INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Rethinking Evaluation in ASR: Are Our Models Robust Enough?

Tatiana Likhomanenko[1*], *Qiantong Xu*[1*], *Vineel Pratap*[1], *Paden Tomasello*[1], *Jacob Kahn*[1], *Gilad Avidov*[1], *Ronan Collobert*[1], *Gabriel Synnaeve*[2]

[1]Facebook AI Research, USA
[2]Facebook AI Research, France
{antares,qiantong}@fb.com

Abstract

Is pushing numbers on a single benchmark valuable in automatic speech recognition? Research results in acoustic modeling are typically evaluated based on performance on a single dataset. While the research community has coalesced around various benchmarks, we set out to understand generalization performance in acoustic modeling across datasets – in particular, if models trained on a single dataset transfer to other (possibly out-of-domain) datasets. Further, we demonstrate that when a large enough set of benchmarks is used, average word error rate (WER) performance over them provides a good proxy for performance on real-world data. Finally, we show that training a single acoustic model on the most widely-used datasets – combined – reaches competitive performance on both research and real-world benchmarks.

Index Terms: automatic speech recognition, generalization, transfer, robustness

1. Introduction and Related Work

Progress in automatic speech recognition (ASR) is measured on the validation and test sets of standard datasets. However, most acoustic models (AMs) are often developed and tuned on a single dataset and transfer poorly to other datasets. Moreover, most large standard benchmarks have similar domains and recording conditions. These factors lead to siloed ASR research. A unified benchmark comprised of conversational, oratory, and read speech with varied recording conditions and noise would certainly serve the research community well; here, however, we study how the currently-popular public benchmarks can be used to gauge model generalization performance.

Our approach constructs a validation procedure – using only public datasets – that is a better predictor of overall and domain transfer performance than datasets taken in isolation. We train the same state-of-the-art model architecture on different benchmarks pushing for best performance on each benchmark separately. We also jointly train a model on all datasets. Given the transfer performance on test sets, we can ascertain which test sets are good proxies for transfer performance as well as which training sets can produce the best-performing models. This informs us on the robustness of various datasets in transfer and which test sets are the best predictors of ASR performance in others. Finally, we look at the performance, in transfer only, on our in-house ASR datasets. This informs us about which sets of test sets should be used if one wants to transfer to a wide range of conditions of speech.

Previous works that study transfer in ASR include [1] that studied transferring varying number of layers trained out-of-domain, from SwitchBoard to AMI-IHM or from LibriSpeech to AMI-IHM. In this paper as in ours, a joint model trained on multiple out-of-domain datasets exhibits better transfer. In the context of the Arabic MGB-3 challenge, the authors of [2] transferred acoustic models trained on broadcast TV to Youtube videos, with a different setting than here as the training transcriptions were noisily labeled. Distillation was used to improved transfer in [3], where the soft-target part of the distillation loss may help with regularization. For another kind of transfer in [4], the authors transferred LibriSpeech trained wav2letter [5] models to German by fine-tuning them on German, with better performance than training from scratch. Very recently, authors of [6] point out some limitations of current ASR benchmarks, and propose guidelines to create multi-domain datasets. Finally, while Deep-Speech 2 [7] did not focus their study on transfer, we train a single acoustic model on multiple datasets at once, as they did.

[*] Equal contribution.

2. Domain Transfer

In order to study transfer across datasets and conditions, we do a systematic analysis. In all our experiments, we use a single Transformer-based acoustic model architecture with 270M parameters, to make our results comparable across the board. We train multiple single-dataset baselines as well as one joint model trained on all datasets at once. We then evaluate this set of models on all the validation and test sets, to measure how each "in-domain" model transfers to "out-of-domain" datasets. From this, we analyze which datasets suffer more acutely from "domain overfitting." Evidently, it is difficult to separate the "in-domainness" and size of a dataset; e.g., we cannot directly compare results on WSJ (80h) to ones on LibriSpeech (960h). We also fine-tune our joint model on the transfer dataset with 1h, 10h, and 100h of in-domain data. Finally, we examine how our models transfer to real data and in the process observe that public validation and test sets performance is predictive of the transfer performance of a model to real data.

3. Experiments

3.1. Datasets

To measure domain transfer, we restrict experiments to use only datasets in English, for which there exist several commonly-used and publicly available datasets with hundreds hours of transcribed audio. Validation sets from each dataset are used to optimize model configurations and to perform all hyper-parameter tuning, while test sets are used for final evaluation only.

LibriSpeech (LS) [8] consists of read speech from audiobook recordings. We use standard split of train, validation (*dev-clean*, *dev-other*) and test sets (*test-clean*, *test-other*).

SwitchBoard & Fisher (SB+FSH) consists of conversational telephone speech. To create a training set, we combine Switchboard [9] and Fisher [10, 11]. We use RT-03S [12] as the vali-

http://dx.doi.org/10.21437/Interspeech.2021-1758

dation set; test sets are the Hub5 Eval2000 [13] data with two subsets, SwitchBoard (SB) and CallHome (CH). For the data processing and evaluation, we follow the recipe provided by Kaldi [14].

Wall Street Journal (WSJ) [15, 16, 17]. We consider the standard subsets *si284*, *nov93dev* and *nov92* for training, validation and test, respectively. We remove any punctuation tokens from *si284* transcriptions when used for training.

Mozilla Common Voice (CV) project [18]. The CV dataset consists of transcribed audio in various languages where speakers record text from Wikipedia. Anyone can submit recorded contributions; as a result, the dataset has a large variation in quality and speakers. We use the English dataset[1], where data splits are provided therein.

TED-LIUM v3 (TL) [19] is based on TED conference videos. We use the last edition of the training set from this dataset (v3), for which the validation and test sets are kept consistent (and thus numbers are comparable) with the earlier releases. We follow the Kaldi recipe [14] for data preparation.

Robust Video (RV) is our in-house English video dataset, which are sampled from public social media videos and aggregated and deidentified before transcription. These videos contain a diverse range of speakers, accents, topics, and acoustic conditions making ASR difficult. The test sets are composed of *clean*, *noisy* and *extreme* with *extreme* being the most acoustically challenging subset among them. The validation set comprises of data from *clean* and *noisy* subsets.

Table 1: *Statistics on datasets.*

Data	kHz	Train (h)	Valid (h)	Test (h)	Speech
WSJ	16	81.5	1.1	0.7	read
TL	16	452	1.6	2.6	oratory
CV	48	693	27.1	25.8	read
LS	16	960	5.1+5.4	5.4+5.4	read
SB+FSH	8	300+2k	6.3	1.7+2.1	convers.
RV	16	5k	14.4	18.8+19.5+37.2	diverse

Table 2: *Statistics on datasets: mean sample duration (in seconds) and mean sample transcription length (in words).*

Data	Train $\mu \pm \sigma$ (s)	Valid $\mu \pm \sigma$ (s)	Test $\mu \pm \sigma$ (s)	Train $\mu \pm \sigma$ (wrd)	Valid $\mu \pm \sigma$ (wrd)	Test $\mu \pm \sigma$ (wrd)
WSJ	7.8 ± 2.9	7.8 ± 2.9	7.6 ± 2.5	17 ± 7	16 ± 7	17 ± 6
TL	6 ± 3	11.3 ± 5.7	8.1 ± 4.3	17 ± 10	35 ± 20	24 ± 15
CV	5.7 ± 1.6	6.1 ± 1.8	5.8 ± 2.6	10 ± 3	10 ± 3	9 ± 3
LS	12.3 ± 3.8	6.8 ± 4.5	7 ± 4.8	33 ± 12	19 ± 13	19 ± 13
SB+FSH	3.7 ± 3.2	4 ± 3.1	2.1 ± 1.7	11 ± 12	12 ± 12	8 ± 8
RV	8.5 ± 1.9	11.6 ± 2.8	11.6 ± 2.7	21 ± 10	25 ± 13	29 ± 12

3.2. Unifying Audio

The datasets used in our work have different sample rate and varied input lengths as shown in Tables 1 and 2. One challenge when training joint models on combined audio data is determining the frequency range that the filterbanks should span to compute log-mel spectrogram features. For example, as SB+FSH has the lowest sample rate of 8kHz, filterbanks can span up to 4kHz and any spectrogram features beyond 4kHz can't be determined accurately [20], as shown in Figure 1. Since we require the same set of filterbanks for joint training across all datasets, we use the minimum sample rate over all datasets (8kHz) and use this setup for training both baseline models on individual datasets as well as joint models. In this case the distribution of mean normalized energy of filterbanks for all the samples in each dataset is similar. For datasets with higher original sample rates, downsampling negatively affects performance; for LS, for example, the WER is absolute 0.3% and 0.2% worse on dev-other, without and with beam-search decoding, respectively. For SB+FSH with lower original sample rate, upsampling to 16kHz also negatively affects performance: up to 1% absolute worse WER. For all experiments we compute 80 log-mel spectrogram features for a 25ms sliding window, strided by 10ms. All features are normalized to have zero mean and unit variance per input sequence before feeding into the neural network.

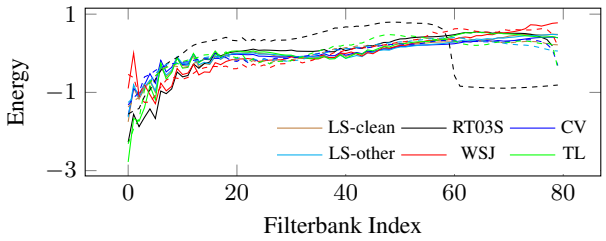

Figure 1: *Distribution of the mean normalized energy of 80 filterbanks on all the public validation sets we used, for 16kHz audio (dashed) and 8kHz audio (solid).*

3.3. Baselines and Joint Model

Table 3: *Perplexity (including out-of-vocabulary words) of word-level language models. We use 4-gram language model for WSJ, LS, SB+FSH, and 5-gram for TL, CV.*

Data/Vocab	in-dom. n-gram		in-dom. Transf.		CC 4-gram	
	Valid	Test	Valid	Test	Valid	Test
WSJ/162K	159	134	83	65	297	285
TL/200k	119	149	79	81	142	136
CV/168K	359	329	256	240	213	157
LS/200K	155/147	164/154	48/50	52/50	258/258	244/249
SB+FSH/64K	124	114/112	91	82/85	221	199/153
RV/200K	158	146	-	-	249	204

3.3.1. Acoustic Model (AM)

All models are trained with Connectionist Temporal Classification [25] and the network architecture follows [26]: the encoder of our acoustic models is composed of a convolutional frontend (1-D convolution with kernel-width 7 and stride 3 followed by GLU activation) followed by 36 4-heads Transformer blocks [27]. The self-attention dimension is 768 and the feed-forward network (FFN) dimension is 3072 in each Transformer block. The output of the encoder is followed by a linear layer to the output classes. We use dropout after the convolution layer. For all Transformer layers, we use dropout on the self-attention and on the FFN, and layer drop [28], dropping entire layers at the FFN level. Dropout and layer dropout values are tuned for each model separately. Token set for all acoustic models consists of 26 English alphabet letters, augmented with the apostrophe and a word boundary token. The popular approach with word-

[1]June 22nd 2020's snapshot: `https://tinyurl.com/cvjune2020`. Transcriptions contain upper-case and non-English characters and punctuation. To have similar transcription normalization as in other datasets, we normalize the text for all splits: lower-casing, Unicode normalization, removing punctuation and non-English tokens, and mapping common abbreviations (e.g. "mr." to "mister").

Table 4: *WER of models evaluated on all datasets (downsampled to 8kHz) with a greedy decoding and no LM (top), with in-domain n-gram LM beam-search decoding (middle) and with additional second-pass rescoring by in-domain Transformer LM (below), with beam-search decoding and 4-gram CC LM for joint model (joint CC). SOTA models are given from WSJ [21], TED-LIUM [22], LibriSpeech [23], SwitchBoard & Fisher [24]. The average is computed as average of averages for LibriSpeech's validations/tests, and SwitchBoard's tests (SB, CH) sets, so as not to weight them more heavily.*

Train	WSJ		TL		CV		LS				SB+FSH			average	
	nov93	nov92	valid	test	valid	test	dev-c	test-c	dev-o	test-o	RT03S	SB	CH	valid	test
SOTA		2.8	5.1	5.6				1.9		3.9	8.0	5.0	9.1		
WSJ	13.3	11.5	42.9	41.7	70.7	76.3	31.1	30.6	52.2	53.5	65.9	57.3	63.1	46.9	46.4
	8.1	6.4	28.4	28.9	54.5	61.7	16.4	16.7	36.8	38.7	52.3	44.2	49.7	34.0	34.3
	6.4	5.2	26.7	26.8	52.8	60.2	12.8	13.3	33.8	35.9	49.8	42.2	47.2	31.8	32.3
TL	12.9	10.7	7.4	7.5	30.8	34.7	9.7	9.8	20.0	20.4	28.3	20.0	28.4	18.9	18.4
	10.0	6.2	6.1	6.4	23.0	27.1	5.7	6.1	13.5	14.3	23.9	16.5	24.5	14.5	14.1
	6.9	5.4	5.8	6.0	22.0	26.1	4.0	4.5	10.1	11.7	23.3	16.6	24.8	13.0	13.3
CV	12.1	9.0	46.4	30.0	13.1	16.9	19.2	20.9	25.3	27.0	47.8	39.7	43.6	28.3	24.3
	6.7	4.1	38.2	23.4	10.8	13.8	14.3	16.1	18.3	20.1	37.1	29.9	34.2	21.8	18.3
	5.7	3.6	37.7	21.8	10.7	13.6	12.6	14.5	15.9	17.7	35.3	28.0	32.9	20.7	17.1
LS-960	13.6	11.0	12.7	13.4	30.0	34.1	2.8	2.8	7.1	7.1	36.4	27.1	33.8	19.5	18.8
	7.1	3.8	7.8	9.4	18.8	22.5	2.0	2.5	5.3	5.6	27.5	19.3	26.4	13.0	12.5
	4.9	3.6	7.3	8.6	18.1	22.0	1.5	2.1	4.3	4.7	25.9	18.3	25.3	11.8	11.9
SB+FSH	12.1	11.5	14.9	12.8	42.6	45.7	14.1	15.0	28.6	29.2	12.8	7.7	12.0	20.8	20.4
	6.4	5.2	8.5	8.8	31.7	36.0	7.1	7.9	19.1	20.4	10.4	6.5	10.3	14.0	14.5
	5.1	3.9	8.1	8.2	29.8	34.3	4.6	5.7	16.1	17.5	10.3	6.4	10.4	12.7	13.3
Joint	4.5	3.4	6.9	6.9	13.1	15.5	3.0	3.0	7.3	7.3	11.7	6.3	10.7	8.3	7.9
	3.1	2.0	5.4	5.7	10.5	12.6	2.0	2.5	5.2	5.6	9.8	5.9	9.5	6.5	6.4
	2.9	2.1	5.1	5.2	10.3	12.3	1.4	2.1	4.1	4.4	9.7	5.8	9.3	6.2	6.1
Joint CC	4.0	2.8	5.6	5.7	8.9	10.6	3.1	3.0	6.0	6.0	10.0	5.5	9.1	6.6	6.2

pieces as tokens set we found to be not suited as intersection between word-pieces constructed on every training set less than 50%. Thus the question what word-pieces set should be used for the joint model is still open. SpecAugment [29] is used for data augmentation in training: there are two frequency masks, and ten time masks with maximum time mask ratio of $p = 0.1$; frequency and time mask parameters are tuned separately for each model; time warping is not used. In the joint model, the maximum frequency bands masked by one frequency mask is 30, and the maximum frames masked by the time mask is 30, too. We use the Adagrad optimizer [30] and decay learning rate by a factor of 2 each time the WER reaches a plateau on the validation sets. All experiments are implemented within flashlight[2] and wav2letter++ pratap2018wav2letter. All models are trained with dynamic batching (effective average batch size is 240s per GPU) and mixed-precision computations on 16 GPUs (Volta 32GB) for 1-5 days for single dataset baselines and 21 days for joint training.

3.3.2. Language Model (LM) and Beam-search Decoding

We train an n-gram LM using KenLM toolkit [31] and a Transformer LM similar to [26] for each dataset independently using their in-domain LM training corpus. Specifically, we use the training transcriptions as LM corpus for domains like SB+FSH and RV; while for TL, both training transcriptions and the original LM corpus are combined together to train its LM. All the Transformer LMs share the similar architecture as [32]'s Google Billion Words model: we use 8 attention heads; 8 (WSJ, CV and SB+FSH), 16 (TL) or 20 (LS) decoder layers with embedding, input and output dimensions of 512 (CV), 1024 (WSJ and SB+FHS) or 1280 (TL and LS); feed-forward layer dimension is set to 1024 (CV), 2048 (WSJ and SB+FHS) or 6144 (TL and LS); dropout is 0.3 (WSJ, CV and SB+FHS), 0.15 (TL) or 0.1 (LS). Number of decoder layers, embedding dimensions as well as dropout were tuned on each dataset depending on the amount of training data.

We also train a 4-gram and a Transformer LMs on Common Crawl (CC) data [33]. Before any training we perform the following text normalization for CC data: splitting paragraphs into separate sentences, punctuation removal, mapping of common abbreviations, converting latin and roman numbers into the text. We keep a dictionary of 200k most-common words. For 4-gram training we prune all 3,4-grams appearing once and use only 10% of the CC data. The perplexity of all LMs is shown in Table 3.

To integrate LMs with AMs, we use one-pass beam-search decoder from the wav2letter++ [5] (lexicon-based with an n-gram LM) and an additional second-pass rescoring with a Transformer LM following [26].

3.3.3. Joint Model

We adopt the same acoustic model architecture described above but with less regularization when training on the combination of all the datasets. We weight each sample equally, i.e. each sample from each dataset is fed into the model once in each epoch. In Table 4 for each dataset, we report known state-of-the-art models with in-domain language models.

3.4. Acoustic Model Transfer

In general, an acoustic model trained in isolation on a single dataset performs poorly on other datasets, as shown in Table 4.

[2] https://github.com/flashlight/flashlight

The model trained on WSJ performs the worst (part of the reason could be the smaller amount of training data) for transfer, while other models transfer very well to WSJ. All models transfer poorly to CV and the CV model transfers poorly to other datasets, which may indicate that CV is very different from other benchmarks. From the results on LS, TL and SB+FSH there is a similarity between LS and TL (they transfer the best to each other). There is also a similarity in transfer between SB+FSH and TL benchmarks, however, LS and SB+FSH do not transfer well to each other. When training on all datasets at once, the joint model in Table 4 performs better or close to a single dataset training. This behaviour compared to results on a single dataset training indicates that i) datasets differ from each other and ii) a robust model scoring well on all these benchmarks exists.

In Table 5 we report results of transfer, of those same models trained on public datasets, to our in-house RV dataset. We also report numbers from a baseline system that is trained in-domain on a corresponding training set of 5000h. As for other benchmarks, single dataset training transfers poorly to in-house data, however, the transfer quality varies a lot, having the best results from the TL model. At the same time our joint model, which performs well on each benchmark, transfers really well, stating that i) public datasets could be the good proxy of training data for real-world ASR, ii) improving average performance on public benchmarks leads to improving performance on real-world noisy data. Moreover, fine-tuning of this joint model on 1h closes the gap with the RV baseline model and fine-tuning with 10h or 100h of data and decoding with Common Crawl language model surpasses WER compared to the RV baseline model decoded with in-domain language model.

3.5. Language Model Transfer

Single-dataset acoustic models get a boost in WER performance when decoding/rescoring with an in-domain language model, as shown in Table 4. These acoustic models perform however poorly in transfer domain conditions (see Tables 4 and 5). In contrast, the joint model transfers well to in-house RV data, when decoded with an in-domain language model (see Table 5). Decoding the joint model with the large generic Common Crawl language model leads to WER performance which is overall improved, on both public and in-house RV datasets.

3.6. Predictors of Transfer

We performed single variable linear regressions using data from Table 4: lines as datapoints, test set score columns as features, and labels being the same models' performance in transfer on the average of test clean, noisy, and extreme, from Table 5. Across all datasets, and taken over all trained models, the best "single test set" predictor for out-of-domain performance on RV data is SB with an $r^2 = 0.8$ (rejecting the null hypothesis with $p < 0.001$), the worst single predictor being WSJ's nov92 with $r^2 = 0.5$ ($p < 0.001$). We also performed multivariate regressions using all the test results from Table 4 and only the results for the models decoded with n-gram language models. This gives an overspecified problem (more variables: 7, than models: 6), so OLS gives a "perfect" (overfitted, $r^2 = 1$) solution which weights nov92, test-clean, test-other *negatively*. We repeat this regression with heavy L1 regularization (Lasso, as proxy for L0 norm regularization) and it yields a regression with $r^2 = 0.98$ (although only 6 datapoints) with only 3 test sets weighted non-zero, and positively: TL, CV, and SB. We can conclude that those test sets are the most predictive of the performance in transfer on RV of our Transformer-based acoustic models decoded with n-grams. A larger study across acoustic models and language models variants should provide a more robust conclusion.

Table 5: *WER comparison with a greedy decoding and with a 5-gram in-domain LM and/or the 4-gram CC LM beam-search decoding on RV validation and test data from videos. Except for the "RV" training and for models with "+finetune", all other models correspond to models in Table 4.*

Train	LM	Valid	Test		
			clean	noisy	extreme
RV	-	18.4	17.1	22.4	31.8
	in-dom.	12.8	15.7	20.9	29.8
WSJ	-	69.6	67.7	74.3	84.8
	in-dom.	56	54.9	62.4	71.8
TL	-	29.5	26	34.4	46.5
	in-dom.	22.1	21.4	29.4	40.6
CV	-	42.2	34.7	45.7	58
	in-dom.	31.6	27.3	37.7	49.4
LS-960	-	36.9	32.7	42.7	58.3
	in-dom.	24.4	24.6	33.5	45
SB+FSH	-	35.7	31.6	37.0	45.3
	in-dom.	28.6	26.6	32.5	41.0
Joint	-	23.6	19.2	25.5	35.0
	in-dom.	17.9	16.1	21.9	31.4
	CC	20.6	15.8	21.7	31.2
Joint + finetune RV-1h	-	22.5	18.4	23.6	34.3
	in-dom.	16.7	15.2	21.2	30.3
	CC	19.5	15.0	20.9	30.1
Joint + finetune RV-10h	-	20.8	17.1	23.4	33.0
	in-dom.	15.7	14.6	20.5	29.8
	CC	18.5	14.1	20.2	29.5
Joint + finetune RV-100h	-	18.9	15.5	21.2	31.4
	in-dom.	14.3	13.3	18.7	28.2
	CC	16.8	12.9	18.2	27.7

4. Conclusion

We studied transfer across five public datasets, as well as transfer to out-of-domain, real-world audio data, for a single acoustic model architecture based on Transformers and with n-gram and Transformer-based language models for decoding. We showed that no single validation or test set from public datasets is sufficient to measure transfer to other public datasets or to real-world audio data. Our results suggests that ASR researchers interested in producing transferable acoustic models should, at the very least, report results on SwitchBoard, CommonVoice, and TED-LIUM (v3). Finally, we provided a recipe for a community-reproducible robust ASR model, which can be trained with a couple of public audio datasets, and a language model built on Common Crawl.

5. References

[1] P. Ghahremani, V. Manohar, H. Hadian, D. Povey, and S. Khudanpur, "Investigation of transfer learning for ASR using LF-MMI trained neural networks," in *ASRU*, 2017.

[2] V. Manohar, D. Povey, and S. Khudanpur, "JHU Kaldi system for

Arabic MGB-3 ASR challenge using diarization, audio-transcript alignment and transfer learning," in *ASRU*, 2017.

[3] T. Asami, R. Masumura, Y. Yamaguchi, H. Masataki, and Y. Aono, "Domain adaptation of dnn acoustic models using knowledge distillation," in *ICASSP*, 2017.

[4] J. Kunze, L. Kirsch, I. Kurenkov, A. Krug, J. Johannsmeier, and S. Stober, "Transfer learning for speech recognition on a budget," in *ACL Workshop on Representation Learning for NLP*, 2017.

[5] R. Collobert, C. Puhrsch, and G. Synnaeve, "Wav2letter: an end-to-end convnet-based speech recognition system," *arXiv preprint arXiv:1609.03193*, 2016.

[6] Szymański et al., "Wer we are and wer we think we are," *arXiv preprint arXiv:2010.03432*, 2020.

[7] D. Amodei et al., "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin," in *ICML*, 2016.

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

[9] John Godfrey and Edward Holliman, "Switchboard-1 release 2 LDC97S62," *Philadelphia: LDC*, 1993.

[10] Christopher Cieri, , David Graff, Owen Kimball, Dave Miller, and Kevin Walker, "Fisher english training speech parts 1 and 2 transcripts LDC200{4,5}T19," *Philadelphia: LDC*, 2004, 2005.

[11] Christopher Cieri, David Miller, and Kevin Walker, "Fisher english training speech parts 1 and 2 LDC200{4,5}S13," *Philadelphia: LDC*, 2004, 2005.

[12] Jonathan G. Fiscus et al., "2003 nist rich transcription evaluation data LDC2007S10," *Web Download. Philadelphia: LDC*, 2007.

[13] LDC et al., "2000 hub5 english evaluation speech LDC2002S09 and transcripts LDC2002T43," *Web Download. Philadelphia: LDC*, 2002.

[14] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, et al., "The Kaldi speech recognition toolkit," in *ASRU*, 2011.

[15] J. Garofolo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) complete LDC93S6A," *Web Download. Philadelphia: LDC*, 1993.

[16] LDC and NIST Multimodal Information Group, "CSR-II (WSJ1) complete LDC94S13A," *Web Download. Philadelphia: LDC*, 1994.

[17] P. C. Woodland, J. J. Odell, V. Valtchev, and S. J. Young, "Large vocabulary continuous speech recognition using HTK," in *ICASSP*, 1994.

[18] R. Ardila, , M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *LREC*, 2020.

[19] F. Hernandez et al., "TED-LIUM 3: twice as much data and corpus repartition for experiments on speaker adaptation," in *SPECOM*, 2018.

[20] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, 1949.

[21] H. Hadian, H. Sameti, D. Povey, and S. Khudanpur, "End-to-end speech recognition using lattice-free MMI," in *Interspeech*, 2018.

[22] W. Zhou, W. Michel, K. Irie, M. Kitza, R. Schlüter, and H. Ney, "The RWTH ASR system for TED-LIUM release 2: Improving hybrid HMM with specaugment," in *ICASSP*, 2020.

[23] Anmol Gulati et al., "Conformer: Convolution-augmented transformer for speech recognition," in *Interspeech*, 2020.

[24] K. J. Han, A. Chandrashekaran, J. Kim, and I. Lane, "The CAPIO 2017 conversational speech recognition system," *arXiv preprint arXiv:1801.00059*, 2017.

[25] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006.

[26] G. Synnaeve, , Q. Xu, J. Kahn, T. Likhomanenko, E. Grave, V. Pratap, A. Sriram, V. Liptchinsky, and R. Collobert, "End-to-end ASR: from supervised to semi-supervised learning with modern architectures," *arXiv preprint arXiv:1911.08460*, 2019.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[28] A. Fan, E. Grave, and A. Joulin, "Reducing transformer depth on demand with structured dropout," in *ICML*, 2020.

[29] 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.

[30] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of machine learning research*, vol. 12, 2011.

[31] K. Heafield, "KenLM: Faster and smaller language model queries," in *Proceedings of the sixth workshop on statistical machine translation*. Association for Computational Linguistics, 2011.

[32] Alexei Baevski and Michael Auli, "Adaptive input representations for neural language modeling," in *International Conference on Learning Representations*, 2019.

[33] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Édouard Grave, "Ccnet: Extracting high quality monolingual datasets from web crawl data," in *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 4003–4012.