



On Minimum Word Error Rate Training of the Hybrid Autoregressive Transducer

Liang Lu, Zhong Meng, Naoyuki Kanda, Jinyu Li, and Yifan Gong

Microsoft Corp., USA

{liang.lu, nakanda, zhong.meng, jinyuli, yifan.gong}@microsoft.com

Abstract

Hybrid Autoregressive Transducer (HAT) is a recently proposed end-to-end acoustic model that extends the standard Recurrent Neural Network Transducer (RNN-T) for the purpose of the external language model (LM) fusion. In HAT, the blank probability and the label probability are estimated using two separate probability distributions, which provides a more accurate solution for internal LM score estimation, and thus works better when combining with an external LM. Previous work mainly focuses on HAT model training with the negative log-likelihood loss, while in this paper, we study the minimum word error rate (MWER) training of HAT – a criterion that is closer to the evaluation metric for speech recognition, and has been successfully applied to other types of end-to-end models such as sequence-to-sequence (S2S) and RNN-T models. From experiments with around 30,000 hours of training data, we show that MWER training can improve the accuracy of HAT models, while at the same time, improving the robustness of the model against the decoding hyper-parameters such as length normalization and decoding beam during inference.

Index Terms: Hybrid autoregressive transducer, Minimum word error rate loss, End-to-end speech recognition

1. Introduction

End-to-end (E2E) speech recognition has been an active research topic for the past few years, and there has been a tremendous progress in this field. In particular, two categories of E2E models have been explored for speech recognition tasks, i.e., label-synchronous model such as sequence-to-sequence with attention model (S2S) [1, 2, 3], and time-synchronous model such as connectionist temporal classification (CTC) [4] and recurrent neural network transducer (RNN-T) [5]. Compared to CTC, RNN-T can perform joint acoustic and language modeling with the augmented prediction network, and therefore it usually yields higher recognition accuracy. Since the time-synchronous model is more natural and straightforward for streaming speech recognition, most recent research efforts have been focusing on RNN-T and its variants, and the model has been successfully applied for on-device speech recognition [6].

The conventional hybrid speech recognition system integrates individual acoustic model and language model (LM) by the Bayes rule during decoding. It usually requires a large decoding graph for large-vocabulary speech recognition problems, and therefore is not very feasible to be deployed to computation and memory constrained platforms such as mobile devices. In contrast, the E2E approach folds the acoustic model, lexicon and LM component into a single model, which enjoys the advantage of a smaller memory footprint. This feature makes E2E models more preferable for on-device speech recognition scenarios with very limited computational capacity and memory. However, by performing joint acoustic and language mod-

eling, E2E models have a critical drawback that they lose the flexibility for target-domain LM adaption. Indeed, external LM fusion has been a long lasting research problem since the emergence of the E2E approach in speech recognition, and numerous approaches have been proposed to tackle this problem [7, 8, 9, 10, 11, 12]. More recently, McDermott et al. [13] proposed the density ratio approach for LM fusion that demonstrates higher accuracy compared to the popular shallow fusion approach.

Instead of LM fusion, another approach is to modularize the E2E models so that the acoustic model and LM factor can be more disentangled. Along this line, the hybrid autoregressive transducer (HAT) [14] builds upon the standard RNN-T model by introducing two distinct distributions for labels and the blank token respectively. The advantage is that the internal LM score is readily available by ignoring the blank probabilities, which can be used for the external LM fusion by a scoring function similar to the one used in the density ratio approach. In [14], the authors mainly investigated HAT with the negative log-likelihood (NLL) training criterion. In this paper, we study the minimum word error rate (MWER) training criterion for HAT. While there have been a few studies of MWER for E2E models such as S2S [15, 16] and RNN-T [17, 18], this work concerns the HAT model, which may be viewed as a variant of RNN-T, and we discuss the robustness of MWER training against the decoding hyperparameters such as length normalization and the decoding beam that has not been well covered in the existing literature. The contributions of the paper are summarized as follows.

1. We study the MWER training of HAT models, and discuss the robustness of MWER training against length normalization and the size of inference beam.
2. We perform a comparison of HAT and RNN-T with and without length normalization
3. We investigate the HAT inference with an external LM with and without MWER training.

2. Related work

Minimum Bayesian Risk (MBR) training has been successfully applied to speech recognition within the hybrid framework [19, 20, 21, 22] with a long history, which can operate at different granularities such as. phone-level MPE [19] or state-level sMBR [20, 21]. Word-level MBR, which is also referred to as MWER, is firstly proposed for the hybrid model [23], but the same type of criterion was also shown to improve end-to-end models such as sequence-to-sequence model [15, 16] and RNN-T [17, 18]. Since the exact sequence-level posterior is computationally intractable, Shannon [23] proposed a sampling approach, while in [15], the authors showed that computing the posterior from an N-best list worked equally well or even better

with a much simpler implementation. As for RNN-T models, a single alignment of each hypothesis was used to derive the gradients of the MWER loss [17], while in [18], the gradients were computed by marginalizing all the possible alignments. In this work, we study the MWER training of HAT, which is a variant of RNN-T model that uses two distributions to estimate the blank token probability and label probability, respectively. We used the N-best list approach to estimate the sequence-level posteriors by marginalizing over all the possible alignments as in [18], and to compute the gradients. We study the robustness of MWER training against the decoding hyper-parameters that was briefly discussed in [24] in the context of speech recognition of rare words with LM fusion.

3. RNN-T and HAT

3.1. Recurrent Neural Network Transducer

RNN-T performs sequence transduction in a time-synchronous fashion. Given an acoustic feature sequence $\mathbf{x} = \{x_1, \dots, x_T\}$ and its corresponding label sequence $\mathbf{y} = \{y_1, \dots, y_U\}$, where T is the length of the acoustic sequence, and U is the length of the label sequence, RNN-T defines the conditional probability as

$$P(\mathbf{y} | \mathbf{x}) = \sum_{\tilde{\mathbf{y}} \in \mathcal{B}^{-1}(\mathbf{y})} P(\tilde{\mathbf{y}} | \mathbf{x}), \quad (1)$$

where $\tilde{\mathbf{y}}$ is a path that contains the blank token \emptyset , and the function \mathcal{B} denotes mapping the path to \mathbf{y} by removing the blank tokens in $\tilde{\mathbf{y}}$. Essentially, the probability $P(\mathbf{y} | \mathbf{x})$ is calculated by summing over the probabilities of all the possible paths that can be mapped to the label sequence after applying the function \mathcal{B} . The probability can be efficiently computed by the forward-backward algorithm, which requires to compute the probability of each step, i.e.,

$$P(k | x_{[1:t]}, y_{[1:u]}) = \frac{\exp(J(f_t^k + g_u^k)/Z)}{\sum_{k' \in \tilde{\mathcal{V}}} \exp(J(f_t^{k'} + g_u^{k'}/Z))}, \quad (2)$$

where \mathbf{f}_t and \mathbf{g}_u are the output vectors from the audio encoder network and the transcription network at the time step t and u respectively, and $J(\cdot)$ denotes a non-linear function followed by an affine transform. $\tilde{\mathcal{V}}$ denotes the set of the vocabulary \mathcal{V} with an additional blank token, i.e., $\tilde{\mathcal{V}} = \mathcal{V} \cup \emptyset$. Z is the temperature for the Softmax function, which is usually set to be 1. Given the distribution of each timestep (t, u) , the sequence-level conditional probability as Eq. (1) can be obtained by the forward-backward algorithm, where the forward variable is defined as

$$\begin{aligned} \alpha(t, u) &= \alpha(t-1, u)P(\emptyset | x_{[1:t-1]}, y_{[1:u]}) \\ &\quad + \alpha(t, u-1)P(y_u | x_{[1:t]}, y_{[1:u-1]}), \end{aligned}$$

while the backward variable can be defined similarly. The probability $P(\mathbf{y} | \mathbf{x})$ can be computed as

$$P(\mathbf{y} | \mathbf{x}) = \alpha(T, U)P(\emptyset | x_{[1:T]}, y_{[1:U]}). \quad (3)$$

The model is usually trained by minimizing the negative log-likelihood (NLL) as

$$\mathcal{L}_{\text{NLL}} = -\log P(\mathbf{y} | \mathbf{x}). \quad (4)$$

Details about the derivatives of the loss can be found in [5].

3.2. Hybrid Autoregressive Transducer

The hybrid autoregressive transducer (HAT) [14] is proposed as an extension of the standard RNN-T, which aims to disentangle the LM factor and the acoustic model factor, and approximate the internal LM score for external LM fusion similar to the form of the density ratio approach [13]. Approximating the internal LM score in the standard RNN-T model is not straightforward, as the token probability is computed over the extended vocabulary set $\tilde{\mathcal{V}}$ instead of \mathcal{V} , indicating that each label probability is biased even after we remove the acoustic feature f_t^k in Eq. (1). HAT reformulates the token probability, and uses two distributions for the blank token and label tokens, i.e.,

$$P(k | x_{[1:t]}, y_{[1:u]}) = \begin{cases} b_{t,u} & k = \emptyset \\ (1 - b_{t,u})\tilde{P}(k | x_{[1:t]}, y_{[1:u]}) & k = y_{u+1} \end{cases} \quad (5)$$

where $b_{t,u}$ is the blank token probability, which can be drawn from a Bernoulli distribution that can be estimated using a Sigmoid function. $\tilde{P}(k | x_{[1:t]}, y_{[1:u]})$ is the label distribution, which is obtained by a Softmax function over the vocabulary set \mathcal{V} instead of $\tilde{\mathcal{V}}$, i.e.,

$$\tilde{P}(k | x_{[1:t]}, y_{[1:u]}) = \frac{\exp(J(f_t^k + g_u^k))}{\sum_{k' \in \mathcal{V}} \exp(J(f_t^{k'} + g_u^{k'}))}. \quad (6)$$

Given the new probability definition as Eq. (5), we can follow the same forward-backward algorithm to compute the sequence-level conditional probability and train the model with NLL loss as Eq. (4). Since the blank distribution and the label distribution are now disentangled, we can approximate the internal LM score of the HAT model by feeding in the transcription network outputs only to the Softmax function, i.e.,

$$\tilde{P}_{\text{ILM}}(\mathbf{y}) = \prod_{u=1}^U \tilde{P}(y_u | y_{1:u-1}) \quad (7)$$

$$= \prod_{u=1}^U \frac{\exp(J(g_{u-1}^{y_u}))}{\sum_{k' \in \mathcal{V}} \exp(J(g_{u-1}^{k'}))}. \quad (8)$$

During inference, HAT adopts the similar decoding rule as in the density ratio approach [13] when the external LM is available. The best path is obtained by maximizing the interpolated score as

$$\begin{aligned} \mathbf{y}^* &= \arg \max_{\mathbf{y}} \left((\log P(\mathbf{y} | \mathbf{x}) - \lambda_1 \log \tilde{P}_{\text{ILM}}(\mathbf{y}) \right. \\ &\quad \left. + \lambda_2 \log P_{\text{LM}}(\mathbf{y})) / |\mathbf{y}| \right), \end{aligned} \quad (9)$$

where λ_1 and λ_2 are interpolation weights, and $P_{\text{LM}}(\mathbf{y})$ denotes the probability from the external LM, which may be trained on a large amount of text-only data, or from a specific domain. The final score is normalized by the length of \mathbf{y} , denoted as $|\mathbf{y}|$, which is referred to as length normalization. As explained in [5], without length normalization the decoder tends to prefer short transcription, and thus resulting in high deletion errors. In the case that the external LM is not available, the HAT model can be decoded in the same way as the standard RNN-T model, i.e.,

$$\mathbf{y}^* = \arg \max_{\mathbf{y}} ((\log P(\mathbf{y} | \mathbf{x})) / |\mathbf{y}|). \quad (10)$$

However, the length normalization approach is not favorable to the streaming scenario, especially with the best-first beam search algorithm [5] as the best path may alter more frequently. We will discuss its impact for both RNN-T and HAT in our experimental section.

4. Minimum Word Error Rate Training

The MWER training criterion, which is also referred to as word-level MBR, aims to mitigate the mismatch between the training criterion and the evaluation metric of a speech recognition model. Instead of minimizing the NLL loss as Eq. (4), MWER minimizes the expected word errors [23]. In the context of RNN-T and HAT, the MWER loss can be formulated as

$$\mathcal{L}_{\text{MWER}} = \sum_{\mathbf{y}_i} \hat{P}(\mathbf{y}_i | \mathbf{x}) R(\mathbf{y}_i, \mathbf{y}^r), \quad (11)$$

where $\hat{P}(\mathbf{y}_i | \mathbf{x})$ denotes the posterior probability of an hypothesis \mathbf{y}_i , and $R(\cdot)$ denotes the risk function, which measures the edit-distance between the hypothesis \mathbf{y}_i and the reference transcription \mathbf{y}^r at the word-level in MWER. While the exact posterior probability is computationally intractable, in practice, an N-best list of hypotheses from beam search decoding are used to compute the empirical posterior probability [15], which can be expressed as

$$\hat{P}(\mathbf{y}_i | \mathbf{x}) = \frac{P(\mathbf{y}_i | \mathbf{x})}{\sum_{\mathbf{y}_i} P(\mathbf{y}_i | \mathbf{x})}, \quad (12)$$

where $P(\mathbf{y}_i | \mathbf{x})$ is the conditional probability defined as Eq. (1) for RNN-T and HAT. To train the model by gradient descent algorithm, we can compute the derivative of $\mathcal{L}_{\text{MWER}}$ with respect to $\log P(\mathbf{y}_i | \mathbf{x})$ instead of the log probability of a specific alignment [17], which is the same algorithm used in [18], i.e.,

$$\frac{\partial \mathcal{L}_{\text{MWER}}}{\partial \log P(\mathbf{y}_i | \mathbf{x})} = \hat{P}(\mathbf{y}_i | \mathbf{x}) (R(\mathbf{y}_i, \mathbf{y}^r) - \bar{R}), \quad (13)$$

where \bar{R} denotes the expected risk as $\bar{R} = \sum_{\mathbf{y}_i} \hat{P}(\mathbf{y}_i | \mathbf{x}) R(\mathbf{y}_i, \mathbf{y}^r)$, which is the weighted average word error in the case of MWER. As pointed out in [23, 15], intuitively the MWER criterion increases the posterior probability of the hypotheses whose word errors are smaller than the expected risk, while decreases the posterior probabilities of those whose word errors are larger than the expected risk. Given Eq. (13), we can compute the derivative of the MWER loss with respect to the probability of each token according to the chain rule, i.e.,

$$\begin{aligned} \frac{\partial \mathcal{L}_{\text{MWER}}}{\partial P(k | x_{[1:t]}, y_{i[1:u]})} &= \frac{\partial \mathcal{L}_{\text{MWER}}}{\partial \log P(\mathbf{y}_i | \mathbf{x})} \\ &\times \frac{\partial \log P(\mathbf{y}_i | \mathbf{x})}{\partial P(k | x_{[1:t]}, y_{i[1:u]})}. \end{aligned} \quad (14)$$

Detailed information regarding the computation of $\frac{\partial \log P(\mathbf{y}_i | \mathbf{x})}{\partial P(k | x_{[1:t]}, y_{i[1:u]})}$ can be found in [5].

5. Experiments and Results

In our experiments, the models were trained with around 30,000 hours of anonymized and transcribed Microsoft data, recorded in various conditions. Our evaluation dataset has around 260,000 utterances from various domains including both reading and conversational speech recorded in close-talk and far-field microphones. We used 4,000 word-piece units including the blank token for tokenization. The RNN-T and HAT model investigated in our experiments share the same neural network architecture, namely, a 6-layer unidirectional LSTM as the audio encoder, and a 2-layer unidirectional LSTM as the transcription network. We used a linear layer followed by the ReLU activation function as the joint network. The number of cell

Table 1: Baseline RNN-T and HAT word error rates (WERs) without an external LM. The decoding beam size was set to 8.

Model	Temperature	Length Norm	WER
RNN-T	1.0	✓	16.6
RNN-T	1.2	✓	16.4
RNN-T	1.2	✗	16.8
HAT	1.0	✓	16.4
HAT	1.2	✓	16.3
HAT	1.2	✗	17.2

Table 2: Comparison of NLL training and MWER training of HAT with/without length normalization during decoding. The temperature in the Softmax was 1.2 during inference, and the decoding beam was 8.

Loss	Length Norm	WER
NLL	✓	16.3
MWER	✓	15.9
NLL	✗	17.2
MWER	✗	16.5

units in both audio encoder and transcription network is 1024. In terms of acoustic features, we used 80-dimensional log-mel filter banks, which are sampled at a 10 millisecond frame rate. We then spliced the consecutive frames with the context window of 8 to generate the super-frames, and then downsampled the acoustic sequences by the factor of 3.

5.1. Baseline Results without External LM

We first show the baseline results of RNN-T and HAT in terms of word error rate (WER) in Table 1. In this experiment, we did not use any external LM, and therefore both RNN-T and HAT models were evaluated following Eq. (10). We used the same configuration to train both models, e.g, the same learning rate scheduler, minibatch size, etc. We trained the model using 32 GPUs with data parallelism, and it took around 4 days for the model to converge. The results in Table 1 show that without the external LM, RNN-T and HAT can achieve very comparable WERs. The average WER of the 26,000 utterances evaluation set is around 16 - 17%. Following [18], we also evaluated the impact of the temperature, and observed that by increasing the temperature to 1.2 during inference, we can slightly improve the accuracy.

We also evaluated the models without length normalization during inference. The results are shown in the 3rd and 6th rows of Table 1. which were obtained by applying length normalization at the end of the decoding. In the streaming scenario, it is unclear if applying length normalization at each decoding step will introduce difference in terms of WER, and it may also complicate the decoding path expansion depending on the beam search algorithm been used. One solution is to first obtain transcription without length normalization, and when the end of sentence token is detected, the system may update the transcription by applying length normalization. However, drastic changes of the transcription may hurt the user experience, and it is desirable that the model is robust to length normalization. From our results, with or without length normalization during inference, the difference in terms of WERs is only around 2% relative for the standard RNN-T model, while for the HAT model, the difference is over 5%. The difference may stem from the way HAT estimates the blank probability, which may incur larger variance of the sequence-level probability compared to the standard RNN-T. However, detailed reason requires further investigation.

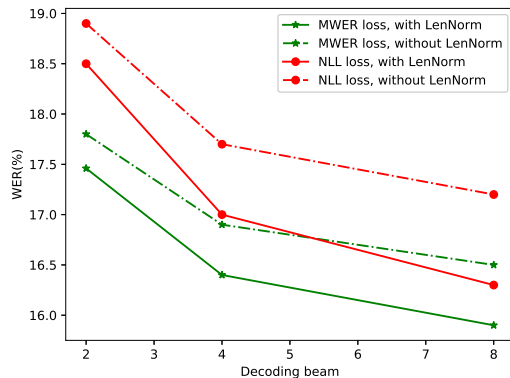


Fig. 1: Comparison of the models trained with NLL loss and MWER loss using different decoding beams. The dashed lines correspond to the results without length normalization (LenNorm) during inference, while the solid lines corresponds to those with LenNorm.

5.2. MWER Training, Length Normalization and Decoding Beam

We then study MWER training for the HAT model. We used the model from Table 1 as the seed model, and then trained the model with the MWER criterion as Eq. (13) using a small fixed learning rate as 1×10^{-6} . We used the N-best list approach to compute the posteriors as in Eq. (12), for which the N-best list were generated on-the-fly with decoding beam size of 4. For MWER training, we used 32 GPUs, and due to the memory constraint, we set the minibatch size as 1 for each GPU, so for each model update, the effective minibatch size becomes 32. We trained the model with around 6,000 hours of data until the model was fully converged according to the validation loss. We evaluated the MWER trained model with and without length normalization during decoding, and the results are given in Table 2. Without length normalization, MWER training can reduce the WER by around 4% relative compared with the result of the NLL trained model (16.5 vs. 17.2). The gain is comparable to the results reported in [18] with an RNN-T model, which did not apply length normalization during decoding either. However, the gain after length normalization during inference is slightly smaller (15.9 vs. 16.3).

We also evaluated the models with different decoding beams to study the robustness of the model against this decoding hyperparameter. The results are shown in Figure 1. We observe that the model trained with the NLL loss is more sensitive to the decoding beam, i.e., the WER increases rapidly when the decoding beam goes smaller from 8. In contrast, the MWER trained model is more robust against the decoding beam size. In addition, the gap between decoding with or without length normalization is much smaller compared with the NLL trained model. Overall, our results demonstrate that MWER training can improve the robustness of the model against the decoding hyperparameters, which is a desirable feature in real applications.

5.3. LM Fusion

As discussed before, HAT is a more modularized E2E model compared with the standard RNN-T in the sense that the acoustic and LM factors can be approximately disentangled, enabling more efficient and accurate LM fusion. To study this aspect of HAT in the context of MWER training, we performed a set of

Table 3: Results from evaluation with the external LM following Eq. (9). When both λ_1 and λ_2 are 0, it corresponds to the baseline results without the external LM. When only $\lambda_1 = 0$, the method corresponds to the shallow fusion approach.

Model	LM weights	Decoding Beam			
		4	8	12	16
NLL	(0, 0)	15.9	15.5	15.5	15.6
	(0, λ_2)	15.7	15.1	15.1	15.2
	(λ_1 , λ_2)	14.7	14.3	14.3	14.3
MWER	(0, 0)	15.6	15.4	15.3	15.4
	(0, λ_2)	15.3	14.9	14.9	15.0
	(λ_1 , λ_2)	14.7	14.2	14.1	14.2

experiments with an external LM. Our LM is based on RNNs, which has two LSTM layers, and each layer has 2048 hidden units. The model was trained on text including short message dictation and conversational data, which has around 2 billion words in total. We used the same word-piece tokenization as used for the HAT model training. Following [14], we computed the internal LM scores according to Eq. (7) and performed decoding according to Eq. (9). In order to tune the hyperparameters λ_1 and λ_2 in Eq. (9), we performed grid search of the two hyperparameters with beam size as 12 on a development dataset. In order to reduce the computational cost, we fixed the hyperparameters for evaluation with all other decoding beams. Since the external LM does not match all the domains in our evaluation set, we only choose a subset from our evaluation set based on the content for the domain adaptation experiments, which has around 2,000 utterances, and are all dictation speech.

The results are shown in Table 3. When both λ_1 and λ_2 are 0, it corresponds to the baseline system without an external LM, and we observe that the WERs vary slightly when using different decoding beams. When $\lambda_1 = 0$, the method corresponds to the widely used shallow fusion approach [7], and the results show that this approach can achieve consistent WER reduction for both NLL and MWER trained models. However, the gain is relatively small, which is only up to 2% relative. When both λ_1 and λ_2 are active, the method corresponds to the default HAT model evaluation with an external LM. The results show that this approach can achieve over 8% relative WER reduction compared with the baseline system, and over 5% relative WER reduction compared with the shallow fusion approach. Across all the three decoding conditions, the MWER trained model resulted in lower WER compared with the NLL trained model. For this particular domain, the MWER model may be further improved by using the external LM to generate the N-best list, which will be investigated in our future work.

6. Conclusion

In this work, we presented the MWER training of the recently proposed HAT model for E2E speech recognition. From our results, HAT is more sensitive to length normalization during inference compared to the standard RNN-T, and MWER training can narrow down the gap between the results with or without length normalization. In addition, we also showed that MWER training can improve the robustness of the model with different decoding beam sizes during inference. In terms of LM fusion, our results show that the HAT approach that integrates both the internal and the external LM during scoring works significantly better than the shallow fusion approach when the internal LM component is disabled, while MWER training can further improve the accuracy.

7. References

- [1] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio, “Attention-based models for speech recognition,” in *Advances in Neural Information Processing Systems*, 2015, pp. 577–585.
- [2] Liang Lu, Xingxing Zhang, Kyunghyun Cho, and Steve Renals, “A study of the recurrent neural network encoder-decoder for large vocabulary speech recognition,” in *Proc. Interspeech*, 2015.
- [3] 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. IEEE*, 2016, pp. 4960–4964.
- [4] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proc. ICML. ACM*, 2006, pp. 369–376.
- [5] Alex Graves, “Sequence transduction with recurrent neural networks,” *arXiv preprint arXiv:1211.3711*, 2012.
- [6] Yanzhang He, Tara N Sainath, Rohit Prabhavalkar, Ian McGraw, Raziq Alvarez, Ding Zhao, David Rybach, Anjuli Kannan, Yonghui Wu, Ruoming Pang, et al., “Streaming end-to-end speech recognition for mobile devices,” in *Proc. ICASSP. IEEE*, 2019, pp. 6381–6385.
- [7] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, et al., “Deep Speech: Scaling up end-to-end speech recognition,” in *arXiv preprint arXiv:1412.5567*, 2014.
- [8] 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,” *arXiv preprint arXiv:1503.03535*, 2015.
- [9] Jan Chorowski and Navdeep Jaitly, “Towards better decoding and language model integration in sequence to sequence models,” *arXiv preprint arXiv:1612.02695*, 2016.
- [10] Naoyuki Kanda, Xugang Lu, and Hisashi Kawai, “Maximum a posteriori based decoding for CTC acoustic models,” in *Proc. Interspeech*, 2016, pp. 1868–1872.
- [11] Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates, “Cold fusion: Training seq2seq models together with language models,” *arXiv preprint arXiv:1708.06426*, 2017.
- [12] Naoyuki Kanda, Xugang Lu, and Hisashi Kawai, “Maximum-a-posteriori-based decoding for end-to-end acoustic models,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 1023–1034, 2017.
- [13] Erik McDermott, Hasim Sak, and Ehsan Variani, “A density ratio approach to language model fusion in end-to-end automatic speech recognition,” in *Proc. ASRU. IEEE*, 2019, pp. 434–441.
- [14] Ehsan Variani, David Rybach, Cyril Allauzen, and Michael Riley, “Hybrid autoregressive transducer (HAT),” in *Proc. ICASSP. IEEE*, 2020, pp. 6139–6143.
- [15] Rohit Prabhavalkar, Tara N Sainath, Yonghui Wu, Patrick Nguyen, Zhifeng Chen, Chung-Cheng Chiu, and Anjuli Kannan, “Minimum word error rate training for attention-based sequence-to-sequence models,” in *Proc. ICASSP. IEEE*, 2018, pp. 4839–4843.
- [16] Chao Weng, Jia Cui, Guangsen Wang, Jun Wang, Chengzhu Yu, Dan Su, and Dong Yu, “Improving attention based sequence-to-sequence models for end-to-end english conversational speech recognition,” in *Interspeech*, 2018, pp. 761–765.
- [17] Chao Weng, Chengzhu Yu, Jia Cui, Chunlei Zhang, and Dong Yu, “Minimum Bayes risk training of RNN-Transducer for end-to-end speech recognition,” *arXiv preprint arXiv:1911.12487*, 2019.
- [18] Jinxi Guo, Gautam Tiwari, Jasha Droppo, Maarten Van Segbroeck, Che-Wei Huang, Andreas Stolcke, and Roland Maas, “Efficient minimum word error rate training of RNN-Transducer for end-to-end speech recognition,” *arXiv preprint arXiv:2007.13802*, 2020.
- [19] D. Povey and PC Woodland, “Minimum phone error and I-smoothing for improved discriminative training,” in *Proc. ICASSP. IEEE*, 1999, 2002, vol. 1.
- [20] Matthew Gibson and Thomas Hain, “Hypothesis spaces for minimum Bayes risk training in large vocabulary speech recognition,” in *Proc. Interspeech. Citeseer*, 2006.
- [21] Brian Kingsbury, Tara N Sainath, and Hagen Soltau, “Scalable minimum Bayes risk training of deep neural network acoustic models using distributed hessian-free optimization,” in *Proc. Interspeech*, 2012.
- [22] K Veselý, A Ghoshal, L Burget, and D Povey, “Sequence-discriminative training of deep neural networks,” in *Proc. Interspeech*, 2013.
- [23] Matt Shannon, “Optimizing expected word error rate via sampling for speech recognition,” in *Proc. Interspeech*, 2016.
- [24] Cal Peyser, Sepand Mavandadi, Tara N Sainath, James Apfel, Ruoming Pang, and Shankar Kumar, “Improving tail performance of a deliberation E2E ASR model using a large text corpus,” *arXiv preprint arXiv:2008.10491*, 2020.