SEQUENCE-LEVEL CONSISTENCY TRAINING FOR SEMI-SUPERVISED END-TO-END AUTOMATIC SPEECH RECOGNITION

Ryo Masumura, Mana Ihori, Akihiko Takashima, Takafumi Moriya, Atsushi Ando, Yusuke Shinohara

NTT Media Intelligence Laboratories, NTT Corporation, Japan

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

ABSTRACT

This paper presents a novel semi-supervised end-to-end automatic speech recognition (ASR) method that employs consistency training with the use of unlabeled data. In consistency training, unlabeled data can be utilized for constraining a model such that it becomes invariant to small deformation. In fact, considering consistency can make the model robust to a variety of input examples. While previous studies have applied consistency training to primitive classification problems, no studies have employed consistency training to tackle sequence-to-sequence generation problems including end-to-end ASR. One problem is that existing consistency training schemes cannot take sequence-level generation consistency into consideration. In this paper, we propose a sequence-level consistency training scheme specialized to handle sequence-to-sequence generation problems. Our key idea is to consider the consistency of the generation function by utilizing beam search decoding results. For semi-supervised learning, we adopt Transformer as the end-to-end ASR model, and SpecAugment as the deformation function in consistency training. Our experiments show that our semi-supervised learning proposal with sequence-level consistency training can efficiently improve ASR performance using unlabeled speech data.

Index Terms— semi-supervised learning, end-to-end automatic speech recognition, sequence-level consistency training, Transformer, SpecAugment

1. INTRODUCTION

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

Recent studies have introduced various modeling methods including connectionist temporal classification [1, 2], recurrent neural aligner [3], recurrent neural network (RNN) transducer [4], and RNN encoder-decoder [5–8]. In particular, Transformer-based modeling methods have shown the most powerful performance in recent studies [9–13]. In addition, a couple of effective training techniques specific to sequence-to-sequence learning have been presented [14]. In particular, SpecAugment, which augments input acoustic feature representations through time warping, time masking, and frequency masking, has yielded significant performance improvements [15].

One main problem with end-to-end ASR systems is that labeled data (speech-to-text paired data) is essential to carry out end-to-end optimization. However, it is difficult to form large labeled data sets in practical use cases. To mitigate this problem, semi-supervised learning that utilizes not only labeled data but also unlabeled data is being examined. The currently dominant semi-supervised learning approaches use text-to-speech networks or sequential auto-encoder networks [16–21]. These additional networks are often utilized for considering speech chain modeling [16], back-translation modeling [17,18], reconstruction modeling [19–21] and so on. However, these methods demand the joint construction of additional networks with a main end-to-end ASR model. This often creates difficulties in semi-supervised learning.

In order to achieve much simpler semi-supervised learning, we focus on consistency training [22–25]. The main strategy of consistency training is to constrain the classification model so that it becomes invariant to small deformation. In semi-supervised learning, the model is trained so that the classification results for an unlabeled input are constant even if small deformation is added to the input. It is known that considering consistency can create a model that is robust to various input examples. While above semi-supervised learning methods require additional modules, consistency training dispenses with them other than a main classification model. While previous studies have applied consistency training to primitive classification problems such as image classification and text categorization, no study has introduced consistency training to sequence-to-sequence generation problems including end-to-end ASR. In fact, existing consistency training schemes cannot take sequence-level generation consistency into consideration.

In this paper, we propose sequence-level consistency training specific to handling sequence-to-sequence generation problems. Sequence-level consistency training is inspired by sequence-level knowledge distillation, a teacher-student learning method for sequence-to-sequence generation problems [26–28]. Unlike conventional consistency training, sequence-level consistency training can consider the consistency of the generation function. In our semi-supervised learning proposal, we first construct a teacher end-to-end ASR model from labeled data. Next, we train a student end-to-end ASR model from both the labeled data and unlabeled data. We train the student end-to-end ASR model so that generation results of the unlabeled data using the student model are consistent with those using the teacher model even if the data are slightly deformed. The generation consistency can be taken into consideration by utilizing beam search decoding results of the unlabeled data. Our semi-supervised learning can be leveraged in two applications. One is unsupervised data augmentation where the target domain data consists of both labeled and unlabeled data [25]. The other is unsupervised domain adaptation where labeled source domain data and unlabeled target domain data are used [29]. In this paper, we adopt Transformer as the end-to-end ASR model, and SpecAugment as the deformation function in consistency training.

Experiments on a corpus of spontaneous Japanese [30] show that our proposal, semi-supervised learning based on sequence-level

consistency training, can improve ASR performance using unlabeled data in both unsupervised data augmentation and unsupervised domain adaptation conditions.

2. TRANSFORMER-BASED END-TO-END AUTOMATIC SPEECH RECOGNITION MODEL

This section briefly describes end-to-end ASR using a Transformer based auto-regressive generative model [9–13]. This model predicts the generation probability of text $\boldsymbol{W} = \{w_1, \cdots, w_N\}$ given speech $\boldsymbol{X} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_M\}$, where w_n is the n-th token in the text and $\boldsymbol{x}_m$ is the m-th acoustic feature in the speech. N is the number of tokens in the text and M is the number of acoustic features in the speech. Auto-regressive generative models define the generation probability of $\boldsymbol{W}$ as

$$P(\boldsymbol{W}|\boldsymbol{X};\boldsymbol{\Theta}) = \prod_{n=1}^{N} P(w_n|\boldsymbol{W}_{1:n-1}, \boldsymbol{X};\boldsymbol{\Theta}), \quad (1)$$

where $\boldsymbol{\Theta}$ represents the model parameter sets and $\boldsymbol{W}_{1:n-1} = \{w_1, \cdots, w_{n-1}\}$.

2.1. Network structure

In our Transformer-based end-to-end ASR model, $P(w_n|\boldsymbol{W}_{1:n-1}\boldsymbol{X};\boldsymbol{\Theta})$ can be computed using a speech encoder and a text decoder, both of which are composed of a couple of Transformer blocks. The model parameter sets are split into those for the speech encoder, $\boldsymbol{\theta}_{\texttt{enc}}$, and those for the text encoder, $\boldsymbol{\theta}_{\texttt{dec}}$.

Speech encoder: The speech encoder converts input acoustic features into hidden representations $\boldsymbol{H}^{(I)}$ using I Transformer encoder blocks. The i-th Transformer encoder block composes i-th hidden representations $\boldsymbol{H}^{(i)}$ from the lower layer inputs $\boldsymbol{H}^{(i-1)}$ as indicated by

$$\boldsymbol{H}^{(i)} = \texttt{TransformerEncoderBlock}(\boldsymbol{H}^{(i-1)};\boldsymbol{\theta}_{\texttt{enc}}), \quad (2)$$

where $\texttt{TransformerEncoderBlock}()$ is a Transformer encoder block that consists of a scaled dot product multi-head self-attention layer and a position-wise feed-forward network [9]. The hidden representation $\boldsymbol{H}^{(0)} = \{\boldsymbol{h}_1^{(0)}, \cdots, \boldsymbol{h}_{M'}^{(0)}\}$ is produced by

$$\boldsymbol{h}_{m'}^{(0)} = \texttt{AddPostionalEncoding}(\boldsymbol{h}_{m'}), \quad (3)$$

where $\texttt{AddPositionalEncoding}()$ is a function that adds a continuous vector in which position information is embedded. $\boldsymbol{H} = \{\boldsymbol{h}_1, \cdots, \boldsymbol{h}_{M'}\}$ is produced by

$$\boldsymbol{H} = \texttt{ConvolutionPooling}(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_M;\boldsymbol{\theta}_{\texttt{enc}}), \quad (4)$$

where $\texttt{ConvolutionPooling}()$ is a function composed of convolution layers and pooling layers. M' is the subsampled sequence length, which depends on the function.

Text decoder: The text decoder computes the generation probability of a token from preceding tokens and the hidden representations of the speech. The predicted probabilities of the n-th token w_n are calculated as

$$P(w_n|\boldsymbol{W}_{1:n-1}, \boldsymbol{X};\boldsymbol{\Theta}) = \texttt{Softmax}(\boldsymbol{u}_{n-1}^{(J)};\boldsymbol{\theta}_{\texttt{dec}}), \quad (5)$$

where $\texttt{Softmax}()$ is a softmax layer with a linear transformation. The input hidden vector $\boldsymbol{u}_{n-1}^{(J)}$ is computed from J Transformer decoder blocks. The j-th Transformer decoder block composes j-th hidden representation $\boldsymbol{u}_{n-1}^{(j)}$ from the lower layer inputs $\boldsymbol{U}_{1:n-1}^{(j-1)} = \{\boldsymbol{u}_1^{(j-1)}, \cdots, \boldsymbol{u}_{n-1}^{(j-1)}\}$ as

$$\boldsymbol{u}_{n-1}^{(j)} = \texttt{TransformerDecoderBlock}(\boldsymbol{U}_{1:n-1}^{(j-1)}, \boldsymbol{H}^{(I)};\boldsymbol{\theta}_{\texttt{dec}}), \quad (6)$$

where $\texttt{TransformerDecoderBlock}()$ is a Transformer decoder block that consists of a scaled dot product multi-head self-attention layer, a scaled dot product multi-head source-target attention layer, and a position-wise feed-forward network [9]. The hidden representation $\boldsymbol{U}_{1:n-1}^{(0)} = \{\boldsymbol{u}_1^{(0)}, \cdots, \boldsymbol{u}_{n-1}^{(0)}\}$ is produced by

$$\boldsymbol{u}_{n-1}^{(0)} = \texttt{AddPositionalEncoding}(\boldsymbol{w}_{n-1}), \quad (7)$$

$$\boldsymbol{w}_{n-1} = \texttt{Embedding}(w_{n-1};\boldsymbol{\theta}_{\texttt{dec}}), \quad (8)$$

where $\texttt{Embedding}()$ is a linear layer that embeds an input token into a continuous vector.

2.2. Supervised learning

In end-to-end ASR, a model parameter set can be optimized from the utterance-level labeled data (speech-to-text paired data) as

$$\mathcal{D}_{\texttt{pair}} = \{(\boldsymbol{X}^1, \boldsymbol{W}^1), \cdots, (\boldsymbol{X}^T, \boldsymbol{W}^T)\}, \quad (9)$$

where T is the number of utterances in the training data set. The objective function based on maximum likelihood estimation is defined as

$$\mathcal{L}_{\texttt{mle}}(\boldsymbol{\theta}_{\texttt{enc}}, \boldsymbol{\theta}_{\texttt{dec}}) = -\sum_{t=1}^{T}\sum_{n=1}^{N^t} \log P(w_n^t|\boldsymbol{W}_{1:n-1}^t, \boldsymbol{X}^t;\boldsymbol{\theta}_{\texttt{enc}}, \boldsymbol{\theta}_{\texttt{dec}}), \quad (10)$$

where w_n^t is the n-th token for the t-th utterance and $\boldsymbol{W}_{1:n-1}^t = \{w_1^t, \cdots, w_{n-1}^t\}$. N^t is the number of tokens in the t-th utterance.

In addition, we can apply SpecAugment, which augments supervised learning with input acoustic feature representations [15]. It consists of three kinds of deformations: time warping, time masking, and frequency masking. Time warping is deformation of input acoustic features in the time direction. Time masking and the frequency masking mask a block of consecutive time steps or frequency channels, respectively. An objective function using SpecAugment is defined as

$$\mathcal{L}_{\texttt{sa}}(\boldsymbol{\theta}_{\texttt{enc}}, \boldsymbol{\theta}_{\texttt{dec}}) = -\sum_{t=1}^{T}\sum_{n=1}^{N^t} \log P(w_n^t|\boldsymbol{W}_{1:n-1}^t, \mathcal{G}(\boldsymbol{X}^t);\boldsymbol{\theta}_{\texttt{enc}}, \boldsymbol{\theta}_{\texttt{dec}}), \quad (11)$$

where $\mathcal{G}()$ is the SpecAugment deformation function with random behavior for the input acoustic features.

3. SEMI-SUPERVISED LEARNING BASED ON SEQUENCE-LEVEL CONSISTENCY TRAINING

This section details sequence-level consistency training for semi-supervised end-to-end ASR models. Our semi-supervised settings assume that two kinds of data can be used for building end-to-end ASR models. One is labeled data as defined in Eq. (9). The other is unlabeled data,

$$\mathcal{D}_{\texttt{unpair}} = \{\boldsymbol{X}_{T+1}, \cdots, \boldsymbol{X}_{T+L}\}, \quad (12)$$

where L is the number of utterances in the unlabeled data set. Our semi-supervised learning is conducted in teacher-student training style. The training steps are follows.

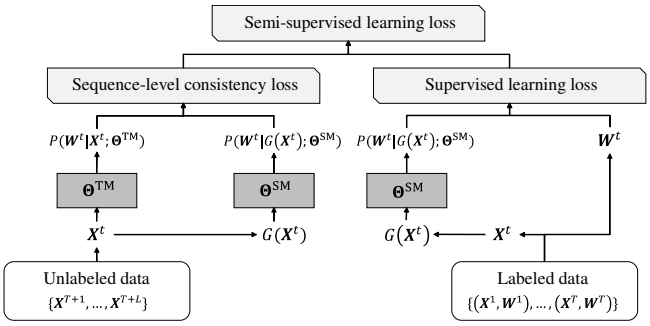

Fig. 1. Semi-supervised learning based on sequence-level consistency training.

1. A teacher model is trained via supervised training using $\mathcal{D}_{\texttt{pair}}$. Model parameter sets of the teacher model are defined as $\mathbf{\Theta}^{\texttt{TM}} = \{\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{enc}}, \boldsymbol{\theta}^{\texttt{TM}}_{\texttt{dec}}\}$. An objective function using SpecAugment is defined as
$$\mathcal{L}(\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{enc}}, \boldsymbol{\theta}^{\texttt{TM}}_{\texttt{dec}}) = \mathcal{L}_{\texttt{sa}}(\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{enc}}, \boldsymbol{\theta}^{\texttt{TM}}_{\texttt{dec}}). \quad (13)$$
The teacher model performs the role of assigning sequence-level generation probabilities of ASR results to the unlabeled data.
2. The student model is trained via semi-supervised learning using both $\mathcal{D}_{\texttt{pair}}$ and $\mathcal{D}_{\texttt{unpair}}$. Model parameter sets of the student model are defined as $\mathbf{\Theta}^{\texttt{SM}} = \{\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}}, \boldsymbol{\theta}^{\texttt{SM}}_{\texttt{dec}}\}$. An objective function for the semi-supervised learning is defined as
$$\mathcal{L}(\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}}, \boldsymbol{\theta}^{\texttt{SM}}_{\texttt{dec}}) = (1-\lambda)\mathcal{L}_{\texttt{sa}}(\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}}, \boldsymbol{\theta}^{\texttt{SM}}_{\texttt{dec}}) + \lambda\mathcal{L}_{\texttt{sc}}(\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}}), \quad (14)$$
where $\mathcal{L}_{\texttt{sc}}$ is the sequence-level consistency loss and λ is the hyper parameter to adjust the influence of the consistency loss. Note that we use the consistency loss only in learning the encoder network so that targets estimated by the teacher network do not adversely affect the decoder network.

Figure 1 shows the procedure of semi-supervised learning based on sequence-level consistency training. In the following subsections, we detail the sequence-level consistency loss and mini-batch training strategies using both labeled and unlabeled data.

3.1. Sequence-level consistency loss

The sequence-level consistency loss considers the sequence-level distribution specified by the teacher model over all possible sequences $\boldsymbol{W}^t \in \mathcal{T}^t$. The consistency loss is defined as
$$\mathcal{L}_{\texttt{sc}}(\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}}) = -\sum_{t=T+1}^{T+L}\sum_{\boldsymbol{W}^t\in\mathcal{T}^t} P(\boldsymbol{W}^t|\boldsymbol{X}^t;\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{enc}},\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{dec}}) \log P(\boldsymbol{W}^t|\mathcal{G}(\boldsymbol{X}^t);\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}},\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{dec}}). \quad (15)$$
In this case, the student end-to-end ASR model is trained so that generation results for the unlabeled data are constant even if slight deformation is added. In fact, it is impossible to consider all possible sequences, so we introduce a heuristic for determining the sequence-level consistency loss; only K-best hypotheses from beam search decoding results are used. The approximated loss is defined as
$$\mathcal{L}_{\texttt{sc}}(\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}}) \approx -\sum_{t=T+1}^{T+L}\sum_{\boldsymbol{W}^t\in\mathcal{T}^t_K} Q(\boldsymbol{W}^t|\boldsymbol{X}^t;\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{enc}},\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{dec}}) \log P(\boldsymbol{W}^t|\mathcal{G}(\boldsymbol{X}^t);\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}},\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{dec}}), \quad (16)$$
where $\mathcal{T}^t_K$ is the K-best hypotheses present in the beam search decoding results. $Q(\boldsymbol{W}^t)$ is the normalized probability which is computed from
$$Q(\boldsymbol{W}^t|\boldsymbol{X}^t;\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{enc}},\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{dec}}) = \frac{P(\boldsymbol{W}^t|\boldsymbol{X}^t;\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{enc}},\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{dec}})}{\sum_{\bar{\boldsymbol{W}}^t\in\mathcal{T}^t_K} P(\bar{\boldsymbol{W}}^t|\boldsymbol{X}^t;\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{enc}},\boldsymbol{\theta}^{\texttt{TM}}_{\texttt{dec}})}. \quad (17)$$
The simplest approximation is to use 1-best result present in the beam search decoding results based on the teacher model. We define the 1-best generation result of the t-th utterance as $\hat{\boldsymbol{W}}^t = \{\hat{w}^t_1, \cdots, \hat{w}^t_{\hat{N}^t}\}$. In this case, the consistency loss is approximated as
$$\mathcal{L}_{\texttt{sc}}(\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}}) \approx -\sum_{t=T+1}^{T+L}\sum_{n=1}^{\hat{N}^t} \log P(\hat{w}^t_n|\hat{\boldsymbol{W}}^t_{1:n-1},\mathcal{G}(\boldsymbol{X}^t);\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{enc}},\boldsymbol{\theta}^{\texttt{SM}}_{\texttt{dec}}). \quad (18)$$
In this paper, we elucidate the effectiveness of both 1-best based consistency loss and K-best based consistency loss.

3.2. Mini-batch training strategies

Our semi-supervised learning approach can be applied in two ways. One is unsupervised data augmentation, where both labeled and unlabeled data belong to the same domain. The other is unsupervised domain adaptation, where labeled source domain data and unlabeled target domain data are used. For each, we introduce the following mini-batch training strategies.

- **Unsupervised data augmentation:** In each epoch, we update the student model by randomly feeding mini-batches drawn from either the labeled or unlabeled data set. This helps to fully leverage both data sets for optimization against the target domain.
- **Unsupervised domain adaptation:** In each epoch, we first update the student model from mini-batches drawn from the labeled data. After that, we update it from mini-batches drawn from the unlabeled data. This helps to adapt the student model into the domain of the unlabeled data at the end of the epoch.

4. EXPERIMENTS

Our experiments used the Corpus of Spontaneous Japanese, which includes academic presentations and extemporaneous presentations [30]. Two semi-supervised learning conditions were assessed: unsupervised data augmentation and unsupervised domain adaptation. In the unsupervised data augmentation setup, the labeled data and unlabeled data were drawn from academic presentations. On the other hand, in the unsupervised domain adaptation setup, the labeled data was drawn from academic presentations, and the unlabeled data was drawn from the extemporaneous presentations. In addition, we prepared three test sets; (Test 1, 2, and 3). Test 1 and 2 are the academic presentations and Test 3 is the extemporaneous presentation. This paper uses characters as the tokens. Details of the data sets are shown in Table 1. The labeled and unlabeled data 1 were used for examining the unsupervised data augmentation setup. The labeled and unlabeled data 2 were used for examining the unsupervised domain adaptation setup. Note that the number of characters of the unlabeled data sets is calculated from their manual transcriptions that cannot be used in the semi-supervised learning absolutely.

Table 1. Experimental data sets.

	Domain	Data size (Hours)	Number of characters
Labeled data 1	Academic	82.9	2,222,994
Unlabeled data 1	Academic	169.6	4,524,392
Labeled data 2	Academic	252.5	6,747,386
Unlabeled data 2	Extemporaneous	263.1	6,679,489
Validation data	Academic	4.8	122,097
Test data 1	Academic	1.8	48,064
Test data 2	Academic	1.9	47,970
Test data 3	Extemporaneous	1.3	32,089

4.1. Conditions

Our experiments introduced Transformer based encoder-decoder for the end-to-end ASR models. We set $I = 8$ for the encoder blocks and $J = 6$ for the decoder blocks. The Transformer blocks were composed using the following conditions; output continuous representations had dimensions of 256, inner outputs in the position-wise feed forward networks had dimensions of 2,048, and the number of heads in multi-head attentions was set to 4. For the speech encoder, we used 40 log mel-scale filterbank coefficients appended with delta and acceleration coefficients as acoustic features; the frame shift was 10 ms. The acoustic features were passed through two convolution and max pooling layers with a stride of 2, so we downsampled them to $1/4$ along the time-axis. In the text decoder, we used 256 dimensional word embeddings. Vocabulary size was set to 2,476 for the unsupervised data augmentation setup and 2,628 for the unsupervised domain adaptation setup. For evaluation, we introduced the following five setups.

- **Supervised**: Supervised learning using the labeled data. The loss is defined by Eq. (10).
- **Supervised+SpecAug**: Supervised learning with SpecAugment using the labeled data. The loss is defined by Eq. (11).
- **Semi-supervised (1-best)**: Semi-supervised learning based on sequence-level consistency training using both the labeled and unlabeled data. λ in Eq. (14) was set to 0.5. We used the 1-best based consistency loss defined by Eq. (18).
- **Semi-supervised (5-best)**: Semi-supervised learning based on sequence-level consistency training using both the labeled and unlabeled data. λ in Eq. (14) was set to 0.5. We used the 5-best based consistency loss defined by Eq. (16).
- **Ideal semi-supervised**: Pseudo semi-supervised learning based on sequence-level consistency training using both the labeled and unlabeled data sets. λ in Eq. (14) was set to 0.5. For this, we used manual transcriptions instead of using the 1-best results in computing consistency loss defined by Eq. (18). Note that decoder network was not trained from the manual transcriptions. This can be regarded as the upper bound of our semi-supervised learning.

For the optimization, we used the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-9}$ and varied the learning rate based on update rule as presented in previous studies [9]. Mini-batch size was set to 32 utterances; dropout rate in Transformer blocks was set to 0.1. We introduced different mini-batch training strategies (described in Section 3.2.) for each semi-supervised learning setup. Our SpecAugment applied only frequency masking and time masking where both used two masks each; frequency masking width was randomly chosen from 0 to 20 frequency bins, and time masking width was randomly chosen from 0 to 100 frames. For ASR decoding, we used a beam search algorithm in which the beam size was set to 20.

Table 2. CER results (%) in unsupervised data augmentation setup.

	Test 1 (Target)	Test 2 (Target)	Test 3 (Unknown)
Supervised	25.88	21.88	34.76
Supervised+SpecAug	21.23	17.23	30.49
Semi-supervised (1-best)	18.35	15.03	27.60
Semi-supervised (5-best)	**17.86**	**14.62**	**27.03**
Ideal semi-supervised	15.60	10.13	20.78

Table 3. CER results (%) in unsupervised domain adaptation setup.

	Test 1 (Source)	Test 2 (Source)	Test 3 (Target)
Supervised	11.21	8.28	17.37
Supervised+SpecAug	9.41	6.97	15.50
Semi-supervised (1-best)	9.32	6.79	12.70
Semi-supervised (5-best)	**9.31**	**6.76**	**12.36**
Ideal semi-supervised	9.25	6.66	9.12

4.2. Results

We evaluate results of unsupervised data augmentation and unsupervised domain adaptation. Experimental results in terms of character error rate (CER) of both setups are shown in Tables 2 and 3, respectively.

First, in both setups, SupecAugment improved ASR performance in supervised learning. This indicates that it is important to make a model robust to small deformation in supervised learning. Next, in the unsupervised data augmentation setup, the results show that semi-supervised learning methods outperformed supervised learning methods in both target domain test data sets and unknown domain test data set. It is thought that the performance improvements were created by making the model robust to various input examples. This suggests that unlabeled data can be leveraged for constraining the model to be invariant to small deformation. In particular, semi-supervised learning with 5-best based sequence-level consistency loss outperformed that with 1-best based sequence-level consistency loss. This indicates that considering sequence-level generation consistency precisely is effective in improving ASR performance. In addition, in the unsupervised domain adaptation setup, the results show that semi-supervised learning methods improved ASR performance in the target domain test data set while maintain ASR performance in source domain test data sets. These results confirm that the proposed semi-supervised learning can effectively improve ASR performance on unlabeled data in both unsupervised data augmentation and unsupervised domain adaptation.

5. CONCLUSIONS

This paper proposed a sequence-level consistency training scheme for enhanced semi-supervised learning of end-to-end ASR systems. The main strength of sequence-level consistency training is that unlabeled speech data can be leveraged for constraining end-to-end ASR models so that they are invariant to small deformation as added by SpecAugemnt. This efficiently yields models that are robust to various input speech examples. Our experiments using Transformer-based end-to-end ASR models showed our semi-supervised learning with sequence-level consistency training can efficiently improve ASR performance using unlabeled speech data in both unsupervised data augmentation and unsupervised domain adaptation setups.

6. REFERENCES

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

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

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

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

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

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

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

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

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *In Proc. Advances in Neural Information Processing Systems (NIPS)*, pp. 5998–6008, 2017.

[10] Linhao Dong, Shuang Xu, and Bo Xu, "Speech-Transformer: A no-recurrence sequence-to-sequence model for speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5884–5888, 2018.

[11] Yuanyuan Zhao, Jie Li, Xiaorui Wang, and Yan Li, "The SpeechTransformer for large-scale mandarin chinese speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 7095–7099, 2019.

[12] Sheng Li, Dabre Raj, Xugang Lu, Peng Shen, Tatsuya Kawahara, and Hisashi Kawai, "Improving Transformer-based speech recognition systems with compressed structure and speech attribute augmentation," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 4400–4404, 2019.

[13] Shigeki Karita, Nelson Enrique Yalta Soplin, Shinji Watanabe, Marc Delcroix, Atsunori Ogawa, and Tomohiro Nakatani, "Improving Transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1408–1412, 2019.

[14] Chung-Cheng Chiu, Tara N. Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J. Weiss, Kanishka Rao, Ekaterina Gonina, Navdeep Jaitly, Bo Li, Jan Chorowski, and Michiel Bacchiani, "State-of-the-art speech recognition with sequence-to-sequence models," .

[15] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2613–2617, 2019.

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

[17] Masato Mimura, Sei Ueno, Hirofumi Inaguma, Shinsuke Sakai, and Tatsuya Kawahara, "Leveraging sequence-to-sequence speech synthesis for enhancing acoustic-to-word speech recognition," *In Proc. Spoken Language Technology Workshop (SLT)*, pp. 477–484, 2018.

[18] Tomoki Hayashi, Shinji Watanabe, Yu Zhang, Tomoki Toda, Takaaki Hori, Ramon Astudillo, and Kazuya Takeda, "Back-translation-style data augmentation for end-to-end ASR," *In Proc. Spoken Language Technology Workshop (SLT)*, pp. 426–432, 2018.

[19] Shigeki Karita, Shinji Watanabe, Tomoharu Iwata, Atsunori Ogawa, and Marc Delcroix, "Semi-supervised end-to-end speech recognition," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2–6, 2018.

[20] Takaaki Hori, Ramon Astudillo, Tomoki Hayashi, Yu Zhang, Shinji Watanabe, and Jonathan Le Roux, "Cycle-consistency training for end-to-end speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 6271–6275, 2019.

[21] Murali Karthick Baskar, Shinji Watanabe, Ramon Astudillo, Takaaki Hori, Lukas Burget, and Jan Cernocky, "Semi-supervised sequence-to-sequence asr using unpaired speech and text," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 3790–3794, 2019.

[22] Samuli Laine and Timo Aila, "Temporal ensembling for semi-supervised learning," *In Proc. International Conference on Learning Representations (ICLR)*, 2017.

[23] Antti Tarvainen and Harri Valpola, "Weight-averaged consistency targets improve semi-supervised deep learning results," *In Proc. Advances in Neural Information Processing Systems (NIPS)*, pp. 1195–1204, 2017.

[24] Takeru Miyato, Shin ichi Maeda, Shin Ishii, , and Masanori Koyama, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[25] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le, "Unsupervised data augmentation for consistency training," *arXiv:1904.12848*, 2019.

[26] Yoon Kim and Alexander M. Rush, "Sequence-level knowledge distillation," *In Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1317–1327, 2016.

[27] Mingkun Huang, Yongbin You, Zhehuai Chen, Yanmin Qian, and Kai Yu, "Knowledge distillation for sequence model," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 3703–3707, 2018.

[28] Raden Mu'az Mun'im, Nakamasa Inoue, and Koichi Shinoda, "Sequence-level knowledge distillation for model compression of attention-based sequence-to-sequence speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 6151–6155, 2019.

[29] Pengcheng Guo, Sining Sun, and Lei Xie, "Unsupervised adaptation with adversarial dropout regularization for robust speech recognition," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 749–753, 2019.

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