ON LANGUAGE MODEL INTEGRATION FOR RNN TRANSDUCER BASED SPEECH RECOGNITION

Wei Zhou[1,2], Zuoyun Zheng[1], Ralf Schlüter[1,2], Hermann Ney[1,2]

[1]Human Language Technology and Pattern Recognition, Computer Science Department,
RWTH Aachen University, 52074 Aachen, Germany
[2]AppTek GmbH, 52062 Aachen, Germany

ABSTRACT

The mismatch between an external language model (LM) and the implicitly learned internal LM (ILM) of RNN-Transducer (RNN-T) can limit the performance of LM integration such as simple shallow fusion. A Bayesian interpretation suggests to remove this sequence prior as ILM correction. In this work, we study various ILM correction-based LM integration methods formulated in a common RNN-T framework. We provide a decoding interpretation on two major reasons for performance improvement with ILM correction, which is further experimentally verified with detailed analysis. We also propose an exact-ILM training framework by extending the proof given in the hybrid autoregressive transducer, which enables a theoretical justification for other ILM approaches. Systematic comparison is conducted for both in-domain and cross-domain evaluation on the Librispeech and TED-LIUM Release 2 corpora, respectively. Our proposed exact-ILM training can further improve the best ILM method.

Index Terms— speech recognition, transducer, language model

1. INTRODUCTION & RELATED WORK

End-to-end (E2E) speech recognition has shown great simplicity and state-of-the-art performance [1, 2]. Common E2E approaches include connectionist temporal classification (CTC) [3], recurrent neural network transducer (RNN-T) [4] and attention-based encoder-decoder models [5, 6]. While E2E models are only trained on paired audio-transcriptions in general, an external language model (LM) trained on much larger amount of text data (from possibly better-matched domain) can further boost the performance. Without modifying the model structure, shallow fusion (SF) [7] has been a widely-used effective LM integration approach for E2E models, which simply applies a log-linear model combination.

However, with context dependency directly included in the posterior distribution, both RNN-T and attention models implicitly learn an internal LM (ILM) as a sequence prior restricted to the audio transcription only. This ILM usually has a strong mismatch to the external LM, which can limit the performance of LM integration such as simple SF. Existing approaches to handle the ILM fall into 3 major categories:

- ILM suppression: The ILM can be suppressed during the E2E model training via limiting context/model size [8] or introducing an external LM at early stage [9].
- ILM correction: The ILM can be estimated in various ways [10, 11, 12, 13, 8] and then corrected from the posterior in decoding, which fits into a Bayesian interpretation.
- ILM adaptation: The ILM can be adapted on the same text data used by the external LM to alleviate the mismatch. This can be done via text to speech [14, 15] to train the E2E model or partial model update directly on the text [16].

ILM suppression requires training modification and performs similarly as ILM correction [8]. ILM adaptation has a even higher complexity and may still need ILM correction for further improvement [14]. Therefore, ILM correction appears to be the most simple and effective approach for LM integration, which also has a better mathematical justification [10, 11].

The density ratio [10] approach estimates the ILM directly from the transcription statistics. The hybrid autoregressive transducer (HAT) [11] proposed to estimate the ILM from the transducer neural network (NN) by excluding the impact of the encoder, which is justified with a detailed proof under certain approximation. This approach is further investigated in [12] and extended by including the average encoder output in [13]. In [11, 17], ILM training (ILMT) is applied to include the ILM into the E2E model training for joint optimization. Recently, [8] conducted a comprehensive study on ILM methods for the attention model, and proposed the mini-LSTM approach that outperforms all previous methods.

In this work, we study various ILM correction-based LM integration methods for the RNN-T model. We formulate different ILM approaches proposed for different E2E systems into a common RNN-T framework. We provide a decoding interpretation on two major reasons for performance improvement with ILM correction, which is further experimentally verified with detailed analysis. Additionally, we extend the HAT proof [11] and propose an exact-ILM training framework which enables a theoretical justification for other ILM approaches. Systematic comparison is conducted for both in-domain and cross-domain evaluation on the Librispeech [18] and TED-LIUM Release 2 (TLv2) [19] corpora, respectively. The effect of ILMT on these approaches is also investigated.

2. RNN-TRANSDUCER

Given a speech utterance, let a_1^S denote the output (sub)word sequence of length S from a vocabulary V. Let X denote the corresponding acoustic feature sequence and $h_1^T = f^{\text{enc}}(X)$ denote the encoder output, which transforms the input into high-level representations. The RNN-T model [4] defines the sequence posterior as:

$$P_{\text{RNNT}}(a_1^S|X) = \sum_{y_1^{U=T+S}:\mathcal{B}^{-1}(a_1^S)} P_{\text{RNNT}}(y_1^U|h_1^T)$$

$$= \sum_{y_1^U:\mathcal{B}^{-1}(a_1^S)} \prod_{u=1}^{U=T+S} P_{\text{RNNT}}(y_u|\mathcal{B}(y_1^{u-1}), h_1^T) \quad (1)$$

Here y_1^U is the blank ϵ-augmented alignment sequence, which is uniquely mapped to a_1^S via the collapsing function $\mathcal{B}$ to remove all blanks. The common NN structure of RNN-T contains an encoder f^{enc}, a prediction network f^{pred} and a joint network J followed by a softmax activation. We denote θ_{RNNT} as the RNN-T NN parameters. The probability in Eq. (1) can also be represented from the lattice representation of the RNN-T topology [4]. By denoting y_1^{u-1} as a path reaching a node $(t, s-1)$, we have:

$$P_{\text{RNNT}}(y_u|\mathcal{B}(y_1^{u-1}), h_1^T) = P_{\text{RNNT}}(y_u|a_1^{s-1}, h_t) \quad (2)$$

$$= \text{Softmax}\left[J\left(f^{\text{pred}}(a_1^{s-1}), f_t^{\text{enc}}(X) \right) \right]$$

where y_1^u reaches $(t+1, s-1)$ if $y_u = \epsilon$ or (t, s) otherwise.

Without an external LM, $P_{\text{RNNT}}(a_1^S|X)$ can be directly applied into the maximum a posteriori (MAP) decoding to obtain the optimal output sequence:

$$X \rightarrow \tilde{a}_1^{\tilde{S}} = \arg\max_{a_1^S, S} \; P(a_1^S|X) \quad (3)$$

Within a Bayesian framework, to integrate the RNN-T model and an external LM jointly into Eq. (3) as modularized components, the Bayes' theorem needs to be applied:

$$X \rightarrow \tilde{a}_1^{\tilde{S}} = \arg\max_{a_1^S, S} \; P(a_1^S) \cdot P(X|a_1^S) \quad (4)$$

$$= \arg\max_{a_1^S, S} \; P_{\text{LM}}^{\lambda 1}(a_1^S) \cdot \frac{P_{\text{RNNT}}(a_1^S|X)}{P_{\text{RNNT-ILM}}^{\lambda 2}(a_1^S)} \quad (5)$$

which suggests the removal of the RNN-T model's internal sequence prior $P_{\text{RNNT-ILM}}$. Here $\lambda 1$ and $\lambda 2$ are scales applied in common practice. The SF approach [7] essentially omits $P_{\text{RNNT-ILM}}$ completely with $\lambda 2 = 0$.

3. INTERNAL LM

$P_{\text{RNNT-ILM}}$ is effectively the ILM, which is implicitly learned and contained in P_{RNNT} after training. It should be most accurately obtained by marginalization $\sum_X P_{\text{RNNT}}(a_1^S|X)P(X)$. Since the exact summation is intractable, an estimated P_{ILM} is usually applied for approximation.

3.1. ILM Estimation

One straightforward way is to assume that $P_{\text{RNNT-ILM}}$ closely captures the statistics of the acoustic training transcription.

This is the density ratio approach [10] which trains a separate P_{ILM} on the audio transcription.

To be more consistent with the P_{RNNT} computation, another popular direction is to partially reuse the RNN-T NN for computing P_{ILM}. A general formulation can be given as:

$$P_{\text{ILM}}(a_1^S) = \prod_{s=1}^S P_{\text{ILM}}(a_s|a_1^{s-1}) = \prod_{s=1}^S P'(a_s|a_1^{s-1}, h') \quad (6)$$

where P' is defined over V and h' is some global representation. Here P' shows a strong correspondence to Eq. (2) except that P_{RNNT} is defined over $V \cup \{\epsilon\}$ in general. In [11], a separate blank distribution in P_{RNNT} is proposed to directly use its label distribution for P'. However, the same can always be achieved by a simple renormalization:

$$P'(a_s|a_1^{s-1}, h') = \frac{P_{\text{RNNT}}(a_s|a_1^{s-1}, h')}{1 - P_{\text{RNNT}}(\epsilon|a_1^{s-1}, h')} \quad (7)$$

$$= \text{Softmax}\left[J_{\backslash\epsilon}\left(f^{\text{pred}}(a_1^{s-1}), h' \right) \right]$$

where $J_{\backslash\epsilon}$ stands for the joint network J excluding the blank logit output. This is also done in [12]. In fact, we recommend to always use a single distribution for P_{RNNT}, since it is equivalent to the separate distributions in [11] after renormalization, but has the advantage of further discrimination between blank and speech labels. This can partially prevent blank being too dominant which may lead to a sensitive decoding behavior with high deletion errors as observed in [20]. Existing ILM estimation approaches are then categorized by the way of representing h':

1. $h'_{\text{zero}} : h' = \vec{0}$ [11, 12]
2. $h'_{\text{avg}} : h' = \text{mean}(h_1^T)$ [13, 8]
3. $h'_{a_1^{s-1}} : h' = f_{\theta_{\text{ILM}}}(a_1^{s-1})$ where an additional NN $f_{\theta_{\text{ILM}}}$ is introduced to generate h' based on a_1^{s-1}. The mini-LSTM method [8] falls into this category (denoted as $h'_{\text{mini-LSTM}}$).

All these approaches are based on fixed θ_{RNNT}. For the $h'_{a_1^{s-1}}$ approach, an LM-like loss $\mathcal{L}_{\text{ILM}} = -\log P_{\text{ILM}}(a_1^S)$ based on Eq. (6) and Eq. (7) is used to train the additional $f_{\theta_{\text{ILM}}}$ over the audio transcription. This effectively combines the advantage of using transcription statistics and partial RNN-T NN.

3.2. ILM Training

The RNN-T model is commonly trained with a full-sum loss $\mathcal{L}_{\text{RNNT}} = -\log P_{\text{RNNT}}(a_1^S|X)$ over all alignment paths as in Eq. (1). When reusing the RNN-T NN for P_{ILM}, one can also combine $\mathcal{L}_{\text{RNNT}}$ and $\mathcal{L}_{\text{ILM}}$ as a multi-task training to train all parameters including θ_{RNNT} jointly:

$$\mathcal{L}_{\text{ILMT}} = \mathcal{L}_{\text{RNNT}} + \alpha \mathcal{L}_{\text{ILM}}$$

where α is a scaling factor. This joint training is applied to the h'_{zero} approach in [11, 17], which we also apply to the h'_{avg} and $h'_{a_1^{s-1}}$ approaches in this work.

3.3. Decoding Interpretation

Since both P_{LM} and P_{ILM} are only defined over V, we can further expand Eq. (5) as:

$$\arg\max_{a_1^S, S} \sum_{y_1^U : \mathcal{B}^{-1}(a_1^S)} \prod_{u=1}^{U} P_{\text{RNNT}}(y_u|\mathcal{B}(y_1^{u-1}), h_1^T) \cdot Q(y_u|\mathcal{B}(y_1^{u-1}))$$

$$\text{with } Q(y_u|\mathcal{B}(y_1^{u-1})) = \begin{cases} 1, & y_u = \epsilon \\ \frac{P_{\text{LM}}^{\lambda 1}(y_u|\mathcal{B}(y_1^{u-1}))}{P_{\text{ILM}}^{\lambda 2}(y_u|\mathcal{B}(y_1^{u-1}))}, & y_u \neq \epsilon \end{cases}$$

which reflects the exact scoring used at each search step. This also reveals two reasons from decoding perspective why applying ILM can improve the recognition performance:

R1. The label distribution of P_{RNNT} is rebalanced with the prior removal, so that we rely more on the external LM for context modeling, which is a desired behavior.

R2. The division by P_{ILM} boosts the label probability against the (usually high) blank probability, so that the importance of the external LM ($\lambda 1$) can be increased without suffering a large increment of deletion errors.

The R2 explains why SF ($\lambda 2 = 0$) can only achieve a limited performance. It may also alleviate the necessity of heuristic approaches in decoding such as length-normalization [4, 20] and length-reward [21, 10, 22]. However, an increasing $\lambda 2$ with more boosting can also lead to an increment of insertion and/or substitution errors. Therefore, both scales require careful tuning in practice.

3.4. Exact-ILM Training

In the appendix A of [11], a detailed proof is given to show:

$$P_{\text{RNNT-ILM}}(a_s|a_1^{s-1}) \propto \exp\left[J_{\backslash\epsilon}\left(f^{\text{pred}}(a_1^{s-1})\right)\right] \quad (8)$$

if the following assumption can hold:

$$J_{\backslash\epsilon}\left(f^{\text{pred}}(a_1^{s-1}), f_t^{\text{enc}}(X)\right) = J_{\backslash\epsilon}\left(f^{\text{pred}}(a_1^{s-1})\right) + J_{\backslash\epsilon}\left(f_t^{\text{enc}}(X)\right)$$

Eq. (8) then leads to the h'_{zero} ILM approach. However, this assumption does not always hold if J contains some non-linearity, which is mostly the case for better performance.

Here we propose to further generalize the assumption:

$$J_{\backslash\epsilon}\left(f^{\text{pred}}(a_1^{s-1}), f_t^{\text{enc}}(X)\right) = J'(a_1^{s-1}) + J_{\backslash\epsilon}\left(f_t^{\text{enc}}(X)\right) \quad (9)$$

where J' can be any function with output size $|V|$. As long as J' is independent of X, the proof still holds and we have:

$$P_{\text{RNNT-ILM}}(a_s|a_1^{s-1}) \propto \exp\left[J'(a_1^{s-1})\right] \quad (10)$$

Instead of relying on the assumption, we can train J' to fulfill Eq. (9) towards an exact ILM estimation eventually. Besides the mean squared error loss, this exact-ILM training objective $\mathcal{L}_{J'}$ can also be the cross entropy (CE) loss, since both sides of Eq. (9) are actually the logits output over V, which we can directly apply softmax. The CE loss may introduce a constant shift to Eq. (9), which is still valid for the proof in HAT [11] to achieve Eq. (10). Additionally, by performing Viterbi alignment of the acoustic data with a (pre)trained RNN-T model, $\mathcal{L}_{J'}$ can be further simplified to consider only those h_t where a_s occurs.

Among various possibilities, we can directly apply this exact-ILM training to the $h'_{a_1^{s-1}}$ approach by defining:

$$J'(a_1^{s-1}) = J_{\backslash\epsilon}\left(f^{\text{pred}}(a_1^{s-1}), f_{\theta_{\text{ILM}}}(a_1^{s-1})\right)$$

and train $f_{\theta_{\text{ILM}}}$ by the following multi-task training:

$$\mathcal{L}_{\text{ILM}}^{\text{exact}} = \mathcal{L}_{\text{ILM}} + \alpha\mathcal{L}_{J'}$$

This extension directly enables a theoretical justification for the $h'_{a_1^{s-1}}$ ILM approach, which is trained towards Eq. (10). Similar as $\mathcal{L}_{\text{ILMT}}$, we can also combine $\mathcal{L}_{\text{RNNT}}$ and $\mathcal{L}_{J'}$ to have a joint training of all parameters, where the model is additionally forced to better fulfill the assumption towards an exact ILM estimation. This may require a careful design of J', which is not investigated in this work.

4. EXPERIMENTS

4.1. Setup

Experimental verification is done on the 960h Librispeech corpus [18] for in-domain evaluation, and on the TLv2 corpus [19] for out-of-domain evaluation. We use 5k acoustic data-driven subword modeling (ADSM) units [23] obtained from the Librispeech corpus. We follow the ADSM text segmentation and train two individual external LMs on the corresponding LM data of each corpus. The Librispeech LM contains 32 Transformer layers [24] and the TLv2 LM contains 4 long short-term memory (LSTM) [25] layers with 2048 units.

We use 50-dim gammatone features [26] and a strictly monotonic ($U = T$) version of RNN-T [27]. The encoder f^{enc} contains 2 convolutional layers followed by 6 bidirectional-LSTM (BLSTM) layers with 640 units for each direction. A subsampling of factor 4 is applied via 2 max-pooling layers in the middle of BLSTM stacks. The prediction network f^{pred} contains an embedding layer of size 256 and 2 LSTM layers with 640 units. We use the standard additive joint network for J which contains 1 linear layer of size 1024 followed by the tanh activation, and another linear layer followed by the final softmax. The RNN-T model is only trained on the Librispeech corpus. We firstly apply a Viterbi training variant [28] for 30 full epochs and then fine-tune the model with $\mathcal{L}_{\text{RNNT}}$ for 15 full epochs. This model is used as the base model for all further experiments.

The density ratio [10] LM uses the same structure as f^{pred}. For the $h'_{a_1^{s-1}}$ approach, we apply the mini-LSTM network [8] for $f_{\theta_{\text{ILM}}}$ which is trained with either $\mathcal{L}_{\text{ILM}}$ or $\mathcal{L}_{\text{ILM}}^{\text{exact}}$ for 0.5-1 full epoch on Librispeech. For the $\mathcal{L}_{\text{ILM}}^{\text{exact}}$, we use the CE loss for $\mathcal{L}_{J'}$, and $\alpha = 1.0$ and 2.0 for in-domain and cross-domain evaluation, respectively. The base RNN-T model is used to generate the Viterbi alignment for $\mathcal{L}_{J'}$ simplification.

For $\mathcal{L}_{\text{ILMT}}$, we follow [17] to initialize with the base model and adjust $\alpha = 0.2$. To avoid the potential improvement just form a much longer re-training with learning rate reset [29], we only apply fine-tuning with $\mathcal{L}_{\text{ILMT}}$ for upto 10 more full epochs on Librispeech. Since $\mathcal{L}_{\text{ILM}}$ is only relevant for f^{pred} and J, we also freeze f^{enc} during this procedure. All the best epochs based on no-LM recognition show the same or slightly better performance than the base model, which are used for

Table 1: *WER[%] results of LM integration evaluation on the in-domain Librispeech and out-of-domain TLv2 corpora.*

Model Train	Evaluation	Librispeech				TLv2	
		dev		test		dev	test
		clean	other	clean	other		
$\mathcal{L}_{\text{RNNT}}$	no LM	3.3	9.7	3.6	9.5	19.8	20.3
	SF	2.0	5.1	2.2	5.5	15.5	16.4
	density ratio	1.9	4.8	2.1	5.2	14.1	15.0
	h'_{zero}	1.8	4.4	2.0	4.8	13.6	14.4
	h'_{avg}	1.8	4.4	2.0	4.9	13.5	14.6
+ $\mathcal{L}_{\text{ILM}}$	$h'_{\text{mini-LSTM}}$	1.8	4.3	1.9	4.7	13.4	14.4
+ $\mathcal{L}_{\text{ILM}}^{\text{exact}}$		**1.8**	**4.2**	**1.9**	**4.6**	**13.2**	**14.0**
$\mathcal{L}_{\text{ILMT}}$	h'_{zero}	1.8	4.4	2.0	4.8	13.3	14.2
	h'_{avg}	1.9	4.5	2.1	4.9	13.5	14.4
	$h'_{\text{mini-LSTM}}$	1.8	4.4	2.0	4.8	13.2	14.1

further LM integration evaluation.

The decoding follows the description in Section 3.3. We apply alignment-synchronous search [30] with score-based pruning and a beam limit of 128. We explicitly do not apply any heuristic approach for decoding to better reflect the effect of each LM integration method. All scales are optimized on the dev sets using grid search.

4.2. LM Integration Evaluation

Table 1 shows the word error rate (WER) results of the aforementioned LM integration methods evaluated on the in-domain Librispeech and out-of-domain TLv2 tasks. As expected, the external LMs bring significant improvement over the standalone RNN-T, and all ILM correction approaches improves further over the simple SF. For both tasks, all h'-based methods outperform the density ratio approach. Both h'_{zero} and h'_{avg} show similar performance, and the $h'_{\text{mini-LSTM}}$ trained with $\mathcal{L}_{\text{ILM}}$ performs slightly better than the two. The proposed $\mathcal{L}_{\text{ILM}}^{\text{exact}}$ further improves the $h'_{\text{mini-LSTM}}$ approach.

The $\mathcal{L}_{\text{ILMT}}$ improves all three h'-based approaches on the cross-domain TLv2 task, while no positive effect is obtained on the in-domain Librispeech task. This is in line with the observation in HAT [11], where a decreasing $\mathcal{L}_{\text{ILM}}$ leads to no improvement on the overall performance. The $\mathcal{L}_{\text{ILM}}^{\text{exact}}$ trained $h'_{\text{mini-LSTM}}$ approach performs the best in all cases.

4.3. Analysis

To verify the 2 decoding-perspective benefits of ILM correction as claimed in Section 3.3, we conduct additional analytical experiments on the Librispeech dev-other set using the base RNN-T model. To simulate the effect of boosting label probability (R2) without the effect of rebalanced label distribution (R1), we apply a constant length reward upon SF for decoding. To simulate the effect of R1 without the effect of R2, we apply a modified h'_{zero} evaluation as following. For each $y_u \neq \epsilon$, we firstly apply a renormalization as:

$$P_{\text{norm}}(y_u) = \frac{P_{\text{RNNT}}(y_u|\mathcal{B}(y_1^{u-1}), h_1^T)/P_{\text{ILM}}^{\lambda 2}(y_u|\mathcal{B}(y_1^{u-1}))}{\sum_{a \in V} P_{\text{RNNT}}(a|\mathcal{B}(y_1^{u-1}), h_1^T)/P_{\text{ILM}}^{\lambda 2}(a|\mathcal{B}(y_1^{u-1}))}$$

Then we modify the probability of $y_u \neq \epsilon$ used for search as:

Table 2: *WER[%] and Sub,Del,Ins error rate[%] results on the dev-other set of the in-domain Librispeech corpus. Analytical evaluation using the base RNN-T model.*

Evaluation	$\lambda 1$	$\lambda 2$	Librispeech dev-other			
			WER	Sub	Del	Ins
SF	0.61	0	5.1	3.8	0.9	0.4
+ length reward	0.65		4.8	3.8	0.5	0.5
h'_{zero} + renorm-ϵ	0.61	0.35	4.9	3.6	0.9	0.4
+ length reward	0.65		4.6	3.7	0.5	0.4
h'_{zero}	0.85	0.4	**4.4**	**3.5**	**0.5**	**0.4**
+ length reward	0.95		4.5	3.6	0.5	0.4

$$\left(1 - P_{\text{RNNT}}(\epsilon|\mathcal{B}(y_1^{u-1}), h_1^T)\right) \cdot P_{\text{norm}}(y_u) \cdot P_{\text{LM}}^{\lambda 1}(y_u|\mathcal{B}(y_1^{u-1}))$$

The first two terms of the product effectively restrict the label probability back to a single distribution w.r.t. the blank probability of P_{RNNT}, but still maintain the rebalanced label distribution to some extent. We denote this operation as renorm-ϵ. Table 2 shows the WER, substitution (Sub), deletion (Del) and insertion (Ins) error rate results together with the optimized scales for these experiments.

For the baseline SF, the optimal $\lambda 1$ already leads to a high Del error and we can not increase it further. Boosting the label probability with length reward largely reduces the Del error to the same level as Ins. It also allows a slight increase of the external LM importance ($\lambda 1$) for better performance. This verifies the individual effect of R2. Rebalancing the label distribution with h'_{zero} + renorm-ϵ reduces the Sub error as we rely more on the external LM for context modeling. However, it still suffers the high Del error without the boosting effect. This verifies the individual effect of R1. When combining length reward and h'_{zero} + renorm-ϵ, we see that the benefits are complementary. Finally, applying the h'_{zero} ILM correction allows further enlarging the effect of R1 and R2 with larger scales, and thus achieves further improvement. It also eliminates the need of length reward.

5. CONCLUSION

In this work, we provided a detailed formulation to compare various ILM correction-based LM integration methods in a common RNN-T framework. We explained two major reasons for performance improvement with ILM correction from a decoding interpretation, which are experimentally verified with detailed analysis. Moreover, we proposed an exact-ILM training framework by extending the proof in HAT [11], which enables a theoretical justification for other ILM approaches. All investigated LM integration methods are systematically compared on the in-domain Librispeech and out-of-domain TLv2 tasks. The recently proposed $h'_{\text{mini-LSTM}}$ ILM approach for the attention model also performs the best for the RNN-T model. Our proposed exact-ILM training can further improve its performance.

6. REFERENCES

[1] Zoltán Tüske, George Saon, Kartik Audhkhasi, and Brian Kingsbury, "Single Headed Attention based Sequence-to-sequence Model for State-of-the-Art Results on Switchboard," in *Proc. Interspeech*, 2020, pp. 551–555.

[2] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech*, 2020, pp. 5036–5040.

[3] Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber, "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks," in *Proc. ICML*, 2006, pp. 369–376.

[4] Alex Graves, "Sequence Transduction with Recurrent Neural Networks," 2012, https://arxiv.org/abs/1211.3711.

[5] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio, "End-to-End Attention-based Large Vocabulary Speech Recognition," in *Proc. ICASSP*, 2016, pp. 4945–4949.

[6] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals, "Listen, Attend and Spell: A Neural Network for Large Vocabulary Conversational Speech Recognition," in *Proc. ICASSP*, 2016, pp. 4960–4964.

[7] Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "On Using Monolingual Corpora in Neural Machine Translation," 2015, http://arxiv.org/abs/1503.03535.

[8] Mohammad Zeineldeen, Aleksandr Glushko, Wilfried Michel, Albert Zeyer, Ralf Schlüter, and Hermann Ney, "Investigating Methods to Improve Language Model Integration for Attention-based Encoder-Decoder ASR Models," in *Proc. Interspeech*, 2021, pp. 2856–2860.

[9] Wilfried Michel, Ralf Schlüter, and Hermann Ney, "Early Stage LM Integration Using Local and Global Log-Linear Combination," in *Proc. Interspeech*, 2020, pp. 3605–3609.

[10] Erik McDermott, Hasim Sak, and Ehsan Variani, "A Density Ratio Approach to Language Model Fusion in End-to-End Automatic Speech Recognition," in *IEEE ASRU*, 2019, pp. 434–441.

[11] Ehsan Variani, David Rybach, Cyril Allauzen, and Michael Riley, "Hybrid Autoregressive Transducer (HAT)," in *Proc. ICASSP*, 2020, pp. 6139–6143.

[12] Zhong Meng, Sarangarajan Parthasarathy, Eric Sun, Yashesh Gaur, Naoyuki Kanda, Liang Lu, Xie Chen, Rui Zhao, Jinyu Li, and Yifan Gong, "Internal Language Model Estimation for Domain-Adaptive End-to-End Speech Recognition," in *IEEE SLT*, 2021, pp. 243–250.

[13] Albert Zeyer, André Merboldt, Wilfried Michel, Ralf Schlüter, and Hermann Ney, "Librispeech Transducer Model with Internal Language Model Prior Correction," in *Proc. Interspeech*, 2021.

[14] Yan Deng, Rui Zhao, Zhong Meng, Xie Chen, Bing Liu, Jinyu Li, Yifan Gong, and Lei He, "Improving RNN-T for Domain Scaling Using Semi-Supervised Training with Neural TTS," in *Proc. Interspeech*, 2021, pp. 751–755.

[15] Gakuto Kurata, George Saon, Brian Kingsbury, David Haws, and Zoltán Tüske, "Improving Customization of Neural Transducers by Mitigating Acoustic Mismatch of Synthesized Audio," in *Proc. Interspeech*, 2021, pp. 2027–2031.

[16] Janne Pylkkönen, Antti Ukkonen, Juho Kilpikoski, Samu Tamminen, and Hannes Heikinheimo, "Fast Text-Only Domain Adaptation of RNN-Transducer Prediction Network," in *Proc. Interspeech*, 2021, pp. 1882–1886.

[17] Zhong Meng, Naoyuki Kanda, Yashesh Gaur, Sarangarajan Parthasarathy, Eric Sun, Liang Lu, Xie Chen, Jinyu Li, and Yifan Gong, "Internal Language Model Training for Domain-Adaptive End-To-End Speech Recognition," in *Proc. ICASSP*, 2021, pp. 7338–7342.

[18] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. ICASSP*, 2015, pp. 5206–5210.

[19] Anthony Rousseau, Paul Deléglise, and Yannick Estève, "Enhancing the TED-LIUM Corpus with Selected Data for Language Modeling and More TED Talks," in *Proc. LREC*, 2014, pp. 3935–3939.

[20] Liang Lu, Zhong Meng, Naoyuki Kanda, Jinyu Li, and Yifan Gong, "On Minimum Word Error Rate Training of the Hybrid Autoregressive Transducer," in *Proc. Interspeech*, 2021, pp. 3435–3439.

[21] Awni Y. Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng, "Deep Speech: Scaling up end-to-end speech recognition," 2014, http://arxiv.org/abs/1412.5567.

[22] George Saon, Zoltán Tüske, Daniel Bolaños, and Brian Kingsbury, "Advancing RNN Transducer Technology for Speech Recognition," in *Proc. ICASSP*, 2021, pp. 5654–5658.

[23] Wei Zhou, Mohammad Zeineldeen, Zuoyun Zheng, Ralf Schlüter, and Hermann Ney, "Acoustic Data-Driven Subword Modeling for End-to-End Speech Recognition," in *Proc. Interspeech*, 2021, pp. 2886–2890.

[24] Kazuki Irie, Albert Zeyer, Ralf Schlüter, and Hermann Ney, "Language Modeling with Deep Transformers," in *Proc. Interspeech*, Graz, Austria, 2019, pp. 3905–3909.

[25] Sepp Hochreiter and Jürgen Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[26] Ralf Schlüter, Ilja Bezrukov, Hermann Wagner, and Hermann Ney, "Gammatone Features and Feature Combination for Large Vocabulary Speech Recognition," in *Proc. ICASSP*, 2007, pp. 649–652.

[27] Anshuman Tripathi, Han Lu, Hasim Sak, and Hagen Soltau, "Monotonic Recurrent Neural Network Transducer and Decoding Strategies," in *IEEE ASRU*, 2019, pp. 944–948.

[28] Wei Zhou, Simon Berger, Ralf Schlüter, and Hermann Ney, "Phoneme Based Neural Transducer for Large Vocabulary Speech Recognition," in *Proc. ICASSP*, 2021, pp. 5644–5648.

[29] Peter Vieting, Christoph Lüscher, Wilfried Michel, Ralf Schlüter, and Hermann Ney, "On Architectures and Training for Raw Waveform Feature Extraction in ASR," in *IEEE ASRU*, 2021.

[30] George Saon, Zoltán Tüske, and Kartik Audhkhasi, "Alignment-Length Synchronous Decoding for RNN Transducer," in *Proc. ICASSP*, 2020, pp. 7804–7808.