Uncertainty-Aware Self-Training for CTC-Based Automatic Speech Recognition

Eungbeom Kim[1], **Kyogu Lee**[1,2,3]

[1]Interdisciplinary Program in Artificial Intelligence, Seoul National University
[2]Artificial Intelligence Institute, Seoul National University
[3]Department of Intelligence and Information, Seoul National University
{eb.kim, kglee}@snu.ac.kr

Abstract

Uncertainty estimation has been widely applied for trustworthy automatic speech recognition (ASR) systems across training and inference stages. In the training stage, previous studies show that uncertainty can facilitate self-training by filtering out unlabeled data samples with high uncertainty. However, the current sequence-level uncertainty estimation method for connectionist temporal classification (CTC) based ASR models drops the output probability information and depends only on the textual distance of decoded predictions. In this study, we argue that this results in limited performance improvement and propose a novel output probability-based sequence-level uncertainty estimation method. We also categorize uncertainty as pseudo-label uncertainty and in-training uncertainty for the self-training process. Finally, we present uncertainty-aware self-training for CTC-based ASR models and experimentally show the effectiveness of the proposed method compared to the baselines.

1 Introduction

Transformer-based models have been successfully adopted in various domains such as language, vision, and audio (Vaswani et al. 2017; Dosovitskiy et al. 2021; Gong, Chung, and Glass 2021). Based on this progress, Transformer has also been popularized in end-to-end automatic speech recognition (ASR) by allowing scalable architectures (Dong, Xu, and Xu 2018; Baevski et al. 2020; Radford et al. 2023). However, training large ASR models requires a large amount of data samples. Moreover, to fully utilize the generalization ability of Transformer models, simultaneous scaling of the dataset and the architecture is regarded as a key element (Zhai et al. 2022).

Semi-supervised learning leverages unlabeled data to alleviate the limited scalability of labeled data. For instance, self-training aims to train the potent student model by pseudo-labeling unlabeled data using the teacher model at the fine-tuning stage. Due to its simplicity, self-training has been widely applied to various ASR architectures such as connectionist temporal classification (CTC) based models (Chen, Wang, and Wang 2020; Xu et al. 2020; Zhang et al. 2022; Singh, Hou, and Wang 2023; Peng et al. 2024), sequence-to-sequence models (Kahn, Lee, and Hannun 2020; Park et al. 2020; Radford et al. 2023), and RNN-Transducer models (Zhang et al. 2022; Hwang et al. 2022).

Since pseudo-labels are estimated by the teacher model at self-training, the student model is easily misguided under the erroneous teacher model. For this reason, Kahn, Lee, and Hannun (2020); Park et al. (2020); Zhang et al. (2022) filter out unreliable pseudo-labels for self-training of ASR models and observe that this prevents performance degradation from the erroneous pseudo-labels and contributes to performance enhancement. Likewise, due to the lack of reliability of the pseudo-labels, pseudo-label filtering for self-training has been widely explored for ASR, including uncertainty-based pseudo-label filtering (Khurana et al. 2021; Khurana, Laurent, and Glass 2022; Dawalatabad et al. 2023).

Uncertainty estimation aims to quantify the reliability of the output prediction. Given this purpose, employing uncertainty estimation is intuitive for measuring the quality of pseudo-labels and filtering pseudo-labels for self-training. Nevertheless, most of the uncertainty estimation methods focus on restricted tasks such as classification or regression and are not directly applicable to a sequence prediction such as ASR (Patel, Allebach, and Qiu 2023). Although Dey et al. (2019); Jiao et al. (2021); Patel, Allebach, and Qiu (2023) employ entropy-based uncertainty during a self-training process on various sequence prediction tasks, they are limited to sequence-to-sequence models.

For Encoder-only ASR models using a CTC mechanism, Vyas et al. (2019) presents an uncertainty estimation method using Monte Carlo Dropout (MCD) (Gal and Ghahramani 2016). In this method, the textual edit distance of the decoded MCD predictions is utilized for uncertainty estimation and Khurana et al. (2021) successfully adopts this method to the self-training process. However, this approach abandons flourishing output probability information and only utilizes the decoded transcriptions for uncertainty estimation. Although the output probability-based uncertainty estimation and its utilization for semi-supervised learning are recently proposed for CTC-based ASR models (Rumberg et al. 2023; Zhu et al. 2023), they focus on token-level uncertainty estimation, not sequence-level uncertainty estimation. In addition, the previous sequence-level uncertainty estimation methods for CTC-based ASR models have mainly focused on model uncertainty, and data uncertainty has hardly been

explored.

To challenge these issues, we introduce the output probability-based sequence-level uncertainty estimation method for CTC-based ASR models, consisting of model uncertainty and data uncertainty. Based on this, we present a novel uncertainty-aware self-training method, UNCAST. UNCAST discriminates reliable and unreliable pseudo-labels by the teacher model's uncertainty. Also, since the proposed uncertainty estimation method is differentiable unlike the textual edit distance-based uncertainty estimation, the student model's uncertainty is easily adaptable to the training loss. Thus, UNCAST incorporates the teacher and the student model's uncertainty estimation results for the uncertainty-aware self-training loss, different from the previous uncertainty-based self-training methods (Khurana et al. 2021; Khurana, Laurent, and Glass 2022; Dawalatabad et al. 2023) focusing only on the teacher model's uncertainty. We observe that the proposed method has a fine-grained uncertainty estimation compared to the textual edit distance-based uncertainty estimation. Building upon this advantage, we observe the effectiveness of UNCAST for semi-supervised learning, particularly within the self-training process. To fully exploit the proposed method, we additionally employ iterative unlabeled data refinement similar to the traditional self-training methods and confirm the extra performance improvement.

2 Related Work

2.1 Self-Training

Self-training has been widely studied to train the student model using unlabeled data. In self-training, the predictions of the teacher model are regarded as pseudo-labels, and then the student model is trained with the labeled dataset and the pseudo-labeled dataset originating from the unlabeled dataset. However, the student model is misled into undesirable confirmation bias where the pseudo-labels are incorrect. Previous self-training methods study two lines of approaches to handle this issue, namely, estimating reliable pseudo-labels (Laine and Aila 2017; Tarvainen and Valpola 2017) and filtering unreliable pseudo-labels (Xie et al. 2020; Sohn et al. 2020; Zhang et al. 2021). These approaches have also been studied for ASR. Specifically, Manohar et al. (2021); Higuchi et al. (2021) utilize a moving average model for the teacher model and Park et al. (2020); Higuchi et al. (2021); Khurana et al. (2021) adopt beam search to generate accurate pseudo-labels. In addition, Tripathi et al. (2024) proposes a Monte Carlo self-training method for pseudo-label sampling. Various pseudo-label filtering methods for ASR self-training have also been studied. For instance, Park et al. (2020) proposes a gradual filtering threshold relaxation method. Also, Dey et al. (2019) compares confidence and entropy for filtering criterion, and Khurana et al. (2021); Dawalatabad et al. (2023); Khurana, Laurent, and Glass (2022) explore uncertainty-driven pseudo-label filtering. Likewise, uncertainty estimation is applied to filtering unreliable pseudo-labels because pseudo-label quality can be evaluated by uncertainty of the teacher model's prediction without the need for a ground-truth label. Thus, in this work, we focus on a pseudo-label filtering method using uncertainty estimation.

2.2 Uncertainty Estimation

Sequence-level uncertainty estimation has been broadly studied across various sequence prediction tasks such as machine translation (Zhou et al. 2020; Jiao et al. 2021) or text recognition (Patel, Allebach, and Qiu 2023) for self-training. For CTC-based ASR models, Vyas et al. (2019) introduces the uncertainty estimation method which is a variant of Monte Carlo Dropout (MCD) (Gal and Ghahramani 2016). This method estimates uncertainty based on a textual edit distance of the decoded sequence predictions from the Dropout implementations at the inference stage and is successfully applied to pseudo-label filtering for the self-training process (Khurana et al. 2021; Dawalatabad et al. 2023; Khurana, Laurent, and Glass 2022). These approaches for CTC-based ASR uncertainty estimation focus on model uncertainty, which denotes uncertainty from model parameters. However, uncertainty of the model prediction is also affected by data uncertainty which stems from inherent noise within the data itself (Kendall and Gal 2017). Also, the textual edit distance does not fully utilize output probability-level information. To address these issues, we aim to propose an output probability-based data uncertainty estimation method and model uncertainty estimation method for CTC-based ASR models in this study.

3 Background

3.1 Connectionist Temporal Classification

ASR models aim to translate an input utterance x into a transcription y. Although an input-transcription alignment is unknown in the training dataset, a Transformer encoder-based ASR model can be efficiently trained with the CTC framework (Graves et al. 2006). A CTC output probability of a label transcription given an input utterance is defined as follows:

$$P_{\text{CTC}}(y|f(x)) := \sum_{a \in \beta^{-1}(y)} P(a|f(x)), \tag{1}$$

where $f(\cdot)$ is a model which outputs the frame-level output probability, a is an frame-level alignment, and $\beta^{-1}(y)$ denotes a possible alignment set of y. That is, CTC enables us to estimate the probability of the label given the input by the sum of the probabilities for every possible alignment. Based on this, the CTC loss function $\mathcal{L}(\cdot)$ is defined as follows:

$$\mathcal{L}(x, y, f) = -\log P_{\text{CTC}}(y|f(x)). \tag{2}$$

Since the CTC loss function is easily computed by the forward-backward algorithm, the Transformer encoder-based ASR models are efficiently trained with the CTC algorithm.

3.2 Self-Training

Self-training based semi-supervised learning is designed to train the student model f^S utilizing a pseudo-labeling process for an unlabeled dataset. In this study, we consider the Noisy Student (Xie et al. 2020) method, which is a popular self-training framework. In the first stage, the teacher

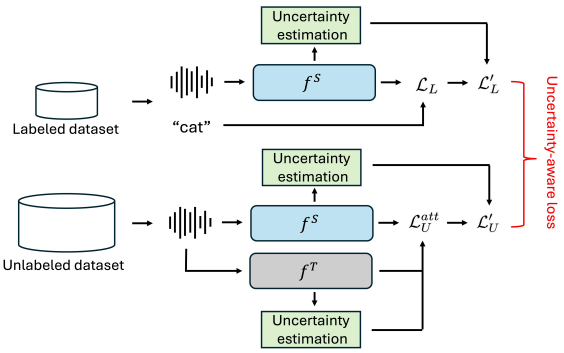


Figure 1: Overview of the proposed uncertainty-aware self-training framework, UNCAST. UNCAST includes two types of uncertainty-aware loss. For the labeled dataset, in-training uncertainty of the student model is utilized to augment the loss function. For the unlabeled dataset, the loss function is augmented from pseudo-label uncertainty originating from the teacher model and in-training uncertainty of the student model.

model f^T is trained on the labeled dataset L, and the trained teacher model is applied to pseudo-labeling the unlabeled data samples. Then, the student model is trained on the labeled dataset and the unlabeled dataset with pseudo-labels. In this stage, noise based on Dropout (Srivastava et al. 2014) and SpecAugment (Park et al. 2019) is injected into the student model, unlike the unnoised teacher model for accurate pseudo-labeling. The total self-training loss $\mathcal{L}_{\text{ST}}$ for the student model is formulated as follows:

$$\mathcal{L}_L = \frac{1}{|L|} \sum_{x_l \in L} \mathcal{L}(x_l, y_l, f^S), \tag{3}$$

$$\mathcal{L}_U = \frac{1}{|U|} \sum_{x_u \in U} \mathcal{L}(x_u, \hat{y}(x_u, f^T), f^S), \tag{4}$$

$$\mathcal{L}_{\text{ST}} = \mathcal{L}_L + \lambda \mathcal{L}_U, \tag{5}$$

where $\mathcal{L}_L$ is the loss for the labeled dataset L, $\mathcal{L}_U$ is the loss for the unlabeled dataset U, $\hat{y}(x, f^T)$ is the output transcription from the teacher model f^T, and λ is a hyperparameter. Since unlabeled data x_u does not have a paired transcription, the pseudo-labeling process $\hat{y}(x, f^T) = \text{decode}(f^T(x))$ is applied where $\text{decode}(\cdot)$ denotes a decoding process such as greedy decoding or beam search.

4 Method

4.1 Data Uncertainty

In this study, we propose an output probability-based sequence-level uncertainty for CTC-based ASR models, which consists of data uncertainty and model uncertainty following Kendall and Gal (2017). For CTC-based ASR models, Rumberg et al. (2023) estimates frame-level uncertainty as the sum of the frame's output probabilities by which the decoded output sequence is changed. By extending this, we present sequence-level data uncertainty u_d based on the sequence probability by which the output sequence $\hat{y}(x, f)$ is changed given the input x and the model f, i.e., $1 - P(\hat{y}(x, f)|f(x))$. Since prediction confidence of CTC-based ASR models can be modeled by CTC probability modeling, accordingly, uncertainty estimation by the proposed method is efficiently computed by the CTC loss function where the label y is replaced by the decoded output sequence prediction $\hat{y}(x, f)$. Finally, by removing the constant and applying logarithmic-scale probability, sequence-level data uncertainty is formulated as follows:

$$u_d(x, f) = -\frac{1}{|\hat{y}(x, f)|} \log P_{\text{CTC}}(\hat{y}(x, f)|f(x)). \tag{6}$$

We normalize the length of the decoded output sequence to minimize the effect of the sequence length. The proposed method is also similar to the perplexity-based sequence-level data uncertainty estimation introduced in Zhou et al. (2020) and the confidence-based data uncertainty estimation (Lakshminarayanan, Pritzel, and Blundell 2017; Mukhoti et al. 2023; Sun et al. 2022).

4.2 Model Uncertainty

Model uncertainty denotes uncertainty originating from model parameters. To capture model uncertainty, MCD (Gal and Ghahramani 2016) has been widely applied due to its simplicity and effectiveness. MCD approximates Bayesian inference with Dropout (Srivastava et al. 2014) at the inference stage and utilizes its variance to estimate model uncertainty. Based on this, Vyas et al. (2019) explores model uncertainty estimation for CTC-based ASR models. They measure the maximum textual edit distance between the decoded output sequence without Dropout and the decoded output sequences sampled multiple times from Monte Carlo Dropout without considering output probability information.

We present the output probability-based MCD extension of model uncertainty estimation for CTC-based ASR models. Given the input x and the model f, we decode the output sequence $\hat{y}(x, f)$ without Dropout. Then, we produce the multiple MCD output probabilities $f_i(x)$ for $i = 1, ..., N$, where $f_i(\cdot)$ denotes the i-th sampled model using Dropout and N is the total number of Dropout implementations. We compute the confidences with respect to the decoded output sequence $\hat{y}(x, f)$ for each MCD output $f_i(x)$. This can be efficiently calculated by CTC probability modeling as $P_{\text{CTC}}(\hat{y}(x, f)|f_i(x))$. Finally, we implement normalization by the sequence length and derive model uncertainty u_m of CTC-based ASR models as follows:

$$u_m^i(x, f) = -\frac{1}{|\hat{y}(x, f)|} \log P_{\text{CTC}}(\hat{y}(x, f)|f_i(x)), \tag{7}$$

$$u_m(x, f, N) = \max_{i=1,2,...,N} \{u_m^i(x, f)\}. \tag{8}$$

In short, the worst estimated probability from the MCD implementations with respect to the decoded output sequence without Dropout is defined as model uncertainty while computing the output probability under CTC probability modeling.

4.3 Uncertainty-Aware Self-Training

In this section, we introduce the uncertainty-aware self-training (UNCAST) framework based on the proposed un-

certainty estimation method. UNCAST utilizes a novel uncertainty-aware loss function that considers two types of uncertainties, the teacher model's uncertainty and the student model's uncertainty, as illustrated in Figure 1. In the rest of this section, we provide a detailed explanation of UNCAST.

Pseudo-label uncertainty. Pseudo-label uncertainty denotes the inherent uncertainty of the pseudo-labels. In self-training, thus, the uncertainty of the teacher model is interpreted as pseudo-label uncertainty. Given the N times Dropout implementations in the inference stage, we estimate pseudo-label uncertainty $u_{\text{pl}}(x, f^T, N)$ of the input x and the pseudo-label $\hat{y}(x, f^T)$ predicted by the teacher model f^T trained on labeled data as follows:

$$u_{\text{pl}}(x, f^T, N) = u_d(x, f^T) + u_m(x, f^T, N), \quad (9)$$

which incorporates the proposed output probability-based sequence-level data uncertainty $u_d(x, f^T)$ and model uncertainty $u_m(x, f^T, N)$.

We leverage pseudo-label uncertainty u_{pl} for loss attenuation to minimize the negative effect of the uncertain pseudo-labels in self-training, motivated by Kendall and Gal (2017). The attenuated loss function $\mathcal{L}_U^{att}$ originating from the loss function $\mathcal{L}_U$ of the unlabeled dataset U is formulated as follows:

$$\mathcal{L}_U^{att} = \frac{1}{|U|} \sum_{x_u \in U} \frac{1}{u_{\text{pl}}(x_u, f^T, N)} \mathcal{L}(x_u, \hat{y}(x_u, f^T), f^S). \quad (10)$$

We aim to strengthen the training effect of the certainly pseudo-labeled data samples, i.e., data samples with low pseudo-label uncertainty u_{pl}, and weaken the training effect of the uncertain pseudo-labeled data samples, i.e., data samples with high pseudo-label uncertainty u_{pl}, by reweighting the loss of each pseudo-labeled sample $x_u \in U$. Unlike previous self-training methods for CTC-based ASR models (Khurana et al. 2021; Khurana, Laurent, and Glass 2022) that adopt hard 0-1 data selection for unlabeled data filtering, we present a soft loss attenuation function to fully exploit the information of estimated uncertainty as shown in Equation 10. A soft loss attenuation method is particularly advantageous for the proposed uncertainty estimation because our approach is based on the output probability-level uncertainty with fine-grained resolution compared to the text-based uncertainty estimation method. Detailed experimental analysis is introduced in Section 6.2. The loss function for the labeled dataset $\mathcal{L}_L$ does not require loss attenuation as illustrated in Figure 1 because the data samples in the labeled dataset are labeled with ground-truth transcriptions.

In-training uncertainty. Herein, we concentrate on in-training uncertainty of the student model in the training stage. We propose the in-training uncertainty-aware loss function by directly including the uncertainty of the student model for a regularization term. The regularization term is formulated as model uncertainty of the student model normalized by its data uncertainty. Formally, the uncertainty-aware loss $\mathcal{L}'_L$ and $\mathcal{L}'_U$ based on data uncertainty $u_d(x, f^S)$ and model uncertainty $u_m(x, f^S, N)$ of the student model is formulated as follows:

$$\mathcal{L}'_L = \mathcal{L}_L + \alpha \sum_{x_l \in L} \frac{u_m(x_l, f^S, N)}{\text{sg}(u_d(x_l, f^S))}, \quad (11)$$

$$\mathcal{L}'_U = \mathcal{L}_U^{att} + \alpha \sum_{x_u \in U} \frac{u_m(x_u, f^S, N)}{\text{sg}(u_d(x_u, f^S))}, \quad (12)$$

where $\text{sg}(\cdot)$ is a stop gradient operation and α is a hyperparameter that decides the strength of the in-training uncertainty minimization objective. Since our uncertainty metric is differentiable, the in-training uncertainty loss functions are directly added to the optimization objective.

Interestingly, in-training uncertainty-aware loss can be interpreted as a variant of a consistency regularization-based semi-supervised learning method such as Mixmatch (Berthelot et al. 2019), RemixMatch (Berthelot et al. 2020), or FixMatch (Sohn et al. 2020) which enforces the outputs of the model are unchanged after the input perturbations. In this point of view, minimizing the proposed model uncertainty is referred to as consistency regularization in that the proposed model uncertainty measures the disagreement between the output with no perturbation and the outputs with Dropout for perturbation in addition to the input perturbation such as data augmentation. However, model uncertainty is also minimized where MCD outputs show a wrong but consistent transcription, which corresponds to confirmation bias in consistency regularization. To prevent this issue, we regulate the in-training uncertainty loss using the reliability of the prediction and approximate this using data uncertainty $u_d(x, f^S)$. Nonetheless, incorporating data uncertainty into the loss function's denominator hinders the training process; therefore, we employ a stop gradient operation for data uncertainty at the proposed loss function.

Training. Overall, the total loss function $\mathcal{L}_{\text{UNCAST}}$ of the proposed method, UNCAST, is formulated as follows:

$$\mathcal{L}_{\text{UNCAST}} = \mathcal{L}'_L + \lambda \mathcal{L}'_U, \quad (13)$$

where λ is a hyperparameter balancing the labeled dataset loss and the unlabeled dataset loss.

In summary, we propose the output probability-based sequence-level uncertainty estimation method including data uncertainty and model uncertainty. Based on this, we present uncertainty estimation for pseudo-label (teacher) uncertainty estimation and in-training (student) uncertainty estimation; we finally incorporate these uncertainty estimations and propose a novel self-training framework, UNCAST.

5 Experiments

Dataset. For semi-supervised learning, the LibriSpeech dataset (Panayotov et al. 2015) is considered to follow the previous ASR semi-supervised learning methods (Kahn, Lee, and Hannun 2020; Park et al. 2020; Xu et al. 2021; Kim et al. 2023; Li, Meng, and Sun 2023; Higuchi et al. 2023). For the labeled training dataset, the 100 hours LibriSpeech train-clean dataset (LS-100) is utilized. For the unlabeled training dataset, we consider two datasets: the 360

U = LS-360	Iteration=1				Iteration=2			
Method	dev-clean	dev-other	test-clean	test-other	dev-clean	dev-other	test-clean	test-other
Seed	5.25	11.90	5.26	11.95	-	-	-	-
Noisy Student	4.66	10.22	4.71	10.25	4.75	10.41	4.73	10.38
DUST	4.47	9.82	4.67	9.85	4.44	9.61	4.48	9.50
UNCAST (*ours*)	**4.21**	**9.60**	**4.31**	**9.35**	**4.23**	**9.42**	**4.26**	**9.32**
Oracle	3.32	8.86	3.60	8.73	-	-	-	-

Table 1: WER on the LibriSpeech dev and test datasets. The seed model is trained only with the labeled LS-100 dataset. The oracle model is trained with the labeled LS-100 and LS-360 datasets. The baseline method (Noisy Student), the previous method (DUST), and the proposed uncertainty-aware self-training (UNCAST) method are trained with the labeled LS-100 dataset and the unlabeled LS-360 dataset. The baseline model is trained without a pseudo-label filtering process. We also experiment with the iterative self-training method by refining pseudo-labels of the unlabeled dataset.

U = LS-500	Iteration=1				Iteration=2			
Method	dev-clean	dev-other	test-clean	test-other	dev-clean	dev-other	test-clean	test-other
Seed	5.25	11.90	5.26	11.95	-	-	-	-
Noisy Student	4.82	10.38	4.87	10.32	4.79	10.35	4.82	10.29
DUST	4.62	**9.61**	4.64	9.63	4.66	9.33	4.57	9.46
UNCAST (*ours*)	**4.34**	9.65	**4.41**	**9.56**	**4.25**	**9.05**	**4.24**	**9.16**
Oracle	3.30	7.73	3.36	7.61	-	-	-	-

Table 2: WER on the LibriSpeech dev and test datasets. The seed model is trained only with the labeled LS-100 dataset. The oracle model is trained with the labeled LS-100 and LS-500 dataset. The baseline method (Noisy Student), the previous method (DUST), and the proposed uncertainty-aware self-training (UNCAST) method are trained with the labeled LS-100 dataset and the unlabeled LS-500 dataset. The baseline model is trained without a pseudo-label filtering process. We also experiment with an iterative self-training method by refining pseudo-labels of the unlabeled dataset.

hours LibriSpeech train-clean dataset (LS-360) and the 500 hours LibriSpeech train-other dataset (LS-500).

Implementation details. We utilize a 12-layer Transformer encoder-based end-to-end ASR model, WavLM Base+ (Chen et al. 2022), under the CTC framework. We use a learning rate of 3e-5 with the Adam optimizer (Kingma and Ba 2015) with 10% warmup stages out of 100 total epochs. Also, the model is frozen for 12.5% of the training except for a newly initialized linear CTC layer. A batch size of 128 and the CTC loss function is utilized for optimization. Following the original specifications, a character-level tokenizer is utilized. All of the results in this study are reproduced for a fair comparison.

To stabilize UNCAST, we clip the uncertainty value u_{pl} lower than 1% of the training dataset to stabilize the loss attenuation in Equation 10 and set λ as the clipped criterion. This can be interpreted as a labeled data sample is regarded as important as the top 1% of the unlabeled dataset. For validation, we conduct a hyperparameter search for UNCAST based on the word error rates (WER) of the dev-clean dataset using the model trained on the labeled LS-100 dataset and the unlabeled LS-360 dataset. We set $N = 3$ for the number of Dropout implementations.

Method	test-clean	test-other
UNCAST (*ours*)	4.31	9.34
- in-training uncertainty	4.36	9.43
- soft loss attenuation	4.47	9.80
- data uncertainty	4.52	9.84

Table 3: Ablation effects of the proposed method evaluated on WER (%) of the LibriSpeech dataset. All of the methods are trained on the labeled LS-100 dataset and the unlabeled LS-360 dataset.

6 Results

In this section, we present the experimental results of semi-supervised learning in Section 6.1 and the experimental analysis of uncertainty estimation in Section 6.2.

6.1 Semi-Supervised Learning

Main results. Table 1 and 2 show the results of the semi-supervised learning methods concerning the labeled LS-100 dataset with the unlabeled LS-360 dataset or the unlabeled LS-500 dataset. We train our seed model on the labeled dataset, the LS-100 dataset. We experiment with Noisy Student models without unlabeled data filtering as our self-training baseline. For the oracle performance, we experiment on the LS-100 dataset with the transcribed LS-

Method	test-clean	test-other
w/ linear-scale	4.34	9.51
w/ log-scale	4.36	9.43

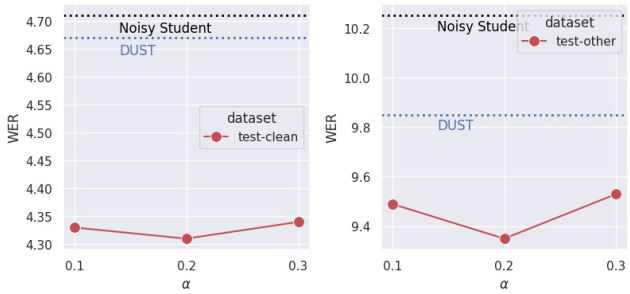
Table 4: Scaling effects for soft loss attenuation of the proposed method. The models are evaluated on WER (%) of the LibriSpeech test datasets and trained on the labeled LS-100 dataset and the unlabeled LS-360 dataset.

360 dataset or the LS-500 dataset, which is expected to show the upper bound of a performance. Our goal is to achieve the performance as close as the oracle models using semi-supervised learning. For the uncertainty-based self-training baseline, DUST (Khurana et al. 2021; Dawalatabad et al. 2023), which utilizes pseudo-label filtering based on MCD with textual edit distance (MCD-ED), is implemented. The unlabeled data filtering threshold of DUST is selected among $\tau \in \{25, 50, 75\}$ using the dev-clean dataset where τ denotes the percentage of unlabeled dataset filtering.

As shown in Table 1 and 2, it is observed that the model trained without the filtering on self-training (Noisy Student) reduces WER of the test-other dataset from 11.95% to 10.25% and 10.32% for the two unlabeled datasets, LS-360 and LS-500, respectively, although the uncertainty-based methods, UNCAST and DUST, consistently show the outperforming performances. The proposed method, UNCAST, surpasses the other methods at every test-clean and test-other dataset including the seed, Noisy Student, and DUST models for both the unlabeled LS-360 dataset and the unlabeled LS-500 dataset cases. As an example, Table 1 shows that UNCAST achieves 9.35% WER on the test-other dataset, which surpasses the seed, Noisy Student, and DUST methods achieving WER of 11.95%, 10.25%, and 9.85%, respectively, where the LS-360 dataset is utilized for the unlabeled dataset. Overall, the results successfully confirm the usefulness of the proposed uncertainty estimation method and self-training.

Iterative self-training. We also explore the iterative self-training in which the unlabeled data samples are re-labeled from the trained student model. Due to the computation complexity, we experiment with self-training for up to two iterations. For the second iteration, we re-initialize the student models following the previous methods (Khurana et al. 2021; Singh, Hou, and Wang 2023). UNCAST for the first and the second iteration achieves 9.35% and 9.32% WER for the test-other dataset, outperforming DUST which achieves 9.85% and 9.50%, as shown in Table 1. Contrary to this, the model trained without unlabeled dataset filtering (Noisy Student) even deteriorates ASR performance. The improvement is more drastic where the LS-500 dataset is utilized as the unlabeled dataset showing WER of 9.56% and 9.16% for the test-other dataset, compared to DUST which shows 9.63% and 9.46% for the test-other dataset, as shown in Table 2.

We observe that applying the LS-360 dataset as the unlabeled dataset for the first iteration of UNCAST shows the surpassing performance compared to the LS-500 dataset al-



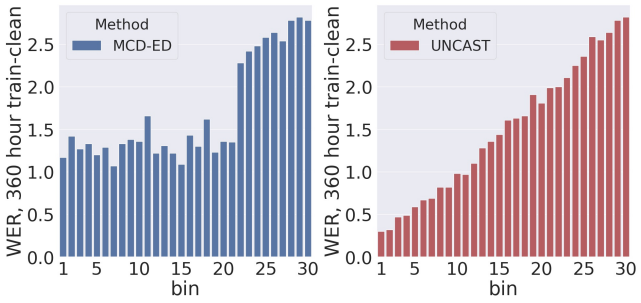
(a) test-clean dataset (b) test-other dataset

Figure 2: WER (%) with hyperparameter α. α decides the degree of in-training uncertainty-based regularization for UNCAST. $\alpha \in \{0.1, 0.2, 0.3\}$ is evaluated by the LibriSpeech test-clean and test-other datasets.

though the LS-500 dataset is larger than the LS-360 dataset. Moreover, this also holds for the baseline methods. We believe that this is because the incorrect pseudo-labels of the LS-500 dataset hinder the model training since the LS-500 dataset is a relatively difficult dataset causing erroneous pseudo-labels. At the second iteration, however, the incorrect pseudo-labels are decreased; this is beneficial for the large unlabeled dataset such as the LS-500 dataset, thus resulting in drastic WER reduction from 9.56% to the 9.16% on the test-other dataset.

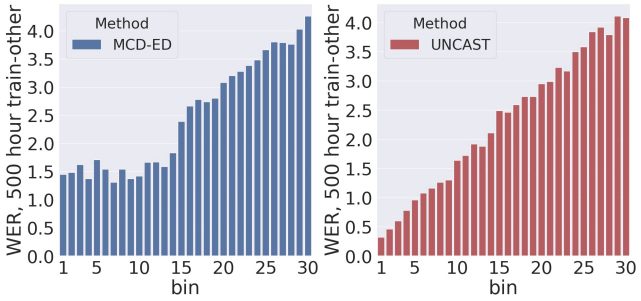
Ablation study. We conduct an ablation study to analyze the effect of each component of the proposed method. As shown in Table 3, it is observed that all of the components contribute to the performance. Specifically, combining pseudo-label uncertainty and in-training uncertainty for self-training, i.e., UNCAST, achieves 4.31% WER on the test-clean dataset, which outperforms the pseudo-label uncertainty-only model achieving 4.36% WER. We observe that applying soft loss attenuation is crucial for UNCAST. Since the proposed uncertainty estimation method includes fine-grained information based on an output probability, soft loss attenuation might be advantageous compared to hard (0-1) attenuation. Lastly, utilizing both model uncertainty and data uncertainty for pseudo-label uncertainty obtains 4.47% WER on the test-clean dataset, and the model attains 4.52% WER on the test-clean dataset without data uncertainty. Note that the simplest version of the proposed method without in-training uncertainty, soft loss attenuation, and data uncertainty even surpasses the DUST method which attains 4.67% WER on the test-clean dataset.

Hyperparameter search. We experiment on the hyperparameter $\alpha \in \{0.1, 0.2, 0.3\}$ for Equation 11, 12. α controls the contribution of the in-training uncertainty-based regularization term concerning the CTC-based loss functions. As shown in Figure 2, setting α low as 0.1 leads to the student model's little concentration on uncertainty-aware training and thus results in decreased performances compared to $\alpha = 0.2$. As opposed to this, $\alpha = 0.3$ starts to distract the training process of CTC loss showing a higher WER



(a) MCD-ED test-clean dataset (b) UNCAST test-clean dataset

Figure 3: Uncertainty estimation results for the LS-360 dataset with the seed model. MCD-ED and the proposed uncertainty estimation method are evaluated. The sorted data samples based on uncertainty are binned, and then WER (%) for each bin is evaluated.



(a) MCD-ED test-other dataset (b) UNCAST test-other dataset

Figure 4: Uncertainty estimation results of the LS-500 dataset with the seed model. MCD-ED and the proposed uncertainty estimation method are evaluated. The sorted data samples based on uncertainty are binned, and then WER (%) for each bin is evaluated.

than $\alpha = 0.2$. However, for all $\alpha \in \{0.1, 0.2, 0.3\}$, UNCAST achieves the lowest WER by outperforming the baseline method and DUST for both the test-clean dataset and the test-other dataset; this shows stability and effectiveness of the proposed method. Following the results, we use $\alpha = 0.2$ for the other experiments.

Design choice. Table 4 shows the experimental results of the scaling effect for soft loss attenuation of the proposed method. Following the CTC loss function which is based on negative log probability, the proposed method directly adopts the CTC loss function for soft loss attenuation as shown in Equation 6 and 7. We compare logarithmic scale-based and linear scale-based soft loss attenuation. As a result, we observe consistent performance improvements across both scaling methods, which supports the stability of the proposed method.

6.2 Uncertainty Analysis

We analyze the proposed uncertainty estimation method in terms of the predictability of the performance. Since we focus on ASR, WER is utilized to present the performance of the output prediction in the analysis. We utilize the seed model which is trained on the labeled LS-100 dataset to estimate uncertainty of the two unlabeled datasets, LS-360 and LS-500. First of all, the seed model predicts the transcription of the given input, and MCD-ED or the proposed uncertainty estimation method estimate uncertainty of each predicted transcription. Then, we sort the data samples using the estimated uncertainty of each sample. Since uncertainty is believed to estimate the quality of the prediction, low uncertainty should indicate low WER. To quantify this, we utilize binning on the sorted samples, followed by measuring WER within each bin. We use 70 bins for experiments. As shown in Figure 3 and 4, the proposed uncertainty estimation method shows aligned WER along with the order of bins for both the LS-360 and LS-500 datasets. However, MCD-ED outputs zero uncertainty until the 21st bin for the LS-360 dataset and the 14th bin for the LS-500 dataset, resulting in the same WER across those bins. Since the MCD-ED method is based on textual edit distance, MCD-ED is relatively coarse-grained. On the contrary, the proposed method shows fine-grained uncertainty estimation results because it is based on output probability which includes rich information. The fine-granularity of the proposed method is particularly advantageous for soft loss attenuation of UNCAST as shown in Table 3 because estimated uncertainties of pseudo-labels are directly applied to re-weight the loss function.

In addition, the proposed uncertainty estimation method is differentiable so that the estimated uncertainty of the student model (i.e., in-training uncertainty) is optimized instantly by being utilized for the loss function. In short, the proposed uncertainty estimation method is fine-grained and differentiable, which is useful for self-training in that the estimated uncertainty can be directly and effectively applied to the loss function of self-training.

7 Conclusion

We propose a sequence-level uncertainty estimation method for CTC-based ASR models based on the output probability. We consider data uncertainty and model uncertainty simultaneously, which is the first attempt for CTC-based ASR models to the best of our knowledge. Also, we categorize teacher uncertainty and student uncertainty and incorporate these into the uncertainty-aware self-training framework. We experimentally verify the effectiveness of the proposed method for ASR semi-supervised learning. Future work can analyze the disentanglement of data uncertainty and model uncertainty or applicability for other sequence prediction tasks such as text recognition.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) [No.RS-2021-II211343, Artificial Intelligence Graduate School Program (Seoul National University), 1/3], [No.RS-2022-II220320, Artificial intelligence research about cross-modal dialogue modeling for one-on-one multi-modal interactions, 1/3], and

[No.RS-2022-II220641, XVoice: Multi-Modal Voice Meta Learning, 1/3].

References

Baevski, A.; Zhou, Y.; Mohamed, A.; and Auli, M. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33: 12449–12460.

Berthelot, D.; Carlini, N.; Cubuk, E. D.; Kurakin, A.; Sohn, K.; Zhang, H.; and Raffel, C. 2020. ReMixMatch: Semi-Supervised Learning with Distribution Matching and Augmentation Anchoring. In *International Conference on Learning Representations*.

Berthelot, D.; Carlini, N.; Goodfellow, I.; Papernot, N.; Oliver, A.; and Raffel, C. A. 2019. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32.

Chen, S.; Wang, C.; Chen, Z.; Wu, Y.; Liu, S.; Chen, Z.; Li, J.; Kanda, N.; Yoshioka, T.; Xiao, X.; et al. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6): 1505–1518.

Chen, Y.; Wang, W.; and Wang, C. 2020. Semi-Supervised ASR by End-to-End Self-Training. In *Proc. Interspeech 2020*, 2787–2791.

Dawalatabad, N.; Khurana, S.; Laurent, A.; and Glass, J. 2023. On unsupervised uncertainty-driven speech pseudo-label filtering and model calibration. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

Dey, S.; Motlicek, P.; Bui, T.; and Dernoncourt, F. 2019. Exploiting Semi-Supervised Training Through a Dropout Regularization in End-to-End Speech Recognition. In *Proc. Interspeech 2019*, 734–738.

Dong, L.; Xu, S.; and Xu, B. 2018. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 5884–5888. IEEE.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, 1050–1059. PMLR.

Gong, Y.; Chung, Y.-A.; and Glass, J. 2021. AST: Audio Spectrogram Transformer. In *Proc. Interspeech 2021*, 571–575.

Graves, A.; Fernández, S.; Gomez, F.; and Schmidhuber, J. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, 369–376.

Higuchi, Y.; Moritz, N.; Roux, J. L.; and Hori, T. 2021. Momentum Pseudo-Labeling for Semi-Supervised Speech Recognition. In *Proc. Interspeech 2021*, 726–730.

Higuchi, Y.; Ogawa, T.; Kobayashi, T.; and Watanabe, S. 2023. Intermpl: Momentum Pseudo-Labeling With Intermediate CTC Loss. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

Hwang, D.; Misra, A.; Huo, Z.; Siddhartha, N.; Garg, S.; Qiu, D.; Sim, K. C.; Strohman, T.; Beaufays, F.; and He, Y. 2022. Large-scale asr domain adaptation using self- and semi-supervised learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6627–6631. IEEE.

Jiao, W.; Wang, X.; Tu, Z.; Shi, S.; Lyu, M.; and King, I. 2021. Self-Training Sampling with Monolingual Data Uncertainty for Neural Machine Translation. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2840–2850. Online: Association for Computational Linguistics.

Kahn, J.; Lee, A.; and Hannun, A. 2020. Self-training for end-to-end speech recognition. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7084–7088. IEEE.

Kendall, A.; and Gal, Y. 2017. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30.

Khurana, S.; Laurent, A.; and Glass, J. 2022. Magic dust for cross-lingual adaptation of monolingual wav2vec-2.0. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6647–6651. IEEE.

Khurana, S.; Moritz, N.; Hori, T.; and Le Roux, J. 2021. Unsupervised domain adaptation for speech recognition via uncertainty driven self-training. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6553–6557. IEEE.

Kim, H. Y.; Kim, B.-Y.; Yoo, S. W.; Lim, Y.; Lim, Y.; and Lee, H. 2023. ASBERT: Asr-specific self-supervised learning with self-training. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, 9–14. IEEE.

Kingma, D.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*. San Diega, CA, USA.

Laine, S.; and Aila, T. 2017. Temporal Ensembling for Semi-Supervised Learning. In *International Conference on Learning Representations*.

Lakshminarayanan, B.; Pritzel, A.; and Blundell, C. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30.

Li, T.; Meng, Q.; and Sun, Y. 2023. Improved Noisy Iterative Pseudo-Labeling for Semi-Supervised Speech Recognition. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, 167–173. IEEE.

Manohar, V.; Likhomanenko, T.; Xu, Q.; Hsu, W.-N.; Collobert, R.; Saraf, Y.; Zweig, G.; and Mohamed, A. 2021. Kaizen: Continuously improving teacher using exponential moving average for semi-supervised speech recognition. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 518–525. IEEE.

Mukhoti, J.; Kirsch, A.; van Amersfoort, J.; Torr, P. H.; and Gal, Y. 2023. Deep deterministic uncertainty: A new simple baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24384–24394.

Panayotov, V.; Chen, G.; Povey, D.; and Khudanpur, S. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 5206–5210. IEEE.

Park, D. S.; Chan, W.; Zhang, Y.; Chiu, C.-C.; Zoph, B.; Cubuk, E. D.; and Le, Q. V. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*.

Park, D. S.; Zhang, Y.; Jia, Y.; Han, W.; Chiu, C.-C.; Li, B.; Wu, Y.; and Le, Q. V. 2020. Improved Noisy Student Training for Automatic Speech Recognition. In *Proc. Interspeech 2020*, 2817–2821.

Patel, G.; Allebach, J. P.; and Qiu, Q. 2023. Sequps: Sequential uncertainty-aware pseudo-label selection for semi-supervised text recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 6180–6190.

Peng, Y.; Sudo, Y.; Shakeel, M.; and Watanabe, S. 2024. OWSM-CTC: An Open Encoder-Only Speech Foundation Model for Speech Recognition, Translation, and Language Identification. *arXiv preprint arXiv:2402.12654*.

Radford, A.; Kim, J. W.; Xu, T.; Brockman, G.; McLeavey, C.; and Sutskever, I. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, 28492–28518. PMLR.

Rumberg, L.; Gebauer, C.; Ehlert, H.; Wallbaum, M.; Lüdtke, U.; and Ostermann, J. 2023. Uncertainty Estimation for Connectionist Temporal Classification Based Automatic Speech Recognition. In *Proc. INTERSPEECH 2023*, 4583–4587.

Singh, S.; Hou, F.; and Wang, R. 2023. A Novel Self-training Approach for Low-resource Speech Recognition. In *Proc. INTERSPEECH 2023*, 1588–1592.

Sohn, K.; Berthelot, D.; Carlini, N.; Zhang, Z.; Zhang, H.; Raffel, C. A.; Cubuk, E. D.; Kurakin, A.; and Li, C.-L. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33: 596–608.

Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1): 1929–1958.

Sun, C.; Song, M.; Cai, D.; Zhang, B.; Hong, S.; and Li, H. 2022. Confidence-guided learning process for continuous classification of time series. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 4525–4529.

Tarvainen, A.; and Valpola, H. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30.

Tripathi, A.; Khorram, S.; Lu, H.; Kim, J.; Zhang, Q.; and Sak, H. 2024. Monte Carlo Self-Training for Speech Recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 12802–12806. IEEE.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Vyas, A.; Dighe, P.; Tong, S.; and Bourlard, H. 2019. Analyzing uncertainties in speech recognition using dropout. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6730–6734. IEEE.

Xie, Q.; Luong, M.-T.; Hovy, E.; and Le, Q. V. 2020. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10687–10698.

Xu, Q.; Baevski, A.; Likhomanenko, T.; Tomasello, P.; Conneau, A.; Collobert, R.; Synnaeve, G.; and Auli, M. 2021. Self-training and pre-training are complementary for speech recognition. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3030–3034. IEEE.

Xu, Q.; Likhomanenko, T.; Kahn, J.; Hannun, A.; Synnaeve, G.; and Collobert, R. 2020. Iterative Pseudo-Labeling for Speech Recognition. In *Proc. Interspeech 2020*, 1006–1010.

Zhai, X.; Kolesnikov, A.; Houlsby, N.; and Beyer, L. 2022. Scaling vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12104–12113.

Zhang, B.; Wang, Y.; Hou, W.; Wu, H.; Wang, J.; Okumura, M.; and Shinozaki, T. 2021. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34: 18408–18419.

Zhang, Y.; Park, D. S.; Han, W.; Qin, J.; Gulati, A.; Shor, J.; Jansen, A.; Xu, Y.; Huang, Y.; Wang, S.; et al. 2022. Bigssl: Exploring the frontier of large-scale semi-supervised learning for automatic speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 16(6): 1519–1532.

Zhou, Y.; Yang, B.; Wong, D. F.; Wan, Y.; and Chao, L. S. 2020. Uncertainty-Aware Curriculum Learning for Neural Machine Translation. In Jurafsky, D.; Chai, J.; Schluter, N.; and Tetreault, J., eds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6934–6944. Online: Association for Computational Linguistics.

Zhu, H.; Gao, D.; Cheng, G.; Povey, D.; Zhang, P.; and Yan, Y. 2023. Alternative pseudo-labeling for semi-supervised automatic speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.