Interspeech 2022
18-22 September 2022, Incheon, Korea

An Empirical Study of Language Model Integration for Transducer based Speech Recognition

Huahuan Zheng[1], *Keyu An*[1], *Zhijian Ou*[1†], *Chen Huang*[2], *Ke Ding*[2], *Guanglu Wan*[2]

[1]Speech Processing and Machine Intelligence (SPMI) Lab, Tsinghua University, China
[2]Meituan, China

{zhh20,aky19}@mails.tsinghua.edu.cn, ozj@tsinghua.edu.cn,
{huangchen09,dingke02,wanguanglu}@meituan.com

Abstract

Utilizing text-only data with an external language model (ELM) in end-to-end RNN-Transducer (RNN-T) for speech recognition is challenging. Recently, a class of methods such as density ratio (DR) and internal language model estimation (ILME) have been developed, outperforming the classic shallow fusion (SF) method. The basic idea behind these methods is that RNN-T posterior should first subtract the implicitly learned internal language model (ILM) prior, in order to integrate the ELM. While recent studies suggest that RNN-T only learns some low-order language model information, the DR method uses a well-trained neural language model with full context, which may be inappropriate for the estimation of ILM and deteriorate the integration performance. Based on the DR method, we propose a low-order density ratio method (LODR) by replacing the estimation with a low-order weak language model. Extensive empirical experiments are conducted on both in-domain and cross-domain scenarios on English LibriSpeech & Tedlium-2 and Chinese WenetSpeech & AISHELL-1 datasets. It is shown that LODR consistently outperforms SF in all tasks, while performing generally close to ILME and better than DR in most tests.

Index Terms: ASR, language model, transducer

1. Introduction

A trend of speech recognition research is moving from hybrid models to end-to-end (E2E) models such as connectionist temporal classification (CTC) [1], recurrent neural network transducer (RNN-T) [2] and attention-based encoder-decoder (AED) [3]. E2E models have been shown to outperform hybrid ones in in-domain tasks [4, 5, 6, 7], when trained on large amounts of paired speech-text data.

However, in practice, text-only data is easily available, allowing improving the performance of ASR models in lower cost, compared to increasing the paired labeled speech-text data. Language model (LM) integration is a widely-used approach to boost ASR performance with text-only data for both in-domain test and cross-domain adaptation. ASR decoding amounts to maximizing the posterior probability $P(Y|X)$, where Y is token sequence represented by words, sub-words or characters, and X is the observed speech data. Since a hybrid system explicitly learns the acoustic model (AM), integrating the external language model (ELM) is straightforward, by following the Bayes rule:

$$\hat{Y} = \arg\max_{Y} \left[P_{\text{AM}}(X|Y) P_{\text{ELM}}(Y) \right] \tag{1}$$

† Corresponding author, also affiliated with Beijing National Research Center for Information Science and Technology, China. This work is supported by NSFC 61976122 and Tsinghua University - Meituan Joint Institute for Digital Life.

However, E2E systems like RNN-T jointly learn the posterior probability $P(Y|X)$ in a unified model, with no separation of AM and LM, making the LM integration for E2E models not as easy as that of hybrid models.

A number of "fusion" approaches have been proposed to address the problem of LM integration for E2E models, such as shallow fusion [8, 9, 10], deep fusion [11] and cold fusion [12], among which the shallow fusion is the most popular one. SF just conducts log-linear interpolation between the scores of the E2E model and the ELM. Recently, a class of internal language model (ILM) estimation methods have been developed, outperforming SF, which were initially studied for RNN-T [13] and later extended to AED [14, 15]. In this paper, we constrain our study of ILM to RNN-T.

The basic idea of ILM is that an RNN-T model $P_{\text{RNN-T}}(Y|X)$, through training over labeled data, has captured LM information from the transcripts, and such information is thought to be encapsulated in the so-called ILM. Thus, if the ILM from the RNN-T can be estimated in some way, the LM integration for the RNN-T model can be achieved by subtracting the ILM and integrating the ELM in the manner of Eq. (1) for hybrid models:

$$\hat{Y} = \arg\max_{Y} \left[\frac{P_{\text{RNN-T}}(Y|X)}{P_{\text{ILM}}(Y)} P_{\text{ELM}}(Y) \right] \tag{2}$$

Based on the above idea, many ILM estimation methods are proposed, differing mainly in the manner how the ILM is estimated. In hybrid autoregressive transducer (HAT) [13], the distributions for the blank label `<blk>` and normal labels are re-defined via a Sigmoid function and a Softmax function separately. [14] proposes the "internal lanugage model estimation" (ILME) method, which zeros out the acoustic hidden state and normalizes the logits from the joint network just over non-blank labels. In a more complex method, called mini-LSTM [16], the acoustic hidden state is calculated via an additional LSTM. In these methods, the ILM probability is the product of the resulting label probabilities, which basically still come from the RNN-T model. Instead, the density ratio (DR) method [17] uses a neural LM separately trained over the E2E training transcripts as an estimate of ILM, whose structure and size are usually set to match the prediction network in RNN-T.

As reported in [13, 14], the perplexities (PPLs) of the estimated ILMs during RNN-T training first decrease and then increasingly converge to rather larger PPLs than those of the well-learned neural LMs used in the DR method (e.g., 99.4 and 30.1 in [14]). Moreover, it is found in [13, 18] that RNN-T using only one history label for prediction network can achieve competitive performance to the one with full context, which also suggests RNN-T only learns some low-order language information, but not a strong LM. Inspired by these previous findings,

Copyright © 2022 ISCA

10.21437/Interspeech.2022-10576

we propose to train a low-order weak LM (such as a bi-gram LM) over the transcripts[1], use it as the estimate of ILM, and substitute to Eq. (2) to perform LM integration. The proposed method, which we refer to as low-order density ratio (LODR), can be viewed as an extension of the DR method, estimating the ILM with low-order information, instead of training well-learned neural LMs as in standard DR method.

Extensive experiments are conducted in this paper to compare the performances of SF, DR, ILME, and LODR on RNN-T, evaluated on both in-domain and cross-domain tasks with text-only data. It is shown in this study that ILME and LODR performs close to each other, both better than SF (in all tests) and DR (in most tests). This further consolidates and deepens our understanding of the ILM in RNN-T, and would be helpful for future work to further advance LM integration for RNN-T. The code will be released upon the acceptance of the paper.

All comparisons are made in decode-time manner, thus the works like [13, 16, 19, 20] that modify the architecture or training objective of RNN-T are not included in this work. It should also be clarified that, when discussing "cross-domain", we assume the source domain and target domain are matched in acoustics, otherwise Eq. (2) cannot be adopted.

2. LM Integration Methods for RNN-T

2.1. RNN Transducer

The RNN-T model [2] consists of an acoustic encoder (a.k.a. the transcription network), a prediction network (PN) and a joint network. With given acoustic features $X = (\mathbf{x}_1, ..., \mathbf{x}_T)$ and the token sequence $Y = (y_0, ..., y_U)$, the posterior probability is defined as:

$$\begin{aligned} P_{\text{RNN-T}}(Y|X) &= \sum_{\tilde{Y} \in \mathcal{B}^{-1}(Y)} P(\tilde{Y}|X) \\ &= \sum_{\tilde{Y} \in \mathcal{B}^{-1}(Y)} \prod_{i=1}^{T+U} P(\tilde{y}_i|X, y_{0:u}) \end{aligned} \tag{3}$$

$$P(\tilde{y}_i|X, y_{0:u}) = \text{Softmax}\left[J\left(\mathbf{g}_u, \mathbf{f}_t\right)\right] \tag{4}$$

Here $\tilde{Y} = (\tilde{y}_1, ..., \tilde{y}_{T+U})$ represent the alignment sequence including the blank label <blk>. $t \in [1, T]$ and $u \in [0, U]$ denote the number of speech frames and that of non-blank labels up to emitting $\tilde{y}_i$. $\mathcal{B}(\cdot)$ represents the alignment mapping of $\tilde{Y}$ to Y by removing <blk>. The token-level probabilities are computed by the joint network as in Eq. (4), where $\mathbf{g}_u$ and $\mathbf{f}_t$ denotes the output hidden features of the PN and encoder. $J(\cdot)$ denotes the joint network, which consists of linear layers and non-linear activations in the common RNN-T architecture.

2.2. Shallow Fusion

The shallow fusion (SF) method takes the linear combination of log scores of E2E models and the ELMs, with one extra parameter β as reward for the sequence length $|Y|$. The hypothesis $\hat{Y}$ is selected as follows:

$$\hat{Y} = \arg\max_{Y} \left[\log P_{\text{E2E}}(Y|X) + \lambda \log P_{\text{ELM}}(Y) + \beta|Y|\right] \tag{5}$$

2.3. Density Ratio

The density ratio (DR) method [17] is proposed as an extension of SF, and makes the assumption that the model distribution $P(Y|X)$ captured by RNN-T can be factorized into acoustic and linguistic parts as in the hybrid system:

$$P_{\text{RNN-T}}(X|Y) = P_{\text{RNN-T}}(X) \frac{P_{\text{RNN-T}}(Y|X)}{P_{\text{RNN-T}}(Y)} \tag{6}$$

where $P_{\text{RNN-T}}(X)$ models the marginal distribution of data. The DR method further assumes: (1) the acoustic models of the source domain (where the RNN-T is trained) and the target domain are consistent, both modeled by $P_{\text{RNN-T}}(X|Y)$; (2) the linguistic distribution for the two domains can be separately estimated via LMs $P_{ILM}(Y) \approx P_{\text{RNN-T}}(Y)$ and $P_{ELM}(Y)$, respectively. According to [17], $P_{ILM}(Y)$ is estimated by a separately trained neural LM over source domain transcripts. $P_{ELM}(Y)$ can be trained over the target domain corpus. With these assumptions, decoding for recognizing utterances from the target domain can be derived the same as shown in Eq. (2). Introducing LM weights (λ_0, λ_1) and length reward β, we have

$$\begin{aligned} \hat{Y} = \arg\max_{Y} \big[&\log P_{\text{RNN-T}}(Y|X) + \lambda_0 \log P_{\text{ILM}}(Y) \\ &+ \lambda_1 \log P_{\text{ELM}}(Y) + \beta|Y|\big] \end{aligned} \tag{7}$$

When $\lambda_0 = -1, \lambda_1 = 1$ and $\beta = 0$, Eq. (7) is equivalent to Eq. (2), in a strict manner of Bayes rule.

2.4. ILME

Inspired by [13], ILME [14] applies Proposition 1 in [13, Appendix. A] to estimate the ILM via zeroing out the acoustic part in Eq. (4). The proposition claims that, if $J\left(\mathbf{g}_u, \mathbf{f}_t\right) \approx J\left(\mathbf{g}_u, \mathbf{0}\right) + J\left(\mathbf{0}, \mathbf{f}_t\right)$ is satisfied, we have

$$P_{\text{RNN-T}}(y_{u+1}|y_{0:u}) \propto \exp\left(J(\mathbf{g}_u)\right) \tag{8}$$

where we omit the $\mathbf{0}$ to simplify the notations. So the ILM of RNN-T can be computed by applying Softmax to the token logits excluding <blk>, i.e.

$$P_{\text{ILM}}(Y) = \prod_{u=0}^{U} \text{Softmax}\left[J_{\backslash\texttt{<blk>}}(\mathbf{g}_u)\right] \tag{9}$$

The ILME decoding basically follows Eq. (7), except that the ILM is estimated from RNN-T itself. It is reported that the simple joint network described in Sec. 2.1 makes RNN-T possible to roughly satisfy the condition of the proposition [13].

3. Low-order Density Ratio Estimate

3.1. Design of LODR

As discussed in Sec. 1, previous works have shown that the ILM of RNN-T actually captures small amount of LM information from the transcripts [13, 14], and RNN-T only makes use of limited context and low-order information in the prediction network [13, 18]. In contrast, the DR method uses a well-learned LM with full context as the estimation of ILM. Inspired by the findings, we hypothesize that the original DR setting is inappropriate and may deteriorate the performance of integration.

To investigate the performance of DR using a low-order weak ILM, we separately train a low-order LM on the transcripts and follow the Eq. (7) to do the integration. In implementation, we train a bi-gram LM on the training transcripts, and prune the bi-grams except for the most frequent 20k ones. This would produce a very small model, typically around 250 kB on disk. Note that if the modeling units are in small granularity like alphabets, the number of bi-grams is probably not enough up to 20k; in that case, one may need to use relatively higher order of n-gram. In our experiments, we use 1024 word-pieces for English and around 5k characters for Chinese as the modeling units, where the numbers of bi-grams are both more than 100k. We leave further discussion to the Sec. 4.3.

[1]When the transcripts are not available, training a low-order LM over some other text corpus also produces competitive performance, as shown later in our experiments.

Table 1: *Performance of LM integration methods, represented as word error rate (WER %) on LibriSpeech and character error rate (CER %) on WenetSpeech. The perplexity (PPL) of the ILM is computed on the transcript of each dataset, which could roughly measure the similarity of the ILM and the transcript corpus distributions. λ_0 is the weight of the ILM in each method. For all methods, λ_1 denotes the weight of the shared ELM and β denotes the length reward. "avg." is the average WER/CER on all dev and test datasets. "Rel %" measures the relative reduction of WER (CER) compared to "No LM" setup.*

Method	ILM PPL	λ_0	λ_1	β	LibriSpeech					
					dev		test		avg.	Rel %
					clean	other	clean	other		
No LM	-	-	-	-	2.18	5.33	2.40	5.42	3.81	-
SF	-	-	0.625	1.0	1.82	4.06	1.96	4.42	3.04	20.2
DR	24.72	-0.125	0.75	0.5	1.79	4.00	1.97	4.31	3.00	21.3
ILME	50.21	-0.125	0.75	1.0	1.78	3.99	1.92	4.35	**2.99**	**21.5**
LODR	100.94	-0.125	0.75	0.75	1.83	4.00	1.94	4.34	3.01	21.0

Method	ILM PPL	λ_0	λ_1	β	WenetSpeech				
					dev	test		avg.	Rel %
						net	meeting		
No LM	-	-	-	-	11.14	12.75	20.88	14.05	-
SF	-	-	0.25	3.125	9.19	11.73	18.36	12.37	12.0
DR	37.89	0.0	0.25	3.125	9.19	11.73	18.36	12.37	12.0
ILME	94.32	-0.125	0.375	3.0	9.10	11.56	18.26	12.25	12.8
LODR	79.33	-0.125	0.375	3.125	9.07	11.54	18.23	**12.22**	**13.0**

3.2. Implementation of Score Interpolation

As is shown in Eq. (5) and Eq. (7), SF, DR, ILME and LODR are all essentially conducting linear interpolation between scores of the RNN-T and the LMs, where the hyperparameters, i.e., the λ_0, λ_1 and β, are shown to be important to the integration performance [14, 16, 17]. Since lacking of common hyperparameter tuning setup in the literature and grid-search becomes too expensive as the number of hyperparameters exceeds two, in this work, we detail our hyperparameter searching method, which is coordinate descent [21] combined with binary search:

1. Initialize the searching ranges and minimum interval sizes for each hyperparameter.
2. Tune one hyperparameter per iteration. In an iteration, fix other hyperparameters and only search for one. Binary search is used for tuning one hyperparameter, and is done when the searching range is smaller than the minimum interval size.
3. Continue the loop until there is not any better combination. Once the tuned hyperparameters are at the boundary of the initial range, we extend the range and continue the loop.

The hyperparameters are first tuned on a held-out validation set, then evaluated on the test sets. In the experiments, all LM integration methods follow the same steps to tune their hyperparameters, except that there are two hyperparameters for SF as Eq. (5), while three for DR, ILME and LODR as Eq. (7).

4. Experiment

Experiments are taken on 1000-hour WenetSpeech [22] Chinese dataset[2], 960-hour LibriSpeech [23] English dataset and an in-house Chinese dataset *Tasi* of around 4.5k hours. The 200M-char WenetSpeech corpus and 800M-word LibriSpeech corpus are taken for in-domain test. For cross-domain scenarios, we use Tedlium-2 [24], AISHELL-1 [25] and an in-house TV-news dataset to evaluate on the three training sets respectively.

For Chinese datasets, we use character-based modeling units, where WenetSpeech dataset has around 5k chars and our in-house one has around 6k chars. The English LibriSpeech dataset is trained with 1024 word-piece units, obtained with SentencePiece toolkit [26]. Speech data in our experiments are transformed into 80-dimensional raw FBank features, CMVN [27] is applied after that. In the experiments, the encoder of RNN-T is Conformer [5] with $1/4$ subsampling inserted before. The prediction network is 1-layer unidirectional LSTM. And the joint network is standard fully-connected layers with $\tanh(\cdot)$ activation. The design of RNN-T components follows [5], but some of the hyperparameters (e.g., hidden dimensions and number of layers of encoders) may differ in experiments. The RNN-Ts are trained with Adam optimizer and the Transformer learning rate scheduler [28]. At the convergence, we take best 10 of the checkpoint models and do model averaging to get the one for further evaluation.

For leveraging the performance and decoding speed, we run the decoding in monotonic topology, basically following [29], with beam size limited to 128. The LM integration methods are evaluated in the manner of rescoring. Hyperparameters of all LM integration methods are tuned as described in Sec. 3.2. Following [17], LMs serving as the ILM in the DR method are 6-layer LSTM with 512-dimensional hidden size, trained on the transcripts. We train N-gram models with modified Kneser-Ney smoothing using the toolkit KenLM [30].

4.1. In-domain evaluation

The hyperparameters of LM integration are tuned on: *dev* for WenetSpeech; *dev-clean+dev-other* for LibriSpeech. We set the initial range [0, 1] and minimum interval size 0.1 for parameter tuning. Though the scales of ILMs λ_0 are suggested to take negative in [17, 14], we do not add such restriction. With the range extending mechanism descried in Sec. 3.2, λ_0 is still possible to be negative.

The RNN-T model in WenetSpeech experiment is of around 90M parameters, while the one in LibriSpeech is Conformer-L with 120M parameters, following [5]. The ELM used in LibriSpeech experiment is a Transformer LM with 87M parameters, trained on the 800M-word LibriSpeech corpus. No extra data is used, so our LibriSpeech results in Table. 1 are comparable to those in literature [5, 16]. WenetSpeech experiments take the 200M-char WenetSpeech corpus to train the ELM. The

[2]The full dataset includes 10k hours of labeled data. We just take the 1000-hour train-M subset.

Table 2: *Performance of LM integration methods evaluated on cross-domain scenarios.*

Method	λ_0	λ_1	β	LibriSpeech → Tedlium-2			
				dev	test	avg.	Rel %
No LM	-	-	-	11.67	11.41	11.51	-
SF	-	0.625	1.5	10.26	10.05	10.13	12.0
DR	-0.125	0.625	1.5	10.21	9.85	**9.99**	**13.2**
ILME	-0.125	0.5	1.0	10.23	9.87	10.01	13.0
LODR	-0.125	0.625	1.5	10.25	9.97	10.08	12.4

Method	λ_0	λ_1	β	WenetSpeech → AISHELL-1			
				dev	test	avg.	Rel %
No LM	-	-	-	6.32	7.22	6.63	-
SF	-	0.5	1.375	5.11	5.56	5.26	20.7
DR	-0.125	0.5	1.375	5.10	5.65	5.28	20.4
ILME	-0.125	0.5	1.125	4.99	5.55	5.18	21.9
LODR	-0.375	0.625	0.375	4.76	5.33	**4.95**	**25.3**

Method	λ_0	λ_1	β	Tasi → TV-news	
				tv-news test	Rel %
No LM	-	-	-	13.41	-
SF	-	0.25	2.125	11.67	13.0
DR	-0.125	0.25	1.625	11.61	13.4
ILME	-0.125	0.25	1.375	11.46	14.5
LODR	-0.125	0.25	1.25	**11.44**	**14.7**

ELMs are shared for all methods on each dataset.

As Table. 1 shows, all methods bring significant improvement over the baseline (standalone RNN-T without LM). The three methods with ILM estimated (DR, ILME and LODR) are of close performance, and all outperform the SF method.

Note that all λ_0 in the Table. 1 are 0 or -0.125. Considering that the minimum interval size in hyperparameter searching is 0.1, this could reveal that for in-domain evaluation, the subtracted part might be unimportant. This somehow shows the SF method is sufficiently good for in-domain test.

Among the methods estimating ILMs, in LibriSpeech experiment, DR, ILME and LODR perform considerably close on average; in WenetSpeech experiment, our LODR method gains 1.2% relative CER reduction over DR and slightly outperforms ILME.

It is also shown in the Table. 1 that the estimated ILM of DR simulate the transcript best (with the smallest perplexities on transcripts, in both datasets), but the evaluation performance does not consistently surpass the ILME and LODR, which further verifies our analysis in Sec. 3.1.

4.2. Cross-domain evaluation

In cross-domain evaluation, we follow most of the settings in in-domain test, except that the evaluation is on a new domain and the ELM is trained on the target domain corpus.

Table. 2 shows the performance of the integration methods in domain adaptation. When adapting the RNN-T trained on LibriSpeech to Tedlium-2, DR gets the best WER on both dev and test sets. DR, ILME and LODR obtain 1.4%, 1.2% and 0.5% relative WER reduction on the Tedlium-2 *test* compared to SF. This relative reduction is much smaller than the literature results [16], which reports the DR gains 8.5% (16.4 → 15.0) relative WER reduction and ILME gains 12.2% (16.4 → 14.4) compared to SF. We argue that, this is probably due to our significantly lower baseline (No LM: 11.41 vs. 20.3 [16]).

In the adaptation from WenetSpeech to AISHELL-1, our proposed LODR method obtains 6.2% and 4.4% relative CER reduction over DR and ILME methods, respectively. Overfitting of weights tuning is observed in the adaptation from WenetSpeech to AISHELL-1: compared to SF, the DR method obtains lower CER on dev set, but performs worse on test set with 1.6% relative CER increase; ILME gains 2.2% relative CER reduction on dev set, while only 0.2% on test. In contrast, our LODR method obtains 4.6% and 4.1% relative CER reduction on the two sets over SF, showing that LODR is a very promising method for cross-domain LM adaptation. When adapted to TV-news from our in-house Tasi dataset, LODR also outperforms all SF, DR and ILME methods.

Table 3: *Comparisons of the low-order LM trained on transcripts and external corpus.*

Data	λ_0	λ_1	β	dev	test
LibriSpeech → Tedlium-2					
transcripts	-0.125	0.625	1.5	10.25	9.97
external corpus	-0.125	0.625	1.5	10.22	9.95
WenetSpeech → AISHELL-1					
transcripts	-0.375	0.625	0.375	4.76	5.33
external corpus	-0.375	0.625	0.375	4.75	5.33

4.3. Discussion

As we explained in Sec. 3.1, LODR is driven by estimating the ILM for RNN-T with the cognition that the ILM is indeed a low-order one.

To further investigate whether LODR really benefit from the transcript information, we do an ablation study that train the low-order LM from an external corpus. Here the external corpus is randomly taken from the ELMs training corpus described in Sec. 4.1, keeping the size the same as the transcripts. It is interesting that the low-order LMs trained on transcript and external corpus have consistent performance in cross-domain adaptation on both Chinese and English tasks. An intuitive interpretation is that, under the low-order bi-gram modeling and with most of the bi-grams pruned, the LM only learns very basic statistics about the language itself, which are, to some extent, consistent across corpus.

Reviewing the results in Table. 1 and Table. 2, it seems LODR performs consistently better in Chinese tasks, but not as in English. This is possibly due to the smaller number of modeling units in English tasks, which may require more than 20k bi-grams in the pruned low-order LM. Note that the number "20k" is an empirically set value, may not be optimal.

5. Conclusion

In this work, we first review existing LM integration methods, including SF, DR and ILME, in the common RNN-T framework. As recent studies suggest that RNN-T only learns some low-order LM information, we hypothesize that the ILM used in the original DR method is unduly strong and may deteriorate the performance. A low-order density ratio method (LODR) is proposed by training a low-order weak ILM for DR. Extensive empirical experiments are conducted on both in-domain and cross-domain scenarios on English LibriSpeech & Tedlium-2 and Chinese WenetSpeech & AISHELL-1 datasets. It is shown in this study that our proposed LODR consistently outperforms the SF, and performs better than the original DR in most tests with less extra parameters introduced. As compared to ILME, our LODR method has close performance and avoids feeding the labels to the text encoder twice. This verifies our hypothesis and deepens our understanding of the ILM in RNN-T, and would be helpful for future work to further advance LM integration for RNN-T.

6. References

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

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

[3] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *Advances in neural information processing systems*, vol. 28, 2015.

[4] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6381–6385.

[5] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[6] J. Li, R. Zhao, Z. Meng, Y. Liu, W. Wei, S. Parthasarathy, V. Mazalov, Z. Wang, L. He, S. Zhao *et al.*, "Developing rnn-t models surpassing high-performance hybrid models with customization capability," *arXiv preprint arXiv:2007.15188*, 2020.

[7] J. Li, Y. Wu, Y. Gaur, C. Wang, R. Zhao, and S. Liu, "On the comparison of popular end-to-end models for large scale speech recognition," *arXiv preprint arXiv:2005.14327*, 2020.

[8] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model." in *Interspeech*, vol. 2, no. 3. Makuhari, 2010, pp. 1045–1048.

[9] 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.

[10] A. Kannan, Y. Wu, P. Nguyen, T. N. Sainath, Z. Chen, and R. Prabhavalkar, "An analysis of incorporating an external language model into a sequence-to-sequence model," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1–5828.

[11] C. Gulcehre, O. Firat, K. Xu, K. Cho, L. Barrault, H.-C. Lin, F. Bougares, H. Schwenk, and Y. Bengio, "On using monolingual corpora in neural machine translation," *arXiv preprint arXiv:1503.03535*, 2015.

[12] A. Sriram, H. Jun, S. Satheesh, and A. Coates, "Cold fusion: Training seq2seq models together with language models," *arXiv preprint arXiv:1708.06426*, 2017.

[13] E. Variani, D. Rybach, C. Allauzen, and M. Riley, "Hybrid autoregressive transducer (hat)," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6139–6143.

[14] 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 *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 243–250.

[15] 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.

[16] W. Zhou, Z. Zheng, R. Schlüter, and H. Ney, "On language model integration for rnn transducer based speech recognition," *arXiv preprint arXiv:2110.06841*, 2021.

[17] E. McDermott, H. Sak, and E. Variani, "A density ratio approach to language model fusion in end-to-end automatic speech recognition," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 434–441.

[18] M. Ghodsi, X. Liu, J. Apfel, R. Cabrera, and E. Weinstein, "Rnn-transducer with stateless prediction network," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7049–7053.

[19] Z. Meng, N. Kanda, Y. Gaur, S. Parthasarathy, E. Sun, L. Lu, X. Chen, J. Li, and Y. Gong, "Internal language model training for domain-adaptive end-to-end speech recognition," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 7338–7342.

[20] Z. Meng, Y. Wu, N. Kanda, L. Lu, X. Chen, G. Ye, E. Sun, J. Li, and Y. Gong, "Minimum word error rate training with language model fusion for end-to-end speech recognition," *arXiv preprint arXiv:2106.02302*, 2021.

[21] S. J. Wright, "Coordinate descent algorithms," *Mathematical Programming*, vol. 151, no. 1, pp. 3–34, 2015.

[22] B. Zhang, H. Lv, P. Guo, Q. Shao, C. Yang, L. Xie, X. Xu, H. Bu, X. Chen, C. Zeng *et al.*, "Wenetspeech: A 10000+ hours multi-domain mandarin corpus for speech recognition," *arXiv preprint arXiv:2110.03370*, 2021.

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

[24] A. Rousseau, P. Deléglise, Y. Esteve *et al.*, "Enhancing the ted-lium corpus with selected data for language modeling and more ted talks." in *LREC*, 2014, pp. 3935–3939.

[25] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, "Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline," in *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*. IEEE, 2017, pp. 1–5.

[26] T. Kudo and J. Richardson, "Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," *arXiv preprint arXiv:1808.06226*, 2018.

[27] O. Viikki and K. Laurila, "Cepstral domain segmental feature vector normalization for noise robust speech recognition," *Speech Communication*, vol. 25, no. 1-3, pp. 133–147, 1998.

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

[29] A. Tripathi, H. Lu, H. Sak, and H. Soltau, "Monotonic recurrent neural network transducer and decoding strategies," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 944–948.

[30] K. Heafield, "Kenlm: Faster and smaller language model queries," in *Proceedings of the sixth workshop on statistical machine translation*, 2011, pp. 187–197.