INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Improving RNN Transducer Acoustic Models for English Conversational Speech Recognition

Xiaodong Cui, George Saon, Brian Kingsbury

IBM Research AI

{cuix,gsaon,bedk}@us.ibm.com

Abstract

In this paper we investigate several techniques for improving the performance of RNN transducer (RNNT) acoustic models for conversational speech recognition and report state-of-the-art word error rates (WERs) on the 2000-hour Switchboard dataset. We show that n-best label smoothing and length perturbation which show improved performance on the smaller 300-hour dataset are also very effective on large datasets. We further give a rigorous theoretical interpretation of the n-best label smoothing based on stochastic approximation for training RNNT under the maximum likelihood criterion. Random quantization is also introduced to improve the generalization of RNNT models. On the 2000-hour Switchboard dataset, we report a single model performance of 4.9% and 7.7% WERs on the Switchboard and CallHome portions of NIST Hub5 2000, 7.1% on NIST Hub5 2001 and 6.8% on NIST RT03, without using external LMs.
Index Terms: RNN transducer, label smoothing, length perturbation, random quantization, Switchboard dataset

1. Introduction

End-to-end (E2E) automatic speech recognition (ASR) systems based on deep neural networks (DNNs) have made great progress in recent years [1, 2, 3, 4]. Among the E2E ASR frameworks, RNN transducer (RNNT) models [5] emerge as a promising option due to their competitive performance and streaming friendly nature, which makes them attractive in real-world deployment [6, 7, 8, 9].

In this work we focus on improving the performance of RNNT acoustic models on large scale English conversational ASR. First, we investigate n-best label smoothing and length perturbation and show their effectiveness on the 2000-hour Switchboard dataset. The two techniques were previously shown to be helpful on the 300-hour Switchboard task [10]. We show in this work that they generalize well on 2000-hour Switchboard. Furthermore, we give a rigorous mathematical formulation and theoretical interpretation of n-best label smoothing from the perspective of stochastic approximation under the maximum likelihood (ML) training criterion. We demonstrate that n-best label smoothing along with iterative stochastic gradient descent (SGD) amounts to a doubly stochastic approximation optimization process. Other than the above two techniques, we also introduce random quantization in the input logmel feature space to improve the robustness and generalization of the RNNT acoustic models. With these techniques, we are able to improve upon a high-performing RNNT model and achieve state-of-the-art word error rates (WERs) on the 2000-hour Switchboard dataset without using external LMs.

The remainder of the paper is organized as follows. Section 2 describes the RNNT framework. Section 3 gives the mathematical formulation of n-best label smoothing. Sections 4 and 5 give the details of length perturbation and random quantization. Training and decoding recipes are presented in Section 6 and experimental results are reported in Section 7. Finally, Section 8 concludes the paper with a summary.

2. RNN Transducers

We follow the notation in [5]. Let $\mathcal{X}$ denote the input space and $\mathcal{Y}$ the output space. Let $\boldsymbol{x} = (x_1, x_2, \cdots, x_T)$ be the input acoustic sequence of length T where $x_t \in \mathcal{X}$ and $\boldsymbol{y} = (y_1, y_2, \cdots, y_U)$ be the output label sequence of length U where $y_u \in \mathcal{Y}$. Define the extended output space

$$\bar{\mathcal{Y}} = \mathcal{Y} \cup \{\varnothing\} \tag{1}$$

where $\varnothing$ represents a null output. The acoustic features x_t are embedded in a latent space by a convolution-augmented transformer (Conformer) [11], referred to as the transcription network $\mathcal{T}$

$$f_t = \mathcal{T}(\boldsymbol{x}_{1:T}, t). \tag{2}$$

The label tokens y_u are embedded in a latent space by a unidirectional LSTM network, referred to as the prediction network $\mathcal{P}$:

$$g_u = \mathcal{P}(\boldsymbol{y}_{[1:u-1]}, u). \tag{3}$$

Given the acoustic embedding f_t and the label token embedding g_u, the predictive output probability at (t, u) is implemented as

$$p(\cdot|t,u) = \text{softmax}[\mathbf{W}^{\text{out}}\text{tanh}(\mathbf{W}^{\text{enc}} f_t \odot \mathbf{W}^{\text{pred}} g_u + b)] \tag{4}$$

where matrices $\mathbf{W}^{\text{enc}}$ and $\mathbf{W}^{\text{pred}}$ are linear transforms that project f_t and g_u into the same joint latent space. The fusion of embeddings from the transcription and prediction networks is carried out through elementwise multiplicative integration [8, 12]. Compared to additive integration, multiplicative integration promotes high-order interaction and gives superior gating property. After the fusion of embeddings, a hyperbolic tangent nonlinearity is applied and then projected to the output space by a linear transform $\mathbf{W}^{\text{out}}$ and normalized by softmax, producing a predictive probability estimate.

3. N-best Label Smoothing

N-best label smoothing was used in [10] for the 300-hour Switchboard dataset. In this paper, we show that it generalizes well to the 2000-hour Switchboard dataset. In this section we give its rigorous mathematical formulation and its theoretical interpretation behind stochastic approximation [13].

10.21437/Interspeech.2023-2207

Conventional label smoothing is applied to classification problems under a cross-entropy loss to avoid over-confidence. In this type of problem, labels are typically provided as one-hot vectors. Suppose y is a class label for a sample x and there are K classes in total. Label smoothing smooths the label with a uniform distribution over K classes weighted by ϵ $(0 \leq \epsilon \leq 1)$ as shown in Eq.5 where $\mathbf{1}$ is an all-ones vector

$$\tilde{y} = (1-\epsilon) \cdot y + \epsilon \cdot \tfrac{1}{K} \cdot \mathbf{1} \tag{5}$$

Suppose p is the ground truth (one-hot) distribution, q is the distribution to be learned and u is the uniform distribution. Label smoothing imposes a regularization term $\sum_{i=1}^{n} H_i(u,q)$ to the original cross-entropy term $\sum_{i=1}^{n} H_i(p,q)$:

$$\mathcal{L} = (1-\epsilon) \textstyle\sum_{i=1}^{n} H_i(p,q) + \epsilon \sum_{i=1}^{n} H_i(u,q). \tag{6}$$

where n is the total number of samples.

In RNNT, the input sequence $\boldsymbol{x}$ belongs to the set $\mathcal{X}^*$ of all sequences over the input space $\mathcal{X}$ and the the output label sequence $\boldsymbol{y}$ belongs to the set $\mathcal{Y}^*$ of all sequences over the output space $\mathcal{Y}$. To differentiate the ground truth label sequence and alternative label sequence for future discussion, we use $\bar{\boldsymbol{y}}$ for the ground truth label sequence and $\hat{\boldsymbol{y}}$ for alternative label sequences. RNNTs are trained under the ML loss function [5]

$$\mathcal{L} = -\log P(\bar{\boldsymbol{y}}|\boldsymbol{x}). \tag{7}$$

A natural way of applying label smoothing is to treat the output symbols in $\mathcal{Y}$ as classes and regularize the output distribution of the softmax function with a uniform distribution as Eq.6. However, this scheme only smooths local decisions while RNNT learning is on the whole sequence globally. Our pilot experiments using this local smoothing approach give rise to degraded performance.

We then approach this sequence label smoothing problem from a sequence classification perspective. Each sequence may represent a class and all sequences in $\mathcal{Y}^*$ form a countably infinite set of classes in that label sequence space. Therefore, we minimize the following loss function

$$\mathcal{L}_s = \mathbb{E}_{f(\boldsymbol{y}_k|\boldsymbol{x})}[-\log P(\boldsymbol{y}_k|\boldsymbol{x})] \tag{8}$$

where the expectation is taken over the distribution of all label sequences $f(\boldsymbol{y}_k|\boldsymbol{x})$ in the label space $\mathcal{Y}^*$. So Eq.8 is a smoothed version of the conventional ML loss function for RNNT with respect to all label sequences.

We approximate the label sequence space $\mathcal{Y}^*$ which consists of countably infinite number of classes by a space $\Omega_{\boldsymbol{y}|\boldsymbol{x}}$ which consists of the ground truth label $\bar{\boldsymbol{y}}$ and K other representative samples:

$$\mathcal{Y}^* \approx \Omega_{\boldsymbol{y}|\boldsymbol{x}} \triangleq \{\bar{\boldsymbol{y}}\} \cup \{\hat{\boldsymbol{y}}_1, \cdots, \hat{\boldsymbol{y}}_K\} \tag{9}$$

Under this approximation, the loss function in Eq.8 is changed to

$$\mathcal{L}_s = \mathbb{E}_{q(\boldsymbol{y}_k|\boldsymbol{x})}[-\log P(\boldsymbol{y}_k|\boldsymbol{x})]. \tag{10}$$

The expectation is evaluated with respect to the distribution $q(\boldsymbol{y}_k|\boldsymbol{x})$ of $K+1$ classes in the approximated label space $\Omega_{\boldsymbol{y}|\boldsymbol{x}}$ in Eq.9.

We choose the distribution $q(\boldsymbol{y}_k|\boldsymbol{x})$ of $K+1$ classes analogously to the conventional label smoothing in Eq.5:

$$q(\boldsymbol{y}_k|\boldsymbol{x}) = \begin{cases} 1-\epsilon, & \boldsymbol{y}_k \in \{\bar{\boldsymbol{y}}\} \\ \epsilon/(K+1), & \boldsymbol{y}_k \in \{\hat{\boldsymbol{y}}_1, \cdots, \hat{\boldsymbol{y}}_K\} \end{cases} \tag{11}$$

If the distribution $q(\boldsymbol{y}_k|\boldsymbol{x})$ is a delta function on the ground truth label $\bar{\boldsymbol{y}}$

$$q(\boldsymbol{y}_k|\boldsymbol{x}) = \delta(\boldsymbol{y} - \bar{\boldsymbol{y}}) \tag{12}$$

then the loss function is equivalent to the conventional ML loss function

$$\mathcal{L}_s = \mathbb{E}_{\delta(\boldsymbol{y}-\bar{\boldsymbol{y}})}[-\log P(\boldsymbol{y}_k|\boldsymbol{x})] = -\log P(\bar{\boldsymbol{y}}|\boldsymbol{x}). \tag{13}$$

The expectation term in Eq.10 is computationally demanding. We carry out the minimization using stochastic approximation [13] in the SGD-based iterative optimization framework. In each SGD iteration we draw a random sample $\tilde{\boldsymbol{y}}$ from the distribution $q(\boldsymbol{y}_k|\boldsymbol{x})$ and use it to approximate the expectation

$$\mathbb{E}_{q(\boldsymbol{y}_k|\boldsymbol{x})}[-\log P(\boldsymbol{y}_k|\boldsymbol{x})] \approx -\log P(\tilde{\boldsymbol{y}}|\boldsymbol{x}). \tag{14}$$

Specifically, with probability $1-\epsilon$ we choose the ground truth label $\bar{\boldsymbol{y}}$ and with probability ϵ we choose uniformly at random a label $\hat{\boldsymbol{y}}_i$ from the K alternative samples $\{\hat{\boldsymbol{y}}_1, \cdots, \hat{\boldsymbol{y}}_K\}$.

Note that since SGD itself is based on stochastic approximation where the expectation of the gradient over all data samples is approximated by a stochastic gradient evaluated from a small number of random data samples in each iteration, the proposed sequence label smoothing amounts to a doubly stochastic approximation strategy.

In this work, the alternative samples are chosen from n-best hypotheses. There are a few advantages to using the n-best hypotheses (along with the ground truth) to approximate the label sequence space $\mathcal{Y}^*$ given an input sequence $\boldsymbol{x}$. First, the space $\Omega_{\boldsymbol{y}|\boldsymbol{x}}$ with n-best hypotheses and the ground truth label gives a good approximation to the expectation over the space $\mathcal{Y}^*$ in terms of likelihood as n-best hypotheses contribute high likelihood compared to some random label sequences. Second, n-best hypotheses give a reasonable representation of insertion, deletion and substitution patterns as label sequences. Third, since each n-best hypothesis does not significantly differ from the ground truth label sequence, their gradients do not significantly deviate from each other either. The gradient evaluated from a competing n-best hypothesis is equivalent to a small perturbation to the one using the ground truth label. This makes the training more stable in practice.

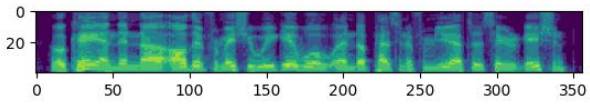
4. Length Perturbation

We implement length perturbation following the algorithm in [10] which randomly drops and inserts a number of frames in an input acoustic feature sequence. Compared to other data augmentation techniques such as SpliceOut [14] and Fill-in-frames [15], length perturbation perturbs an utterance both ways.

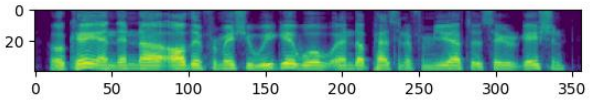
Length perturbation as a data augmentation technique perturbs the length of an utterance. Moreover, it perturbs the "memory" of a sequence model to avoid simply memorizing the history of the sequence in the training and hence improves the generalization capability of the models. In this work we confirm that length perturbation is not only helpful when the data size is modest (300-hour Switchboard dataset) but also helpful when the data size is large (2000-hour Switchboard dataset).

5. Random Quantization

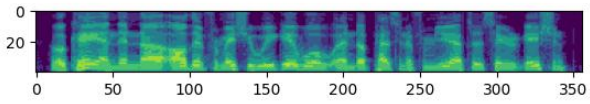
In this section we introduce random quantization to the input feature space to improve model robustness. For an input feature sequence $\boldsymbol{x} = (x_1, x_2, \cdots, x_T)$ and $x_t \in \mathbb{R}^d$, we first evaluate



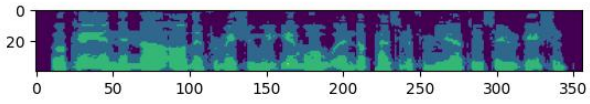
(a) *logmel of original utterance*



(b) *logmel after 16-level quantization*



(c) *logmel after 8-level quantization*



(d) *logmel after 4-level quantization*

Figure 1: *Illustration of logmel features after quantization using various quantization levels.*

the range $[x_{\text{min}}, x_{\text{max}}]$ of feature values in this $T \times d$ array and make a perturbation to the endpoints of the interval

$$x'_{\text{min}} = x_{\text{min}} \times (1 + \delta_1) \tag{15}$$

$$x'_{\text{max}} = x_{\text{max}} \times (1 + \delta_2) \tag{16}$$

where δ_1 and δ_2 are random variables in $[-0.5, 0.5]$. Next, we quantize each dimension with random levels that are uniform in $[x'_{\text{min}}, x'_{\text{max}}]$. Fig.1 illustrates the original logmel features of an input utterance and the logmel features after quantization with various quantization levels L

$$\tilde{x}_t = \mathcal{Q}_L(x_t). \tag{17}$$

where $\mathcal{Q}$ is a uniform quantizer on $[x'_{\text{min}}, x'_{\text{max}}]$ and L is a random variable. In the training, random quantization is performed with a probability which is a hyper-parameter.

Quantization results in a distorted version of the original feature sequence but still keeps the topographical correlation between the dimensions and frames. The distortion depends on the number of quantization levels. Intuitively the quantization will tolerate to certain degree the mismatch of features as the mismatched features may fall into the same quantization bin and give rise to the same output. Therefore random quantization can promote feature invariance and improve robustness of the model. Although random vector quantization has been used in the literature [16, 17], it is not applied for the same reasons nor in the same context as in this paper where the proposed random scalar quantization of input features is applied as a data augmentation technique to improve ASR robustness.

6. Training and Decoding Settings

Dataset We train RNNT acoustic models on the 2000-hour Switchboard dataset. It consists of 2000 hours of English conversational speech sampled at 8KHz. Speed and tempo perturbation is conducted offline on the 2000-hour speech, which gives rise to 4 extra replicas of the original speech data. Therefore, the total amount of training data is 10,000 hours. We measure WERs on the NIST Hub5 2000 (Switchboard and CallHome), NIST Hub5 2001 and NIST RT03 evaluation test sets. The Hub5 2000 test set consists of 3.8 hours of speech; the Hub5 2001 test set consists of 6.2 hours of speech; and the RT03 test set consists of 6.3 hours of speech. The Hub5 2001 and RT03 evaluation sets are larger and represent mismatched scenarios from training. We include them to ensure that the acoustic models are not over-tuned to the extensively tested Hub5 2000 test set.

RNNT The transcription network is a Conformer. The input to the transcription network is 40-dimensional logmel features and their first and second order derivatives. Features of every two adjacent frames are concatenated which results in 240-dimensional input vectors. There are 512 hidden units and 8 64-dimensional attention heads in each conformer block. The convolution kernel size is 31. The prediction network is a single-layer uni-directional LSTM with 1024 cells. The outputs of the transcription network and the prediction network are projected down to a 256-dimensional latent space in the joint network. The softmax layer contains 43 output units which correspond to 42 characters and the null symbol.

Training Schedule The AdamW optimizer [18] is used in training. The training data is divided into 100 chunks and the training is conducted sequentially by chunks in a randomized fashion in each epoch. In each chunk the utterances are organized in a sorted order. This is equivalent to a curriculum learning strategy that starts with short utterances to stabilize the training early on before gradually introducing more difficult, longer utterances. The batch size is 128 utterances which are distributed to 8 V100 GPUs. We study two learning schedules: OneCycleLR [19] and long warmup/long hold (LWLH). The training takes 30 epochs. In the OneCycleLR policy, the maximum learning rate is 5e-4 and it starts with a linear warmup phase from 5e-5 to 5e-4 over the first 9 epochs followed by a linear annealing phase to 0 for the next 21 epochs. In LWLH, the maximum learning rate is 1e-3 and it starts at 1e-4 in the first epoch and then linearly scales up to 1e-3 in the first 10 epochs. It holds for another 6 epochs before being annealed by $\frac{1}{\sqrt{2}}$ every epoch after the 16th epoch. The learning rate changes in each iteration in OneCycleLR but only changes across epochs in LWLH. The hyper-parameters (e.g. maximum learning rate and number of epochs) of both learning rate schedules have been optimized.

Data Augmentation and Regularization Other than the offline speed and tempo perturbation, two additional data augmentation techniques are conducted on the fly in the data loader. One is sequence noise injection [20], where a training utterance is artificially corrupted by adding a randomly selected down-scaled training utterance from the training set, and the other is SpecAug [21], where the spectrum of a training utterance is randomly masked in blocks in both the time and frequency domains. The training is also regularized by dropout with a dropout rate of 0.25 for the LSTM, 0.1 for the conformer and 0.05 for the embedding. In addition, DropConnect [22] is applied with a rate of 0.25, which randomly zeros out elements of the LSTM hidden-to-hidden transition matrices.

Investigated Techniques in This Work In n-best label smoothing, the probability of applying label smoothing is $p = 0.2$ and the number of n-best hypotheses is $K = 20$. The n-best hypotheses are generated by the baseline RNNT models. In length perturbation, the probability of applying frame skipping and insertion is $p_s = p_p = 0.6$, the fraction of utterances for frame skipping and insertion is $r_s = r_p = 0.1$, the maximum number of frames to skip is $T_s = 7$ and the maximum number of frames

to insert is $T_p = 3$. In random quantization, the probability of applying quantization is $p = 0.2$. There are three possible quantization levels $\{16, 8, 4\}$. If an utterance is chosen for quantization, one of the quantization levels is randomly selected from this set and feature values are quantized accordingly.

Decoding Inference uses alignment-length synchronous decoding [23], which only allows hypotheses with the same alignment length in the beam for the beam search. No external LMs are used.

7. Experimental Results

In Table 1, we report the WERs on n-best label smoothing, length perturbation and random quantization, respectively, and compare them with the baseline model without using three techniques under the two learning schedules, OneCycleLR and LWLH. Note that the baseline model is already a high-performing RNNT model with very competitive WERs. It can be seen that all three techniques improve the performance by themselves under the two learning schedules. Although there is no significant difference between the WERs using the two learning schedules, LWLH appears to give slightly superior performance. Therefore, in the following experiments, we will use the LWLH learning schedule to report WERs.

	Schedule	Hub5'00			Hub5'01	RT'03
		SWB	CH	Avg		
baseline	OneCycleLR	5.5	8.5	7.0	7.9	8.4
	LWLH	5.5	8.5	7.0	7.9	8.0
n-best	OneCycleLR	5.2	8.1	6.7	7.6	7.8
lab. smth.	LWLH	5.0	8.1	6.6	7.4	7.8
len.	OneCycleLR	5.1	8.1	6.6	7.3	7.3
perturb.	LWLH	5.1	8.1	6.6	7.2	7.4
random	OneCycleLR	5.4	8.5	7.0	7.7	8.1
quant.	LWLH	5.5	8.3	7.0	7.5	7.7

Table 1: *WERs on three test sets using n-best label smoothing, length perturbation and random quantization under OneCycleLR and LWLH learning schedules.*

	params	Hub5'00			Hub5'01	RT'03
		SWB	CH	Avg		
10 conf. blocks	74.1M	5.2	7.9	6.6	7.3	7.1
12 conf. blocks	**87.9M**	**4.9**	**7.7**	**6.3**	**7.1**	**6.8**
13 conf. blocks	94.8M	5.0	7.6	6.3	7.0	7.1

Table 2: *WERs on three test sets using n-best label smoothing, length perturbation and random quantization with 10, 12 and 13 conformer blocks.*

In Table 2, we report the WERs on the three test sets when all three investigated techniques are applied sequentially. The training is extended from 30 epochs to 33 epochs under LWLH. We apply n-best label smoothing in the first 20 epochs, length perturbation in the next 10 epochs and random quantization in the 3 final epochs. We also evaluate the performance by varying the model size using 10, 12 and 13 conformer blocks, respectively. The best result is obtained when using 12 conformer blocks. Under this condition, the WERs are 4.9% and 7.7% on the Switchboard and CallHome test subsets of the NIST Hub5 2000 evaluation, 7.1% on NIST Hub5 2001 evaluation and 6.8% NIST RT03 evaluation, respectively.

To the best of our knowledge, this is the state of the art for a single model on the 2000-hour Switchboard dataset without using any external LMs. It is also better than many of the reported results using external LMs.

An ablation study on the three investigated techniques is carried out and results are shown in Table 3 which demonstrates the impact of each technique to the overall improved performance. N-best label smoothing and length perturbation contribute on all three test sets in both matched (e.g. Switchboard in Hub5'00) and mismatched (e.g. CallHome in Hub5'00 and RT03) conditions. Random quantization mainly contributes to the mismatched conditions and therefore makes the model more robust and improves generalization. This confirms the intuition when designing the random quantization technique. This ablation study is also in line with the results in Table 1 with individual contributions from each technique.

	Hub5'00			Hub5'01	RT'03
	SWB	CH	Avg		
baseline	5.5	8.5	7.0	7.9	8.0
three tech. combined	4.9	7.7	6.3	7.1	6.8
w/o lab. smth.	5.1	8.3	6.7	7.2	7.1
w/o len. pertb.	5.0	8.1	6.6	7.3	7.7
w/o rand. quant.	4.9	7.7	6.3	7.2	7.0

Table 3: *Ablation study on the n-best label smoothing, length perturbation and random quantization.*

We compare our results with best single models reported in literature on the 2000-hour Switchboard dataset without using external LMs in Table 4.

Model	Hub5'00			Hub5'01	RT'03
	SWB	CH	Avg		
AED ([24])	4.8	8.0	6.4	7.6	7.8
AED ([25])	4.8	8.0	6.4	7.3	7.5
DSSformer ([26])	5.2	8.2	6.7	7.2	7.5
RNNT (this work)	**4.9**	**7.7**	**6.3**	**7.1**	**6.8**

Table 4: *WERs of best single models reported in literature on 2000-hour Switchboard dataset without using external LMs. AED stands for attention-based encoder-decoder architecture and DSSformer stands for diagonal state space augmented transformer architecture*

8. Conclusion

In this paper we investigate n-best label smoothing, length perturbation and random quantization for improving the performance of RNNT acoustic models for conversational speech recognition on the 2000-hour Switchboard dataset.

We show that n-best label smoothing and length perturbation generalize well to large training datasets and yield good improvement on the 2000-hour Switchboard dataset. We give a rigorous mathematical formulation of the n-best label smoothing and show that it amounts to a doubly stochastic approximation strategy in the SGD framework optimizing a smoothed version of the conventional ML loss function for RNNT training. In addition, random quantization helps to improve the model robustness and generalization to mismatched conditions. With the three techniques combined, we report 4.9% and 7.7% WERs on the Switchboard and CallHome of the NIST Hub5 2000, 7.1% on NIST Hub5 2001 and 6.8% NIST RT03 evaluation test sets, respectively, which, to the best of our knowledge, is the state of the art of a single system without using external LMs on the 2000-hour Switchboard dataset.

9. References

[1] A. Graves, S. Fernandez, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2006, pp. 369–376.

[2] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: a neural network for large vocabulary conversational speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4960–4964.

[3] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4945–4949.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhit, "Attention is all you need," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 6000–6010.

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

[6] A. Graves and A.-r. Mohamed and G. Hinton, "Speech recognition with deep recurrent neural networks," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 6645–6649.

[7] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, Q. Liang, D. Bhatia, S. Yuan, B. Li, G. Pundak, K. C. Sim, T. Bagby, S. y. Chang, K. Rao, and A. Gruenstein, "Streaming end-to-end speech recognition for mobile devices," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6381–6385.

[8] G. Saon, Z. Tueske, D. Bolanos, and B. Kingsbury, "Advancing RNN transducer technology for speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[9] J. Li, R. Zhao, H. Hu, and Y. Gong, "Improving RNN transducer modeling for end-to-end speech recognition," in *Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019.

[10] X. Cui, G. Saon, T. Nagano, M. Suzuki, T. Fukuda, B. Kingsbury, and G. Kurata, "Improving generalization of deep neural network acoustic models with length perturbation and n-best based label smoothing," in *Interspeech*, 2022, pp. 2638–2642.

[11] 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 *Interspeech*, 2020, pp. 5036–5040.

[12] Y. Wu, S. Zhang, Y. Zhang, Y. Bengio, and R. Salakhutdinov, "On multiplicative integration with recurrent neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2016, pp. 2864–2872.

[13] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, 2003.

[14] A. Jain, P. R. Samala, D. Mittal, and P. Jyothi, "SpliceOut: a simple and efficient audio augmentation method," *arXiv preprint arXiv:2110.00046*, 2021.

[15] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "MaskCycleGAN-VC: learning non-parallel voice conversion with filling in frames," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5919–5923.

[16] C.-C. Chiu, J. Qin, Y. Zhang, J. Yu, and Y. Wu, "Self-supervised learning with random-projection quantizer for speech recognition," in *International Conference on Machine Learning (ICML)*, 2022, pp. 3915–3924.

[17] H. Wu, C. Lei, X. Sun, P.-S. Wang, Q. Chen, K.-T. Cheng, S. Lin, and Z. Wu, "Randomized quantization for data agnostic representation learning," *arXiv:2212.08663*, 2022.

[18] D. P. Kingma and J. L. Ba, "ADAM: a method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.

[19] L. N. Smith and N. Topin, "Super-convergence: very fast training of neural networks using large learning rates," in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, 2019.

[20] G. Saon, Z. Tuske, K. Audhkhasi, and B. Kingsbury, "Sequence noise injected training for end-to-end speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6261–6265.

[21] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Interspeech*, 2019, pp. 2613–2617.

[22] L. Wan, M. Zeiler, S. Zhang, Y. LeCun, and R. Fergus, "Regularization of neural networks using DropConnect," in *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2013, pp. 1058–1066.

[23] G. Saon, Z. Tuske, and K. Audhkhasi, "Alignment-length synchronous decoding for RNN transducer," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7804–7808.

[24] Z. Tuske, G. Saon, K. Audhkhasi, and B. Kingsbury, "Single headed attention based sequence-to-sequence model for state-of-the-art results on switchboard," in *Interspeech*, 2020, pp. 551–555.

[25] Z. Tuske, G. Saon, and B. Kingsbury, "On the limit of english conversational speech recognition," in *Interspeech*, 2021, pp. 2062–2066.

[26] G. Saon, A. Gupta, and X. Cui, "Diagonal state space augmented transformers for speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.