INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Minimum Word Error Rate Training with Language Model Fusion for End-to-End Speech Recognition

Zhong Meng[1], *Yu Wu*[2], *Naoyuki Kanda*[1], *Liang Lu*[1], *Xie Chen*[1], *Guoli Ye*[1],
Eric Sun[1], *Jinyu Li*[1], *Yifan Gong*[1]

[1]Microsoft Corporation, Redmond, WA, USA
[2]Microsoft Research Asia, Beijing, China

{zhme,yuwu1,nakanda,lial,xieche,guoye,ersun,jinyli,ygong}@microsoft.com

Abstract

Integrating external language models (LMs) into end-to-end (E2E) models remains a challenging task for domain-adaptive speech recognition. Recently, internal language model estimation (ILME)-based LM fusion has shown significant word error rate (WER) reduction from Shallow Fusion by subtracting a weighted internal LM score from an interpolation of E2E model and external LM scores during beam search. However, on different test sets, the optimal LM interpolation weights vary over a wide range and have to be tuned extensively on well-matched validation sets. In this work, we perform LM fusion in the minimum WER (MWER) training of an E2E model to obviate the need for LM weights tuning during inference. Besides MWER training with Shallow Fusion (MWER-SF), we propose a novel MWER training with ILME (MWER-ILME) where the ILME-based fusion is conducted to generate N-best hypotheses and their posteriors. Additional gradient is induced when internal LM is engaged in MWER-ILME loss computation. During inference, LM weights pre-determined in MWER training enable robust LM integrations on test sets from different domains. Experimented with 30K-hour trained transformer transducers, MWER-ILME achieves on average 8.8% and 5.8% relative WER reductions from MWER and MWER-SF training, respectively, on 6 different test sets.

Index Terms: speech recognition, language model, transformer-transducer, end-to-end models

1. Introduction

End-to-end (E2E) models have achieved state-of-the-art performance for automatic speech recognition (ASR) [1, 2, 3]. The most popular E2E models include connectionist temporal classification (CTC) [4, 5, 6], recurrent neural network transducer (RNN-T) [7, 8, 9] and attention-based encoder-decoder (AED) models [10, 11, 12, 13]. E2E models are commonly trained to maximize the log posteriors of token sequences given speech sequences while the ASR performance is measured by the word error rate (WER). Therefore, a minimum WER (MWER) criterion was proposed to train CTC [14], AED [15], RNN-T [16, 17] and hybrid autoregressive transducer (HAT) [18] models, leading to improved ASR performance.

However, an E2E model is more susceptible to domain shift [19] from training to testing than a hybrid system [20]. Numerous methods have been proposed to adapt ASR models, such as regularization methods [21, 22, 23, 24], teacher-student learning [25, 26, 27, 28], transformation methods [29, 30], and adversarial learning [31, 32, 33, 34]. However, all these methods require audio as the adaptation data when applied to E2E models [35, 36, 37]. A promising solution is to integrate an external language model (LM) into the E2E model during inference or during MWER training [38, 16, 39]. With no clear separation of acoustic and language models in an E2E model, LM fusion remains to be a challenging task.

Shallow Fusion [5, 40, 41] is a simple yet effective LM fusion method which combines the probabilities of an E2E model and an LM through a log-linear interpolation at each step of beam search. The Density Ratio method [42, 43] improves Shallow Fusion by subtracting a source-domain LM score from the Shallow Fusion. Recently, inspired by HAT [44], we proposed an internal LM estimation (ILME)-based Fusion [45] which estimates the internal LM score of an E2E model and subtracts it from the Shallow Fusion score. ILME-based Fusion has shown consistent WER reductions from both Shallow Fusion and Density Ratio methods. Further, we proposed an internal LM training (ILMT) [46] that minimizes an internal LM loss in addition to the standard E2E loss. ILMT enables a more effective ILME-based Fusion during inference. However, with these methods, large WER reductions are only obtained when we carefully tune the LM interpolation weights on well-matched validation sets because the optimal LM weights often fluctuate dramatically on different test sets.

In this work, we perform LM fusion in the MWER training of E2E models to obviate the need for LM weights tuning. We first apply Shallow Fusion to generate the N-best hypotheses and compute their posteriors for MWER training (i.e., MWER-SF). Note that our MWER-SF differs from [38] in that the E2E and LM scores are interpolated in the log domain and the combined scores are re-normalized over N-best hypotheses. Further, we propose a MWER training with ILME (MWER-ILME) in which the N-best hypotheses are generated by an ILME-based Fusion and their posteriors are computed by the probabilities of the E2E model, internal LM and external LM. The participation of internal LM in MWER-ILME loss computation induces additional gradient. During inference, the LM fusion with LM weights preset in training are expected to achieve steady performance improvement on multiple different test sets. Experimented with a 30 thousand (K)-hour trained transformer transducer, MWER-ILME achieves on average 8.8% and 5.8% relative WER reductions from MWER and MWER-SF, respectively, on a multi-domain test set, and the reductions are consistent over all subsets from different domains.

2. Related Work

2.1. Minimum Word Error Rate Training

Given a sequence of speech features $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$ where $\mathbf{x}_t \in \mathbb{R}^d$, an E2E model is typically trained to minimize the negative log posterior of the reference token sequence $\mathbf{Y}^* = \{y_1^*, \ldots, y_U^*\}$ where $y_u^* \in \mathcal{V}$ and $\mathcal{V}$ is the set of output tokens

$$\mathcal{L}_{\text{E2E}}(\mathbf{X}, \mathbf{Y}^*) = -\log P(\mathbf{Y}^*|\mathbf{X}; \theta_{\text{E2E}}^{\text{S}}), \qquad (1)$$

where $\theta_{\text{E2E}}^{\text{S}}$ is the parameters of a source-domain E2E model.

http://dx.doi.org/10.21437/Interspeech.2021-2075

With MWER training [14, 15, 16], the E2E model is further fine-tuned to directly minimize the expected number of word errors on the training corpus. As a closer objective to the ASR performance metric, the MWER training is expected to outperform the standard E2E training. The MWER loss is commonly approximated as the expected number of word errors over the top N hypotheses $\{\mathbf{Y}^1, \ldots, \mathbf{Y}^N\}$ of $\mathbf{X}$ as follows

$$\mathcal{L}_{\text{MWER}} = \sum_{n=1}^{N} \bar{P}(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}) R(\mathbf{Y}^n, \mathbf{Y}^*), \tag{2}$$

where $\bar{P}(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}) = \frac{P(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}})}{\sum_{n=1}^{N} P(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}})}$ is the re-normalized posterior over the N-best hypotheses, and $R(\mathbf{Y}^n, \mathbf{Y}^*)$ is the number of word errors in a hypothesis $\mathbf{Y}^n$ compared to the reference $\mathbf{Y}^*$.

2.2. Shallow Fusion

During inference, Shallow Fusion [5, 40] searches for an optimal hypothesis $\hat{\mathbf{Y}}$ that maximizes a log-linear interpolation between the E2E model and the external LM probabilities.

$$\hat{\mathbf{Y}} = \arg\max_{\mathbf{Y}} \Big[\log P(\mathbf{Y}|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}) + \lambda_{\text{T}} \log P(\mathbf{Y}; \theta^{\text{T}}_{\text{LM}})\Big], \tag{3}$$

where $P(\mathbf{Y}; \theta^{\text{T}}_{\text{LM}})$ is the probability of an external LM trained with target-domain text, and λ_{T} is the interpolation weight for the external LM.

2.3. Internal LM Estimation-based Fusion

An E2E model implicitly learns an internal LM $P(\mathbf{Y}^n; \theta^{\text{S}}_{\text{E2E}})$ that characterizes the distribution of source-domain training text. The internal LM probability $P(\mathbf{Y}; \theta^{\text{S}}_{\text{E2E}})$ can be estimated as the output of an E2E model by removing the contribution of the acoustic encoder (with parameters $\theta^{\text{S}}_{\text{enc}}$) [44, 45].

With the internal LM, we are able to compute the hypothesis posterior predicted by a target-domain E2E model $\theta^{\text{T}}_{\text{E2E}}$ with Bayes' Theorem as follows [45]:

$$P(\mathbf{Y}|\mathbf{X}; \theta^{\text{T}}_{\text{E2E}}) = P(\mathbf{Y}|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}) \frac{P(\mathbf{Y}; \theta^{\text{T}}_{\text{LM}})^{\lambda_{\text{T}}}}{P(\mathbf{Y}; \theta^{\text{S}}_{\text{E2E}})^{\lambda_{\text{S}}}} g(\mathbf{X}), \tag{4}$$

where $g(\mathbf{X}) = P(\mathbf{X}; \theta^{\text{S}}_{\text{E2E}})/P(\mathbf{X}; \theta^{\text{T}}_{\text{E2E}}) = P_{\text{S}}(\mathbf{X})/P_{\text{T}}(\mathbf{X})$ is the ratio of acoustic priors shared by all hypotheses. ILME-based Fusion [45] searches for the optimal hypothesis $\hat{\mathbf{Y}}$ with highest target-domain posterior $P(\mathbf{Y}|\mathbf{X}; \theta^{\text{T}}_{\text{E2E}})$ as follows:

$$\hat{\mathbf{Y}} = \arg\max_{\mathbf{Y}} \Big[\log P(\mathbf{Y}|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}) + \lambda_{\text{T}} \log P(\mathbf{Y}; \theta^{\text{T}}_{\text{LM}}) - \lambda_{\text{S}} \log P(\mathbf{Y}; \theta^{\text{S}}_{\text{E2E}})\Big], \tag{5}$$

where λ_{S} is the interpolation weight for the internal LM.

In [46], ILMT was proposed to encourage the acoustically-conditioned LM of an E2E model with parameters $\theta^{\text{S}}_{\text{E2E}} \backslash \theta^{\text{S}}_{\text{enc}}$ (e.g., the predictor and joint network of a transducer [47], the decoder of AED [48, 49]) to also behave like a standalone LM, and thus facilitates a better external LM integration. In ILMT training, an internal LM loss below is jointly minimized with the E2E loss $\mathcal{L}_{\text{E2E}}(\mathbf{X}, \mathbf{Y}^n)$ by updating the E2E model

$$\begin{aligned}\mathcal{L}_{\text{ILM}}(\mathbf{Y}^n) &= -\log P(\mathbf{Y}; \theta^{\text{S}}_{\text{E2E}}) \\ &= -\sum_{u=1}^{U+1} \log P(y_u|\mathbf{Y}_{0:u-1}; \theta^{\text{S}}_{\text{E2E}} \setminus \theta^{\text{S}}_{\text{enc}}),\end{aligned} \tag{6}$$

3. Minimum Word Error Rate Training with LM Fusion

LM Fusion has remarkably improved the ASR performance of an E2E model with optimal LM weights carefully tuned on a validation set. However, the optimal LM interpolation weights fluctuate dramatically on different test sets. It may even happen that the optimal LM weights on one test set degrade the E2E model performance on the other set. Therefore, for each test set, the search for optimal LM weights relies on extensive tuning on a well-matched validation set, taking huge computational resources and human efforts. In many scenarios, such compatible validation sets are not even available. To obviate the need for LM weight tuning, we perform LM fusion in MWER training of E2E models.

3.1. MWER Training with Shallow Fusion

In [38], Shallow Fusion is performed during MWER training to generate N-best hypotheses $\{\mathbf{Y}^1, \ldots, \mathbf{Y}^N\}$ of $\mathbf{X}$. The posterior of each hypothesis is computed by a linear combination of E2E model and LM probabilities. Different from [38], the hypothesis posterior in this work is obtained by interpolating the log-probabilities of the E2E model and the external LM to better match the Shallow Fusion inference, and the word errors of each hypothesis is weighted by a re-normalized probability. Therefore, the MWER-SF loss is formulated as

$$\mathcal{L}^{\text{SF}}_{\text{MWER}} = \sum_{n=1}^{N} \bar{P}(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}, \theta^{\text{T}}_{\text{LM}}) R(\mathbf{Y}^n, \mathbf{Y}^*), \tag{7}$$

where $\bar{P}(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{T}}_{\text{E2E}}, \theta^{\text{T}}_{\text{LM}})$ is the re-normalized Shallow Fusion probability over N-best hypotheses below

$$\bar{P}(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}, \theta^{\text{T}}_{\text{LM}}) = \frac{P(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}) P(\mathbf{Y}^n|\theta^{\text{T}}_{\text{LM}})^{\lambda_{\text{T}}}}{\sum_{i=1}^{N} P(\mathbf{Y}^i|\mathbf{X}; \theta^{\text{S}}_{\text{E2E}}) P(\mathbf{Y}^i|\theta^{\text{T}}_{\text{LM}})^{\lambda_{\text{T}}}}. \tag{8}$$

During inference, the pre-defined LM weight in MWER-SF training is used for Shallow Fusion. MWER-SF adapts the E2E model to a fixed external LM and its interpolation weight, and mitigates the importance of the LM weight tuning.

3.2. MWER Training with Internal LM Estimation

We propose an MWER training of E2E models with ILME (MWER-ILME) by performing ILME-based Fusion in MWER training. MWER-ILME differs from standard MWER training in that: 1) we apply ILME-based Fusion to generate the N-best hypotheses of training utterances; 2) we compute hypothesis posteriors using probabilities of the E2E model, internal LM and external LM.

Therefore, an MWER-ILME loss function is defined as:

$$\mathcal{L}^{\text{ILME}}_{\text{MWER}} = \sum_{n=1}^{N} \bar{P}(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{T}}_{\text{E2E}}) R(\mathbf{Y}^n, \mathbf{Y}^*). \tag{9}$$

Here, $\{\mathbf{Y}^1, \ldots, \mathbf{Y}^N\}$ are the N-best hypotheses of $\mathbf{X}$ generated by ILME-based Fusion performed with a source-domain E2E model, an internal LM and a fixed external LM. $\bar{P}(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{T}}_{\text{E2E}})$ is the re-normalized posterior over N-best hypotheses below:

$$\bar{P}(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{T}}_{\text{E2E}}) = \frac{P(\mathbf{Y}^n|\mathbf{X}; \theta^{\text{T}}_{\text{E2E}})}{\sum_{i=1}^{N} P(\mathbf{Y}^i|\mathbf{X}; \theta^{\text{T}}_{\text{E2E}})}. \tag{10}$$

The posteriors of N-best hypotheses are computed by Eq. (4) as

$$P(\mathbf{Y}^n|\mathbf{X};\theta^{\text{T}}_{\text{E2E}}) = P(\mathbf{Y}^n|\mathbf{X};\theta^{\text{S}}_{\text{E2E}})\frac{P(\mathbf{Y}^n;\theta^{\text{T}}_{\text{LM}})^{\lambda_{\text{T}}}}{P(\mathbf{Y}^n;\theta^{\text{S}}_{\text{E2E}})^{\lambda_{\text{S}}}}g(\mathbf{X}). \quad (11)$$

MWER-ILME training is essentially minimizing the expected number of word errors over the N-best hypotheses generated by a target-domain E2E model $\theta^{\text{T}}_{\text{E2E}}$.

With Eqs. (9) to (11), we express the MWER-ILME loss by $P(\mathbf{Y}^n|\mathbf{X};\theta^{\text{S}}_{\text{E2E}})$, $P(\mathbf{Y}^n;\theta^{\text{T}}_{\text{LM}})$ and $P(\mathbf{Y}^n;\theta^{\text{S}}_{\text{E2E}})$. Note that, during MWER-ILME training, the only learnable parameters are θ^{S}_{E2E}, and the inclusion of internal LM $P(\mathbf{Y}^n;\theta^{\text{S}}_{\text{E2E}})$ in loss function results in additional gradient compared to MWER training.

The derivative of MWER-ILME loss with respect to $\theta^{\text{S}}_{\text{E2E}}$ is

$$\frac{\partial \mathcal{L}^{\text{ILME}}_{\text{MWER}}}{\partial \theta^{\text{S}}_{\text{E2E}}} = \sum_{n=1}^{N} \frac{\partial \mathcal{L}^{\text{ILME}}_{\text{MWER}}}{\partial \log P(\mathbf{Y}^n|\mathbf{X};\theta^{\text{T}}_{\text{E2E}})} \frac{\partial \log P(\mathbf{Y}^n|\mathbf{X};\theta^{\text{T}}_{\text{E2E}})}{\partial \theta^{S}_{\text{E2E}}}. \quad (12)$$

In Eq. (12), the gradient of MWER-ILME loss with respect to the target-domain hypothesis posterior is computed as

$$\frac{\partial \mathcal{L}^{\text{ILME}}_{\text{MWER}}}{\partial \log P(\mathbf{Y}^n|\mathbf{X};\theta^{\text{T}}_{\text{E2E}})} = \bar{P}(\mathbf{Y}^n|\mathbf{X};\theta^{\text{T}}_{\text{E2E}})\left[R(\mathbf{Y}^n,\mathbf{Y}^*) - \bar{R}\right], \quad (13)$$

where $\bar{R} = \sum_{n=1}^{N} \bar{P}(\mathbf{Y}^n|\mathbf{X};\theta^{\text{T}}_{\text{E2E}})R(\mathbf{Y}^n,\mathbf{Y}^*)$ is the expected number of word errors over the N-best hypotheses. Given Eq. (4), the gradient of the target-domain hypothesis posterior with respect to θ^{S}_{E2E} in Eq. (12) can be expressed as

$$\begin{aligned} &\frac{\partial \log P(\mathbf{Y}^n|\mathbf{X};\theta^{\text{T}}_{\text{E2E}})}{\partial \theta^{\text{S}}_{\text{E2E}}} \\ &= \frac{\partial \log P(\mathbf{Y}^n|\mathbf{X};\theta^{\text{S}}_{\text{E2E}})}{\partial \theta^{\text{S}}_{\text{E2E}}} - \lambda_{\text{S}} \frac{\partial \log P(\mathbf{Y}^n;\theta^{\text{S}}_{\text{E2E}})}{\partial \theta^{\text{S}}_{\text{E2E}}} \\ &= -\frac{\partial \mathcal{L}_{\text{E2E}}(\mathbf{X},\mathbf{Y}^n)}{\partial \theta^{\text{S}}_{\text{E2E}}} + \lambda_{\text{S}} \frac{\partial \mathcal{L}_{\text{ILM}}(\mathbf{Y}^n)}{\partial \left(\theta^{\text{S}}_{\text{E2E}} \setminus \theta^{\text{S}}_{\text{enc}}\right)}, \end{aligned} \quad (14)$$

where $\mathcal{L}_{\text{E2E}}(\mathbf{X},\mathbf{Y}^n)$ and $\mathcal{L}_{\text{ILM}}(\mathbf{Y}^n)$ are E2E and ILM losses in Eq. (1) and Eq. (6), respectively. $\mathcal{L}_{\text{E2E}}$ for AED and $\mathcal{L}_{\text{ILM}}$ are both cross-entropy losses with simple gradients. For a transducer model, the gradient of $\mathcal{L}_{\text{E2E}}$ is derived in [7]. Therefore, the gradient of MWER-ILME loss with respect to θ^{S}_{E2E} is obtained by substituting Eqs. (13) and (14) into Eq. (12).

Through MWER-ILME training, we adapt the E2E model towards a fixed external LM along with the internal LM weight λ_{S} and external LM weight λ_{T}. Therefore, during evaluation, the E2E model adapted by a *multi-domain* LM is expected to get robust WER reductions on multiple different test sets if ILME-based Fusion is performed with the same set of LM weights.

4. Experiments

In this work, we train streaming transformer transducer (T-T) models [47] to minimize transducer [7], MWER [15], MWER-SF and MWER-ILME losses, and evaluate it on a multi-domain test set. A multi-domain external LM is used in LM fusion for both MWER-SF and MWER-ILME training and evaluation. We also investigate the generalizability of MWER-SF and MWER-ILME training by evaluating with an out-of-domain LM on an out-of-domain test set.

In all experiments, beam search and N-best generation are both performed with a beam size of 5. 3999 word-piece units generated by byte-pair encoding [50] are used as the output token set $\mathcal{V}$ for both T-T and the external LM. A word-count weighted average of subset WERs is computed for each test set.

4.1. Language Models Training

We use the same external LMs as in [45, 46].

Multi-domain LM: We first train a uni-directional long short-term memory (LSTM) [51, 52, 53] LM on 2 billion words of anonymized text corpus comprising conversational data such as talks, interviews, meeting transcripts, and short message dictation from Microsoft services. The multi-domain LM has 2 hidden layers with 2048 units for each layer and in total 58 million (M) parameters.

LibriSpeech LM: We train another LSTM-LM with the 9.4M-word transcript of the 960-hour training speech and the 813M-word text in LibriSpeech corpus [54]. The LibriSpeech LM has exactly the same architecture as the multi-domain LM.

4.2. Transformer Transducer Models Training

We train T-T models with 30 thousand (K) hours of anonymized and transcribed data as in [45, 46] collected from Microsoft services, including voice search, short message dictation, conversations, command and control recorded in various conditions.

The T-T encoder starts with four 2D convolution layers that sub-sample the input 80-dimensional (dim) log Mel-filter bank features in time by a factor of 4 and project them to 512 dimensions. This is followed by a transformer with 18 layers, each layer has an 8-head attention sub-layer with relative positional encoding [55] and a 2048-dim fully-connected sub-layer. The encoder has a look-ahead of 360 ms on average. The T-T predictor is a 2-layer transformer with each layer containing a 4-head attention sub-layer followed by a 1024-dim fully-connected sub-layer. The input to the predictor are 512-dim word-piece embedding vectors with positional encoding. The attention dimension is fixed at 512 for both encoder and predictor. The outputs of encoder and predictor are projected to 512-dim vectors. Dropout [56] with a probability of 0.1 is deployed in both the encoder and the predictor. The T-T model has 67 million (M) parameters.

First, a T-T model is trained to minimize the transducer loss with the 30K-hour data. The baseline T-T is then fine-tuned with the same data to minimize MWER, MWER-SF and MWER-ILME losses using Adam optimizer [57] with a constant learning rate of 10^{-5}. The multi-domain LM is used as the external LM for both MWER-SF and MWER-ILME training. In MWER-SF training, the external LM weight is set to 0.25. In MWER-ILME training, the external and internal LM weights are set to 0.25 and 0.05, respectively.

4.3. Multi-Domain Evaluations

We collect a multi-domain test set containing 6 subsets from domains covered by the multi-domain LM to evaluate T-Ts trained in Section 4.2 as follows.

Call: 30K-word conversational speech collected from real online phone calls.

Meeting: 5K-word conversational speech collected from real meetings.

Search: 29K-word voice search speech collected from multiple microphone arrays.

Keyboard: 15K-word dictated speech collected from keyboard voice input.

Email: 9K-word dictated speech from email applications.

Common: the Common Voice [58] test set consisting of 125k-word spoken text from blog posts, old books, movies, etc. Except for Common, the other five anonymized subsets are all collected from Microsoft services.

Table 1: *WERs (%) of T-T models trained with 4 different objectives on a multi-domain test set including 6 subsets. WERR (%) is the relative WER reduction of MWER-ILME from MWER.*

Test Subset	T-T	MWER	MWER-SF	MWER-ILME	WERR
Call	8.70	8.37	7.70	**7.58**	9.4
Meeting	16.34	16.07	17.54	**15.87**	1.2
Search	12.56	12.35	12.13	**11.73**	5.0
Keyboard	7.95	7.56	7.60	**7.37**	2.5
Email	19.16	17.67	16.94	**16.31**	7.7
Common	12.43	11.75	11.34	**10.51**	10.6
Avg.	12.00	11.43	11.07	**10.43**	8.8

Table 2: *WERs (%) of MWER trained T-T model on a multi-domain test set. "Oracle Call" uses optimal LM weights tuned on Call subset for LM fusion on all 6 subsets. "Oracle All" uses optimal LM weights tuned on each subset for LM fusion.*

Test Subset	No LM	Oracle Call		Oracle All	
		SF	ILME	SF	ILME
Call	8.37	7.74	7.33	7.74	7.33
Meeting	16.07	18.99	17.98	15.97	15.78
Search	12.35	12.27	12.01	11.97	11.66
Keyboard	7.56	7.92	7.92	7.54	7.37
Email	17.67	17.85	16.55	16.74	16.19
Common	11.75	12.35	11.94	11.33	11.05
Avg.	11.43	11.78	11.37	11.00	10.69

The baseline and MWER-trained T-Ts are evaluated on the muliti-domain test set *without* using external LM. As shown in Table 1, the baseline T-T achieves 12.00% WER on average over 6 subsets. MWER training leads to 4.8% relative WER reduction from the baseline T-T. In Table 2, we evaluate the MWER-trained T-T with Shallow Fusion and ILME-based Fusion. We first tune the LM weights on the Call subset and apply the optimal weights for all 6 subsets (i.e., Oracle Call). Although both LM fusions achieve significant WER reductions on Call and Search, the performance degrades severely on Meeting, Keyboard and Common. This indicates that the optimal LM weights fluctuate dramatically on different test sets. Further, we conduct LM fusions on each subset with the optimal LM weights tuned on itself (i.e., Oracle All). Both LM fusions result in consistent WER reductions on all subsets, with ILME-based Fusion performing better on each subset.

We then evaluate the MWER-SF trained T-T with Shallow Fusion by using the same multi-domain LM and LM weight (0.25) as in training. In Table 1, although MWER-SF improves MWER training by 3.1% relatively in terms of averaged WER, it degrades WERs on Meeting and Keyboard subsets by 9.1% and 5.3% relatively, respectively. Finally, we evaluate the MWER-ILME trained T-T with ILME-based Fusion by using the same multi-domain LM and external/internal LM weights (0.25/0.05) as in training. Among all 4 training objectives, MWER-ILME performs the best on every subset of the multi-domain test set, achieving on average 8.8% and 5.8% relative WER reductions from the MWER and MWER-SF training, respectively. MWER-ILME consistently improves MWER training by 1.2% to 10.6% relatively on 6 subsets with no performance degradation. With MWER-ILME training, ILME-based Fusion is able to achieve significant and robust relative WER reductions on test sets from multiple domains by applying the same multi-domain LM and LM weights during training.

Note that MWER-ILME in Table 1 even outperforms the MWER-trained T-T with ILME-based Fusion and "Oracle All" in Table 2 by 2.4% relatively on average. This shows that, during MWER-ILME, the adaptation of T-T towards external LM also facilitates a more effective fusion between these two models. The need for LM weight tuning is thus entirely eliminated.

Table 3: *WERs (%) of T-T models trained with 4 different objectives on out-of-domain LibriSpeech dev and test sets. WERR (%) is the relative WER reduction of MWER-ILME from MWER.*

Test Set	T-T	MWER	MWER-SF	MWER-ILME	WERR
dev-clean	6.11	6.03	4.46	**4.42**	26.7
dev-other	14.34	13.87	11.29	**11.09**	20.0
test-clean	6.18	5.98	4.46	**4.42**	26.1
test-other	14.51	14.15	11.67	**11.43**	19.2
Avg.	10.21	9.94	7.91	**7.78**	21.7

4.4. Out-of-Domain Evaluations

We further investigate the generalizability of MWER-SF and MWER-ILME trained T-Ts by integrating out-of-domain LMs into them, and performing the evaluation on an out-of-domain test set. We choose LiriSpeech LM and the LibriSpeech test set for out-of-domain evaluation since neither the training data of T-T nor that of the multi-domain LM contains LibriSpeech data. The LibriSpeech test set consists of 4 subsets, "dev-clean", "test-clean", "dev-other", "test-other", including 54K, 53K, 51K and 52K words, respectively [54].

The baseline and MWER-trained T-Ts are evaluated *without* external LM. The MWER-SF and MWER-ILME trained T-Ts are evaluated with Shallow Fusion and ILME-based Fusion, respectively. During evaluation, the LM fusions are performed with LibriSpeech LM and the same LM weights as in training. In Table 3, the MWER trained T-T achieves on average 9.94% WER on LibriSpeech test set, with a 2.6% relative WER reduction from baseline T-T. MWER-SF outperforms MWER training by 20.42% relatively on average, and the improvement is consistent over all subsets. Among all 4 methods, MWER-ILME performs the best on every subset, achieving an on average 21.7% relative WER reduction from MWER training.

The results show that, even performed with a multi-domain LM, neither MWER-SF nor MWER-ILME training biases the T-T entirely towards the domains covered by the external LM. Instead, they equip T-Ts with the generalizability to be integrated with an out-of-domain LM and achieve a significant WER reduction on an out-of-domain test set.

5. Conclusion

In this work, we perform the LM fusion in the MWER training of a T-T model to eliminate the need for LM weight tuning during inference. In addition to Shallow Fusion, we propose to apply ILME-based Fusion for the N-best generation and the hypothesis score computation in the MWER training. With LM weights preset in MWER-ILME training, ILME-based fusion significantly and consistently outperforms MWER and MWER-SF trained models on multiple different test sets, achieving on average 8.8% and 5.8% relative WER reductions, respectively. MWER-ILME even outperforms MWER-trained model with ILME-based Fusion and *oracle* LM weights. With great generalizability, both MWER-SF and MWER-ILME trained E2E models perform more than 20% relatively better than MWER trained model on an out-of-domain test set when integrated with an out-of-domain LM.

6. References

[1] C.-C. Chiu, T. N. Sainath *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models," in *Proc. ICASSP*, 2018.

[2] T. Sainath, Y. He, B. Li *et al.*, "A streaming on-device end-to-end model surpassing server-side conventional model quality and latency," in *Proc. ICASSP*, 2020, pp. 6059–6063.

[3] J. Li, R. Zhao, Z. Meng *et al.*, "Developing RNN-T models surpassing high-performance hybrid models with customization capability," in *Interspeech*, 2020.

[4] A. Graves, S. Fernández, F. Gomez *et al.*, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*. ACM, 2006.

[5] A. Hannun, C. Case *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[6] J. Li, G. Ye, A. Das *et al.*, "Advancing acoustic-to-word CTC model," in *Proc. ICASSP*, 2018.

[7] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[8] Y. He, T. Sainath, R. Prabhavalkar *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *Proc. ICASSP*, 2019.

[9] J. Li, R. Zhao, H. Hu *et al.*, "Improving RNN transducer modeling for end-to-end speech recognition," in *Proc. ASRU*, 2019.

[10] J. K. Chorowski, D. Bahdanau, D. Serdyuk *et al.*, "Attention-based models for speech recognition," in *NIPS*, 2015.

[11] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. ICASSP*. IEEE, 2016.

[12] S. Karita, N. Chen *et al.*, "A comparative study on transformer vs RNN in speech applications," in *Proc. ASRU*, 2019.

[13] J. Li, Y. Wu *et al.*, "On the comparison of popular end-to-end models for large scale speech recognition," in *Proc. Interspeech*, 2020.

[14] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International Conference on Machine Learning*, 2014, pp. 1764–1772.

[15] R. Prabhavalkar, T. N. Sainath, Y. Wu *et al.*, "Minimum word error rate training for attention-based sequence-to-sequence models," in *Proc. ICASSP*. IEEE, 2018.

[16] C. Weng, C. Yu, J. Cui *et al.*, "Minimum bayes risk training of RNN-transducer for end-to-end speech recognition," *arXiv preprint arXiv:1911.12487*, 2019.

[17] J. Guo, G. Tiwari, J. Droppo *et al.*, "Efficient minimum word error rate training of RNN-transducer for end-to-end speech recognition," *arXiv preprint arXiv:2007.13802*, 2020.

[18] L. Lu, Z. Meng, N. Kanda *et al.*, "On minimum word error rate training of the hybrid autoregressive transducer," *arXiv preprint arXiv:2010.12673*, 2020.

[19] P. Bell, J. Fainberg, O. Klejch *et al.*, "Adaptation algorithms for neural network-based speech recognition: An overview," *IEEE Open Journal of Singal Processing*, vol. 2, 2021.

[20] G. Hinton, L. Deng, D. Yu *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, 2012.

[21] D. Yu, K. Yao, H. Su *et al.*, "Kl-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," in *Proc. ICASSP*, May 2013.

[22] Z. Meng, J. Li, and Y. Gong, "Adversarial speaker adaptation," in *Proc. ICASSP*, 2019.

[23] H. Liao, "Speaker adaptation of context dependent deep neural networks," in *Proc. ICASSP*, May 2013.

[24] Z. Meng, H. Hu, J. Li *et al.*, "L-vector: Neural label embedding for domain adaptation," in *Proc. ICASSP*. IEEE, 2020.

[25] J. Li, M. L. Seltzer, X. Wang *et al.*, "Large-scale domain adaptation via teacher-student learning," in *Proc. INTERSPEECH*, 2017.

[26] Z. Meng, J. Li, Y. Gong *et al.*, "Adversarial teacher-student learning for unsupervised domain adaptation," in *Proc. ICASSP*, 2018.

[27] V. Manohar, P. Ghahremani, D. Povey *et al.*, "A teacher-student learning approach for unsupervised domain adaptation of sequence-trained asr models," in *Proc. SLT*. IEEE, 2018.

[28] Z. Meng, J. Li, Y. Zhao, and Y. Gong, "Conditional teacher-student learning," in *Proc. ICASSP*, 2019.

[29] R. Gemello, F. Mana, S. Scanzio *et al.*, "Linear hidden transformations for adaptation of hybrid ANN/HMM models," *Speech Communication*, vol. 49, no. 10, pp. 827 – 835, 2007.

[30] T. Tan, Y. Qian, M. Yin *et al.*, "Cluster adaptive training for deep neural network," in *Proc. ICASSP*. IEEE, 2015.

[31] Y. Shinohara, "Adversarial multi-task learning of deep neural networks for robust speech recognition," in *INTERSPEECH*, 2016.

[32] Z. Meng, J. Li, Z. Chen *et al.*, "Speaker-invariant training via adversarial learning," in *Proc. ICASSP*, 2018.

[33] D. Serdyuk, K. Audhkhasi *et al.*, "Invariant representations for noisy speech recognition," in *NIPS Workshop*, 2016.

[34] Z. Meng, Z. Chen, V. Mazalov, J. Li, and Y. Gong, "Unsupervised adaptation with domain separation networks for robust speech recognition," in *Proc. ASRU*, 2017.

[35] T. Ochiai, S. Watanabe *et al.*, "Speaker adaptation for multichannel end-to-end speech recognition," in *Proc. ICASSP*, 2018.

[36] Z. Meng, Y. Gaur, J. Li *et al.*, "Speaker adaptation for attention-based end-to-end speech recognition," *Proc. Interspeech*, 2019.

[37] Z. Meng, J. Li, Y. Gaur, and Y. Gong, "Domain adaptation via teacher-student learning for end-to-end speech recognition," in *Proc. ASRU*. IEEE, 2019.

[38] C. Peyser, S. Mavandadi, T. Sainath *et al.*, "Improving tail performance of a deliberation e2e asr model using a large text corpus," in *INTERSPEECH*, 2020.

[39] N. Kanda, X. Lu, and H. Kawai, "Minimum bayes risk training of ctc acoustic models in maximum a posteriori based decoding framework," in *Proc. ICASSP*. IEEE, 2017.

[40] C. Gulcehre, O. Firat, K. Xu *et al.*, "On using monolingual corpora in neural machine translation," *CoRR*, 2015.

[41] J. Chorowski and N. Jaitly, "Towards better decoding and language model integration in sequence to sequence models," *CoRR*, 2016.

[42] E. McDermott, H. Sak, and E. Variani, "A density ratio approach to language model fusion in end-to-end automatic speech recognition," in *Proc. ASRU*. IEEE, 2019, pp. 434–441.

[43] N. Kanda, X. Lu, and H. Kawai, "Maximum a posteriori based decoding for CTC acoustic models." in *Interspeech*, 2016.

[44] E. Variani, D. Rybach, C. Allauzen *et al.*, "Hybrid autoregressive transducer (hat)," *arXiv preprint arXiv:2003.07705*, 2020.

[45] Z. Meng, S. Parthasarathy, E. Sun *et al.*, "Internal language model estimation for domain-adaptive end-to-end speech recognition," in *Proc. SLT*. IEEE, 2021.

[46] Z. Meng, N. Kanda, Y. Gaur *et al.*, "Internal language model training for domain-adaptive end-to-end speech recognition," in *Proc. ICASSP*. IEEE, 2021.

[47] X. Chen, Y. Wu, Z. Wang *et al.*, "Developing real-time streaming transformer transducer for speech recognition on large-scale dataset," *arXiv preprint arXiv:2010.11395*, 2020.

[48] Z. Meng, Y. Gaur, J. Li *et al.*, "Character-aware attention-based end-to-end speech recognition," in *Proc. ASRU*, 2019.

[49] Y. Gaur, J. Li, Z. Meng *et al.*, "Acoustic-to-phrase end-to-end speech recognition," in *INTERSPEECH*, 2019.

[50] R. Sennrich *et al.*, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909*, 2015.

[51] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Interspeech*, 2014.

[52] H. Erdogan, T. Hayashi *et al.*, "Multi-channel speech recognition: LSTMs all the way through," in *CHiME-4 workshop*, 2016, pp. 1–4.

[53] Z. Meng, S. Watanabe, J. R. Hershey *et al.*, "Deep long short-term memory adaptive beamforming networks for multichannel robust speech recognition," in *ICASSP*. IEEE, 2017.

[54] V. Panayotov, G. Chen *et al.*, "Librispeech: an asr corpus based on public domain audio books," in *Proc. ICASSP*, 2015.

[55] A. Vaswani, N. Shazeer, N. Parmar *et al.*, "Attention is all you need," in *Proc. NIPS*, vol. 30, 2017.

[56] N. Srivastava, G. Hinton, A. Krizhevsky *et al.*, "Dropout: a simple way to prevent neural networks from overfitting," *in JMLR*, 2014.

[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[58] R. Ardila *et al.*, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.