Interspeech 2022
18-22 September 2022, Incheon, Korea

Combining Simple but Novel Data Augmentation Methods for Improving Low-Resource ASR

Ronit Damania[1], *Christopher Homan*[1], *Emily Prud'hommeaux*[2]

[1]Rochester Institute of Technology, Rochester, NY USA
[2]Boston College, Chestnut Hill, MA USA

rjd2551@rit.edu,cmhvcs@rit.edu,prudhome@bc.edu

Abstract

In this paper, we propose several novel data augmentation methods for improving the performance of automatic speech recognition (ASR) in low-resource settings. Using a 100-hour subset of English LibriSpeech to simulate a low-resource setting, we compare the well-known SpecAugment approach to these new methods, along with several other competitive baselines. We then apply the most promising combinations of models and augmentation methods to three genuinely under-resourced languages using the 40-hour Gujarati, Tamil, Telugu datasets from the 2021 Interspeech Low Resource Automatic Speech Recognition Challenge for Indian Languages. Our data augmentation approaches, coupled with state-of-the-art acoustic model architectures and language models, yield reductions in word error rate over SpecAugment and other competitive baselines for the LibriSpeech-100 dataset, showing a particular advantage over prior models for the "other", more challenging, dev and test sets. Extending this work to the low-resource Indian languages, we see large improvements over the baseline models and results comparable to large multilingual models.

Index Terms: automatic speech recognition, data augmentation, low-resource languages

1. Introduction

Automatic speech recognition (ASR) has been a fundamental problem in artificial intelligence for decades. Recently, the performance of ASR on high-resource languages has benefited enormously from neural models [1, 2, 3, 4], enabling the integration of ASR for into software and devices used every day by the people who speak these languages. However, the vast majority of the world's languages, even those spoken by tens of millions of people, do not have the quantity of transcribed speech necessary to build ASR models of this caliber, making speech-based applications out of reach for billions of people.

A common approach for learning in under-resourced settings is *data augmentation*, in which new training examples are added to the existing training corpus. Augmentation via duplication and modification of the existing acoustic training data has been found to be particularly useful for ASR. Early work in this type of augmentation often focused on changes in speaking rate or pitch in order to accommodate variations in speaking style and voice features [5, 6, 7] or superimposing noises (e.g., background conversation, street sounds) on the acoustic training samples to simulate realistic recording environments [8]. An alternative to modifying raw audio, SpecAugment [9] time-warps and masks regions in the spectral representation of the speech signal, which has been shown to make models more robust to spectral variation and variability in recording quality.

In this paper we introduce new methods of data augmentation that alter, rather than mask, random regions in the spectrum, as well as a method of augmentation that acts on raw audio via concatenation of existing training samples. We first demonstrate the utility of these methods in a simulated low-resource setting using a 100-hour subset of English LibriSpeech [10]. We find that some of our augmentation methods outperform SpecAugment, particularly on the more challenging dev and test sets (i.e., those labeled "other" as opposed to "clean"). Combined with a language model, this approach yields word error rates (WERs) on both the dev and test sets lower than several state-of-the-art architectures for this dataset. We then test the most promising of these methods on three genuinely low-resource datasets – 40 hours each of Gujarati, Tamil, Telugu – using monolingual conformer acoustic models, yielding reductions in WER for all three languages over the Interspeech challenge baselines by 5.0-12.0 (a reduction of 17-33%) and in some cases outperforming larger, multilingual models.

Table 1: *Data splits in the LibriSpeech 100-hour subset [10].*

subset	hours	min/spkr	female	male	total
dev-clean	5.4	8	20	20	40
test-clean	5.4	8	20	20	40
dev-other	5.3	10	16	17	33
test-other	5.1	10	17	16	33
train-clean-100	100.6	25	125	126	251

2. Data

2.1. Simulated low-resource setting: English

In order to simulate a low-resource setting while still having a significant body of prior work from which to gather competitive baselines, we used a 100-hour subset of the English LibriSpeech corpus [10]. LibriSpeech was collected from a corpus of audiobooks that are part of the LibriVox project[1]. The creators of LibriSpeech separated the dev and test data into two categories, *clean* and *other*, where the audio in the latter category is drawn from speakers whose recordings yielded higher WER in the original baseline system, suggesting that these recordings are more challenging. The 100-hour subset of LibriSpeech provides a reasonable surrogate for truly under-resources languages. Table 1 provides information about the sub-corpora of LibriSpeech used in our work.

2.2. Truly low-resource settings: Indian languages

For our truly low-resource settings, we consider three small, monolingual datasets for Gujarati, Tamil, Telugu from the Mul-

[1]https://librivox.org/

10.21437/Interspeech.2022-10835

Table 2: *Data from the Interspeech 2021 multilingual ASR challenge for low-resource Indian languages [11].*

Language	Split	Size(hrs)	Uniq sent	Spkrs
Gujarati	Train	40	20257	94
	Test	5	3069	15
	Blind	5.26	3419	18
Tamil	Train	40	30329	448
	Test	5	3060	118
	Blind	4.41	2584	118
Telugu	Train	40	34176	464
	Test	5	2997	129
	Blind	4.39	2506	129

tilingual and Code-Switching (MUCS) ASR Challenges at Interspeech 2021 [11]. The audio is a combination of conversational speech and read speech. As shown in Table 2, there are three subsets in the data splits for each language: *train*, *test*, and *blind*, which we use for training, validation, and testing, respectively. Each training set contains 40 hours of training data. The *train* and *test* sets are sampled at 16kHz, while the *blind* set is sampled at 8kHz. In addition, 34.1%, 23.8% and 29.0% of the blind data chosen at random from Gujarati, Tamil and Telugu is modified with speed perturbation or noise.

3. Augmentation methods

We introduce three novel data augmentation methods: (1) spectral augmentation by *multiplying* the regions to augment with random values (AugMult); (2) spectral augmentation by *replacing* the regions to augment with random values (AugRepl); and (3) input concatenation (IC).

Recall that in SpecAugment [9], f consecutive mel frequency channels $[f_0, f_0 + f)$ are masked, where f is selected at random from 0 to the selected frequency masking parameter, F, and f_0 is chosen from [0, v - f), where v is the number of mel frequency channels. The masking value of zero is constant for all regions to augment in the input, and the values of the regions after masking do not correlate with the values of those regions before masking. In our novel AugMult augmentation method, rather than masking by replacing with 0, we multiply the regions to augment with a value m, chosen uniformly at random from the range (a, b) for each utterance, where a and b are hyperparameters. This maintains the correlation between the augmented regions before and after masking.

In AugRepl, we instead choose a masking value r uniformly at random from values ranging from the minimum to the maximum value in each batch. There are two ways in which we can do this. The first is to have the same r for all the utterances in the batch (AugReplB), and the second is to select r randomly for each utterance (AugReplU). Figure 1 visualizes the difference between these augmentation methods and SpecAugment.

For input concatenation (IC) we concatenate the raw audio of two training samples and their corresponding transcripts. For a given batch, we create an array of random integers ($randomInt$) whose length equals the length of the batch, and we pick elements of the array from the range $[0, length(batch) - 1]$ with replacement. Given a batch array ($batch$), for index i, we concatenate $batch[i]$ with $batch[randomInt[i]]$. In case of a non-integer value, the ceiling value is taken. Input concatenation can help the acoustic model generalize because the acoustic model has to adapt to different speakers, who could have variability in accent, age, gender, and other vocal qualities. In order to avoid diverging too much from the validation set, we concatenate only a certain percentage of the inputs in any batch.

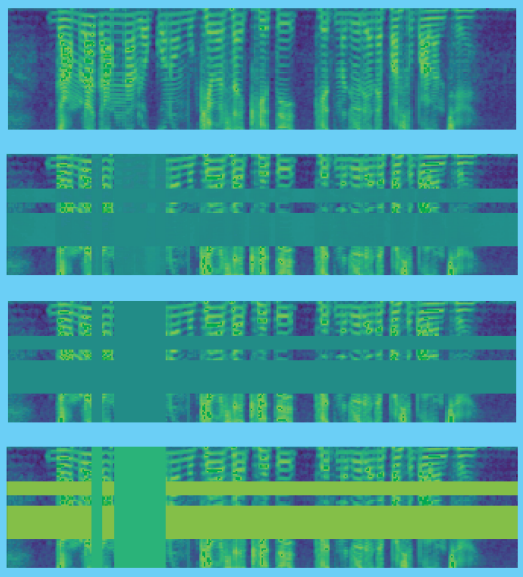

Figure 1: *A log mel spectrogram with different augmentations. From top to bottom: (1) no augmentation; (2) AugMult with multiplier selected uniformly randomly from* $(-0.5, 0.5)$ *yielding here* -0.2048 *for frequency and* -0.3085 *for time; (3) SpecAugment [9] where the masked region's value is* 0*; and (4) AugRepl where the value is selected uniformly randomly from (-4.886, 6.209) (the min and max value of the audio) yielding* 5.2457 *for frequency and* 1.6975 *for time.*

We compare our novel augmentation methods with two baseline augmentation methods: SpecAugment [9] with the parameters $F = 30$, $T = 40$, $m_T = m_F = 2$, and speed perturbation (SP) [5] with perturbation factors of 0.9 and 1.1 to increase the training data by threefold.

4. ASR Architecture

4.1. Data Preparation

Our augmentation experiments are built on the following pipeline. We downsample all of the Indian language audio data to 8kHz because the blind sets are at 8kHz. (We do not downsample the English LibriSpeech data since all of the training, dev, and testing data is provided at 16kHz.). We then use a fast Fourier transform (FFT) to construct mel scale spectrograms [12, 13], with hop length of 256 for Librispeech and 128 for the Indian languages. For the transcripts, we use SentencePiece byte-pair-encodings [14] with a vocabulary size of 5000 for English and 200 each for Gujarati, Tamil and Telugu.

4.2. Our Conformer Model

We use a sequence-to-sequence conformer-based acoustic model using ESPnet2 [15, 16] The encoder has 12 conformer blocks with 8 attention heads and 512 encoder dimensions. The decoder has 6 transformer blocks with 2048 linear units, 8 attention heads, and dropout of 0.1. The model performs mutlitask learning [17] with a connectionist temporal classification

(CTC) weight of 0.3 and attention weight of 0.7. For learning, we use an Adam optimizer [18] with an initial learning rate of 0.0025, $\beta_1 = 0.9$, $\beta_2 = 0.99$ and 40000 warmup steps. We train each model for 150 epochs.

The language model for each Indian language was trained only on the transcripts of the audio files for that language, using a 4-layer long short-term memory (LSTM) model with 2048 nodes per layer. For the English data, we used an existing, transformer-based language model pre-trained on the LibriSpeech training set plus various external text corpora, distributed with the pre-trained conformer acoustic model.

We perform inference in two ways. One is with the best performing model (BEST) on the validation set. The other is with an AVGn (e.g., AVG10) model [19, 20], whose parameters are formed by taking the average of the parameters of the top-n performing models on the validation set at the end of each epoch, where n is a hyperparameter. For example, if $n = 3$ in a ten epoch training, and if during the training the best results in the validation set occur at the ends of epochs 3, 7, and 8, then the AVG3 model takes as its parameters the average of the model parameters at epochs 3, 7, and 8.

4.3. Competitive Prior Baselines

Recall that we use a small subset of the LibriSpeech English train-clean-100 corpus as a **simulated low-resource setting**. This dataset is widely used in the research community, enabling us to compare our system to four competitive baselines reported in prior work. We consider four such baselines: (1) Kaldi's [21] best-performing hybrid deep neural network/hidden Markov model (DNN/HMM) framework [22] with a 4-gram language model; (2) RWTH Aachen University's DNN/HMM hybrid model [23] that uses a bi-directional LSTM architecture [24, 25] of 6 layers with 1000 units for backward and forward directions each; (3) a direct-to-word CTC sequence model [26], which uses a transformer-based acoustic model with a CTC objective and 4-gram language model; additionally for augmentation, it uses SpecAugment [9]; (4) an end-to-end model [27] that uses a transformer-based acoustic model and a recurrent neural network with four LSTM layers with 2048 units for each language model; the augmentation used in this particular model involves speed perturbation with factors of 0.9 and 1.1, along with SpecAugment [9].

To demonstrate the utility of our data augmentation methods in a **truly low-resource setting**, for Gujarati, Tamil and Telugu, the baseline ASR model [11] is a sequence-trained time-delay neural network (TDNN) architecture optimized using the lattice-free maximum mutual information (LF-MMI) objective function [28]. The architecture consists of an acoustic model with 6 TDNN blocks, each of dimension 512, and a 3-gram language model. We also report results from the MUCS21 leaderboard[2], which uses multilingual acoustic training.

5. Results

5.1. Simulated low-resource setting

Table 3 shows the WER for a variety of augmentation methods applied with our conformer framework for the simulated low-resource LibriSpeech train-clean-100 English corpus. Averaging alone yields substantial reductions in WER. IC alone does not appear to improve results. However, our novel method of spectral data augmentation, AugReplB, in which the masking value is the same for all utterances in a given batch, yields WERs lower than those achieved with SpecAugment in dev-other, test-other, and test-clean. The reductions are larger in the "other" condition, suggesting that this approach renders the model more able to generalize to challenging speakers.

Combining AugReplB augmentation averaged over the 20 best models with IC and SP, we further improve performance. Table 4 shows that this combination achieves lower WER than all four state-of-the-art baselines described in Section 4. These reductions in WER are particular noticeable for the "other" data, indicating that the AugRepl augmentation methods produce models that can handle more challenging input.

Table 3: *WER of the described conformer architecture trained on the simulated low-resource setting, LibriSpeech train-clean-100, augmented in various ways, including both our novel approaches and existing approaches (SpecAug, speed perturbation (SP)). For AugMult we use random scaling factors in the range* $(-0.1, 0.1)$. *For the input concatenation (IC) percentage we use* 0.50. *Our AugReplB augmentation method outperforms SpecAug when combined both with IC and SP. Note: all models here use the same language model during decoding.*

Model	dev clean	dev other	test clean	test other
BEST	13.4	34.9	13.8	35.9
AVG10	9.5	28.7	10.0	29.4
+ SpecAug	7.4	20.0	7.9	20.5
+ AugMult	8.6	23.5	8.8	24.2
+ AugReplB	7.5	19.7	7.8	20.1
+ AugReplU	7.5	20.0	7.8	20.3
+ IC	9.6	28.7	10.1	29.0
+ SpecAug + IC	6.4	19.2	7.4	20.1
+ AugReplB + IC	6.5	18.4	6.9	19.0
+ AugReplU + IC	7.6	20.0	7.8	20.3
+ SpecAug + IC +SP	6.0	16.4	6.7	16.7
+ AugReplB + IC + SP	6.0	16.6	6.6	16.8
+ AugReplU + IC + SP	**5.8**	**16.0**	**6.3**	**16.0**

Table 4: *WER of prior competitive baselines LibriSpeech train-clean-100 alongside our conformer architecture trained with our best combination of inference and augmentation methods. For the input concatenation (IC) percentage we use* 0.50. *Our best augmentation combination from Table 3 outperforms all four prior models in the "other" more challenging conditions. Using AVG20 yields additional improvements, yielding superior results in all four dev and test sets.*

Model	dev clean	dev other	test clean	test other
Kaldi [21]	5.9	20.4	6.6	22.5
word-level CTC [26]	6.3	19.1	6.8	19.4
RWTH [23]	5.0	19.5	5.8	18.6
end2end [27]	5.8	16.6	7.0	17.0
AVG10+AugReplU+IC+SP	5.8	16.0	6.3	16.0
AVG20+AugReplB+IC+SP	**4.6**	**13.2**	**5.1**	**13.1**

5.2. Truly low-resource settings

Table 5 shows that, for the Indian languages, any kind of augmentation in combination with AVG10 results in WER decreases over the monolingual baseline, which uses an LM but

[2]https://navana-tech.github.io/MUCS2021/leaderboard.html

Table 5: *WER for the three Indian languages, Gujarati, Tamil and Telugu using our conformer architecture with various combinations of augmentation (SpecAug, AugRepl, IC), inference (BEST, AVG10, AVG20), and language model (no LM and +LM), all with SP. For comparison, we show monolingual baseline results [11] and the best multilingual leaderboard result for each language. For the IC percentage, we use* 0.50*. When using AVGn inference, our novel augmentation methods alone and in combination, outperform the baseline, with our best models yielding results competitive with the multilingual leaderboard.*

Inference	Augmentation	Gujarati	Tamil	Telugu	Average
	Baseline (monolingual + LM)	26.0	35.8	29.4	30.4
BEST (no LM)	SpecAug	30.3	29.0	32.0	30.4
	AugReplB	29.1	29.0	31.6	29.9
	AugReplU	29.9	28.9	32.4	30.4
AVG10 (no LM)	SpecAug	25.3	24.9	26.5	25.6
	AugReplB	25.0	24.7	26.9	25.5
	AugReplU	25.1	24.9	26.7	25.6
AVG10 (no LM)	SpecAug + IC	26.7	25.8	26.5	26.3
	AugReplB + IC	26.6	26.0	26.9	26.5
	AugReplU + IC	26.3	25.9	27.7	26.6
AVG10 + LM	SpecAug	21.9	23.9	24.3	23.4
	AugReplB	22.3	23.9	24.2	23.5
	AugReplU	22.0	24.2	24.2	23.5
AVG10 + LM	SpecAug + IC	20.9	23.8	24.1	22.9
	AugReplB + IC	21.2	24.1	23.9	23.0
	AugReplU + IC	20.9	23.7	24.3	23.0
AVG20 + LM	SpecAug + IC	20.8	**23.6**	**23.8**	**22.7**
	AugReplB + IC	**20.7**	24.1	24.1	23.0
	AugReplU + IC	20.9	23.9	23.9	22.9
	Leaderboard (multilingual + LM)	20.1	18.8	17.0	18.6

not augmentation. The degree to which augmentation improves output appears to be language dependent, with Tamil showing relatively large improvements under all training conditions. Comparing our augmentation methods with SpecAug, AugRepl yields WERs lower than or identical to those produced with SpecAug in all three languages when using BEST inference with no LM. When averaging without an LM, AugRepl yields WERs comparable to SpecAug in two of the three languages. These results suggest that our novel approaches to spectral augmentation that involve masking with non-zero values provide some benefit over SpecAug when not using an LM. Interestingly, combining IC with all three spectral augmentation methods without an LM slightly degrades performance.

Introducing an LM results in lower WER in all cases. Moreover, combining spectral methods of augmentation with IC is noticeably effective when decoding using an LM for Gujarati, where WER drops by a full point in most cases. In Gujarati for all three augmentation methods, utterance-level paired t-tests between AVG10+LM with and without IC were significant ($p < 10^{-6}$). The decrease in WER for the AugReplU case in Tamil was also significant ($p < 0.05$). The final inference model, which combines an LM with AGV20, yields the lowest WER for all three data augmentation methods and all three languages, and, in the case of Gujarati, a WER comparable to the multilingual leaderboard.

6. Conclusions

Although novel neural architectures are responsible for many recent improvements in low-resource ASR, we demonstrate the utility of data augmentation in the acoustic training pipeline. Our novel augmentation methods, which rely both on spectral distortion (AugMult, AugReplB, AugReplU) and combining the raw audio of multiple speakers into a single input example (IC), result in reductions in WER over SpecAugment, the most commonly-used spectral augmentation method. This holds for both the simulated low-resource scenario of the LibriSpeech 100-hour subset and the truly low-resource datasets for Gujarati, Telugu, and Tamil.

Our future work will focus on applying these data augmentation methods in other state-of-the-art low-resource ASR architectures and exploring the extent to which these methods can improve results for extremely small datasets for endangered languages, which typically have fewer than 10 hours of transcribed audio data. We also plan to explore alternative approaches for selecting masking values within the AugRepl method, which can act as random noise in the input. In addition, since the AVGn model generally showed large improvements over BEST, we will carry out additional analyses using weighted averages or applying softmax using the validation set.

7. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1761562. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

8. References

[1] A. Y. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Y. Ng, "Deep Speech: Scaling up end-to-end speech recognition," *CoRR*, vol. abs/1412.5567, 2014. [Online]. Available: http://arxiv.org/abs/1412.5567

[2] D. Amodei and et al., "Deep Speech 2: End-to-end speech recognition in English and Mandarin," *CoRR*, vol. abs/1512.02595, 2015. [Online]. Available: http://arxiv.org/abs/1512.02595

[3] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *ICASSP*. IEEE, 2016, pp. 4960–4964.

[4] 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.

[5] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Interspeech*, 2015.

[6] M. Geng, X. Xie, S. Liu, J. Yu, S. Hu, X. Liu, and H. Meng, "Investigation of data augmentation techniques for disordered speech recognition," in *Interspeech*, 2020, pp. 696–700.

[7] Y. Zhou, C. Xiong, and R. Socher, "Improved regularization techniques for end-to-end speech recognition," *CoRR*, vol. abs/1712.07108, 2017. [Online]. Available: http://arxiv.org/abs/1712.07108

[8] A. Pervaiz, F. Hussain, H. Israr, M. A. Tahir, F. R. Raja, N. K. Baloch, F. Ishmanov, and Y. B. Zikria, "Incorporating noise robustness in speech command recognition by noise augmentation of training data," *Sensors*, vol. 20, no. 8, p. 2326, 2020.

[9] 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," *Interspeech*, 2019. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2680

[10] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *ICASSP*, 2015, pp. 5206–5210.

[11] A. D. et al., "Multilingual and code-switching ASR challenges for low resource Indian languages," *CoRR*, vol. abs/2104.00235, 2021. [Online]. Available: https://arxiv.org/abs/2104.00235

[12] S. S. Stevens, J. Volkmann, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," *Journal of the ASA*, vol. 8, no. 3, pp. 185–190, 1937. [Online]. Available: https://doi.org/10.1121/1.1915893

[13] S. S. Stevens and J. Volkmann, "The relation of pitch to frequency: A revised scale," *The American Journal of Psychology*, vol. 53, no. 3, pp. 329–353, 1940. [Online]. Available: http://www.jstor.org/stable/1417526

[14] T. Kudo and J. Richardson, "Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," *CoRR*, vol. abs/1808.06226, 2018. [Online]. Available: http://arxiv.org/abs/1808.06226

[15] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-end speech processing toolkit," in *Proceedings of Interspeech*, 2018, pp. 2207–2211. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1456

[16] H. Inaguma, S. Kiyono, K. Duh, S. Karita, N. Yalta, T. Hayashi, and S. Watanabe, "ESPnet-ST: All-in-one speech translation toolkit," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Online: Association for Computational Linguistics, Jul. 2020, pp. 302–311. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-demos.34

[17] S. Kim, T. Hori, and S. Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," *CoRR*, vol. abs/1609.06773, 2016. [Online]. Available: http://arxiv.org/abs/1609.06773

[18] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of adam and beyond," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=ryQu7f-RZ

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017, pp. 5998–6008.

[20] S. K. et al., "A comparative study on transformer vs RNN in speech applications," *CoRR*, vol. abs/1909.06317, 2019. [Online]. Available: http://arxiv.org/abs/1909.06317

[21] D. Povey and et al., "The Kaldi Speech Recognition Toolkit," in *ASRU*, 2011.

[22] X. Zhang, J. Trmal, D. Povey, and S. Khudanpur, "Improving deep neural network acoustic models using generalized maxout networks," in *ICASSP*, 2014, pp. 215–219.

[23] C. Lüscher, E. Beck, K. Irie, M. Kitza, W. Michel, A. Zeyer, R. Schlüter, and H. Ney, "RWTH ASR systems for LibriSpeech: Hybrid vs attention," in *Interspeech*, 2019.

[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, p. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[25] A. Graves, N. Jaitly, and A. rahman Mohamed, "Hybrid speech recognition with deep bidirectional lstm," *ASRU*, pp. 273–278, 2013.

[26] R. Collobert, A. Hannun, and G. Synnaeve, "Word-level speech recognition with a letter to word encoder," in *International Conference on Machine Learning*. PMLR, 2020, pp. 2100–2110.

[27] A. Laptev, R. Korostik, A. Svischev, A. Andrusenko, I. Medennikov, and S. Rybin, "You do not need more data: Improving end-to-end speech recognition by text-to-speech data augmentation," *CISP-BMEI*, Oct 2020. [Online]. Available: http://dx.doi.org/10.1109/CISP-BMEI51763.2020.9263564

[28] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Interspeech*, 2016.