Interspeech 2022
18-22 September 2022, Incheon, Korea

UserLibri: A Dataset for ASR Personalization Using Only Text

Theresa Breiner, Swaroop Ramaswamy, Ehsan Variani, Shefali Garg, Rajiv Mathews, Khe Chai Sim, Kilol Gupta, Mingqing Chen, Lara McConnaughey

Google Inc.

{tbreiner,swaroopram,variani,shefgarg,mathews,khechai,
kilolgupta,mingqing,laramcc}@google.com

Abstract

Personalization of speech models on mobile devices (on-device personalization) is an active area of research, but more often than not, mobile devices have more text-only data than paired audio-text data. We explore training a personalized language model on text-only data, used during inference to improve speech recognition performance for that user. We experiment on a user-clustered LibriSpeech corpus, supplemented with personalized text-only data for each user from Project Gutenberg. We release this User-Specific LibriSpeech (UserLibri) dataset to aid future personalization research. LibriSpeech audio-transcript pairs are grouped into 55 users from the test-clean dataset and 52 users from test-other. We are able to lower the average word error rate per user across both sets in streaming and nonstreaming models, including an improvement of 2.5 for the harder set of test-other users when streaming.

Index Terms: speech recognition, personalization, language modeling

1. Introduction

End-to-end (E2E) automatic speech recognition (ASR) systems can now run inference entirely on-device, thereby having important implications for latency, reliability and privacy [1, 2]. The models themselves, which are typically trained server-side on large amounts of audio data, can perform quite well during inference on-device if the target domain is a reasonable match to the training data. However, they may perform poorly on rare words, or in applications where there is not a wealth of similar labeled training data available server-side. New research techniques are being developed to address this, such as by leveraging an external language model (LM) [3].

On-device learning techniques such as Federated Learning (FL) and personalization (p13n[1]) can train models on the actual user data on-device without ever sending the raw user data to a central server. While a few active users of speech recognition services on mobile devices may have significant amounts of audio data on their device ($\approx 40\%$ of the U.S. population uses a digital voice assistant at least once a month according to Statista [4]), users tend to interact more with their mobile devices using virtual keyboards rather than speech recognition. Further, typical supervised training of ASR models requires ground-truth text labels for each utterance, which may not be available on mobile devices to use for p13n. However, by utilizing more of the on-device text data to learn user-preferred words or phrases, we can enable the on-device ASR models to possibly recognize them correctly the first time without access to any audio training examples containing that information.

[1]In this paper, we use "p13n" to mean both "personalization" and "personalized", as in "p13n LM".

In this work, we first generate a simulation dataset of ~100 users, based on the existing LibriSpeech dataset. We then explore the use of shallow fusion with personalized LMs to improve average word error rate (WER) per user on this data.

2. Our Contributions

We present the UserLibri dataset[2], a re-formatting of LibriSpeech containing paired audio-transcripts and extra text-only data, that can be useful in a variety of experiments. We hope that this dataset will help futher personalization research as to our knowledge there is no existing open-source dataset containing both user-specific audio and additional text-only personalized data. We report our initial experiments leveraging this dataset, which aim to improve WER of an ASR system via shallow fusion with personalized (p13n[1]) LMs. Our top findings are:

- p13n LM fusion can greatly improve WER, especially for certain users, and is better than non-p13n LM fusion,
- streaming ASR models, which perform worse than non-streaming models, see greater absolute WER improvements with p13n LM fusion than non-streaming,
- larger LMs perform better than smaller ones,
- the number of LM training examples per user affects fusion performance, but even fine tuning a p13n LM on only 500 examples per user can beat the baseline,
- p13n speech models can be combined with p13n LM fusion for even better results.

Background and related work motivating our research is in Section 3. Dataset creation details are in Section 4, followed by our models and experiments in Section 5. We also include some specific wins and losses in Section 5.5.

3. Background

Machine learning models frequently have to handle the question of bias, where model performance on certain classes of data or certain users may be much worse than others, even despite excellent metrics on large test datasets [5, 6]. Previous work has improved ASR performance for underrepresented classes of users via ASR p13n, by fine tuning a speaker-independent base model on a single speaker's audio examples [7, 8].

One specific type of data that can be difficult for natural language models is less common proper names or other tail words, which may not occur frequently in the training data and occur disproportionately depending on the user [9]. Previous work has used shallow fusion [10] to leverage larger text corpora and combined it with biasing techniques that can target improvements for contexts frequently followed by proper nouns, such

[2]https://www.kaggle.com/datasets/google/userlibri

10.21437/Interspeech.2022-10915

as "call" and "message" [11], or even without such context constraints [12]. Recent work has specifically explored personalizing end-to-end ASR models on-device by incorporating a biased LM on-the-fly, or by model fine-tuning on synthesized speech [13]. These approaches only require the target proper nouns in text form, possibly in a user-provided list. By contrast, we aim to improve ASR performance via personalization of an RNN LM requiring no active input from the user at all, and no real or synthesized user-specific audio data.

There are several ways of using LMs to improve ASR. One can interpolate the scores with the ASR model score, use them during the beam search, or use the LM as a rescoring model after the beam search. In this work, we choose to always interpolate the language model scores prior to the beam search, which is known as shallow fusion [10].

Since one of the advantages of shallow fusion is the ability to leverage a much larger text dataset to improve an ASR model, most literature explores fusing with large LMs trained on the order of a million examples [14]. However, in the on-device LM p13n setting, both training data and model size are much more limited. Streaming models are preferred in the on-device setting due to their low latency at inference, despite their generally worse performance compared to nonstreaming models, which can analyze the entire input sequence before deciding the output. Recent work showed that shallow fusion of an external RNN LM can be successful with streaming models, again training on hundreds of millions or even billions of examples [15].

Related work in personalizing LMs has often focused on the dilemma of mismatched domains and sparseness of personalized text data, and incorporates adaptation or transfer learning techniques during p13n [16]. While our simulation datasets all draw from matching text domains, the work in this paper introduces a new challenge of evaluating p13n LMs not on text-only tasks, where optimizing LM perplexity may suffice, but on ASR. To our knowledge, there is limited literature discussing this topic in the realm of p13n, although there are many works on optimizing or adapting LMs for speech tasks [17, 18, 19].

4. User Specific LibriSpeech Dataset

LibriSpeech [20] is a widely-used dataset containing 970 hours of paired audio-transcript data, which are recordings of various speakers reading aloud from Project Gutenberg e-books [21]. The Project Gutenberg (PG) raw book text data which served as the source for these recordings can also be downloaded from the LibriSpeech resources at https://www.openslr.org/12. The raw metadata files are also available which map audio examples to the corresponding PG source book texts.

The audio recordings are based on only a subset of text from each book, and the rest of the book text will contain similar vocabulary, character names, and writing style as the recordings. Therefore, we can create a dataset for a "user" containing the original paired audio-transcripts from a book as well as additional text-only data from the remainder of the book, which will match the user's domain. This multi-user dataset can be used for a variety of personalization applications beyond shallow fusion with personalized LMs, such as data augmentation studies using Text-to-Speech to create more audio training examples for a user [22]. While the LibriSpeech audio data is stored by speaker ID and chapter ID, we combine audio examples from chapters of the same book, read by the same speaker, into a single "user" dataset. This results in more data per user, with an average of 52 audio examples per user (see more in Table 1).

To process the raw PG book text files into LM train sets of sentences, the text file encodings are standardized and boilerplate is removed. Then, the book text is broken into sentences, ignoring newlines mid-sentence that exist for readability in the raw data. For each sentence, we remove word-leading or trailing punctuation and sentence-leading or trailing whitespace, and uppercase the ascii characters. We then discard any sentences containing a sequence of 80% of the sentence tokens where the same sequence can be found in the test audio examples which came from the same book.

Table 1: *UserLibri dataset utterance & p13n LM text metadata*

Metadata	Test-Other	Test-Clean
# Users	52	55
Audio-Transcript Utts		
Avg. # per User	56.5	47.1
Median # per User	52	46
# Users with $\geq$10	50	52
Max # for 1 User	144	108
LM Train Text Sentences		
Total #	444,520	377,049
Avg. # per User	8,548	6,855
Median # per User	5,299	3,750
# Users with $\geq$3k	43	38
Max # for 1 User	38,498	54,306

In the UserLibri dataset, we only consider the users found in the LibriSpeech test-clean (considered easier for ASR) and test-other (considered harder/more noisy for ASR) datasets, as this allows us to use the LibriSpeech train and dev sets to train the ASR model that we use for our fusion experiments without training on any of the test speakers' data. We note that although LibriSpeech's train, dev, and test audio sets have no overlap in speakers, there are a few speakers in each set who read chapters from the same book, so the ASR model does see some text snippets from these books, read by different speakers. See Appendix **??** for further details on the dataset generation process.

While we focus on the English LibriSpeech here, the Multilingual LibriSpeech could be similarly processed in the future.

5. Experiments

We use the UserLibri dataset to explore whether shallow fusion with p13n LMs can improve per-user WER, as well as the effects of fusing with streaming ASR models vs. nonstreaming, and using different LM sizes trained on various amounts of data. We then combine p13n LM fusion with a p13n ASR model.

5.1. Models

We train 86M parameter Conformer Hybrid Autoregressive Transducer (HAT) models [23] on the 960 hours of audio train data in LibriSpeech. The audio data is preprocessed similarly to [24], by extracting 80-channel filterbanks features computed from a 25ms window with a stride of 10ms. We use SpecAugment [25] with mask parameter ($F = 27$), and ten time masks with maximum time-mask ratio (p_S = 0.05), where the maximum-size of the time mask is set to p_S times the length of the utterance. Our models consist of 12 encoder layers with dimension 512, 4 attention heads, a convolution kernel size of 32, and a HAT decoder like in [23] with a single RNN layer of dimension 640. Each label is embedded with 128 dimen-

sions, and inputs are tokenized with a 1k Word-Piece Model (WPM) trained on the LibriSpeech text-only data. The models are trained with Adam [26] and use GroupNorm [27].

We train two Conformer HAT models, one with streaming where the right context is 0 (no lookahead) which uses causal convolution and local self attention, and one nonstreaming with multi-headed attention. To make sure that the encoder is streaming, we remove all the sub-architecture components which were benefiting from right context. Namely, we remove the convolution sub-sampling layer and also force the stacking layers to only stack within the left context.

For both models, we always use time-synchronous beam search during decoding, resulting in slightly different baseline WERs than similar studies like [24], which uses label-synchronous beam search and a right context of 1 frame.

We experiment with three RNN LM sizes with LSTM cells, all using the same 1024 WPM as the Conformer. The 3M parameter model uses 192 embeddings and 1 RNN layer of size 670. The 10M and 25M models use 384 embeddings and an RNN of size 1340, with 1 and 2 layers respectively. We train general non-p13n LMs on the LibriSpeech LM text data (40M examples)[3] using Adam with a learning rate of 1e-4, and batch size 4096. We then fine tune all weights for 1k steps on the user's p13n LM data, with batch size 32.

5.2. Room for Improvement with Personalization

We ran hyperparameter sweeps over the ASR models above to select the best top-K and beam width for the beam search, and the best smoothing temperature for the baseline models without shallow fusion (baseline-1 or BL1). Additionally, we swept over the HAT decoding hyperparameters from [23] - the internal LM weight, the external LM weight, and internal LM smoothing temperature - for the same models fused with a 10M general LM, trained on the entire LibriSpeech LM text corpus (BL2).

Table 2: *Baseline WERs, for test-clean (CL) and test-other (OT), without p13n LM fusion. Average WERs per user (with 95% CIs) are higher than the WER on the full set of users.*

Model	Set	Full Set WER	Average WER per User
Streaming			
No Fusion (BL1)	CL	5.8	6.0 [5.3, 6.7]
	OT	10.4	11.2 [9.6, 13.0]
Gen. 10M LM (BL2)	CL	5.3	**5.4** [**4.8**, **6.0**]
	OT	9.0	**9.7** [**8.3**, **11.1**]
Nonstreaming			
No Fusion (BL1)	CL	2.4	2.5 [2.1, 2.8]
	OT	5.8	6.8 [5.6, 8.1]
Gen. 10M LM (BL2)	CL	2.1	**2.1** [**1.8**, **2.5**]
	OT	5.0	**5.8** [**4.8**, **6.9**]

With the best hyperparameters for the streaming and the nonstreaming models, both with and without shallow fusion using this non-p13n LM, we get the WERs reported in Table 2. 95% confidence intervals (CIs) for averages throughout the paper are calculated using the bootstrap technique [28], sampling the per-user WERs with replacement 10k times.

Our models achieve similar results on the full LibriSpeech test-other and test-clean sets compared to other studies such as [24] (note the differences as described in Section 5.1). However, average per-user WERs are worse than WERs of the full set, especially in the test-other users, and even in the nonstreaming model despite its generally lower WERs. We hope to improve performance for the high-WER users (Figure 1).

Fusion with the general LM (BL2) improves performance, but still shows higher per-user WER averages than on the full sets. We will now describe how further per-user improvements can be made using p13n LMs.

[3] See Appendix **??** for alternative general LM results.

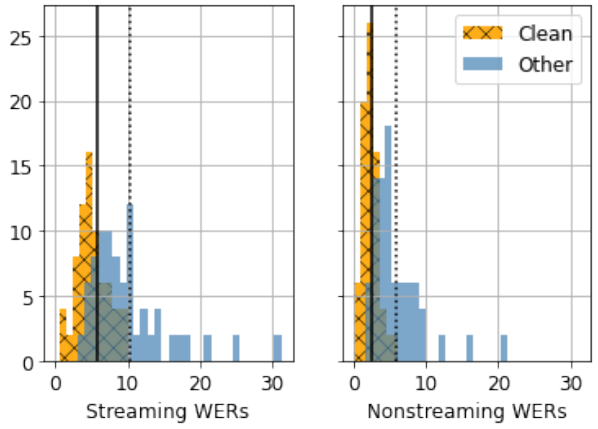

Figure 1: *Histogram of per-user baseline WERs (no fusion). Baseline WER for the full test-clean is shown as a solid vertical line, and for test-other as a dotted vertical line. Many users have much higher baseline WERs, and p13n may help.*

5.3. Results

Table 3: *Average WERs and 95% CIs from fusing with general (Gen) and p13n LMs of different sizes, for streaming (Str.) and nonstreaming (Nonstr.). BL1 is the baseline with no fusion.*

Model	Test-Clean	Test-Other	All Users
Str.			
BL1	6.0 [5.3, 6.7]	11.2 [9.6, 13.0]	8.5 [7.5, 9.6]
3M Gen	5.8 [5.1, 6.4]	10.5 [9.0, 12.0]	8.1 [7.2, 9.0]
3M p13n	5.6 [4.9, 6.2]	10.0 [8.4, 11.6]	7.7 [6.8, 8.7]
10M Gen	5.4 [4.8, 6.0]	9.7 [8.3, 11.1]	7.5 [6.7, 8.4]
10M p13n	5.4 [4.8, 6.1]	9.4 [8.0, 10.9]	7.4 [6.5, 8.3]
25M Gen	**5.2** [**4.7**, **5.9**]	9.1 [7.8, 10.4]	7.1 [6.3, 8.0]
25M p13n	**5.2** [**4.5**, **5.9**]	**8.7** [**7.4**, **10.2**]	**6.9** [**6.1**, **7.8**]
Nonstr.			
BL1	2.5 [2.1, 2.8]	6.8 [5.6, 8.1]	4.5 [3.8, 5.4]
3M Gen	2.3 [2.1, 2.7]	6.4 [5.3, 7.6]	4.3 [3.6, 5.1]
3M p13n	2.0 [1.7, 2.4]	5.8 [4.8, 7.0]	3.9 [3.2, 4.5]
10M Gen	2.1 [1.8, 2.5]	5.8 [4.8, 6.9]	3.9 [3.3, 4.6]
10M p13n	**1.9** [**1.6**, **2.3**]	5.3 [4.3, 6.4]	3.6 [3.0, 4.2]
25M Gen	2.0 [1.7, 2.3]	5.5 [4.6, 6.6]	3.7 [3.2, 4.4]
25M p13n	**1.9** [**1.6**, **2.3**]	**4.6** [**3.8**, **5.6**]	**3.2** [**2.7**, **3.8**]

We sweep external LM weights of [0.15, 0.22, 0.36, 0.45, 0.55] for p13n LM shallow fusion with both the streaming and nonstreaming ASR models and notice that values above 0.22 quickly perform worse than the baseline, and for most users, 0.15 performs better than 0.22. We fuse using LM weight 0.15

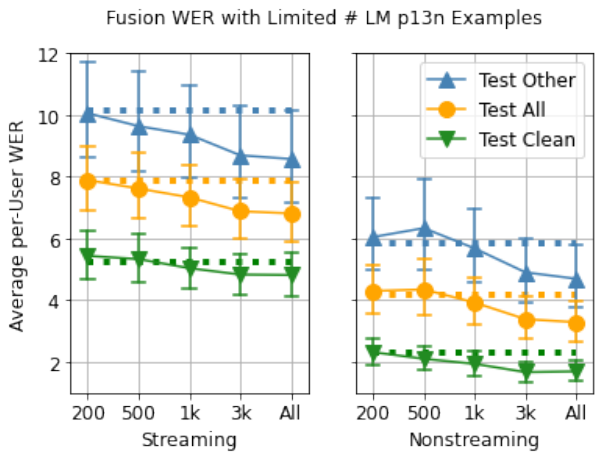

Figure 2: *Average WER per user (and 95% CI) for p13n 10M LM fusion on limited user LM train sets, for Test-Clean, Test-Other and all users combined, for the 81 users with at least 3k LM train examples. Horizontal dotted lines show the baseline WERs (no fusion) for each group.*

for all users and leave the question of p13n acceptance criteria, or if/how to fuse per user, for future work. Results can be found in Table 3, with p13n LMs outperforming the general LMs in almost every category. While the streaming model shows larger effects, even the nonstreaming model benefits from p13n LM fusion. Fusing with the 3M model already surpasses the baselines, with greater gains as model size increases. Per-User WER histograms are included in Appendix **??**.

We also experiment with fine tuning the p13n LMs on varying amounts of user LM examples: train sets of 200, 500, 1k, and 3k examples per user (see Appendix **??**), which we compared to fine tuning on the full amount of LM data we have available for that user, only for the 81 users that have at least 3k examples total. In Figure 2, we see that training with 3k examples performs almost as well as using all available data, and we can beat the baselines as long as we personalize on at least 500 examples for streaming models and 1k for nonstreaming.

5.4. Interaction with Speech Personalization

As mentioned in Section 3, speech p13n by fine tuning full or partial models on-device can greatly improve the recall of proper nouns such as named entities [13]. We briefly try combining speech p13n and p13n LM fusion[4].

We fine tune the joint layer of the streaming ASR model for each user via 5-fold cross validation, for the 102 users with at least 10 audio-transcript pairs, and average the per-fold WERs. Speech p13n alone improves upon the baseline, with non-p13n speech + p13n LM fusion performing a little better. Combining the two approaches gets the best results (Table 4).

5.5. Wins and Losses

If we examine specific cases where the p13n LM is able to correctly predict examples that the baseline or general LM predicted incorrectly (Table 5), we see that proper nouns seen more frequently in the p13n LM training data can be a source of the win, but there are also some losses due to over-predicting words seen in the p13n LM data. Combining speech and LM p13n may allow wins on tail words while avoiding losses on small common words (Table 6). See Appendix **??** for more.

[4]We don't sweep any hyperparameters for these speech experiments.

Table 4: *Average WERs from personalizing the streaming ASR model on each user's test utterances via 5-fold cross validation, with and without p13n LM fusion. No p13n means no speech p13n or p13n LM fusion (BL1). Only the 102 users with at least 10 audio-transcript pairs are included. Reports 95% CIs.*

Model	Test-Clean	Test-Other	All Users
No p13n	6.0 [5.3, 6.6]	11.3 [9.6, 13.1]	8.6 [7.6, 9.7]
+10M LM	5.6 [5.0, 6.3]	9.3 [7.9, 10.8]	7.4 [6.6, 8.3]
p13n ASR	5.7 [5.1, 6.3]	10.4 [9.0, 12.1]	8.0 [7.1, 9.0]
+10M LM	**5.4** [**4.8**, **6.0**]	**8.7** [**7.5**, **10.0**]	**7.0** [**6.2**, **7.8**]

Table 5: *Example predictions. p13n LM fusion can win (W) on names, but may lose (L) if a test word rarely appears in the rest of the p13n LM data, but a similar word appears many times.*

W/L	Baseline	General LM	P13n LM	Count in p13n data
W	king sharkan	king sharkan	king **sharrkan**	sharkan: 0 sharrkan: 336
W	mardock blinked	murdock blinked	**murdoch** blinked	m[a/u]rdock: 0 murdoch: 92
W	thanks is he	thanks is he	thanks **izzy**	is he: 72 izzy: 155
L	on the navel	on the navel	on the **naval**	navel: 0 naval: 2
L	tied to a woman	tied to a woman	**tide** to a woman	tied: 0 tide: 4

Table 6: *Predictions on one utterance for speech p13n combined with LM p13n, which may allow similar tail word wins from Table 5 while avoiding losses on smaller common words.*

Experiment	Prediction
Baseline	ten years farewell vintage is none
General LM	ten years farewell vintage is done
p13n LM only	panniers farewell vintage is none
Speech p13n only	panniers farewell vintage is none
Speech & LM p13n	**panniers** farewell vintage is **done**

6. Conclusions

We share our new personalization simulation dataset, UserLibri, based on LibriSpeech and useful for a variety of personalization research. We demonstrate a use case where we are able to improve average per-user WER in both streaming and nonstreaming models, bringing down streaming WER on the hardest group of users by 1.2 by performing shallow fusion with personalized LMs of only 3M parameters, or an improvement of 2.5 with 25M LMs. We show that the technique can still work with limited p13n LM examples and can be combined with speech p13n for even better results.

7. References

[1] Y. He, T. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, Q. Liang, D. Bhatia, Y. Shangguan, B. Li, G. Pundak, K. C. Sim, T. Bagby, S. yiin Chang, K. Rao, and A. Gruenstein, "Streaming end-to-end speech recognition for mobile devices," 2019. [Online]. Available: https://arxiv.org/abs/1811.06621

[2] T. N. Sainath, Y. He, B. Li, A. Narayanan, R. Pang, A. Bruguier, S. Chang, W. Li, R. Alvarez, Z. Chen, C. Chiu, D. Garcia, A. Gruenstein, K. Hu, M. Jin, A. Kannan, Q. Liang, I. McGraw, C. Peyser, R. Prabhavalkar, G. Pundak, D. Rybach, Y. Shangguan, Y. Sheth, T. Strohman, M. Visontai, Y. Wu, Y. Zhang, and D. Zhao, "A streaming on-device end-to-end model surpassing server-side conventional model quality and latency," *CoRR*, vol. abs/2003.12710, 2020. [Online]. Available: https://arxiv.org/abs/2003.12710

[3] T. N. Sainath, Y. R. He, A. Narayanan, R. Botros, R. Pang, D. J. Rybach, C. Allauzen, E. Variani, J. Qin, Q.-N. Le-The, A. Gruenstein, A. Gulati, B. Li, C. Peyser, C.-C. Chiu, D. A. Caseiro, E. Guzman, I. C. McGraw, J. Yu, M. D. Riley, P. Rondon, Q. Liang, S. Mavandadi, S. yiin Chang, T. D. Strohman, W. R. Huang, W. Li, Y. Wu, and Y. Zhang, "An efficient streaming non-recurrent on-device end-to-end model with improvements to rare-word modeling," 2021.

[4] Statista.com, "Number of voice assistant users int he United States from 2017 to 2022," 2022, accessed March 16, 2022. [Online]. Available: https://www.statista.com/statistics/1029573/us-voice-assistant-users/

[5] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *ACM Comput. Surv.*, vol. 54, no. 6, jul 2021. [Online]. Available: https://doi.org/10.1145/3457607

[6] J. L. Martin, "Spoken corpora data, automatic speech recognition, and bias against african american language: The case of habitual 'be'," in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, ser. FAccT '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 284. [Online]. Available: https://doi.org/10.1145/3442188.3445893

[7] J. R. Green, B. MacDonald, P.-P. Jiang, J. Cattiau, R. Heywood, R. Cave, K. Seaver, M. Ladewig, J. Tobin, M. Brenner, P. Q. Nelson, and K. Tomanek, "Automatic speech recognition of disordered speech: Personalized models outperforming human listeners on short phrases," 2021.

[8] K. Tomanek, F. Beaufays, J. Cattiau, A. Chandorkar, and K. Sim, "On-device personalization of automatic speech recognition models for disordered speech," 06 2021. [Online]. Available: https://arxiv.org/abs/2106.10259

[9] P. Aleksic, C. Allauzen, D. Elson, A. Kracun, D. M. Casado, and P. J. Moreno, "Improved recognition of contact names in voice commands," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5172–5175.

[10] Ç. Gülçehre, O. Firat, K. Xu, K. Cho, L. Barrault, H. Lin, F. Bougares, H. Schwenk, and Y. Bengio, "On using monolingual corpora in neural machine translation," *CoRR*, vol. abs/1503.03535, 2015. [Online]. Available: http://arxiv.org/abs/1503.03535

[11] D. Zhao, T. N. Sainath, D. Rybach, P. Rondon, D. Bhatia, B. Li, and R. Pang, "Shallow-fusion end-to-end contextual biasing," in *INTERSPEECH*, 2019.

[12] D. Le, M. Jain, G. Keren, S. Kim, Y. Shi, J. Mahadeokar, J. Chan, Y. Shangguan, C. Fuegen, O. Kalinli, Y. Saraf, and M. L. Seltzer, "Contextualized streaming end-to-end speech recognition with trie-based deep biasing and shallow fusion," *CoRR*, vol. abs/2104.02194, 2021. [Online]. Available: https://arxiv.org/abs/2104.02194

[13] K. Sim, L. Johnson, G. Motta, L. Zhou, F. Beaufays, A. Benard, D. Guliani, A. Kabel, N. Khare, T. Lucassen, P. Zadrazil, and H. Zhang, "Personalization of end-to-end speech recognition on mobile devices for named entities," 12 2019, pp. 23–30.

[14] A. Kannan, Y. Wu, P. Nguyen, T. N. Sainath, Z. Chen, and R. Prabhavalkar, "An analysis of incorporating an external language model into a sequence-to-sequence model," *ICASSP*, 2018. [Online]. Available: https://arxiv.org/pdf/1712.01996.pdf

[15] R. Cabrera, X. Liu, M. Ghodsi, Z. Matteson, E. Weinstein, and A. Kannan, "Language model fusion for streaming end to end speech recognition," *CoRR*, vol. abs/2104.04487, 2021. [Online]. Available: https://arxiv.org/abs/2104.04487

[16] T. H. Wen, A. Heidel, H.-Y. Lee, Y. Tsao, and L.-S. Lee, "Recurrent neural network based language model personalization by social network crowdsourcing," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pp. 2703–2707, 01 2013.

[17] T. Mikolov, S. Kombrink, L. Burget, J. Černocký, and S. Khudanpur, "Extensions of recurrent neural network language model," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 5528–5531.

[18] K. Li, H. Xu, Y. Wang, D. Povey, and S. Khudanpur, "Recurrent neural network language model adaptation for conversational speech recognition," 09 2018, pp. 3373–3377.

[19] S. Deena, M. Hasan, M. Doulaty, O. Saz, and T. Hain, "Recurrent neural network language model adaptation for multi-genre broadcast speech recognition and alignment," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 3, pp. 572–582, 2019.

[20] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[21] (n.d.), "Project gutenberg," retrieved October 19, 2021. [Online]. Available: www.gutenberg.org

[22] A. Rosenberg, Y. Zhang, B. Ramabhadran, Y. Jia, P. J. Moreno, Y. Wu, and Z. Wu, "Speech recognition with augmented synthesized speech," *CoRR*, vol. abs/1909.11699, 2019. [Online]. Available: http://arxiv.org/abs/1909.11699

[23] E. Variani, D. Rybach, C. Allauzen, and M. Riley, "Hybrid autoregressive transducer (hat)," 05 2020, pp. 6139–6143.

[24] A. Gulati, C.-C. Chiu, J. Qin, J. Yu, N. Parmar, R. Pang, S. Wang, W. Han, Y. Wu, Y. Zhang, and Z. Zhang, Eds., *Conformer: Convolution-augmented Transformer for Speech Recognition*, 2020.

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

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[27] Y. Wu and K. He, "Group normalization," *CoRR*, vol. abs/1803.08494, 2018. [Online]. Available: http://arxiv.org/abs/1803.08494

[28] M. Bisani and H. Ney, "Bootstrap estimates for confidence intervals in asr performance evaluation," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, 2004, pp. I–409.