Looking within the self: Investigating the Impact of Data Augmentation with Self-training on Automatic Speech Recognition for Hupa

Nitin Venkateswaran
University of Florida
venkateswaran.n@ufl.edu

Zoey Liu
University of Florida
liu.ying@ufl.edu

Abstract

We investigate the performance of state-of-the-art neural ASR systems in transcribing audio recordings for Hupa, a critically endangered language of the Hoopa Valley Tribe. We also explore the impact on ASR performance when augmenting a small dataset of gold-standard high-quality transcriptions with a) a larger dataset with transcriptions of lower quality, and b) model-generated transcriptions in a self-training approach. An evaluation of both data augmentation approaches shows that the self-training approach is competitive, producing better WER scores than models trained with no additional data and not lagging far behind models trained with additional lower quality manual transcriptions instead: the deterioration in WER score is just 4.85 points when all the additional data is used in experiments with the best performing system, Wav2Vec. These findings have encouraging implications on the use of ASR systems for transcription and language documentation efforts in the Hupa language.

1 Introduction

Automatic Speech Recognition (ASR) can assist with the manual process of transcribing audio recordings in low-resource and endangered languages, thereby facilitating language documentation efforts in these languages. With neural networks now dominating research in ASR (Baevski et al., 2020; Radford et al., 2023; Gulati et al., 2020), and with related efforts to build and release open-source neural-network based ASR frameworks (Wolf et al., 2020; Watanabe et al., 2018; Amodei et al., 2016), the possibilities for research on ASR for endangered languages have greatly increased; a researcher can now leverage one of the open-source ASR toolkits and apply it to their language of interest. Cross-lingual speech representations currently leveraged by state-of-the-art neural ASR systems (Babu et al., 2021; Conneau et al., 2020) also provide opportunities for knowledge transfer to endangered languages; commonalities across different speech representations can be leveraged to improve ASR performance.

The Hupa language from the Dene/Athabaskan language family is one such language that stands to benefit from these advances in ASR. Hupa is the ancestral language of the Hoopa Valley Tribe residing in Northern California. It is critically endangered with only a handful of first-language (L1) speakers and a number of second-language learners. Since the 1970s, the Hupa speech community has been actively engaged in documentation and reclamation work to preserve their language. ASR systems can be especially beneficial for Hupa, given that there may be low literacy levels among Hupa speakers and learners of the language who focus instead on oral proficiency; these low literacy levels may in turn hinder efforts to transcribe audio recordings.

However, the development of an ASR system for Hupa using supervised learning approaches faces a chicken-and-egg problem: high quality transcriptions are necessary to train a performant ASR system which can be leveraged in a manual transcription process to produce high quality transcriptions. Given the challenges with producing additional manual annotations, data augmentation approaches must instead be relied upon to generate the necessary data to train an ASR system.

In this study, we explore the efficacy of different ASR systems for Hupa, coupled with self-training, a data augmentation method that is favored in the literature due to its simplicity and elegance (Charniak, 1997; Zhang et al., 2022b). The general idea is to apply a trained model to unlabeled data, then combine the automatically annotated data with existing gold-standard training data to build a new model, to see whether the addition of model-generated annotations is helpful towards model performance. With this approach, here we seek to investigate if model-generated au-

Proceedings of the The Seventh Workshop on the Use of Computational Methods in the Study of Endangered Languages, pages 58–66
March 21-22, 2024 ©2024 Association for Computational Linguistics

dio transcripts can be leveraged to improve acoustic model performance on gold-standard data with *high-quality* manual transcriptions. In addition, we compare the self-training approach to an alternative method, where we swap out the additional machine-produced transcripts in the training data with human-annotated data that has overall *lower transcription quality* (Section 3).

2 Related Work

ASR for endangered languages A number of studies use the popular ASR toolkit Kaldi (Povey et al., 2011) to probe how far simple deep neural networks can go when situated in severely resource-constrained settings with less than 10 hours of audio training data; these studies include languages such as Seneca (Jimerson and Prud'hommeaux, 2018), Cherokee (Zhang et al., 2022a) and Hupa (Liu et al., 2022), the last of which is closest to our work. Others apply more recent end-to-end architectures: for instance, Shi et al. (2021) explore models built from ESPnet (Watanabe et al., 2018) for Yoloxóchitl Mixtec, using more than 55h of conversational speech from more than 20 speakers.

In addition, others investigate augmentation methods applied on the acoustic signals to improve ASR performance for endangered languages. These include, but are not limited to, semi-supervised training and vocal tract length perturbation (Ragni et al., 2014), elastic spectral distortion methods (Kanda et al., 2013), creation of synthetic data using voice transformation and signal distortion (Thai et al., 2019) and transfer learning with data augmentation (Thai et al., 2020).

Self-training Some recent studies have begun to apply self-training, or "pseudo-labeling", for ASR, mostly focusing on English (Xu et al., 2021, 2020; Kahn et al., 2020). A few multilingual studies exist (Khurana et al., 2022; Lugosch et al., 2022), including investigations of cross-linguistic low-resource settings (Zhang et al., 2022b). For endangered languages, Bartelds et al. (2023) apply self-training to four minority languages, Gronings, West-Frisian, Besemah, and Nasal, each with 4h of acoustic training data to start with.

In comparison to previous approaches, our work uses the Wav2Vec 2.0 framework (Baevski et al., 2020), and generates transcripts from audio data in a self-training setting while leaving the audio intact. Self-training has not been widely applied to endangered languages with the exception of Bartelds et al. (2023), who do not include Hupa in their study; the amount of gold-standard high quality acoustic training data that we start out with is also quite small in comparison with their work (more details in Section 3).

3 The Hupa ASR Dataset

The audio data for Hupa is a result of continuous linguistic fieldwork since 2005. The spoken records are provided by Mrs. Verdena Parker, an L1 speaker of the language. The audio content is composed of different genres including descriptions of historical tribal events and stories and tales narrated by Mrs. Parker. The transcriptions of the audio files have been carried out by several linguistics researchers over the years with consultation from Mrs. Parker. Each transcript follows the practical orthography developed in Golla (1996), and is time aligned with annotation tools such as ELAN (Brugman and Russel, 2004).

Transcripts go through stages of manual verification to different extents, which are necessary since the recordings come from multiple fieldwork sessions across different years and are transcribed by different researchers. Some transcripts are checked more thoroughly than others, with more checks resulting in better transcription quality. Based solely on transcription quality, we divide the audio data and their corresponding transcripts into two datasets: the "fine" and the "coarse" datasets. The fine data has approximately 1h35m of audio with thoroughly checked transcriptions, and the coarse data has around 7h37m of audio with comparatively lower transcription quality.

4 Experiments

4.1 Training and data setup

We investigate two questions: the first question is whether adding the "coarse" data to the "fine" dataset for training results in better ASR performance than using just the "fine" dataset. We create partitions of the coarse dataset with different data sizes in each partition, to investigate the effect of augmenting data of different sizes on model performance; the data is randomly sampled into each partition. Three partitions are sampled with sizes being a) the same size as the fine dataset [1x], b) three times the size of the fine dataset [3x], and c) five times the size of the fine dataset [5x], which is roughly the same size as the full coarse dataset. Data from each partition is added to the training

portion of the fine dataset (discussed below), and an ASR model is built for each partition. This overall setup lets us compare the performance of the partitions with each other as well as with a baseline consisting of just the fine dataset.

The second question is the impact on ASR performance of a self-training data augmentation approach using model-generated transcripts. An ASR model is first trained on the training portion of the fine dataset, and is then used to produce transcriptions from data in the coarse dataset, simulating the scenario when there are no transcriptions available. To facilitate comparisons, the coarse audio samples from the same partitions detailed in the previous setup are used to produce the transcriptions. Each partition containing model-generated transcriptions is then added to the training portion of the fine dataset, and an ASR model is built using each partition. This lets us directly compare results from using additional model-generated transcripts with results from using additional manually-transcribed coarse data, as well as with a baseline consisting of just the fine dataset.

To measure the quality of the model-generated transcripts as substitutes for the coarse transcripts, we calculate the word error rate (WER) scores of the generated transcripts using the coarse transcripts as references. Figure 1 in Appendix A.2 shows the box-plot distribution of WER scores: the average WER across transcripts is 38.15. To get another perspective, the Levenshtein edit distance is calculated between the coarse and model-generated transcripts to provide the number of character-level operations needed to convert one transcript to the other. The edit distance is normalized by the length of the coarse transcript, and reported as the number of operations per 100-character transcript. Figure 2 in Appendix A.2 shows the distribution of the normalized edit distance: the mean distance across all partitions is 8.58. In addition, Figure 3 in Appendix A.3 compares the distributions of token counts per transcript between the model-generated and coarse data; the distributions are quantitatively similar. Table 3 in the Appendix A.1 provides further comparisons, including type counts and average word length; while there are more types in the model-generated transcripts, the average word lengths are similar.

For evaluation, we use a random split approach: Liu et al. (2023) show random splits can yield reliable estimates of acoustic model performance, and that the WER from a single random split is comparable to that averaged from multiple random splits. Here we apply a single random split to the fine dataset, taking 20% as the test set which is used to evaluate ASR model performance across all experiments. The remaining 80% of the fine dataset is the training portion, used to train the baseline with no additional data. The coarse and model-generated transcripts from their respective partitions are added to the training portion of the fine dataset for the remaining experiments. Given the scarcity of training data, for hyper-parameter tuning we use 5-fold cross validation instead of a held-out development set. The WER and CER (character error rate) on the random test split are reported for all experiments.

4.2 Models

`Kaldi DNN` We use a hybrid fully connected deep neural network (DNN) from the Kaldi toolkit (Povey et al., 2011). Our implementations follow the default sequence training parameters from Karel's DNN recipe[1]. The model architecture has six hidden layers, each with 1024 hidden units. Previous studies (Morris et al., 2021; Morris, 2021) demonstrate that in resource-constrained scenarios, this DNN architecture is capable of yielding competitive performance compared to other neural models such as Whisper (Radford et al., 2023) and time delay neural networks (Peddinti et al., 2015). For decoding, we train trigram language models on the transcripts with Witten-Bell discounting (Witten and Bell, 1991), using the SRILM (Stolcke, 2002) toolkit. The training parameters for the DNN are present in Appendix A.5

`Wav2Vec2` We fine-tune the Wav2Vec XLS-R model with 2 billion parameters (2B), as studies have shown that models with more parameters perform better and are critical for better multi-lingual representations (Babu et al., 2021). Neither Hupa nor any of the other languages in the Athabaskan language family are among the languages used to pre-train the XLS-R models, implying that there is no transfer effect from using the pre-trained model. The XLS-R-2B architecture is based on the Wav2Vec 2.0 framework (Baevski et al., 2020). The training hyper-parameters are presented in Appendix A.5. The HuggingFace transformers library (Wolf et al., 2020) is used for the training setup. Our code for fine-tuning Wav2Vec is available.[2]

[1] https://kaldi-asr.org/doc/dnn1.html

[2] GitHub link: https://github.com/ufcompling/asr_lm.git#hupa-asr-eval

Model	Experiment Setup (Partition Size)	WER	Diff. w/ baseline	Diff. w/ best setup	CER
Kaldi DNN	Fine only (baseline)	42.79	–	(9.01)	14.72
	Fine + Coarse (1x)	39.29	3.5	(5.51)	12.54
	Fine + Coarse (3x)	35.73	7.06	(1.95)	11.10
	Fine + Coarse (5x)	**33.78**	9.01	–	10.26
	Fine + Model-generated (1x)	41.35	1.44	(7.57)	12.56
	Fine + Model-generated (3x)	38.45	4.34	(4.67)	10.74
	Fine + Model-generated (5x)	36.84	5.95	(3.06)	**9.98**
Wav2Vec2	Fine only (baseline)	29.49	–	(8.52)	6.41
	Fine + Coarse (1x)	24.87	4.62	(3.9)	5.77
	Fine + Coarse (3x)	22.25	7.24	(1.28)	5.15
	Fine + Coarse (5x)	**20.97**	8.52	–	**5.10**
	Fine + Model-generated (1x)	27.37	2.12	(6.4)	5.99
	Fine + Model-generated (3x)	26.82	2.67	(5.85)	6.16
	Fine + Model-generated (5x)	25.82	3.67	(4.85)	5.84

Table 1: WER and CER scores on the random test split, by model architecture and experiment setup. The best WER scores are from augmenting the fine dataset with the full coarse dataset (size 5x), and are highlighted in bold.

Model	Partition Size	Coarse WER	Model-generated WER	Diff
Kaldi DNN	1x	39.29	41.35	(2.06)
	3x	35.73	38.45	(2.72)
	5x	33.78	36.84	(3.06)
Wav2Vec2	1x	24.87	27.37	(2.5)
	3x	22.25	26.82	(4.57)
	5x	20.97	25.82	(4.85)

Table 2: Comparison of WER results on the random test split using additional coarse versus model-generated transcripts, across partition sizes and model architectures.

5 Results

It is clear from Table 1 that using any amount of additional data, whether from manually generated coarse transcripts or model-generated ones, improves the WER score over the baseline of using no additional data. Moreover, for both transcript types, using data from larger partition sizes leads to better acoustic model performance. These results are consistent across both Kaldi and Wav2Vec2. The best models are obtained by training on all the coarse transcripts from the size 5x partition together with the fine ones.

Comparing the two model architectures, Wav2Vec2 outperforms Kaldi across all experiments in terms of absolute WER. Both architectures are able to utilize both manually-transcribed and model-generated data to achieve improvements in WER, though the gains are slightly bigger with the former. Interestingly, Kaldi seems to better utilize the model-generated transcripts; looking at the best score with self-training normalized by the baseline, Kaldi shows an improvement of 13.91 percentage points over the baseline compared to 12.44 points for Wav2Vec2. These numbers suggest that Wav2Vec2 is possibly more sensitive to the noise present in model-generated transcripts.

While a self-training setup with model-generated transcripts under-performs versus training with additional coarse transcripts of the same size (Table 2), the differences in scores are not too large. The worst score difference is just 4.85 points for Wav2Vec2 in the size 5x partition when all the model-generated transcripts are used. Interestingly, the 1x partition results in a worsening of just 2.5 points, suggesting that it is possible to use smaller sets of model-generated transcripts to build an ASR system in the absence of any manual transcriptions. These findings suggest that it may be possible to effectively use self-trained ASR models for Hupa without needing large amounts of model-generated transcripts, which is a very encouraging find.

Lastly, the impact of using additional data from self-training versus manual annotation can be viewed from the perspective of an investigation of word types present in the transcriptions of each approach. Specifically, we look at the number of new word types introduced by each approach that are not present in the fine dataset; the coarse dataset from the 5x partition contains 5,521 new types not present in the fine dataset, versus 8,487 in the model-generated transcriptions. Given the similar number of tokens between the two, the model-generated transcripts have a higher type-token percentage of 17.71 when considering only new types

(cf. 11.42 in the coarse transcripts), which seems to correlate with higher WER scores. However, of the 5,521 new types in the coarse transcripts, 2,516 (46%) are also found in the model-generated transcripts. This has implications for new vocabulary discovery in language documentation efforts for Hupa, as up to 46% of new words in the coarse dataset can be discovered through ASR transcriptions instead of solely through manual effort.

6 Conclusion and Future Work

We find that a self-training approach for Hupa ASR is able to produce transcriptions of better quality than one using no additional training data, as seen in the 3.67 point improvement in WER score; moreover, it does not fall far behind when compared to the best-performing setup of using all the additional human-transcribed coarse data (a 4.85 WER difference). Moreover, the availability of manually verified low-quality transcripts in the best performing setup should not be taken for granted; it is not uncommon, in the cases of indigenous and endangered languages, for the audio recordings to be sourced from just one speaker (see also Boulianne (2022)), and the resources needed to produce transcriptions from the audio may be very limited. With that in mind, we believe self-training to be a useful data augmentation method, at least in the initial stages of developing ASR systems for endangered languages when there is very little data available.

An important goal of developing ASR systems for endangered languages is to automate, fully or partially, the transcription of new fieldwork recordings; the automatic transcriptions can be manually corrected by speakers of the language, potentially removing the need to transcribe from scratch (Prud'hommeaux et al., 2021).

It would be interesting to study whether an ASR system trained on additional manually produced transcripts of any quality would be more beneficial to the transcription process than a system trained on additional model-generated transcripts using self-training; or in other words, can transcribers tolerate a 4.85 point degradation in WER score by using model-generated transcripts to train an ASR system, in exchange for not needing to manually produce transcriptions to improve that system? Additionally, would tools such as ELPIS (Foley et al., 2018) that take advantage of speech recognition technologies in their language documentation transcription workflows benefit from the integration of self-training or other data augmentation methods into existing pipelines? We leave these possibilities for future work.

Acknowledgements

We are grateful for the continuous support from the Hupa indigenous community. We thank Mrs. Verdena Parker for her generous and valuable input throughout the years, and Justin Spence for his work on language documentation efforts; this study has been made possible thanks to their work. In addition, we thank the anonymous reviewers for their helpful feedback.

References

Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, Jie Chen, Jingdong Chen, Zhijie Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Ke Ding, Niandong Du, Erich Elsen, Jesse Engel, Weiwei Fang, Linxi Fan, Christopher Fougner, Liang Gao, Caixia Gong, Awni Hannun, Tony Han, Lappi Vaino Johannes, Bing Jiang, Cai Ju, Billy Jun, Patrick LeGresley, Libby Lin, Junjie Liu, Yang Liu, Weigao Li, Xiangang Li, Dongpeng Ma, Sharan Narang, Andrew Ng, Sherjil Ozair, Yiping Peng, Ryan Prenger, Sheng Qian, Zongfeng Quan, Jonathan Raiman, Vinay Rao, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Kavya Srinet, Anuroop Sriram, Haiyuan Tang, Liliang Tang, Chong Wang, Jidong Wang, Kaifu Wang, Yi Wang, Zhijian Wang, Zhiqian Wang, Shuang Wu, Likai Wei, Bo Xiao, Wen Xie, Yan Xie, Dani Yogatama, Bin Yuan, Jun Zhan, and Zhenyao Zhu. 2016. Deep speech 2: End-to-end speech recognition in english and mandarin. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, page 173–182. JMLR.org.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, et al. 2021. Xls-r: Self-supervised cross-lingual speech representation learning at scale. *arXiv e-prints*, pages arXiv–2111.

Alexei Baevski, Henry Zhou, Abdel rahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *ArXiv*, abs/2006.11477.

Martijn Bartelds, Nay San, Bradley McDonnell, Dan Jurafsky, and Martijn Wieling. 2023. Making more of little data: Improving low-resource automatic speech recognition using data augmentation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 715–729, Toronto, Canada. Association for Computational Linguistics.

Gilles Boulianne. 2022. Phoneme transcription of endangered languages: an evaluation of recent ASR architectures in the single speaker scenario. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2301–2308, Dublin, Ireland. Association for Computational Linguistics.

Hennie Brugman and Albert Russel. 2004. Annotating multi-media/multi-modal resources with ELAN. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Eugene Charniak. 1997. Statistical parsing with a context-free grammar and word statistics. *AAAI/IAAI*, 2005(598-603):18.

Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdel rahman Mohamed, and Michael Auli. 2020. Unsupervised cross-lingual representation learning for speech recognition. In *Interspeech*.

Ben Foley, Josh Arnold, Rolando Coto-Solano, Gautier Durantin, T. Mark Ellison, Daan van Esch, Scott Heath, František Kratochvíl, Zara Maxwell-Smith, David Nash, Ola Olsson, Mark Richards, Nay San, Hywel Stoakes, Nick Thieberger, and Janet Wiles. 2018. Building Speech Recognition Systems for Language Documentation: The CoEDL Endangered Language Pipeline and Inference System (ELPIS). In *Proc. 6th Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU 2018)*, pages 205–209.

Victor Golla. 1996. *Hupa Language Dictionary Second Edition*. Hoopa Valley Tribe.

Anmol Gulati, Chung-Cheng Chiu, James Qin, Jiahui Yu, Niki Parmar, Ruoming Pang, Shibo Wang, Wei Han, Yonghui Wu, Yu Zhang, and Zhengdong Zhang, editors. 2020. *Conformer: Convolution-augmented Transformer for Speech Recognition*.

Robbie Jimerson and Emily Prud'hommeaux. 2018. ASR for documenting acutely under-resourced indigenous languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Jacob Kahn, Ann Lee, and Awni Hannun. 2020. Self-training for end-to-end speech recognition. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7084–7088.

Naoyuki Kanda, Ryu Takeda, and Yasunari Obuchi. 2013. Elastic spectral distortion for low resource speech recognition with deep neural networks. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 309–314.

Sameer Khurana, Antoine Laurent, and James Glass. 2022. Magic dust for cross-lingual adaptation of monolingual wav2vec-2.0. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6647–6651.

Zoey Liu, Justin Spence, and Emily Prud'hommeaux. 2022. Enhancing documentation of Hupa with automatic speech recognition. In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 187–192, Dublin, Ireland. Association for Computational Linguistics.

Zoey Liu, Justin Spence, and Emily Prud'hommeaux. 2023. Investigating data partitioning strategies for crosslinguistic low-resource ASR evaluation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 123–131, Dubrovnik, Croatia. Association for Computational Linguistics.

Loren Lugosch, Tatiana Likhomanenko, Gabriel Synnaeve, and Ronan Collobert. 2022. Pseudo-labeling for massively multilingual speech recognition. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7687–7691.

Ethan Morris. 2021. Automatic Speech Recognition for Low-Resource and Morphologically Complex Languages. Master's thesis, Rochester Institute of Technology.

Ethan Morris, Robert Jimerson, and Emily Prud'hommeaux. 2021. One size does not fit all in resource-constrained ASR. In *The Annual Conference of the International Speech Communication Association (Interspeech)*, pages 4354–4358.

Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. 2015. A time delay neural network architecture for efficient modeling of long temporal contexts. In *Proc. Interspeech 2015*, pages 3214–3218.

Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al. 2011. The Kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*, CONF. IEEE Signal Processing Society.

Emily Prud'hommeaux, Robbie Jimerson, Richard Hatcher, and Karin Michelson. 2021. Automatic speech recognition for supporting endangered language documentation. *Language Documentation and Conservation*, 15:491–513.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR.

Anton Ragni, Kate Knill, Shakti Prasad Rath, and Mark John Francis Gales. 2014. Data augmentation for low resource languages. In *Interspeech*.

Jiatong Shi, Jonathan D. Amith, Rey Castillo García, Esteban Guadalupe Sierra, Kevin Duh, and Shinji Watanabe. 2021. Leveraging end-to-end ASR for endangered language documentation: An empirical study on Yoloxóchitl Mixtec. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1134–1145, Online. Association for Computational Linguistics.

Andreas Stolcke. 2002. SRILM-an extensible language modeling toolkit. In *Seventh international conference on spoken language processing*.

Bao Thai, Robert Jimerson, Dominic Arcoraci, Emily Prud'hommeaux, and Raymond Ptucha. 2019. Synthetic data augmentation for improving low-resource ASR. In *2019 IEEE Western New York Image and Signal Processing Workshop (WNYISPW)*, pages 1–9. IEEE.

Bao Thai, Robert Jimerson, Raymond Ptucha, and Emily Prud'hommeaux. 2020. Fully convolutional ASR for less-resourced endangered languages. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 126–130, Marseille, France. European Language Resources association.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. 2018. ESPnet: End-to-End Speech Processing Toolkit. In *Proceedings of Interspeech*, pages 2207–2211.

Ian H Witten and Timothy C Bell. 1991. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *Ieee transactions on information theory*, 37(4):1085–1094.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Qiantong Xu, Alexei Baevski, Tatiana Likhomanenko, Paden Tomasello, Alexis Conneau, Ronan Collobert, Gabriel Synnaeve, and Michael Auli. 2021. Self-training and pre-training are complementary for speech recognition. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3030–3034.

Qiantong Xu, Tatiana Likhomanenko, Jacob Kahn, Awni Hannun, Gabriel Synnaeve, and Ronan Collobert. 2020. Iterative Pseudo-Labeling for Speech Recognition. In *Proc. Interspeech 2020*, pages 1006–1010.

Shiyue Zhang, Ben Frey, and Mohit Bansal. 2022a. How can NLP help revitalize endangered languages? a case study and roadmap for the Cherokee language. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1529–1541, Dublin, Ireland. Association for Computational Linguistics.

Zi-Qiang Zhang, Yan Song, Ming-Hui Wu, Xin Fang, Ian McLoughlin, and Li-Rong Dai. 2022b. Cross-lingual self-training to learn multilingual representation for low-resource speech recognition. *Circuits, Systems, and Signal Processing*, 41(12):6827–6843.

A Appendix

A.1 Partition statistics for all transcripts

Table 3 details the statistics of the training and test partitions for the different experimental setups.

A.2 WER & edit distance distributions

Figure 1 shows the box-plot distribution of WER scores between coarse and model-generated transcript types. Figure 2 shows the normalized edit distance distribution between coarse and model-generated transcript types.

A.3 Distribution of token counts

Figure 3 shows the distributions of token counts per transcript between coarse and model-generated transcript types.

A.4 Distribution of word length

Figure 4 shows the distribution of word length across all manually annotated texts.

A.5 Model training hyper-parameters

Table 4 and Table 5 show the hyper-parameters used to train Wav2Vec2 and Kaldi.

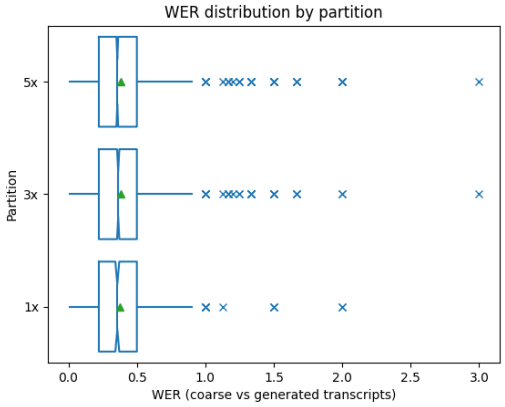

Figure 1: WER scores for coarse versus model-generated transcripts by partition. The mean WER scores for the 5x, 3x and 1x partitions are 0.3819, 0.3848, 0.3779 respectively.

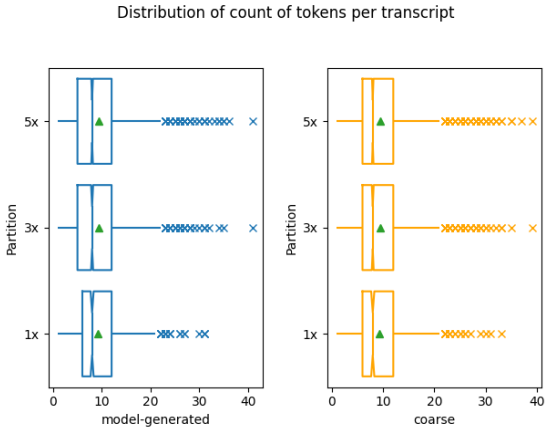

Figure 3: Distribution of token count per transcript, grouped by partition. The distributions appear quantitatively similar across model-generated and coarse transcripts.

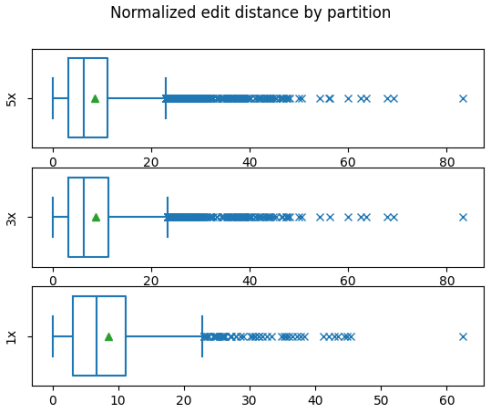

Figure 2: Distribution of normalized edit distances for the 5x, 3x and 1x partitions between the coarse transcripts and the model generated transcripts; the distances are normalized by the length of the coarse transcript and reported as edits per 100-character transcript. The mean normalized edit distances for the 5x, 3x and 1x partitions are 8.51, 8.73 and 8.47 respectively.

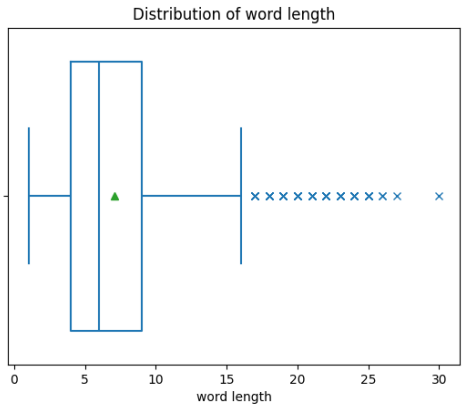

Figure 4: Distribution of word length across all manually-annotated texts. The mean word length is 7.05.

Experiment setup (Partition Size)	Duration	Token count	Type count	Avg. tokens per sentence	Avg. word length (chars)
Fine only (baseline)	1h 16m	7,438	2,028	9.15	7.22
Fine + Coarse (1x)	2h 52m	16,126	3,649	9.26	7.09
Fine + Coarse (3x)	6h 1m	32,828	5,889	9.32	7.04
Fine + Coarse (5x)	9h 12m	48,342	7,549	9.34	7.03
Fine + Model-generated (1x)	2h 52m	16,041	4,255	9.21	7.12
Fine + Model-generated (3x)	6h 1m	32,557	7,774	9.24	7.10
Fine + Model-generated (5x)	9h 12m	47,922	10,515	9.26	7.09
Test Split	19m	1,797	754	8.81	7.42

Table 3: Statistics about the train-test split across fine, coarse, and model-generated transcripts.

Parameter	Value
Number of Epochs	60
Training Batch Size	4
Evaluation Batch Size	8
Warmup Size	0 (no warmup)
Gradient Accumulation Size	2
Learning Rate	3e-5

Table 4: Parameters used to train Wav2Vec XLS-R-2B.

Parameter	Value
Hidden layers	4
Hidden dim	1024
Learning Rate	0.08

Table 5: Parameters used to train Kaldi DNN.