On the limit of English conversational speech recognition

Zoltán Tüske, George Saon, Brian Kingsbury

IBM Research AI, Yorktown Heights, USA

`zoltan.tuske@ibm.com`

Abstract

In our previous work we demonstrated that a single headed attention encoder-decoder model is able to reach state-of-the-art results in conversational speech recognition. In this paper, we further improve the results for both Switchboard 300 and 2000. Through use of an improved optimizer, speaker vector embeddings, and alternative speech representations we reduce the recognition errors of our LSTM system on Switchboard-300 by 4% relative. Compensation of the decoder model with the probability ratio approach allows more efficient integration of an external language model, and we report 5.9% and 11.5% WER on the SWB and CHM parts of Hub5'00 with very simple LSTM models. Our study also considers the recently proposed conformer, and more advanced self-attention based language models. Overall, the conformer shows similar performance to the LSTM; nevertheless, their combination and decoding with an improved LM reaches a new record on Switchboard-300, 5.0% and 10.0% WER on SWB and CHM. Our findings are also confirmed on Switchboard-2000, and a new state of the art is reported, practically reaching the limit of the benchmark.

Index Terms: encoder-decoder, attention, speech recognition, AdamW, Switchboard, i-vector

1. Introduction

End-to-end (E2E) automatic speech recognition (ASR) models directly map an acoustic feature sequence to a word sequence. Due to their universal capability to handle even non-monotonic alignment, attention models are widely used in many machine learning problems, e.g. translation [1]. The extreme flexibility of these models also allows efficient transfer learning to a non-monotonic spoken language understanding problem from a monotonic speech recognition task [2]. In [3], we have shown that attention models are able to reach state-of-the-art recognition performance even if only 300 hours of speech are available, using a long list of various regularization techniques. This work extends the best recipe proposed in that study to further advance recognition quality. The overview of the investigated methods and related works is presented in Section 2. The details about the updated experimental settings and results are presented in Section 3 and 4. In Section 5, the paper closes with conclusions.

2. Methods

Extending our previous study in [3], by default we use various dropout methods, data augmentation and regularization approaches, curriculum learning and scheduled sampling techniques [4–16]. On top of those, the following methods and modeling approaches are investigated in this paper.

AdamW optimizer: The Adam optimizer combines adaptive stochastic gradient descent (SGD) optimization and classic momentum [17]. Recently, a modification has been proposed, in which weight decay is decoupled from the adaptive gradient calculation [18].

I-vector based speaker adaptation estimates an identity vector from incoming acoustic features. This vector transforms the parameters of a universal background model (UBM) in a shared sub-space to match the speaker dependent distribution. I-vectors have been successfully used for neural network based ASR, e.g. in [19–21]. To combine with speed and tempo perturbation, the artificially created recordings of a speaker are treated as coming from a previously unseen speaker.

Alternative log-Mel representations: in the standard feature extraction pipeline, amplitude spectra are usually extracted every 10ms, applying a 25-32ms time window. Studies on acoustic modeling of raw waveforms, however, suggest that higher temporal resolution might be beneficial [22, 23]. Therefore, we investigate a reparametrized log-Mel extraction pipeline in which amplitude spectra are extracted every 2.5ms using a 10ms analysis window. Further, 7th root compression is applied instead of logarithmic, as in [24, 25]. The alternative feature extraction pipeline is also extended with small-energy masking (SEM) perturbation [26]. We also propose to apply high energy clipping (HEC) distortion on the Mel-spectrum, on top of SEM. Denoting the amplitude spectrum as $e_{t,c}$, where t corresponds to the frame index and c the filterbank channel, the following function is applied:

$$e_{t,c}^{(\text{HEC})} = \min(\eta_c, e_{t,c}) \qquad (1)$$

where η_c denotes a randomly chosen, channel dependent threshold above which the amplitude values get clipped, e.g. picking the 80th percentile per channel.

Convolutional self-attention modules: Recurrent hidden layers, e.g. long short-term memory (LSTM), are often chosen to model temporal dependencies in the speech signal [27]. Self-attention based transformer models have been explored as encoder and also as decoder for ASR in [28–31]. Recently, the combination of convolutional neural network and transformer has been proposed and achieved state-of-the-art recognition results on read speech [32]. Our study also investigates the effect of replacing the LSTM based encoder, as well as the replacement of our single-head LSTM decoder with a conformer.

Self-attention based language models can easily capture long term dependencies and can outperform LSTM models [33]. As shown by [34, 35], significant gain has been observed on the Switchboard test sets when the language model (LM) is trained at the conversation level. In this paper, we also investigate whether a cross sentence transformer or conformer LM outperforms the LSTM LM, similar to [36].

External language model fusion through probability ratio: as proposed in [37], to decode with an end-to-end sequence posterior $p^{(\text{E2E})}(w_1^N|x_1^T)$ and external language model $p^{(\text{EXT})}(w_1^N)$, the language model learned by the E2E model $p^{(\text{E2E})}(w_1^N)$ has to be compensated.

$$p(w_1^N|x_1^T) \sim \frac{p^{(\text{E2E})}(w_1^N|x_1^T)}{p^{(\text{E2E})}(w_1^N)} \cdot p^{(\text{EXT})}(w_1^N) \qquad (2)$$

where the fraction is proportional to $p(x_1^T|w_1^N)$, since $p(x_1^T)$ can be ignored during search. Thus, Eq. 2 is basically the famous hybrid model equation applied at the sequence level [38].

Unlike the hybrid approach where the framewise priors can even directly be estimated and integrated into the softmax layer [39], it is difficult to estimate the sequence prior $p^{(\text{E2E})}(w_1^N)$. The usual solution is to estimate an additional, e.g. LSTM, model on the transcription of acoustic data and plug it into Eq. 2 after position- and sequence-level smoothing, according to the next paragraph.

Model combination: an encoder-decoder model provides position-wise normalized scores $p(w_n|w_1^{n-1}, x_1^T)$; thus, it is easy to apply classifier combination rules that use multiple models in making a decision [40, 41]. Log-linear score combination and joint decoding with various models is an old idea in ASR [42], and shallow fusion simply corresponds to sequence-level log-linear combination [43]. Extending the concept to K LM and L E2E models, in this paper the following decision rule is used to find the optimal sequence:

$$\hat{w}_1^{\hat{N}} = \operatorname*{argmax}_{w_1^N, N} \left(\sum_{n=1}^{N} \sum_{k=1}^{K} \lambda_{\text{LM}_k} \log p_k(w_n|w_1^{n-1}) \right.$$
$$+ \sum_{n=1}^{N} \sum_{l=1}^{L} \lambda_{\text{E2E}_l} \log p_l(w_n|w_1^{n-1}, x_1^T) \quad (3)$$
$$\left. + \sum_{l=1}^{L} \lambda_{\text{CT}_l} \sum_{t=1}^{T'} I_{\{\alpha > \tau_l\}} (\max_n \alpha_{l,n,t}) + \lambda_{\text{LEN}} N \right)$$

where the first term corresponds to sequence level log-linear interpolation of multiple external language models, and includes the probability ratio model with negative weight. The second term corresponds to the combination of the end-to-end ASR models. The third term (applied only to single-head decoder) is the coverage term (CT), using decoder specific threshold τ_l the indicator function I gives extra credit to hypotheses which cover the subsampled acoustic feature stream $(T \rightarrow T')$ with sharp decoder attention $\alpha_{n,t}$ [44, 45]. The last term controls the reward for emitting longer hypotheses [46]. In addition, the position-wise priors and posteriors are also smoothed:
$\tilde{p}_m(w_n|\cdot) = p_m(w_n|\cdot)^{\beta_m} / \sum_{w_n} p_m(w_n|\cdot)^{\beta_m}$.

3. Experimental setup

Our research focuses on the standard Switchboard (SWB) English conversational telephony speech recognition benchmark. Data preparation follows the Kaldi `s5c` recipe [47] and the work of [48]. The setup is based on our previous best system; for further details refer to [3]. Below, we focus mainly on the settings of the new components.

3.1. Input features

Unless noted otherwise, 80-dimensional log-Mel features are extracted from a 25ms long window shifted by 10ms. When using the 2.5ms frame rate, the window size is reduced to 10ms, the number of Mel filters to 20, and the SpecAugment parameters are also adjusted accordingly. Results with SEM and HEC are only presented for high temporal resolution log-Mel features; standard features showed marginal improvement together with SpecAugment. SEM is turned on with 10% probability, the peak energy corresponds to the 95th percentile and the energy threshold is chosen randomly between -30 and -20dB. Instead of masking after normalization, we reset the Mel-filter outputs to the channel-wise utterance-level mean values. The high energy clipping is applied 40% of the time, and the channel-wise threshold η_c is randomly selected from the interval of 80-100th percentile. The HEC operation is followed by rescaling to preserve

the total energy of an utterance. Two frames of the high frame rate features are stacked and every second frame is skipped. The 100-dimensional i-vectors are extracted using a 2048-component 40-dimensional diagonal covariance Gaussian mixture UBM trained on PLP features.

3.2. Sequence-to-sequence models

The 300 and 2k-hour systems are built on graphemic units created by 600 and 1000 BPE rules [49]. Our default E2E model has a 6-layer bidirectional LSTM encoder and 2-layer unidirectional LSTM decoder. The conformer based decoder follows the structure of [50]. The second multi-head attention units, which use absolute positional encoding and the encoded sequence as key and value, are inserted after the convolutional block. We note that our LSTM decoder has a single-head location-aware additive attention mechanism, while the conformer decoder has multiple layers of multiplicative multi-head attention [32, 51]. In the SWB-300 experiments, the number of decoder and encoder conformer layers is limited to 3 and 12. The model and the inner dimensions of the feed-forward and convolution modules are set to 384, 1536, and 1280, and the number of heads and head dimension are 6 and 96. In the 2000-hour experiments 16-layer encoder and 6-layer decoder conformer modules are used, and the head and inner dimensions are increased to 128 and 2048. The dropout rates in the conformer modules are set to 15% and 20% for the 300 and 2000-hour setups. The input to the conformer encoder is processed by two 2D convolutional units with 128 and 256 output channels using 5x5 kernels, ReLU non-linearity and 10% dropout. The kernels are strided by two in each direction. Similarly, the LSTM encoder reduces the frame rate by a factor of 4 (or 8) to 1/40ms by pyramidal processing [52]. The i-vectors are concatenated either to the inputs of every LSTM encoder layer or position-wise feed-forward layer of a conformer. In order to use a highly parallelized implementation, zoneout and scheduled sampling are not applied to the conformer decoder. In each model, the batch normalization layers are frozen and turned into global normalization layers in the middle of the training [3]. The SWB-300 models are trained on 6 V100 GPUs with a batch size of 32 sequences per GPU. The SWB-2000 models are optimized on 24 GPUs with variable batch size of up to 128 sequences per GPU. In every case, the first iterations are used to warm up the learning rate [53]. The learning rate is then kept constant and annealed exponentially in the last 25% of the training by a total factor of 256. When switching from SGD to AdamW, the initial learning rate had to be reduced by a factor of 30. Overall 450-500k updates are performed until convergence to maximize sentence posterior probability (cross-entropy training).

3.3. Language models

The language models are trained on the 24M-word Switchboard+Fisher data. As in our previous study, the unidirectional LSTM language model has two layers and 2048 nodes per layer. The transformer-XL language model is based on the implementation of [54, 55], having 10 self-attention layers, 8 heads with 64 dimension per head, and the inner and model dimension is set to 2048 and 512. The conformer LM has a similar structure, but the inner dimension in the feed-forward blocks is reduced to 1536 and 768, and the convolution module's inner dimension is set to 768. Batch normalization and the macaron-like, second feed-forward units are deactivated. For both models, the dropout rates are set to 0.05, and the memory of the self-attention is unconstrained, limited only by the input sequence length. The cross-utterance models are initialized with the model trained on

optimizer	ivec.	w/o LM			w/ LM		
		swb	chm	tot.	swb	chm	tot.
SGD		7.5	14.8	11.2	6.4	13.2	9.8
AdamW		6.9	13.8	10.4	6.1	12.6	9.4
	✓	**6.8**	**13.5**	**10.2**	**6.0**	**12.4**	**9.3**

Table 2: *Experiments with alternative log-Mel speech representation on Switchboard-300 using LSTM model. WER measured after decoding with cross-utterance LM in shallow fusion.*

logMel			SEM HEC	optim.	ivec.	WER [%]		
win. [ms]		dim.				hub5'00		
step	size					swb	chm	tot.
10	25	80		AdamW	✓	**6.0**	**12.4**	**9.3**
2.5	10	20	✓	SGD		6.5	13.1	9.8
						6.2	13.0	9.6
				AdamW		6.2	12.3	9.3
					✓	**6.2**	**12.0**	**9.1**

independent utterances. Their input is constructed by concatenation of successive utterances of a conversation channel up to 150 words. The denominator model of Eq. 2 follows the decoder structure of a corresponding E2E model, and is trained on the reference transcriptions of the acoustic data. All models are trained by SGD optimization with batch size of 256 sequences and Nesterov momentum [56].

The decoding hyper parameters are optimized on Hub5'00, iteratively by one dimensional grid search. During search, the beam size is limited to 16. To validate our findings and avoid learning on the test, we also evaluated the best systems on RT03 and Hub5'01, measuring word error rate (WER).

4. Experimental results

4.1. Comparison of SGD and AdamW optimizers, effect of i-vector on LSTM model

As can be seen in Table 1, AdamW significantly improves the quality of the LSTM model, on average by 7% without LM and 4% after shallow fusion with cross-utterance LM. I-vectors improve the results further, and give small but consistent gain after decoding with LM. We note that the best result is produced with a 57M-parameter E2E model in Table 1, and it already outperforms the 280M-parameter model developed in [3]. In Table 2, we test our alternative, high temporal resolution log-Mel speech representations. Comparing to the best results in Table 1, it can be observed that the two features perform similarly. The systems turned out to be complementary, and their combination resulted in significant gain, see e.g. Table 5.

4.2. Experiments with conformers

In the next set of experiments, we analyzed whether conformer or LSTM elements are more beneficial in the encoder and decoder modules of an attention based E2E model. In Table 3, the conformer-LSTM encoder-decoder shows 0.4% absolute gain over the LSTM-LSTM configuration. I-vectors have inconsistent effects on conformer-LSTM E2E. We also note that the conformer-encoder model ran significantly faster than the LSTM based one. Using beam size of 4, the model achieved 0.12 real-time factor (RTF) without loss of accuracy on a single core of an Intel Xeon E5-2690v4. After 8-bit integer quantization of the feed-forward and LSTM weights, we measured 0.08 RTF on a single core of an Intel Xeon Platinum 8280M

Table 3: *Performance comparison of conformer and LSTM encoder/decoder blocks on Switchboard-300. External LSTM LM operates across utterances using shallow fusion.*

model		optim.	sched. samp.	ivec.	ext. LM	hub5'00		
enc.	dec.					swb	chm	tot.
LSTM	LSTM	AdamW	✓	✓	✓	6.0	12.4	9.3
Conf.	LSTM	SGD				6.8	13.5	10.2
						6.7	13.0	9.8
		AdamW	✓		✓	**5.8**	12.0	**8.9**
				✓		6.5	12.9	9.7
				✓	✓	5.9	**11.8**	**8.9**
Conf.	Conf.	SGD				6.9	14.2	10.5
						7.0	13.6	10.3
		AdamW	✓			6.9	13.4	10.1
					✓	6.3	13.1	9.7
				✓		6.8	13.3	10.1
				✓	✓	6.1	12.9	9.5
LSTM	Conf.	AdamW				7.6	13.9	10.7

Table 4: *Effect of external language model on conformer encoder-decoder model on Switchboard-300.*

LM				PPL	WER [%]		
	#param.	xutt.	prob. ratio	CV	hub5'00		
					swb	chm	tot.
n/a					6.7	13.0	9.8
LSTM	57M			52.9	5.9	12.2	9.0
			✓		5.8	11.8	8.8
		✓		44.1	5.8	12.0	8.9
					5.7	11.4	8.6
TrafoXL	35M	✓	✓	39.3	**5.6**	**11.2**	**8.4**
Conf.	42M			39.0	**5.6**	11.3	8.5
LSTM + TrafoXL					5.5	11.2	8.4
LSTM + TrafoXL + Conf.					**5.4**	**11.1**	**8.3**

CPU and 0.1% WER degradation on Hub5'00. Switching to a conformer based decoder turned out to be less efficient than a single head LSTM decoder, and the performance degraded. For a more fair comparison, we also implemented scheduled sampling (SS) for the conformer decoder, and applied teacher forcing with 0.8 probability. As can be seen, SS indeed improves the conformer decoder results (by 2% relative), nevertheless the training time increases by more than 30%. Surprisingly, after decoding with external LM, the conformer-decoder results improve only 0.4% absolute, compared to the 0.8-0.9% gain we measure with conformer-LSTM model. For the sake of completeness, we also ran experiments with LSTM-encoder conformer-decoder models. Such model gives even worse results than the conformer-conformer architecture, and we measure 0.4% absolute WER degradation.

4.3. Effect of decoding with external language model

Shallow fusion showed large improvement even with the best performing conformer model in Table 3. Additional experiments were designed to investigate the effect of probability ratio fusion and self-attention LMs. As can be seen in Table 4, incorporating the external LM through probability ratio fusion results in 0.2-0.3% absolute WER improvement. The gain is consistent even if cross-utterance external LM is used. Further, transformer-XL LM gives additional 0.2% reduction. Conformer LM turned out to be not better than transformer-XL. Since the probability ratio approach already corresponds to LM combination, we also ran recognition experiments with combined cross-utterance LSTM and attention based LMs. As can be seen, slight improvement is measured over the best "single" LM result in Table 4. Decoding with the best model runs at 0.6 RTF on a K80 GPU.

Table 5: *Overall results on Switchboard 300 and 2000 with single and combined end-to-end and cross-utterance language models. (Ext. LM based SWB-300 and SWB-2k results are obtained with probability ratio and shallow fusion, respectively.)*

		E2E			fea.ext	ivec.	ext. LM	hub5'00		hub5'01			rt03	
	id	enc.	dec.	#parm.	step [ms]			swb	chm	swb	swb2p3	swb2p4	swb	fsh
SWB-300	1	LSTM	LSTM	57M	10		LSTM	6.0	12.0	6.6	8.8	12.8	13.6	7.5
	2			61M		✓		5.8	12.1	6.6	8.6	12.8	14.0	7.7
	3			56M	2.5			6.0	12.0	6.7	8.6	12.5	13.2	7.7
	4			60M		✓	LSTM	5.9	11.5	6.5	8.5	11.6	13.0	7.2
							LSTM+TrafoXL	5.7	11.3	6.1	8.3	11.3	12.6	7.0
	5	Conf.		68M	10		-	6.7	13.0	7.1	9.2	13.7	15.7	9.1
							LSTM	5.7	11.4	6.2	7.8	11.4	12.8	7.2
							LSTM+TrafoXL	5.5	11.2	6.1	7.7	11.4	12.6	7.0
	1+3+5						-	6.1	11.9	6.4	8.8	12.6	13.7	8.0
							LSTM+TrafoXL	5.1	10.1	5.5	**6.9**	10.5	**10.7**	6.1
	2+4+5						LSTM+TrafoXL	**5.0**	**10.0**	**5.3**	7.0	**10.4**	10.8	**6.0**
SWB-2k	6	LSTM	LSTM	661M	10		LSTM	4.6	7.8	5.2	6.1	10.0	8.0	6.6
	7			663M	2.5			4.9	7.7	5.5	6.2	9.8	8.4	6.7
	8	Conf.		99M	10		-	4.8	8.0	5.2	6.4	10.3	8.2	6.7
							LSTM	4.7	7.6	5.0	6.1	9.7	7.7	**5.9**
							LSTM+TrafoXL	4.6	7.6	4.9	6.2	9.5	7.8	**5.9**
	9	Conf.	Conf.	154M			-	4.7	7.6	5.0	6.1	10.0	8.0	7.0
							LSTM+TrafoXL	4.5	7.3	4.9	5.8	9.4	7.6	6.6
	6+7						LSTM+TrafoXL	4.5	7.2	4.9	5.7	9.2	7.6	6.1
	6+7+8+9							**4.3**	**6.8**	**4.6**	**5.5**	**9.0**	**7.2**	**5.9**

We also ran probability ratio LM fusion experiments with conformer decoders (results not presented), in order to get a better understanding of the small effect of the external language model with such decoders we observed in Section 4.2. Although the probability ratio approach improved the conformer decoder result more (0.4% absolute WER improvement without i-vector), the conformer decoder still lagged significantly behind the LSTM decoder. The gap remained roughly the same irrespective of the type and depth of the denominator LM. We hypothesize that a deep multi-headed self-attention LM learns a significantly different mechanism than the decoder of a multi-headed ASR. Thus, a simple model plug-in is not effective in recovering a score proportional to the emission likelihood $p(x_1^T|w_1^N)$.

4.4. Combination experiments

Besides decoding with combined LMs, the application of multiple E2E models is also tested. The detailed results of the single and combined systems are shown in the upper section of Table 5. The superiority of the conformer over LSTM encoder is also confirmed on the Hub5'01 and RT03 sets. The LSTM encoder trained on high temporal-resolution log-Mel and i-vector features is nevertheless fairly competitive, especially after decoding with combined LSTM+TrafoXL LM. The combination of E2E models without external LM resulted in significant gains, in some cases over 10% relative WER improvement over a very strong baseline. E.g. WER on Hub5'00 CHM improved from 13.0% to 11.9%. The i-vector has inconsistent effect on the combination. Surprisingly, the effect of decoding with external language models is not mitigated by the combination of E2E models, and additional 16-30% relative WER improvement is observed. We also note that our best 300-hour results could even match system combination results on 2000 hours published only a few years ago [57].

4.5. Experiments on Switchboard-2000

Building systems on 2000 hours of speech data, we made the following observations. Without retraining the UBM and total variability matrix, i-vectors showed inconsistent gain on the evaluation sets. Scheduled sampling improved conformer decoder based E2E model by 3% relative on the larger dataset. Not surprisingly, probability ratio fusion did not improve the results,

because the E2E and LM were trained on the same dataset. Moreover, switching from SGD to AdamW did not show consistent improvement with large scale models; the objective comparison can be made by contrasting the best LSTM results in [3] and the corresponding row in Table 5.

As results indicate in Table 5, a 100M-parameter conformer-encoder outperforms the large scale LSTM models. On average, the best single-model performance is achieved by conformer-conformer model, but also note the discrepancy on the Fisher subset of RT03. Overall, system and language model combination improves the state-of-the-art by 8-14% relative, compared to the previous best numbers reported in [3]. The improvement is also confirmed on RT02 and RT04 sets, where the best combination achieves 6.3% and 5.2% WER. The performance of the best system combination on SWB part of Hub5'00 (4.3%) is clearly below the human error rate (5.1%) measured in [48]. This can be attributed to fact that most of the speakers appear in the training data, hyperparameters are optimized on Hub5'00, and the human error rate might also have been overestimated. We note that the 4-E2E combination with multiple LMs can run on a V100 GPU at 0.4 RTF. Considering that human performance, and thus transcription error rate, is at 6.8%, 6.0%, 4.5%, 4.7% WER on the CHM subset of Hub5'00 and on the RT0{2,3,4} sets [48], it is surprising to see how close we can actually get with 2000 or even only 300 hours of data using fairly simple and general attention models.

5. Conclusions

We investigated a set of techniques on top of our best recipe for the Switchboard English speech recognition benchmark. We showed that complementary systems with similar performances can be developed using different input features or diverse encoder, decoder and language model structures. Using a more advanced optimizer and score combination techniques, our final Switchboard-300 system achieved 5.0% and 10.0% WER on the SWB and CHM subsets of Hub5'00. Joint decoding with such models trained on Switchboard-2000 achieves a new state of the art: the flexible attention models reach the limit of many of the standard evaluation sets. Future work could focus on achieving such performance with a limited amount of transcriptions.

6. References

[1] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.

[2] H. K. J. Kuo *et al.*, "End-to-end spoken language understanding without full transcripts," in *Interspeech*, 2020, pp. 906–910.

[3] Z. Tüske *et al.*, "Single headed attention based sequence-to-sequence model for state-of-the-art results on Switchboard," in *Interspeech*, 2020, pp. 551–555.

[4] G. E. Hinton *et al.*, "Improving neural networks by preventing co-adaptation of feature detectors," 2012. [Online]. Available: http://arxiv.org/abs/1207.0580

[5] L. Wan *et al.*, "Regularization of neural networks using DropConnect," in *ICML*, vol. 28, no. 3, 2013, pp. 1058–1066.

[6] N. Kanda, R. Takeda, and Y. Obuchi, "Elastic spectral distortion for low resource speech recognition with deep neural networks," in *ASRU*, 2013, pp. 309–314.

[7] T. Ko *et al.*, "Audio augmentation for speech recognition," in *Interspeech*, 2015, pp. 3586–3589.

[8] G. Saon *et al.*, "Sequence noise injected training for end-to-end speech recognition," in *ICASSP*, 2019, pp. 6261–6265.

[9] D. S. Park *et al.*, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Interspeech*, 2019, pp. 2613–2617.

[10] A. F. Murray and P. J. Edwards, "Enhanced MLP performance and fault tolerance resulting from synaptic weight noise during training," *IEEE Trans. on Neural Networks*, vol. 5, no. 5, pp. 792–802, 1994.

[11] D. Krueger *et al.*, "Zoneout: regularizing RNNs by randomly preserving hidden activations," in *ICLR*, 2017.

[12] C. Szegedy *et al.*, "Rethinking the inception architecture for computer vision," in *CVPR*, 2016, pp. 2818–2826.

[13] A. Krogh and J. A. Hertz, "A simple weight decay can improve generalization," in *NIPS*, 1992, pp. 950–957.

[14] Y. Bengio *et al.*, "Curriculum learning," in *ICML*, 2009, pp. 41–48.

[15] S. Bengio *et al.*, "Scheduled sampling for sequence prediction with recurrent neural networks," in *NIPS*, 2015, pp. 1171–1179.

[16] D. Amodei *et al.*, "Deep speech 2: End-to-end speech recognition in English and Mandarin," in *ICML*, 2016, pp. 173–182.

[17] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[18] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *ICLR*, 2019.

[19] G. Saon *et al.*, "Speaker adaptation of neural network acoustic models using i-vectors," in *ASRU*, 2013, pp. 55–59.

[20] M. Kitza *et al.*, "Cumulative adaptation for BLSTM acoustic models," in *Interspeech*, 2019, pp. 754–758.

[21] G. Saon *et al.*, "Advancing RNN transducer technology for speech recognition," in *ICASSP*, 2021.

[22] Z. Zhu, J. H. Engel, and A. Hannun, "Learning multiscale features directly from waveforms," in *Interspeech*, 2016, pp. 1305–1309.

[23] Z. Tüske, R. Schlüter, and H. Ney, "Acoustic modeling of speech waveform based on multi-resolution, neural network signal processing," in *ICASSP*, 2018, pp. 4859–4863.

[24] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, 1990.

[25] R. Schlüter *et al.*, "Gammatone features and feature combination for large vocabulary speech recognition," in *ICASSP*, 2007, pp. 649–652.

[26] C. Kim, K. Kim, and S. R. Indurthi, "Small energy masking for improved neural network training for end-to-end speech recognition," in *ICASSP*, 2020, pp. 7684–7688.

[27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[28] M. Sperber *et al.*, "Self-attentional acoustic models," in *Interspeech*, 2018, pp. 3723–3727.

[29] D. Povey *et al.*, "A time-restricted self-attention layer for ASR," in *ICASSP*, 2018, pp. 5874–5878.

[30] Q. Zhang *et al.*, "Transformer transducer: A streamable speech recognition model with transformer encoders and RNN-T loss," in *Interspeech*, 2020, pp. 7829–7833.

[31] L. Dong, S. Xu, and B. Xu, "Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition," in *ICASSP*, 2018, pp. 5884–5888.

[32] A. Gulati *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," in *Interspeech*, 2020, pp. 5036–5040.

[33] R. Al-Rfou *et al.*, "Character-level language modeling with deeper self-attention," in *AAAI Conf. on Artificial Intelligence*, 2019, pp. 3159–3166.

[34] Z. Tüske, R. Schlüter, and H. Ney, "Investigation on LSTM recurrent n-gram language models for speech recognition," in *Interspeech*, 2018, pp. 3358–3362.

[35] W. Xiong *et al.*, "The Microsoft 2017 conversational speech recognition system," in *ICASSP*, 2018, pp. 5934–5938.

[36] K. Irie *et al.*, "Training language models for long-span cross-sentence evaluation," in *ASRU*, 2019, pp. 419–426.

[37] E. McDermott, H. Sak, and E. Variani, "A density ratio approach to language model fusion in end-to-end automatic speech recognition," in *ASRU*, 2019, pp. 434–441.

[38] H. A. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*. Norwell, MA, USA: Kluwer Academic Publishers, 1993.

[39] V. Manohar, D. Povey, and S. Khudanpur, "Semi-supervised maximum mutual information training of deep neural network acoustic models," in *Interspeech*, 2015, pp. 2630–2634.

[40] J. Kittler *et al.*, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[41] K. Kirchhoff and J. A. Bilmes, "Combination and joint training of acoustic classifiers for speech recognition," in *ASR2000, ISCA Tutorial and Research Workshop*, 2000, pp. 17–23.

[42] C. Dugast, L. Devillers, and X. Aubert, "Combining TDNN and HMM in a hybrid system for improved continuous-speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 1, pp. 217–223, 1994.

[43] Ç. Gülçehre *et al.*, "On using monolingual corpora in neural machine translation," 2015. [Online]. Available: http://arxiv.org/abs/1503.03535

[44] J. Chorowski and N. Jaitly, "Towards better decoding and language model integration in sequence to sequence models," 2016. [Online]. Available: http://arxiv.org/abs/1612.02695

[45] Z. Tu *et al.*, "Modeling coverage for neural machine translation," in *ACL*, 2016, pp. 76–85.

[46] A. Hannun *et al.*, "Deep Speech: Scaling up end-to-end speech recognition," 2014. [Online]. Available: http://arxiv.org/abs/1412.5567

[47] D. Povey *et al.*, "The Kaldi speech recognition toolkit," in *ASRU*, 2011.

[48] G. Saon *et al.*, "English conversational telephone speech recognition by humans and machines," in *Interspeech*, 2017, pp. 132–136.

[49] "https://github.com/rsennrich/subword-nmt."

[50] A. Vaswani *et al.*, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.

[51] J. K. Chorowski *et al.*, "Attention-based models for speech recognition," in *NIPS*, 2015, pp. 577–585.

[52] W. Chan *et al.*, "Listen, attend and spell: a neural network for large vocabulary conversational speech recognition," in *ICASSP*, 2016, pp. 4960–4964.

[53] P. Goyal *et al.*, "Accurate, large minibatch SGD: Training imagenet in 1 hour," 2017. [Online]. Available: http://arxiv.org/abs/1706.02677

[54] "https://github.com/NVIDIA/DeepLearningExamples."

[55] Z. Dai *et al.*, "Transformer-XL: Attentive language models beyond a fixed-length context," in *ACL*, 2019, pp. 2978–2988.

[56] Y. Nesterov, "A method of solving a convex programming problem with convergence rate $O(1/k^2)$," in *Soviet Mathematics Doklady*, vol. 27, no. 2, 1983, pp. 372–376.

[57] K. J. Han *et al.*, "The CAPIO 2017 conversational speech recognition system," 2018. [Online]. Available: http://arxiv.org/abs/1801.00059