

MASK THE BIAS: IMPROVING DOMAIN-ADAPTIVE GENERALIZATION OF CTC-BASED ASR WITH INTERNAL LANGUAGE MODEL ESTIMATION

Nilaksh Das, Monica Sunkara, Sravan Bodapati,
Jinglun Cai, Devang Kulshreshtha, Jeff Farris, Katrin Kirchhoff
AWS AI Labs, USA

ABSTRACT

End-to-end ASR models trained on large amount of data tend to be implicitly biased towards language semantics of the training data. Internal language model estimation (ILME) has been proposed to mitigate this bias for autoregressive models such as attention-based encoder-decoder and RNN-T. Typically, ILME is performed by modularizing the acoustic and language components of the model architecture, and eliminating the acoustic input to perform log-linear interpolation with the text-only posterior. However, for CTC-based ASR, it is not as straightforward to decouple the model into such acoustic and language components, as CTC log-posteriors are computed in a non-autoregressive manner. In this work, we propose a novel ILME technique for CTC-based ASR models. Our method iteratively masks the audio timesteps to estimate a pseudo log-likelihood of the internal LM by accumulating log-posteriors for only the masked timesteps. Extensive evaluation across multiple out-of-domain datasets reveals that the proposed approach improves WER by up to 9.8% and OOV F1-score by up to 24.6% relative to Shallow Fusion, when only text data from target domain is available. In the case of zero-shot domain adaptation, with no access to any target domain data, we demonstrate that removing the source domain bias with ILME can still outperform Shallow Fusion to improve WER by up to 9.3% relative.

Index Terms— internal LM estimation, speech recognition, CTC

1. INTRODUCTION

Automatic speech recognition (ASR) has now become tightly integrated with daily life through commonly used real-world applications such as digital assistants, news transcription and AI-based interactive voice response telephony. Often in such practical scenarios, ASR models that are trained on some source domain training data are deployed for other target domains in which target domain training data, especially paired audio-text data, may be severely limited. External language model (LM) integration techniques have been proposed to overcome such cases of domain mismatch, wherein text-only data from the target domain is leveraged [1, 2, 3, 4, 5, 6]. Complementary to adding an external target domain LM to capture target domain semantics, density ratio estimation techniques have been proposed that aim to minimize any bias from the source domain by utilizing an additional source domain LM [7, 8, 9]. Building further on top of this, a growing body of recent works have also looked into internal LM estimation (ILME), aimed at reducing any internal model bias from the source domain by estimating implicit language semantics learned internally by the model itself [10, 11, 12].

ILME proposes to reduce source domain model bias by modularizing the acoustic model (AM) and language model (LM) components of the model architecture, and eliminating the acoustic input altogether. Then, a log-linear interpolation with the overall log-likelihood scores is performed to subtract away the text-only log-posteriors [10].

Table 1. Example of a CTC model’s internal bias. With words masked from input audio, the model still attempts to decode a coherent output.

REF	he is a member of the republican main street partnership
HYP: greedy	he is a member of the republican main street partnership
HYP: mask “the”	he is a member of public and main street partnership
HYP: mask “street”	he is a member of the republican mainstream partnership

Such an ILME approach is fundamentally limited to autoregressive models such as attention-based encoder-decoder (AED) and recurrent neural network transducer (RNN-T), as it is relatively easy to decouple and approximate such architectures into AM and LM components.

Recently, models based on connectionist temporal classification (CTC) loss have demonstrated state-of-the-art ASR performance with high inference speed and low memory footprint [13, 14, 15, 16]. However, the proposed ILME techniques cannot be formulated readily for CTC-based ASR models. Given CTC’s non-autoregressive architecture, decomposing a CTC model into AM and LM components is ambiguous. In this work, we propose a novel ILME technique for CTC models. Our approach is to iteratively mask the input audio timesteps in order to estimate a pseudo log-likelihood of the internal LM (ILM) by accumulating the log-posteriors for only the masked timesteps across all iterations. This brings forth the ILM bias by eliminating the acoustic input in an iterative non-autoregressive fashion, thus bypassing the need to decouple the model into an AM and LM. Through this work, we make the following contributions:

- **Novel ILME technique for CTC-based ASR.** While ILME has mainly been proposed in the context of autoregressive models like AED and RNN-T, we present a novel masking-based ILME technique for non-autoregressive CTC decoding. We experiment with multiple masking methods including masking the audio at word boundaries. We find that simply dividing the audio into equal lengths and masking each partition yields best results.
- **Extensive evaluation across multiple out-of-domain datasets.** We evaluate the proposed ILME approach across 4 out-of-domain public datasets with varying target domains. Empirical evaluation shows that in comparison to shallow fusion, the proposed method can provide a relative word error rate reduction (WERR) by up to 9.8% on unseen target domains.
- **Zero-shot domain adaptation.** Departing from conventional wisdom that ILME requires a *target* domain LM trained on target domain data for inference, we show that the proposed ILME technique can provide up to 9.3% relative WERR over shallow fusion with only the *source* domain LM, thus not requiring any additional target domain data.
- **ILME improves OOV F1.** We show that the proposed approach not only improves word error rate (WER), but can also consistently improve the F1 score for detecting out-of-vocabulary terms from target domains by up to 24.6% relative to shallow fusion.

2. BACKGROUND

Given an input sequence of speech features $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_T\}$ and the corresponding output token sequence $\mathbf{Y} = \{y_1, \dots, y_L\}$, the objective of an ASR model, θ , is to estimate the posterior distribution $P(\mathbf{Y}|\mathbf{X};\theta)$. Typically, log-posteriors are computed for numeric stability. Finally, the optimal transcription $\hat{\mathbf{Y}}$ is determined from these log-posteriors using an autoregressive beam search algorithm with the criteria: $\hat{\mathbf{Y}} = \arg \max_{\mathbf{Y}} \left[\log P(\mathbf{Y}|\mathbf{X};\theta_{\text{ASR}}^{\text{S}}) \right]$.

LM Fusion for Domain Adaptation

Also referred to as *text-only* adaptation, in LM fusion, a separately trained target domain LM, $\theta_{\text{LM}}^{\text{T}}$, is combined with the source domain ASR model, $\theta_{\text{ASR}}^{\text{S}}$, to bias the system to output text with target domain semantics. Shallow Fusion [17] is one of the most popular methods of LM fusion [18]. In this approach, the log-posteriors of the source domain ASR model are interpolated with target domain LM log-probabilities during inference. The weighted log-scores are then used to select the output token at each step of beam search using the criteria: $\hat{\mathbf{Y}} = \arg \max_{\mathbf{Y}} \left[\log P(\mathbf{Y}|\mathbf{X};\theta_{\text{ASR}}^{\text{S}}) + \lambda_T \log P(\mathbf{Y};\theta_{\text{LM}}^{\text{T}}) \right]$, where $P(\mathbf{Y};\theta_{\text{LM}}^{\text{T}})$ is the probability assigned to the token sequence \mathbf{Y} by the target domain LM, and λ_T is the target LM weight. In this work, we compare the proposed ILME approach with this widely used shallow fusion technique for domain adaptation.

ILM Estimation for AED and RNN-T Models

Internal LM estimation has been proposed to reduce implicitly learned model bias from the source domain [10]. To estimate the internal LM, joint softmax approximation [19] is invoked to decompose the joint model parameters $\theta_{\text{ASR}}^{\text{S}}$ into individual AM and LM components. Assuming domain invariant acoustic conditions, a reformulation [10] of the Bayes' theorem for the posterior yields the inference criteria:

$$\hat{\mathbf{Y}} = \arg \max_{\mathbf{Y}} \left[\log P(\mathbf{Y}|\mathbf{X};\theta_{\text{ASR}}^{\text{S}}) + \lambda_T \log P(\mathbf{Y};\theta_{\text{LM}}^{\text{T}}) - \lambda_I \log P(\mathbf{Y};\theta_{\text{ASR}}^{\text{S}}) \right], \quad (1)$$

where λ_I is internal LM weight, and $\log P(\mathbf{Y};\theta_{\text{ASR}}^{\text{S}})$ is computed in an autoregressive manner by completely eliminating the acoustic input. This approach is limited to AED and RNN-T models as they inherently perform autoregressive decoding, and it is straightforward to decouple the model parameters into AM and LM components. In this work, we propose a novel technique for adapting the ILME algorithm to the non-autoregressive decoding of CTC models.

3. ESTIMATING ILM FOR CTC-BASED ASR

The core principle of the original ILME technique is to estimate the model's internal LM, $P(\mathbf{Y};\theta_{\text{ASR}}^{\text{S}})$, in an autoregressive manner: $P(\mathbf{Y};\theta_{\text{ASR}}^{\text{S}}) = \prod_{l=1}^L P(y_l|\mathbf{Y}_{1:l-1};\theta_{\text{ASR}}^{\text{S}})$, which lends itself naturally to AED and RNN-T models. As these models perform autoregressive decoding, they consist of a parametric component that can be identified as learning probabilities over token sequences — for AED models this would be the decoder network, and for RNN-T models it would be the prediction network. In contrast, CTC models compute log-posteriors in a non-autoregressive fashion, i.e., we have $\mathbf{Y} = \{y_1, \dots, y_T\}$ output tokens (including *blank* tokens) corresponding to each input timestep \mathbf{x}_t . Hence, there is no separate parametric component for tokens. The CTC loss also has an inherent conditional independence between the output tokens \mathbf{Y} , given the input \mathbf{X} . However, the self-attention and convolution mechanisms in transformer and conformer-based neural architectures implicitly

relaxes the conditional independence, due to the sequence-level interdependence of the intermediate features from attention and convolution. Therefore, CTC models still inadvertently learn an internal LM. An example of this is shown in Table 1. Additionally, the non-autoregressive architecture of CTC models makes it inconceivable to structurally decompose the model into corresponding acoustic and language components, which is a precursor to eliminating the acoustic input for conventionally estimating the internal LM.

Fundamentally, the ILM is the model's bias in the absence of any acoustic context. Consequently, we overcome the above challenges for ILME with CTC models by proposing an iterative input masking approach. At a high level, we mask the input audio multiple times and iteratively perform forward pass to determine the CTC log-posteriors for the corresponding masked timesteps. We then accumulate the log-posteriors for different groups of masked timesteps and normalize them in the posterior domain to compute the final internal LM distribution. We now explain our approach in further detail.

Iterative Input Masking

Since it is not straightforward to decouple the acoustic and language components of a CTC model, we cannot simply zero-out the entire acoustic input for estimating the ILM. Hence, instead of completely eliminating the acoustic input, we propose to iteratively mask the input audio for performing ILME. Given an input speech sequence $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_T\}$, we divide the input timesteps into K equal, non-overlapping partitions, and eliminate the input for each partition, yielding a set of K masked sequences $\{\tilde{\mathbf{X}}_1, \dots, \tilde{\mathbf{X}}_K\}$, such that

$$\tilde{\mathbf{x}}_t^{(k)} = \begin{cases} 0, & \text{if } t_{k-1} \leq t < t_k \\ \mathbf{x}_t & \text{otherwise} \end{cases} \quad (2)$$

where $\tilde{\mathbf{x}}_t^{(k)}$ is the input at t for $\tilde{\mathbf{X}}_k$, also $t_0 = 1$ and $t_k = kT/K$.

There are several masking strategies that can be followed here, such as masking at word boundaries, but experimentally we find that simply dividing the audio into equal partitions yields best results. Next, we discuss how we leverage these masked sequences for ILME.

Computing ILM for CTC

Given a vocabulary of N tokens $\{\mu_1, \dots, \mu_N\}$, we first denote the log-posteriors computed by the CTC model at timestep t as:

$$\Psi_t(\mathbf{X}) = \begin{bmatrix} \log P(y_t = \mu_1|\mathbf{X};\theta_{\text{CTC}}^{\text{S}}) \\ \vdots \\ \log P(y_t = \mu_N|\mathbf{X};\theta_{\text{CTC}}^{\text{S}}) \end{bmatrix} \quad (3)$$

Next, we compute the log-posterior distribution vector for a given masked sequence $\Psi_t(\tilde{\mathbf{X}}_k)$. Now, we want to determine whether these log-probabilities correspond to the original acoustic input or if they are affected by the acoustic masking. In the latter case, it would indicate the model's bias when no acoustic information was passed, and hence it would correspond to the model's ILM. For this, we determine the element-wise absolute difference between the log-posterior distributions as $\epsilon_t^k = |\Psi_t(\tilde{\mathbf{X}}_k) - \Psi_t(\mathbf{X})|$. Intuitively, if all elements of the vector ϵ_t^k have relatively low values, say below some threshold γ , it means timestep t is unaffected by the K^{th} masking, and we can ignore this in the ILM computation. Conversely, we want to keep $\Psi_t(\tilde{\mathbf{X}}_k)$ for the ILM computation if ϵ_t^k has any relatively high values, as it indicates the model bias that is affected by the masking.

Hence, we take the maximum of the N values in the vector ϵ_t^k , denote it as δ_t^k , and normalize it across all timesteps to determine a scalar $\hat{\delta}_t^k$ that we can compare against a threshold γ for determining whether $\Psi_t(\tilde{\mathbf{X}}_k)$ will contribute to the ILM. More concretely:

$$\delta_t^k = \max_n (\epsilon_t^k[n]); \quad \hat{\delta}_t^k = \frac{\delta_t^k}{\max_{t'} (\delta_{t'}^k)} \quad (4)$$

Finally, we estimate the ILM for timestep t by computing a pseudo log-likelihood using the log-posterior distributions across all K masks conditioned on whether the corresponding $\hat{\delta}_t^k$ is above a threshold γ . Therefore,

$$\log P_{\text{ILM}}(\mathbf{y}_t; \theta_{\text{CTC}}^S) = \text{LogSoftmax} \left(\sum_{k=1}^K \Psi_t(\tilde{\mathbf{X}}_k) \left[\hat{\delta}_t^k > \gamma \right] \right) \quad (5)$$

Although we show the ILM computation for a single timestep t , all the steps discussed are easily vectorizable in the sequence dimension, and can be computed for all timesteps in parallel using efficient sub-linear methods. The main overhead of our approach comes from the multiple forward passes required to compute the log-posteriors across all masks, which introduces a cost that scales linearly with K . This can be mitigated in part by doing one batched forward pass with the original and masked inputs in the same batch.

Inference with ILM Pseudo Log-Likelihood

For inference with ILME, we follow a similar approach as described in Eq. (1), subtracting the ILM pseudo log-likelihood computed using Eq. (5) from the original CTC log-posterior. However, CTC models have a special *blank* token that is purely functional and does not correlate with any language semantics. Hence, similar to [10], we skip subtracting the ILM for the timesteps where blank token is originally predicted. However, given the peaky behaviour of CTC models [20], we skip the timesteps for blank token only when the model assigns it a high likelihood in the posterior domain, above some threshold β :

$$\text{if } P_{\text{CTC}}(y_t = \langle \text{blank} \rangle | \mathbf{X}) < \beta,$$

$$\text{SCORE}(\mathbf{y}_t) = \log P_{\text{CTC}}(\mathbf{y}_t | \mathbf{X}) - \lambda_I \log P_{\text{ILM}}(\mathbf{y}_t) \quad (6)$$

$$\text{else, SCORE}(\mathbf{y}_t) = \log P_{\text{CTC}}(\mathbf{y}_t | \mathbf{X}) \quad (7)$$

Finally, we pass these modified log-posterior scores, $\text{SCORE}(\mathbf{y}_t)$, to beam search for performing inference combined with shallow fusion.

4. EXPERIMENTS

Model. We perform all our experiments on a conformer-based model [21], trained in a hybrid fashion [22] using joint CTC and decoder attention loss for more stable training. During inference, we only use the non-autoregressive CTC head and discard the shallow decoder. The conformer encoder consists of 20 layers of conformer blocks, where each layer has 8-headed attention and 2048 hidden units followed by a 512-dimensional output projection. The shallow decoder is a single transformer-based layer with a hidden dimension of 2048. The model has approximately 140M parameters, and is trained using the ADAM optimizer with an initial learning rate of 3×10^{-3} for 45 epochs. For tokenization, we train a sentencepiece tokenizer with a vocabulary size of 2048. Our model implementation and training is done by leveraging the widely used ESPnet framework [23]. In addition, we train 4-gram language models (LMs) for shallow fusion experiments and ILME using a modified Kneser-Ney smoothing algorithm implemented in the popular KenLM library [24].

Data. The model is trained on a large English corpus of 50k+ hours paired audio text data. We sample this data from public and in-house paired audio and text, ensuring a good mix of accents, speakers, sampling rates and background noise. The data regime is representative of a wide range of end-to-end ASR systems for various speech applications. To evaluate the proposed ILME approach, we consider four evaluation datasets from varying domains: LibriSpeech, VoxPopuli, Wikipedia and WSJ. We use the official test splits for each datasets in our inference experiments. **LibriSpeech** [25] is a read English speech corpus based on LibriVox audiobooks. We consider the two official

Table 2. Summary of out-of-domain evaluation datasets.

Dataset	Domain	# Hrs.	# OOVs	# Unique
LibriSpeech (test-clean)	Audiobooks	5.4	232	8138
LibriSpeech (test-other)	Audiobooks	5.3	265	7597
VoxPopuli	Parliamentary	4.9	52	5168
Wikipedia	Encyclopedia	32.7	9233	38706
WSJ	News	0.7	0	1834

evaluation sets: *test-clean* and *test-other*, each with 5 hours of test audio. **VoxPopuli** [26] consists of public political speech, sampled from 2009–2020 European Parliament event recordings. For our evaluation purpose, we utilize a 5-hour subset of VoxPopuli English data. The **Wikipedia** evaluation set [27] is a custom text-to-speech (TTS) dataset, which consists of a wide range of spoken named entities. More specifically, 14k sentences are sampled from English Wikipedia and passed through a TTS service to produce synthetic audios of 33 hours with different speakers. The **Wall Street Journal (WSJ)** [28] corpus contains conventional and spontaneous dictation by journalists. The *test_eval92* split of 0.7 hours is selected for our evaluation.

Our AM training data has no overlap whatsoever with data from the evaluation sets. Hence all 4 evaluation sets are regarded as *out-of-domain*. Table 2 shows a summary of these evaluation sets. Also, when we train a *target* LM, only the corresponding training splits of LibriSpeech, VoxPopuli, Wikipedia and WSJ are used respectively.

Experimental Setup. In order to demonstrate the generalizability of the proposed approach, we perform all experiments with a single set of hyperparameters across all datasets, exhaustively tuned on an in-house development set. For all experiments, we perform CTC inference using the beam search algorithm with a beam size of 50. For LM fusion, we use an LM weight of $\lambda_T = 1.0$. While performing ILME, we mask the audios into $K = 5$ partitions and perform inference with an ILM weight of $\lambda_I = 0.1$, a log-posterior delta threshold of $\gamma = 0.25$ and a blank-token filter threshold of $\beta = 0.9$.

5. RESULTS

We showcase the efficacy of the proposed ILME approach on multiple out-of-domain datasets with a baseline of CTC Beam Search (BS) as well as with stronger Shallow Fusion (SF) inference with an external LM. We perform our multifaceted evaluation in two modes:

1. **text-only domain adaptation**, in which we use a *target* LM trained on target domain text. This is typical in studying the efficacy of ILME domain adaptation in previous works [10].
2. **zero-shot domain adaptation**, in which we only use a *source* LM trained on source domain text, i.e., only the audio transcriptions already available for training the ASR model.

Note that the latter case is a much stronger test of generalization as the inference does not include any information about the target domain. To further calibrate generalization performance, we also mine out-of-vocabulary (OOV) terms from each out-of-domain evaluation dataset, and evaluate the F1 score for detecting these OOV terms which were never encountered by the model during training. Table 2 also shows the number of OOV references found in the evaluation datasets.

5.1. Text-only Domain Adaptation

Transcription WER. In Table 3, we show the WER and relative WERR for each inference technique. For text-only domain adaptation using the *target* LM, we clearly see that the proposed ILME approach outperforms the baseline in every case, showing a relative WERR by up to 28.5% (Wikipedia). ILME also outperforms the shallow fusion approach for the LibriSpeech test-{clean, other}, Wikipedia and WSJ datasets, improving the WER relative to SF by up to 9.4%

Table 3. WER (lower is better) and relative WERR (shown in parenthesis).

	Method	LM	LibriSpeech test-clean	LibriSpeech test-other	VoxPopuli	Wikipedia	WSJ
	BS	no LM	4.0 -	8.1 -	9.9 -	13.0 -	3.4 -
Zero-shot Domain Adaptation	SF	source LM	4.1 (-2.5%)	7.9 (+2.5%)	9.3 (+6.1%)	10.8 (+16.9%)	3.6 (-5.9%)
	ILME	source LM	3.7 (+7.5%)	7.6 (+6.2%)	9.3 (+6.1%)	10.7 (+17.7%)	3.0 (+11.8%)
Text-only Domain Adaptation	SF	target LM	3.2 (+20.0%)	6.6 (+18.5%)	7.9 (+20.2%)	9.5 (+26.9%)	3.2 (+5.9%)
	ILME	target LM	2.9 (+27.5%)	6.5 (+19.8%)	7.9 (+20.2%)	9.3 (+28.5%)	2.9 (+14.7%)

Table 4. F1 (higher is better) for detecting OOVs using *target* LM.

Dataset	BS	SF	ILME
LibriSpeech (test-clean)	30.7	32.5	43.1
LibriSpeech (test-other)	33.3	27.4	33.3
VoxPopuli	31.7	34.4	44.1
Wikipedia	39.4	42.4	47.6

Table 5. F1 (higher is better) for detecting OOVs using *source* LM.

Dataset	BS	SF	ILME
LibriSpeech (test-clean)	30.7	36.6	45.8
LibriSpeech (test-other)	33.3	32.8	39.4
VoxPopuli	31.7	26.2	34.4
Wikipedia	39.4	39.5	45.2

(LibriSpeech test-clean and WSJ). For VoxPopuli, even though WER improvement for ILME is very similar to SF, we can see a clear benefit of using ILME in terms of OOV F1 score as seen in Table 4.

F1 score for detecting OOVs. Table 4 shows the F1 score for detecting OOV terms using the *target* LM for the out-of-domain datasets. As there are no OOV terms found in WSJ, we exclude that dataset from this evaluation. We see that ILME outperforms both BS and SF for multiple datasets: LibriSpeech test-clean, VoxPopuli and Wikipedia. ILME improves over the baseline method’s F1 score relatively by up to 40.4% (LibriSpeech test-clean). For VoxPopuli, we see a 39.1% relative improvement in F1 from BS, whereas SF only improves F1 relatively by 8.5% from BS. This shows ILME is clearly a better choice than SF for VoxPopuli even though they have very similar WER. In the case of LibriSpeech test-other, we see that ILME performs the same as BS. However, looking at the shallow fusion F1 score for LibriSpeech test-other, it is evident that the target LM actually degrades the OOV detection performance. This can be attributed to the fact that LibriSpeech test-other is deliberately more challenging [25], and we note that the LibriSpeech training data has extremely sparse representation of the OOV terms. Hence, it is not guaranteed that using a target LM by itself can improve the detection of OOV terms, as these rare terms are not guaranteed to be present in the target domain text. However, ILME is still able to recover the degradation in shallow fusion F1 using the target LM, while also improving WER for LibriSpeech test-other. ILME is able to this by removing the model bias at each timestep, that would allow the OOV tokens to surface up compared to the previously biased tokens.

5.2. Zero-shot Domain Adaptation

In real world ASR applications, it is often hard to determine the specific domain for the speech input apriori. This prompted us to evaluate the generalization of ILME in situations where we are not aware of the target domain, or we have no data from the target domain to train an LM. For this, we use a *source* LM trained on the ASR training transcriptions. In using the source LM directly without ILME, we run the risk of doubly biasing the output due to the internal model bias combined with the external LM bias. Thus, ILME aids zero-

shot domain adaptation by mitigating the internal bias of the source domain, while the external source LM can ensure that higher level semantics pertaining to the language itself are preserved.

Transcription WER. Table 3 shows that in the case of zero-shot domain adaptation using a *source* LM, ILME outperforms BS in all cases with up to 17.7% (Wikipedia) relative WERR. As we are using a *source* LM in this case, simple shallow fusion may not yield better results than BS, as can be seen in the case of LibriSpeech test-clean (-2.5%) and WSJ (-5.9%). This is because the inference may get doubly biased from the source domain data by the internal as well as external LM. ILME can help in such cases by removing the internal bias of the source domain. This can be noticeably seen in Table 3, as ILME outperforms SF using only the *source* LM for multiple datasets: LibriSpeech test-{clean, other}, Wikipedia and WSJ. For VoxPopuli, again, ILME performs similar to SF, but ILME is once more clearly the optimal choice when the OOV F1 score is considered (Table 5).

F1 score for detecting OOVs. In Table 5, we show the F1 score for detecting OOV terms using the *source* LM. We see that ILME unambiguously outperforms the other techniques in all the cases of this zero-shot domain adaptation scenario, improving F1 relatively by up to 49.2% (LibriSpeech test-clean). For the LibriSpeech datasets, we see that ILME as well as SF with the *source* LM is actually better at detecting OOV terms than the *target* LM. This is because there is no guarantee that the OOV terms would be present in the training data for the target LM, as LibriSpeech samples the training data from a separate partition of speakers and audiobooks [25]. For VoxPopuli, again, while ILME performs similar to SF in terms of WER, ILME is clearly the better option as it improves F1 relatively by 8.5% whereas SF actually degrades F1 by 17.4%. Coincidentally for VoxPopuli, ILME with *source* LM performs as well as SF with *target* LM.

5.3. Future Work

We observe that previous ILME methods for AED and RNN-T models only perform decoder-side de-biasing by eliminating the encoder input. In contrast, our CTC-based ILME method works for non-autoregressive decoding using the encoder itself. Hence, we hypothesize that our ILME approach can be extended to AED and RNN-T models as well for complementary encoder-side de-biasing. Furthermore, post-training techniques can be formulated for training student models to estimate the ILM of a CTC model, thus accelerating ILME inference. We aim to pursue these directions in future work.

6. CONCLUSION

In this work, we propose a novel ILME approach for CTC-based ASR models. We extensively evaluate the efficacy of the approach in adapting to multiple unseen target domains. We show that the proposed ILME technique improves WER by up to 9.8% and OOV F1-score by up to 24.6% relative to shallow fusion, when only target domain text data is available. We also demonstrate that applying ILME is beneficial in application to any new target domain even when there is no access to target domain data at all, as it de-biases the encoder and provides out of the box improvement.

7. REFERENCES

- [1] S. Toshiwal, A. Kannan, C.-C. Chiu, Y. Wu, T. N. Sainath, and K. Livescu, "A comparison of techniques for language model integration in encoder-decoder speech recognition," in *IEEE Spoken Language Technology Workshop (SLT)*, 2018.
- [2] C. Shan, C. Weng, G. Wang, D. Su, M. Luo, D. Yu, and L. Xie, "Component fusion: Learning replaceable language model component for end-to-end speech recognition system," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.
- [3] A. Shenoy, S. Bodapati, M. Sunkara, S. Ronanki, and K. Kirchhoff, "Adapting Long Context NLM for ASR Rescoring in Conversational Agents," in *Proc. Interspeech 2021*, 2021.
- [4] A. Shenoy, S. Bodapati, and K. Kirchhoff, "ASR adaptation for E-commerce chatbots using cross-utterance context and multi-task language modeling," in *Proceedings of the 4th Workshop on e-Commerce and NLP (ACL)*, Aug. 2021.
- [5] L. Liu, Y. Gu, A. Gourav, A. Gandhe, S. Kalmene, D. Filimonov, A. Rastrow, and I. Bulyko, "Domain-aware neural language models for speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.
- [6] S. Dingliwal, A. Shenoy, S. Bodapati, A. Gandhe, R. T. Gadge, and K. Kirchhoff, "Domain Prompts: Towards memory and compute efficient domain adaptation of ASR systems," in *Proc. Interspeech 2022*, 2022.
- [7] E. McDermott, H. Sak, and E. Varianni, "A density ratio approach to language model fusion in end-to-end automatic speech recognition," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019.
- [8] C. Choudhury, A. Gandhe, X. Ding, and I. Bulyko, "A likelihood ratio based domain adaptation method for e2e models," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.
- [9] W. Zhou, Z. Zheng, R. Schlüter, and H. Ney, "On language model integration for RNN transducer based speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.
- [10] Z. Meng, S. Parthasarathy, E. Sun, Y. Gaur, N. Kanda, L. Lu, X. Chen, R. Zhao, J. Li, and Y. Gong, "Internal language model estimation for domain-adaptive end-to-end speech recognition," in *IEEE Spoken Language Technology Workshop (SLT)*, 2021.
- [11] M. Zeineldeen, A. Glushko, W. Michel, A. Zeyer, R. Schlüter, and H. Ney, "Investigating methods to improve language model integration for attention-based encoder-decoder ASR models," *arXiv preprint arXiv:2104.05544*, 2021.
- [12] Y. Liu, R. Ma, H. Xu, Y. He, Z. Ma, and W. Zhang, "Internal language model estimation through explicit context vector learning for attention-based encoder-decoder ASR," *arXiv preprint arXiv:2201.11627*, 2022.
- [13] J. Lee and S. Watanabe, "Intermediate loss regularization for CTC-based speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.
- [14] R. Fan, W. Chu, P. Chang, and J. Xiao, "CASS-NAT: CTC alignment-based single step non-autoregressive transformer for speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.
- [15] X. Zhang, F. Zhang, C. Liu, K. Schubert, J. Chan, P. Prakash, J. Liu, C.-F. Yeh, F. Peng, Y. Saraf *et al.*, "Benchmarking LF-MMI, CTC and RNN-T criteria for streaming ASR," in *IEEE Spoken Language Technology Workshop (SLT)*, 2021.
- [16] S. Dingliwal, M. Sunkara, S. Ronanki, J. Farris, K. Kirchhoff, and S. Bodapati, "Personalization of CTC speech recognition models," in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023.
- [17] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.
- [18] D. Le, G. Keren, J. Chan, J. Mahadeokar, C. Fuegen, and M. L. Seltzer, "Deep shallow fusion for RNN-T personalization," in *IEEE Spoken Language Technology Workshop (SLT)*, 2021.
- [19] E. Varianni, D. Rybach, C. Allauzen, and M. Riley, "Hybrid Autoregressive Transducer (HAT)," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.
- [20] A. Zeyer, R. Schlüter, and H. Ney, "Why does CTC result in peaky behavior?" *arXiv preprint arXiv:2105.14849*, 2021.
- [21] A. Gulati, J. Qin, C. C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. Interspeech 2020*, 2020.
- [22] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, 2017.
- [23] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-end speech processing toolkit," in *Proc. Interspeech 2018*, 2018.
- [24] K. Heafield, "Kenlm: Faster and smaller language model queries," in *Proceedings of the Sixth Workshop on Statistical Machine Translation*, 2011.
- [25] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.
- [26] C. Wang, M. Rivière, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, "VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021.
- [27] N. Das, M. Sunkara, D. Bekal, D. H. Chau, S. Bodapati, and K. Kirchhoff, "Listen, know and spell: Knowledge-infused subword modeling for improving ASR performance of OOV named entities," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.
- [28] J. S. Garofolo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) Complete LDC93S6A," *Linguistic Data Consortium*, 1993. [Online]. Available: <https://catalog.ldc.upenn.edu/LDC93S6A>