, and baseline ISR. The ISRs attention matrices were normalized from the original matrices by padding the matrices. Since the ISRs here performed window-based recognition, these models did not attend to the speech segments that were not within the input window. Therefore, we filled each row with zero for the segments outside the input window in the corresponding incremental steps. Furthermore, before the normalization, the baseline's and AT-ISR's attention component has the highest attention score on the last speech block in the input window for each decoding step because the models anticipated the end-of-block token that marks the end of an incremental step. To simplify the comparison with the teacher model, we removed the attention score that corresponded to the last speech block in each incremental step. In all our experimental settings, the last speech block in an incremental step was the last block of the contextual input segment.

Fig. 4 shows that the AT-ISR attention sequences had a similar pattern to the teacher model's attention sequences. AT-ISR's attention concentrated most on the main speech segment for each incremental step, and it also attended the necessary contextual segment to improve the recognition. This figure shows that the attention-transfer training enabled the AT-ISR to mimic how its teacher attends the necessary information in the speech sequence to predict the transcription token. Attention-transfer training also resulted in an ISR with a cleaner and more monotonic attention sequence than the baseline method. Here the AT-ISR with a delay of 0.54 seconds resulted in the most similar attention pattern to the teacher and achieved the best score among the other ISR models. In the teacher's attention matrix, each text token scored high attention scores to several speech blocks consecutively, with an average of three consecutive speech blocks. As a result, the AT-ISR with a short input window size or delay might not receive enough information to predict the token sequences correctly. An example of such a condition can be seen in our ISR result with a delay of 0.24 seconds, in which the input window only consists of two speech blocks.

In the second experiment, we made a deeper analysis of the AT-ISR model by using *TED-LIUM release 1* dataset. The speech recognition performance on *TED-LIUM release 1* is shown in Table 2. We also performed statistical t-test to see the statistical difference between the ISR models with a significance level of 5%. The results are represented as symbols next to the performance score in Table 2. The performance comparison is done based on CER, word error rate (WER), and uncovered-word rate (UCR). An uncovered-word is one that does not exist in the training data because of one or several character-level mistakes in that word. For this reason, an uncovered-word could be a word that does not have linguistical meaning. A lower UCR implies a lower uncovered-word number and a better performance. The UCR of the correct transcription in the evaluation set was 1.55%. We set the AT-ISR input size to one and four main frame blocks, with the addition of two or four look-ahead blocks. The input size here was chosen to keep the output quality with a limited delay.

With the same amount of delay, AT-ISR WER and CER outperformed the baseline. The baseline ISR was better in producing semantically recognizable words, but it struggled to produce the correct words. The performance difference between the baseline ISR and AT-ISR might be caused by the

Table 2 Speech recognition performance on *TED-LIUM release 1*: Symbols in some scores indicate statistical significance test result with $p < 0.05$. (↓ = lower is better; ★ = significantly different from baseline with identical output units; ⋄ = not significantly different from baseline with identical output units; ● = significantly different from character-level model with identical delay and framework; † = not significantly different from character-level model with identical delay and framework; *e2e* = end-to-end model; *m* = main input frame block; *la* = look-ahead input frame block; 1 block = 8 frames = 0.14 sec.)

Model Output Unit (e2e)	**CER ↓ (%)**	**WER ↓ (%)**	**UCR ↓ (%)**
Topline: Teacher Non-incremental ASR (Att Enc-Dec) *delay* = 7.58 sec (avg.)			
Character	15.21	27.37	3.02
Subword	**13.35** ●	**23.98** ●	**0.54** ●
Baseline ISR: Att Enc-Dec ISR + HMM-GMM alignment *delay* = 0.84 sec (4 *m* + 4 *la*)			
Character	27.89	43.10	2.10
Subword	28.43 ●	39.77 ●	**0.37** ●
Proposed ISR: AT-ISR			
delay = 0.44 sec (1 *m* + 2 *la*)			
Character	24.65 ★	46.14 ★	9.95 ★
Subword	27.53 ⋄ ●	45.39 ★ †	0.54 ★ ●
delay = 0.54 sec (1 *m* + 4 *la*)			
Character	21.00 ★	41.10 ★	11.7 ★
Subword	21.28 ★ †	36.78 ★ ●	0.66 ★ ●
delay = 0.84 sec (4 *m* + 4 *la*)			
Character	16.22 ★	31.04 ★	5.19 ★
Subword	**15.20** ★ †	**28.26** ★ ●	1.04 ★ ●

precision difference in the ground alignments of both models. The baseline learned the alignments that were originally at the word-level. Here the precise alignments of character or subword units cannot be inferred, therefore, all units within a word were aligned into a speech segment where that word ends. It implies that some token alignments are delayed by several segments. As a result, if the speech segment window cannot include the necessary segments, the baseline ISR cannot produce the tokens that form correct words. On the other hand, the AT-ISR learns from more precise alignments based on its teacher's attention module, so it can immediately recognize the tokens from a speech segment without delaying it to the next segment. The baseline might have a better UCR because it learned to produce all tokens of a word in one recognition step.

Based on our result in Table 2, the subword-level model outperformed the character-level model in all non-incremental and incremental recognition tasks. Statistically, although the CERs of character- and subword-level AT-ISR models were not different, there were some differences in the WERs and the UCRs, especially in the models with a longer delay. Subword sequence is more reliable in forming correct words because it retains a longer word context than a character. During inference, character-level output might also contain more low-level errors than subwords. As a result, when the characters are concatenated into a word, the chance of forming an uncovered-word will be higher than the concatenation from subwords. In conclusion, subword-level AT-ISR improves speech recognition performance, especially in terms of WER and UCR. We will discuss how these affected the translation task in Section 6.3.

As we expected, non-incremental ASR's performance is better than ISR because the former is allowed to analyze the entire speech sequence. However, non-incremental ASR is not suitable for tasks that require simultaneous or real-time processing because it could cost a high delay. The amount of delay in ISR has to be configured carefully because the output quality might drop if the delay is too short. In this experiment, AT-ISR with a delay of 0.84 seconds achieved a close performance to that of non-incremental ASR that requires a delay of 7.58 seconds on average. It shows that AT-ISR with an appropriate delay could result in output with quality that is close to the non-incremental ASR.

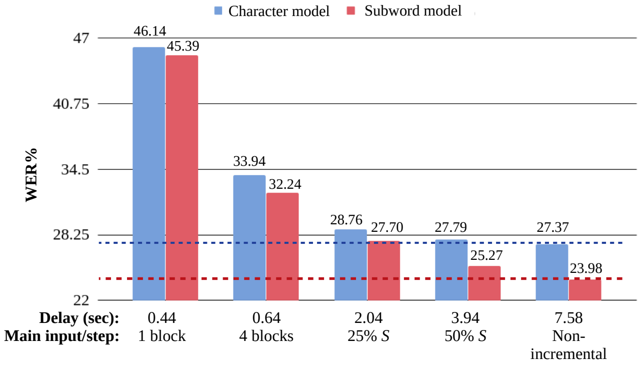

Fig. 5 WER of end-to-end character- and subword-level AT-ISR with various delays. (*S* = average speech utterance length in *TED-LIUM release 1* set (7.58 sec); 1 block = 8 frames = 0.14 sec.)

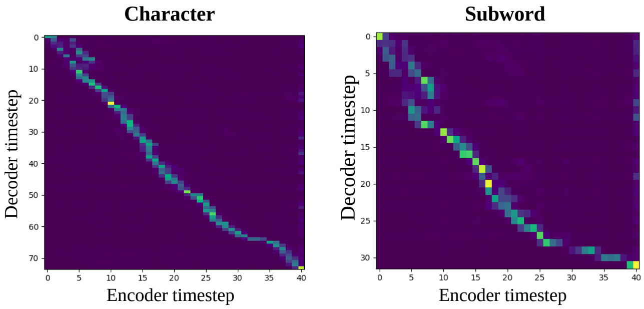

Fig. 6 Examples of attention matrix generated by the non-incremental ASR during inference. From these matrices, in AT-ISRs training, a text token is aligned to a speech segment, which corresponds to an encoder state, with the monotonically highest alignment score.

6.2 ISR Performance in Delay and Output Unit

Fig. 5 shows how the AT-ISR delay and output unit affected the speech recognition performance. Delays shown here are the size of the input segment of an incremental step. All AT-ISR models in this figure included two look-ahead blocks in addition to the main blocks in their input window. Here we made the size of the look-ahead segment shorter than those in Table 2 to limit the delay.

In speech recognition tasks, there is a trade-off between recognition delay and performance. It is shown in Fig. 5, where the AT-ISR WER decreases along with the increase of the delay. Since ISR with a short speech recognition delay and a close performance to the non-incremental ASR is more preferable, we need to find a delay configuration that

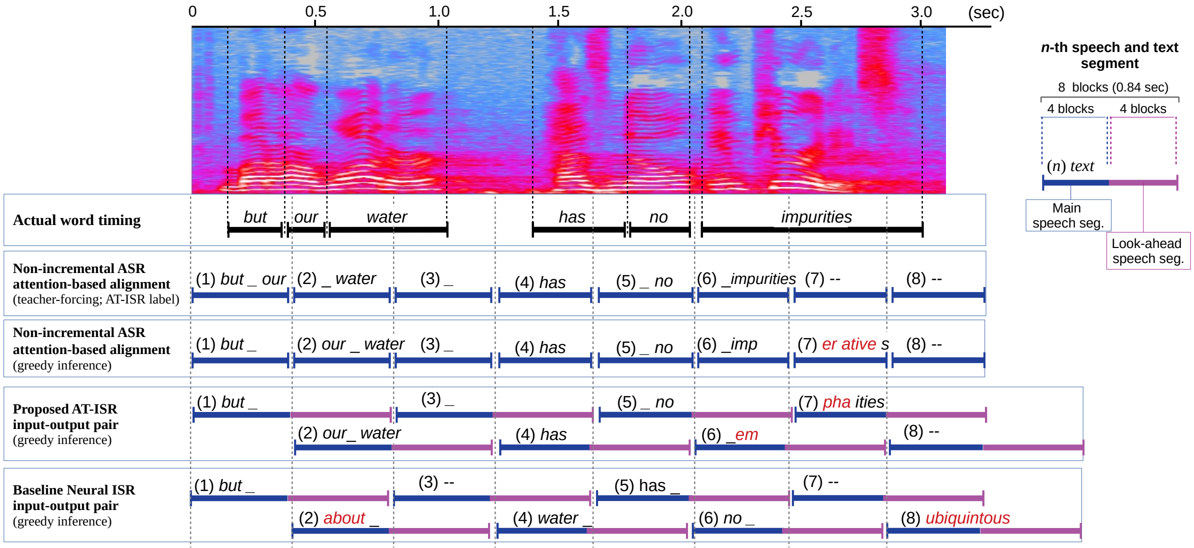

Fig. 7 Examples of speech and subword token alignment of a 3.1 sec speech based on non-incremental ASR and ISR (delay 0.84 sec) inference. The black-colored tokens are the correctly recognized tokens, whereas the red-colored tokens are the incorrect tokens. For the non-incremental ASR alignments, each subword is aligned to the speech part that scored the highest attention score. ('_' = whitespace token; '--' = no text output or only output '*</m>*' token; 1 block = 8 frames = 0.14 sec.)

keeps the balance between the recognition delay and performance. In our delay investigation here, we found that character-level AT-ISR performance improvement did not happen significantly between the following delays: 25% of utterance length, 50% of utterance length, and full-utterance length. Here when the recognition delay was equal to 2.04 seconds or 25% of the full-utterance length, it also started to achieving a comparable WER to the non-incremental ASR that took a full-utterance at once to generate the transcription. This result shows that the character-level model was able to retain the balance between recognition delay and performance when the delay was 2.04 seconds or 25% of the full-utterance length.

Interestingly, the subword-level models outperformed the character-level models in general, but the character-level AT-ISR achieved a closer performance to the teacher with a short delay than the subword-level model. In Fig. 5, when the AT-ISR delay was 25% of the full-utterance length, the WER difference between character-level student and teacher models was 1.38%. In our experiment, this model achieved CER 15.73%, which was higher only 0.52% than the teacher model. With an identical delay, subword-level AT-ISR WER was 3.7% higher than the teacher. In addition, its CER was 15.24% or 1.89% higher than the teacher. In the subsequent delays that we explored, the subword-level also had not shown the balance point between the speed and performance, unlike the character-level AT-ISR. The character-level AT-ISR is better at mimicking the teacher because the necessary information to predict a character token can be satisfied by a shorter speech segment than for predicting a subword token. Fig. 6 shows the examples of character-level and subword-level attention matrices that were generated using non-incremental ASR in the corresponding unit-level. It shows that the subword token scored a high score to several encoder states consecutively more than the character token. This is because a subword token consists of several characters. Therefore, the subword-level ISR's performance cannot approach the teacher's level when the input window cannot include or fails to reach the other speech segments with a high attention score.

Since a subword consists of several characters, the subword-level ISR requires a longer speech context than the character-level ISR. Theoretically, when the input segment is very short, the character-level ISR should be able to result in a better performance than the subword-level ISR. In our experiment, however, the subword-level ISR outperformed the character-level ISR in every delay that we tried. This is because our ISRs looked at look-ahead blocks when taking an input segment. In our data, a subword token consists of seven characters on average, and one speech frame block was aligned to two characters on average. Our shortest input window configuration utilized one main block with two look-ahead blocks, which contained information of six characters on average; it might have a similar amount of information as the character-level recognition. When the recognition delay was below 50% of the average utterance length, the WER difference between the character- and the subword-level ISRs was around 1%. So within that delay, the quality of both models was similar, although the subword-level ISR was slightly better.

Fig. 7 shows the examples of subword-level ISR output sequences that are aligned to the corresponding input speech segment. Many subword tokens resembled words due the size of our subword vocabulary. AT-ISR predicted the subwords well when the input window covered all the speech parts where the subwords were uttered. Mistakes occurred when a subword's speech duration exceeded the length of the window. In this picture, for example, token '*impurities*' was predicted by the AT-ISR as three subwords: ‘*em*’, ‘*pha*’, and ‘*ities*’. The AT-ISR split the prediction of this token into two incremental steps, causing the recognition in both steps to overlook some information and output inaccurate sequences. The non-incremental ASR also struggled to recognize this token correctly but it results in a word with the sound that resembles the target word since it had a longer information context. ISR improvement might be made by splitting a subword token with a long speech duration into several subword tokens when training the ISR.

6.3 Effect of ISR Delay and Output Unit in Speech Translation

We performed additional experiments by exploring the ISR delay and output units that might affect the translation. As developing incremental NMT is not part of this study, we evaluated the ISR using a standard NMT as a downstream task by connecting these systems.

NMT systems generally adopt subwords as the input and output representation [42–44]. Subword representation could avoid out-of-vocabulary condition, which often happens in the word-level model, and preserve the word’s context better than the character-level token. For these factors, our experiment focused on the subword-level translation and see how the ISR affects the translation performance. In this experiment, the AT-ISR models were the models that were trained with *TED-LIUM release 1* dataset.

Since translation by a subword-level NMT requires subword-level tokens as input, we unified the ISR tokens into subwords that can be recognized by the NMT system. We performed two approaches to unify and connect the ISR into a subword-level NMT, which approaches shown in Fig. 8. The first approach converted the character-level ISR output into subwords that were covered by the NMT input vocabulary. Since uniforming the ISR output and NMT input vocabularies with the best performance for both systems might be time-consuming, this approach is suitable when a subword-level ISR with the same output vocabulary as the NMT input vocabulary is unavailable. In this approach, when a character sequence from ISR forms a word, this word is segmented into subwords using a word-to-subword segmentation model. In our experiment, the word-to-subword segmentation model was the SentencePiece model that we used to tokenize the words in the NMT source language training data into subwords. The second approach was a direct ISR and NMT integration, where the ISR is a subword-level model with a matching output vocabulary with the NMT input vocabulary.

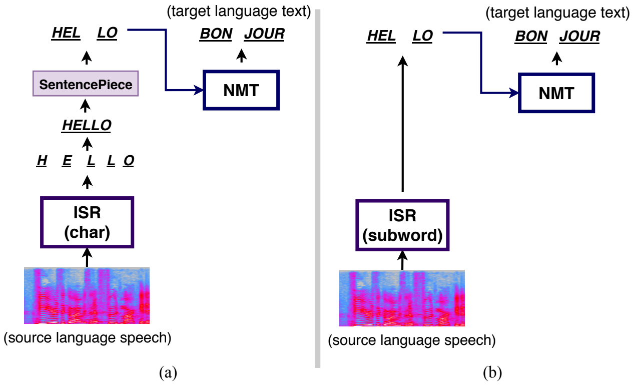

Fig. 8 ISR in speech translation task (*En-Fr*): (a) end-to-end character-level ISR is connected to NMT via character-to-subword conversion with SentencePiece tokenizer; (b) end-to-end subword-level ISR directly connected to subword-level NMT.

The speech recognition performance and translation quality on *tst2010* set can be seen in Table 3. We marked the results based on the statistical significance test result with a significance level of 5%. The translation quality was measured by 1-gram and 4-gram BLEU (bilingual evaluation understudy) [46] scores, NIST [47], TER (translation error rate), and METEOR (metric for evaluation of translation with explicit ordering) [48] scores of the translation result. BLEU score measures the position-independent n-gram word matches between the hypothesis and the reference. NIST evaluation metric is an alteration from BLEU, in which it gives more weight to the correct n-gram that is rare to occur. TER measures the minimum number of edits that are required to change the translation result so it exactly matches the reference, where the possible edits are insertion, deletion, and substitution. METEOR is an evaluation metric that calculates the score based on the harmonic mean of unigram precision and recall. Translation result with the higher BLEU, NIST, and METEOR scores represents a higher performance, while a lower TER is better. Since the ASR results contain errors, the translation quality degrades compared to the translation from the correct transcription. The low translation quality from the ISR output was caused by the nature of incremental recognition, in which the model is forced to produce outputs based on a short input segment. This situation affected the translation quality.

With the condition of low delay speech recognition, end-to-end subword-level AT-ISR resulted in the best speech recognition and translation performance. The translation result from the character-level AT-ISR output that was converted into subwords was less successful than the end-to-end subword-level AT-ISR due to its low speech recognition performance and error propagation. The translation quality from subword-level AT-ISR text also approached those of published system with an NMT, which the results comparison on *IWSLT 2015* English-French speech translation task [49] is shown in Table 4.

AT-ISR delay affected not only the speech recognition but also the translation performance; a higher delay resulted in a lower WER and a higher BLEU score. Fig. 9 shows how the AT-ISR delay and output adaptation approach affected the translation 4-gram BLEU score.

Interestingly, although the character-level AT-ISR and the subword-level AT-ISR might have a similar WER in with the same delay, the translation quality from the subword-level AT-ISR still outperformed the character-level AT-ISR. It can be seen in Fig. 9 at the point of AT-ISR delay of 50% of the total utterance length. In that condition, although the WERs of both AT-ISR systems were close, the ISR UCRs and BLEUs were significantly different; the best UCR and

Table 3 Speech recognition and English-French translation performance on *tst2010* set: Symbols in some scores indicate statistical significance test result with $p < 0.05$. (↓ = lower is better; ↑ = higher is better; *ch-sw (spm)* = character-level ASR with character-to-subword conversion using SentencePiece; *sw (e2e)* = end-to-end subword-level ASR; ⋆ = significantly different from baseline with identical output units; ⋄ = not significantly different from baseline with identical output units; • = significantly different from *ch-sw (spm)* with identical delays; † = not significantly different from *ch-sw (spm)* with identical delays; *m* = main input frame block; *la* = look-ahead input frame block; 1 block = 8 frames = 0.14 sec.)

ASR Output	**Speech Recognition**			**Translation**				
	CER ↓	**WER ↓**	**UCR ↓**	**BLEU1 ↑**	**BLEU4 ↑**	**NIST ↑**	**TER ↓**	**METEOR ↑**
Correct transcription	0.0	0.0	1.36	59.4	31.6	7.34	53.7	52.0
Topline: Non-incremental ASR (*delay* = 7.58 sec (avg.))								
ch-sw (spm)	15.11	26.75	2.67	47.1	21.1	5.37	69.1	39.4
sw (e2e)	**12.39** •	**22.43** •	**0.50** •	**50.0** •	**23.1** •	**5.80** •	**65.4** •	**42.2** •
Baseline ISR: Att Enc-Dec ISR + HMM-GMM alignment								
delay = 0.84 sec (4 *m* + 4 *la*)								
ch-sw (spm)	28.03	42.50	1.74	37.8	13.3	3.93	79.8	28.7
sw (e2e)	31.17 •	38.31 •	0.44 •	41.5 •	15.9 •	4.47 •	74.8 •	31.1 •
Proposed: AT-ISR								
delay = 0.54 sec (1 *m* + 4 *la*)								
ch-sw (spm)	21.56 ⋆	41.39 ⋄	10.07 ⋆	38.0⋄	13.5⋄	4.00 ⋄	80.9 ⋆	29.8 ⋄
sw (e2e)	**21.52** ⋆ †	**36.56** ⋆ •	**0.60** ⋆ •	**42.6** ⋆ •	**16.3** ⋆ •	**4.66** ⋆ •	**74.3** ⋆ •	**33.4** ⋆ •
delay = 0.84 sec (4 *m* + 4 *la*)								
ch-sw (spm)	19.18 ⋆	33.09 ⋆	4.45 ⋆	44.0 ⋆	17.9 ⋆	4.87 ⋆	72.8 ⋆	34.8 ⋆
sw (e2e)	**15.71** ⋆ •	**28.17** ⋆ •	**0.86** ⋆ •	**47.2** ⋆ •	**20.6** ⋆ •	**5.38** ⋆ •	**68.7** ⋆ •	**39.1** ⋆ •

Table 4 Speech translation performance of our system and other works on *IWSLT 2015* English-French translation task tested on *tst2015*. (↓ = the lower the better; ↑ = the higher the better)

System	**Performance**	
	BLEU4 ↑	**TER ↓**
Official IWSLT system result [45]	16.98	80.4
Ours with correct source text (topline)	31.41	54.5
Ours with non-incremental ASR	19.95	72.5
Ours with AT-ISR (proposed)	16.67	75.1

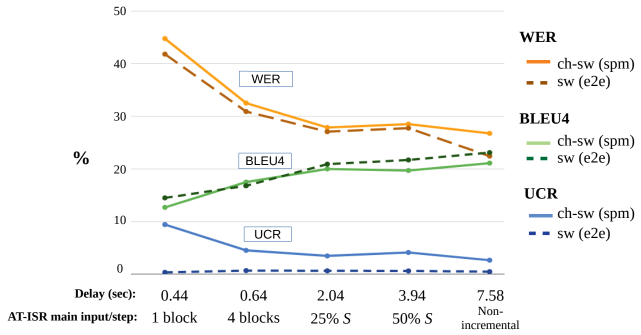

Fig. 9 The effect of AT-ISR delay on the speech translation task. The evaluation was done on *tst2010* set. ('*ch-sw (spm)*' = character-level ASR with character-to-subword conversion using SentencePiece; '*sw (e2e)*' = end-to-end subword-level ASR; *S* = average speech utterance length in *tst2010* set (7.58 sec); 1 block = 8 frames = 0.14 sec.)

BLEU scores were achieved by the end-to-end subword-level AT-ISR. A similar trend was observed in the other translation metrics. It implies that speech translation quality is not only affected by the WER but also by the UCR that represents the number of words that do not exist in the NMT word vocabulary. Since language translation is depends on word semantics, ISR token sequences that did not resemble a word with meaning, which might not appear in NMT training material, could decrease the speech translation performance. Subwords maintains a longer context of a word than characters, so it could resulted in better translation than the character-level model.

7. Related Works

ISR system is a necessary component in a modular real-time speech translation system. The HMM-based ASR [38, 50], the conventional ASR approach, could perform real-time speech recognition because it recognizes the speech incrementally. The HMM-based ASR, however, cannot perform end-to-end recognition, which is the current state-of-the-art approach, although the prediction accuracy could be better than end-to-end systems. End-to-end ASR [21, 22, 28] use attention-based encoder-decoder architecture to do the recognition by combining the acoustic, lexicon, and language model components in the conventional ASR into a single neural network model. Previously, several neural network frameworks that can be applied for ISR tasks were proposed. Hwang et al. [39] proposed a neural ISR that uses unidirectional LSTM with CTC training objective and beam-search mechanism with depth-pruning. Jaitly et al. [7] proposed a neural transducer framework that consists of an attention-based structure. This framework recognizes the speech segment-by-segment with a fixed window. Segment-based speech recognition is achieved by learning the alignment during the training phase. In the original work [7], the alignment can be either generated by an external system, such as HMM-GMM ASR, or with the neural transducer itself by computing and updating the approximately best alignment several times through dynamic programming type of methods. But, as we mentioned earlier, since the neural transducer is required to compute and update the best alignment within the segment in order to lean the incremental recognition, this framework becomes more complicated than the standard attention-based ASR.

After finishing our experiment, we noticed recent papers that also introduced approaches for a neural incremental ASR. For example, Inaguma et al. (2020) proposed an ISR based on a seq2seq model with monotonic chunkwise attention, whose model learns from the alignment taken from a hybrid ASR model [9]. ISR frameworks were also recently proposed that utilize a recurrent neural network transducer (RNN-T) and a frame-synchronous model [10, 11]. An ISR model with an RNN-T consists of an RNN-T encoder, an RNN-T decoder, and a standard attention-based seq2seq rescorer. ISRs based on a neural transformer with CTC were also proposed in which the model did incremental recognition by segmenting the input [12] or by limiting the attention range [13]. The related ISR works generally focused on ISR for mobile-based applications, and utilization on a speech translation task remains uninvestigated. As we mentioned above, the recent ISR frameworks also require a more complicated structure than the standard non-incremental ASR. In our framework, we tackled this problem by tuning the non-incremental ASR for incremental recognition tasks by learning from its attention alignment, allowing an ISR with a simple mechanism and a close performance to the non-incremental recognition task. In this work, we demonstrated it using an LSTM-based seq2seq model. Our approach can also be applied to other attention-based neural network structures, such as Transformer, which we postpone to future work because here we are focusing on an attention transfer mechanism between a teacher and a student model with identical structure to build a simple ISR.

ISR-MT or ASR-MT integration is a challenging problem due to error propagation and the incompatibility of training materials in both modules. By using non-incremental systems, several studies addressed this challenge by adapting the ASR output to MT [51, 52]. Wang et al. [53] previously constructed a real-time system prototype by unifying an HMM-based ASR system and an online MT system [54]. Recently, Ren et al. proposed an end-to-end simultaneous speech translation with wait-*k* strategy [55]. Compared to the number of ISR studies, the study about the utilization of neural ISR in speech translation remains limited.

8. Conclusion

In this work, we constructed neural ISR for low-delay end-to-end speech recognition. We proposed attention-transfer ISR (AT-ISR) that learns attention knowledge from its teacher neural non-incremental ASR and adopts the teacher's structure. Low delay speech recognition is followed by a trade-off between delay and performance. Our character-level AT-ISR showed a comparable performance to the non-incremental ASR when the delay equals to or more than 25% of the total utterance length, therefore, we could use this configuration for this model to keep the balance between speech recognition latency and performance. ISR's performance closeness to the teacher depends on the granularity of the output unit. When the output unit has a coarse granularity, such as subword, it might result in higher recognition performance than the model with finer output unit granularity, such as character. On the other side, ISR with a fine-granulated output unit is faster to achieve a teacher-like performance than ISR with a coarse-granulated output unit. In the downstream task with NMT, our end-to-end subword-level ISR resulted in the best translation quality with the lowest WER and the lowest uncovered-word rate.

For future work, we are interested in the exploration to improve ISR and construct a full-fledged real-time S2ST system.

Acknowledgment

Part of this work was supported by JSPS KAKENHI Grant Numbers JP21H05054 and JP21H03467.

References

[1] S. Nakamura, "Overcoming the language barrier with speech translation technology," Science & Technology Trends - Quarterly Review No.31, April 2009.
[2] E. Matusov, A. Mauser, and H. Ney, "Automatic sentence segmentation and punctuation prediction for spoken language translation," Proc. IWSLT, Kyoto, Japan, pp.158–165, 2006.
[3] M. Pouget, O. Nahorna, T. Hueber, and G. Bailly, "Adaptive latency for part-of-speech tagging in incremental text-to-speech synthesis," Proc. Interspeech, pp.2846–2850, 2016.
[4] T. Kano, S. Sakti, and S. Nakamura, "Structured-based curriculum learning for end-to-end English-Japanese speech translation," Proc. Interspeech, pp.2630–2634, 2017.
[5] L. Cross Vila, C. Escolano, J.A.R. Fonollosa, and M. R. Costa-jussà, "End-to-End Speech Translation with the Transformer," Proc. IberSPEECH, pp.60–63, 2018.
[6] Y. Liu, H. Xiong, J. Zhang, Z. He, H. Wu, H. Wang, and C. Zong, "End-to-end speech translation with knowledge distillation," Proc. Interspeech, pp.1128–1132, 2019.
[7] N. Jaitly, Q.V. Le, O. Vinyals, I. Sutskever, D. Sussillo, and S. Bengio, "An online sequence-to-sequence model using partial conditioning," Proc. NIPS, 2016.
[8] T.N. Sainath, C.C. Chiu, R. Prabhavalkar, A. Kannan, Y. Wu, P. Nguyen, and Z. Chen, "Improving the performance of online neural transducer models," Proc. ICASSP, pp.5864–5868, 2018.
[9] H. Inaguma, Y. Gaur, L. Lu, J. Li, and Y. Gong, "Minimum latency training strategies for streaming sequence-to-sequence ASR," Proc. ICASSP, pp.6064–6068, 2020.
[10] T. Sainath, Y. He, B. Li, A. Narayanan, R. Pang, A. Bruguier, S.y. Chang, W. Li, R. Alvarez, Z. Chen, C.C. Chiu, D. Garcia, A. Gruenstein, K. Hu, A. Kannan, Q. Liang, I. McGraw, C. Peyser, R. Prabhavalkar, and D. Zhao, "A streaming on-device end-to-end model surpassing server-side conventional model quality and latency," Proc. ICASSP, pp.6059–6063, 2020.
[11] B. Li, S.y. Chang, T.N. Sainath, R. Pang, Y. He, T. Strohman, and Y. Wu, "Towards fast and accurate streaming end-to-end ASR," Proc. ICASSP, pp.6069–6073, 2020.
[12] H. Miao, G. Cheng, C. Gao, P. Zhang, and Y. Yan, "Transformer-based online CTC/attention end-to-end speech recognition architecture," Proc. ICASSP, pp.6084–6088, 2020.
[13] N. Moritz, T. Hori, and J. Le, "Streaming automatic speech recognition with the transformer model," Proc. ICASSP, pp.6074–6078, 2020.
[14] S. Yagi, "Studying style in simultaneous interpretation," Meta: Translators' Journal, vol.45, no.3, pp.520–547, 2000.
[15] C. Fügen, A.H. Waibel, and M. Kolss, "Simultaneous translation of lectures and speeches," Machine Translation, vol.21, pp.209–252, 2007.
[16] H.C. Barik, A study of simultaneous interpretation, 1969.
[17] M. Lederer, Language Interpretation and Communication, ch. Si-

multaneous Interpretation — Units of Meaning and other Features, pp.323–332, Springer US, Boston, MA, 1978.

[18] B. Ramabhadran, J. Huang, and M.Picheny, "Towards automatic transcription of large spoken archives - English ASR for the MALACH project," Proc. ICASSP, pp.I–I, 2003.

[19] J. Yuan, M. Liberman, and C. Cieri, "Towards an integrated understanding of speaking rate in conversation," Proc. Interspeech, 2006.

[20] D. Gile, "Methodological aspects of interpretation (and translation) research," Target–International Journal of Translation Studies, vol.3, pp.153–174, 1991.

[21] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," Proc. ICASSP, pp.4960–4964, 2016.

[22] J.K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," Proc. NIPS, pp.577–585, 2015.

[23] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," Proc. ICLR, 2015.

[24] M.T. Luong, H. Pham, and C.D. Manning, "Effective approaches to attention-based neural machine translation," Proc. EMNLP, pp.1412–1421, 2015.

[25] A. Graves, "Supervised sequence labelling," in Supervised sequence labelling with recurrent neural networks, pp.5–13, Springer, 2012.

[26] T. Ochiai, S. Watanabe, T. Hori, and J.R. Hershey, "Multichannel end-to-end speech recognition," Proc. ICML, pp.2632–2641, 2017.

[27] S. Ueno, H. Inaguma, M. Mimura, and T. Kawahara, "Acoustic-to-word attention-based model complemented with character-level CTC-based model," Proc. ICASSP, pp.5804–5808, 2018.

[28] C.C. Chiu, T.N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R.J. Weiss, K. Rao, E. Gonina, *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models," Proc. ICASSP, pp.4774–4778, 2018.

[29] K. Irie, R. Prabhavalkar, A. Kannan, A. Bruguier, D. Rybach, and P. Nguyen, "On the Choice of Modeling Unit for Sequence-to-Sequence Speech Recognition," Proc. Interspeech, pp.3800–3804, 2019.

[30] R.Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," Proc. ACL, pp.1715–1725, 2015.

[31] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," Proc. EMNLP, pp.66–71, 2018.

[32] D.B. Paul and J.M. Baker, "The design for the Wall Street Journal-based CSR corpus," Proc. HLT, pp.357–362, 1992.

[33] A. Rousseau, P. Deléglise, and Y. Estève, "TED-LIUM: an automatic speech recognition dedicated corpus," Proc. LREC, pp.125–129, 2012.

[34] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," Proc. ASRU, 2011.

[35] M. Cettolo, M. Federico, L. Bentivogli, N. Jan, S. Sebastian, S. Katsuitho, Y. Koichiro, and F. Christian, "Overview of the IWSLT 2017 evaluation campaign," Proc. IWSLT, pp.2–14, 2017.

[36] A. Tjandra, S. Sakti, and S. Nakamura, "Multi-scale alignment and contextual history for attention mechanism in sequence-to-sequence model," Proc. SLT, pp.648–655, 2018.

[37] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal Forced Aligner: Trainable text-speech alignment using Kaldi," Proc. Interspeech, pp.498–502, 2017.

[38] M. Gales, S. Young, *et al.*, "The application of hidden Markov models in speech recognition," Foundations and Trends in Signal Processing, vol.1, no.3, pp.195–304, 2008.

[39] K. Hwang and W. Sung, "Character-level incremental speech recognition with recurrent neural networks," Proc. ICASSP, pp.5335–5339, 2016.

[40] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multitask learning," Proc. ICASSP, pp.4835–4839, 2017.

[41] S. Novitasari, A. Tjandra, S. Sakti, and S. Nakamura, "Sequence-to-sequence learning via attention transfer for incremental speech recognition," Proc. Interspeech, pp.3835–3839, 2019.

[42] Y. Wu, M. Schuster, Z. Chen, Q. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean, "Google's neural machine translation system: Bridging the gap between human and machine translation," CoRR, vol.abs/1609.08144, 2016.

[43] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," Proc. ACL, pp.66–75, 2018.

[44] T. Banerjee and P. Bhattacharyya, "Meaningless yet meaningful: Morphology grounded subword-level NMT," Proc. SCLeM, pp.55–60, 2018.

[45] M. Garcia-Martinez, L. Barrault, A. Rousseau, P. Deléglise, and Y. Estève, "The LIUM ASR and SLT systems for IWSLT 2015," Proc. IWSLT, 2015.

[46] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "BLEU: A method for automatic evaluation of machine translation," Proc. ACL, pp.311–318, 2002.

[47] G. Doddington, "Automatic evaluation of machine translation quality using N-gram co-occurrence statistics," Proc. HLT, pp.138–145, 2002.

[48] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," Proc. ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pp.65–72, 2005.

[49] M. Cettolo, J. Niehues, S. Stüker, L. Bentivogli, R. Cattoni, and M. Federico, "The IWSLT 2015 evaluation campaign," Proc. IWSLT, 2015.

[50] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," Proceedings of the IEEE, vol.77, no.2, pp.257–286, Feb 1989.

[51] D. Dechelotte, H. Schwenk, G. Adda, and J. Gauvain, "Improved machine translation of speech-to-text outputs," Proc. Interspeech, pp.2441–2444, 2007.

[52] E. Matusov and H. Ney, "Lattice-based ASR-MT interface for speech translation," IEEE Transactions on Audio, Speech, and Language Processing, vol.19, no.4, pp.721–732, May 2011.

[53] X. Wang, A. Finch, M. Utiyama, and E. Sumita, "A prototype automatic simultaneous interpretation system," Proc. COLING, pp.30–34, 2016.

[54] X. Wang, A. Finch, M. Utiyama, and E. Sumita, "An efficient and effective online sentence segmenter for simultaneous interpretation," Proc. WAT, pp.139–148, 2016.

[55] Y. Ren, J. Liu, X. Tan, C. Zhang, T. QIN, Z. Zhao, and T.Y. Liu, "SimulSpeech: End-to-end simultaneous speech to text translation," Proc. ACL, pp.3787–3796, 2020.

Sashi Novitasari received her B.S. degree in Informatics (cum laude) from Bandung Institute of Technology, Indonesia, in 2018. She continued her studies at the Graduate School of Science and Technology, Nara Institute of Science and Technology, Japan, and received her M.E. in 2020. She is currently taking a doctoral course at Nara Institute of Science and Technology, Japan. She is a recipient of the Japanese Ministry of Education, Culture, Sport, Science, and Technology (MEXT) scholarship. Her research interests include speech recognition and spoken language translation systems.

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). In 2009-2011, she served as a visiting professor of Computer Science Department, University of Indonesia (UI), Indonesia. In 2011-2017, she was an assistant professor at the Augmented Human Communication Laboratory, NAIST, Japan. She served also as a visiting scientific researcher of INRIA Paris-Rocquencourt, France, in 2015-2016, under JSPS Strategic Young Researcher Overseas Visits Program for Accelerating Brain Circulation. Currently, she is a research associate professor at NAIST, as well as a research scientist at RIKEN, Center for Advanced Intelligent Project AIP, Japan. She is a member of JNS, SFN, ASJ, ISCA, IEICE and IEEE. She is also the officer of ELRA/ISCA Special Interest Group on Under-resourced Languages (SIGUL) and a board Member of Spoken Language Technologies for Under-Resourced Languages (SLTU). Her research interests include statistical pattern recognition, graphical modeling framework, deep learning, multilingual speech recognition and synthesis, spoken language translation, affective dialog system, and cognitive communication.

Satoshi Nakamura is Professor of Graduate School of Science and Technology, Nara Institute of Science and Technology, Japan, Project Leader of Tourism Information Analytics Team of RIKEN, Center for Advanced Intelligence Project AIP, 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 Zampolli 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.