FINE-TUNING A SPEECH RECOGNITION MODEL ON DATA FROM THE ORD CORPUS TO AUTOMATE THE CREATION OF DAY-TO-DAY SPEECH CORPORA

CONTENTS

INTRODUCTION

Speech transcription and annotation is a time- and energy-consuming process (Bird 2021). Now that automatic speech recognition systems (henceforth ASR) are achieving close-to-human performance, it is worthwhile to apply them to spoken corpus creation. The goal of this work is to develop an ASR system to automate the transcription of speech for the purposes of creating and scaling a spoken corpus, following the One Day of Speech (henceforth ORD) notation style. The novelty of this thesis is in adapting the latest generation ASR model to a research orthography with limited data and compute to see if it is a viable way of reaching our goal. To achieve this, we fine-tune Whisper on a dataset extracted from ORD. These are the steps we will take to achieve these goals:

1) Create a dataset suitable for training an ASR model from the files of the ORD corpus;

2) Fine-tune Whisper on this dataset;

3) Evaluate the fine-tuned model on the test set;

4) Infer transcriptions on new recordings from the trained model and provide the code for inference.

Chapter 1. Literature Review

The field of automatic speech recognition (ASR) has witnessed advancements in recent years, primarily due to innovations in deep learning, large-scale data availability, and computational power. The main deep learning innovations behind this are the use of self-supervised learning and transformer-based architectures. Wav2vec 2.0 (Baevski et al. 2020) and HuBERT (Hsu et al. 2021) have enabled models to leverage unlabeled audio data, dramatically enhancing transcription accuracy. These advancements, coupled with massively multilingual ASR models (Pratap et al. 2020), have pushed ASR systems to near-human accuracy levels.

There are several works using ASR for the purpose of creating spoken corpora. (Markl 2022) discusses the benefits and challenges of using commercial ASR systems, such as Amazon and Google, in sociolinguistic research. In the Lothian Diary Project, these systems struggled with Scottish English, revealing higher error rates compared to standard English due to algorithmic bias and limited dialectal training data. Therefore, commercial ASR systems need improvements, particularly for minority and regional dialects. (Coats 2024) construct four corpora using automatic transcriptions provided by YouTube, with a minority of videos also having manual transcripts. They train downstream classification models on the transcripts, showing that both transcription methods lead to the same accuracy of classification. Both papers conclude that, while limited and containing errors, the results these systems show are promising.

Whisper (Radford et al. 2022) is an encoder-decoder autoregressive transformer ASR model. It is able to both transcribe and translate speech in 99 languages, detect the spoken language and detect voice activity. For the purposes of this work, it is important that Whisper shows strong performance in Russian transcription, that it outputs text in the orthography it is trained on, with capitalization and punctuation, and that it is specifically created to produce long-form transcription with time alignment. It takes 30s log-magnitude 80-channel mel-frequency spectrograms as inputs and outputs special and word tokens. The special tokens denote the language, absence of speech and the time of the beginning and start of the phrase. Timestamps vary from 0 to 30 seconds with 0.02 second steps. Word tokens are obtained by Byte-Pair Encoding. Whisper comes in several sizes, from tiny (39M parameters) to large (1550M).

A benefit of Whisper is that it is a multilingual model that attracts the attention of contributors from all over the world. As a result, there are many open-source projects building on top of it. They can be used both during training and for the final model. Improvements of such projects are: inference-speedups (“SYSTRAN/Faster-Whisper”), making it multi-platform (Gerganov) and speaker diarization (Bain et al. 2023) among others. It is well supported by the Hugging Face ecosystem of libraries, which make the training process easier. There are already full training scripts available that we base our project on.

The Odin Rechevoj Den (One Day of Speech, ORD) (Asinovsky et al. 2009) corpus contains spontaneous speech by native Russian speakers. It consists of three parts: a database, a collection of audio files, and transcription files, all three are important for this paper. Each file represents a macro-episode of speech with the audio in .wav at 22,050 Hz and its corresponding annotation in an ELAN file (Wittenburg et al. 2006) with various information, including transcription. There are often several speakers in each recording, speaking in varied conditions during varied activities. The database contains, among other things, a label for audio quality that is given an estimate from 1 to 3: "1 – the best quality, suitable for precise phonetic analysis, 2 – rather good quality partially suitable for phonetic analysis, 3 – noisy recordings with low quality which is only partially legible (not suitable for phonetic analysis but suitable for other aspects of research)" (Sherstinova 2009)

The informants are chosen so as to be a representative sample of society, with balanced gender, age, profession and education ratios. The speech in ORD is unique in that it captures natural, spontaneous human communication, not adapted for an audience, which, along with the presence of noise, presents a challenge for ASR systems.

The transcription follows a notation style different from the official Russian orthography. While maintaining the same spelling, it forgoes sentence-initial capitalization, replaces the usual punctuation with one based on prosody and makes use of symbols to represent paralinguistic elements such as throat clearing, sighs and laughter. It has different representations for non-hesitation and hesitation pauses, both filled and unfilled.

Previous research (Sherstinova, Kolobov, and Mikhaylovskiy 2023) showed that Whisper outperforms a different ASR model on ORD, despite showing a currently unimpressive 49% average Word Error Rate (henceforth WER), but achieves 7% WER at its best.

Chapter 2. Data

For each macro-episode, we extract all speech segments longer than 20 milliseconds that contain word tokens, saving their transcription text and speaker, noise and episode information. We skip non-speech segments for easier and more effective training. Segments shorter than 20 milliseconds are often erroneous and can't be timed correctly by Whisper, and hence are omitted. We use pympi (Lubbers and Torreira 2013) to read ELAN files and TorchAudio (Yang et al. 2022) to read, resample, cut and save audio files.

The extraction process resulted in 121,035 audio segments with durations varying from 50 to 65,134 milliseconds, amounting to 82.8 hours of speech data. After that, we consecutively pack these segments into larger files with durations of up to 30 seconds. When constructing the new audio file, we add audio segments from one original macro-episode file, following the original order and skipping segments longer than 30 seconds. Each audio file corresponds to packed untimed and timed text strings. In the untimed text strings, we add the separator "#" between utterances by different speakers. In the timed text strings, each utterance is preceded and followed by a special Whisper token containing the time of the beginning and the end of the utterance in seconds, for example, "<|3.24|>ну //<|3.86|>".

We then create train, validation and test splits of the packed dataset. When creating the evaluation splits, we aim to be able to test whether the amount of speaker audio in the finetuning dataset impacts the quality of the predictions of the finetuned model on utterances by said speaker during inference. We use the informant as an approximation of the speaker. We aim for 80% to remain in the training split. We first construct an intermediate evaluation split consisting of two parts: one containing informants, that will not be "seen" during training, the other with audios by informants that *will* be "heard" during training. In creating the unseen informants portion, we randomly pick 15 informants among the 90 with the lowest amounts of speech in the dataset. This results in 908 audio files. To create the portion with "seen" informants, we do a random split on the files, which are not in the "unseen" informants part. After

merging these two parts, we randomly split off 25% for the validation set and the remainder goes into the test set.

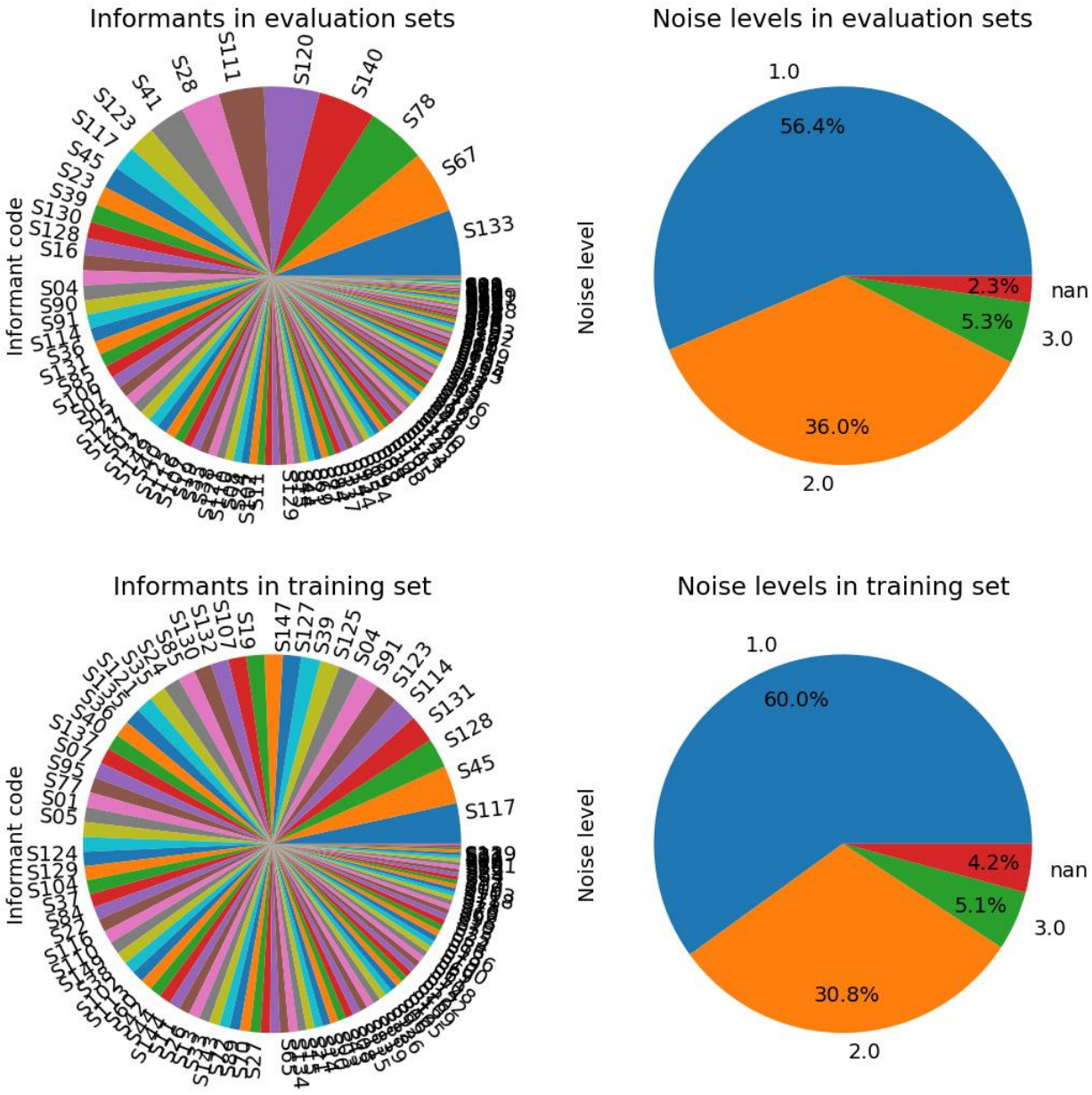

Figure 1. Distribution of noise levels and informants in the dataset

The final packed dataset contains 10,986 audio files with 8,788 files in the train set, and 550 audios in the validation set and 1,648 in the test set. This amounts to 82.1 hours of speech with 65.5, 4.1 and 12.4 hours in the train, validation and test splits accordingly. Of these, 6,510 audios have noise level 1, 3,499 are in level 2 and 561 are level 3.

Chapter 3. Training

We fine-tune Whisper using the Hugging Face transformers library (Wolf et al. 2020). Most of the training code is based on (Gandhi 2022). We add the code for loading our dataset and we make it so that 50% of samples seen during training are timestamped, while others are not, as is done in the original paper. We complicate the process in that way because initial training runs without timestamps showed that the model doesn't transfer its newly-learned notation abilities to time-aligned inference. We also bring the training closer to that of Whisper-large-v2 by adding regularization techniques such as SpecAugment (Park et al. 2019), BPE-Dropout (Provilkov, Emelianenko, and Voita 2020) and LayerDrop (Fan, Grave, and Joulin 2019) as a stand-in for StochasticDepth (Huang et al. 2016).

We train the final model using the parameter efficient fine-tuning method DoRa (weight-decomposed low-rank adaptation) from (Liu et al. 2024). We quantize the pre-trained model into the 4-bit normalized floating point data type (NF4) introduced in (Dettmers et al. 2023) using bitsandbytes, and add 32-rank DoRa to all weights with alpha=64, dropout=0.05 and no bias. We train the DoRa layers for 4 epochs of 2200 update steps with 50 warm-up steps, using the Adam optimizer with a learning rate of 0.001 and weight decay of 0.1, and linear learning rate decay.

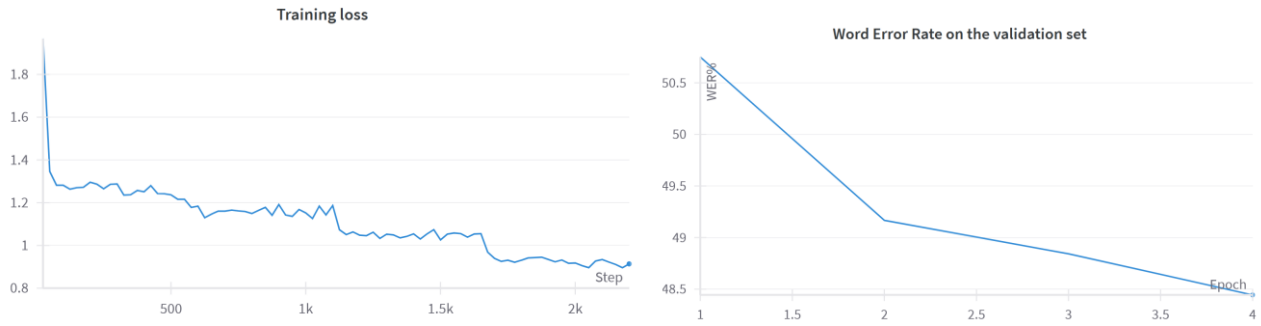

Figure 2. Training statistics

We evaluate the model after each epoch by generating 100% timestamped samples on the validation set without feeding the model the labels. We compute WER

and CER as well as case-insensitive error rates on word tokens only, ignoring ORD-specific notation. We also log 8 generations to Weights and Biases (Biewald 2020) to visually confirm the performance. The final training ran for approximately 24 hours and needed to be resumed from intermediate checkpoints several times because of the limitations on session duration presented by Kaggle.

Even though, taking into consideration the computational resources available for us we use PEFT to train our model instead of full fine-tuning, it is worthwhile to fully fine-tune a new model on the dataset (when computational resources are available), as our experiments with the small model showed, that no matter how efficient LoRA methods are, full fine-tuning is still superior (by about 4 WER% points).

Chapter 4. Results

	Fine-tuned model				**Whisper-large-v2**			
	WER%	**CER%**	**wo-WER%**	**wo-CER%**	**WER%**	**CER%**	**wo-WER%**	**wo-CER%**
mean	45.70	8.33	32.56	7.31	79.41	9.23	49.19	8.11
std	19.59	11.49	22.58	10.81	15.52	11.62	31.45	10.94
min	7.77	0.65	1.64	0.61	43.75	0.71	10.98	0.69
25%	34.69	5.20	19.15	4.48	71.79	6.10	33.33	5.22
50%	44.78	6.38	30.00	5.51	78.80	7.41	44.44	6.40
75%	54.04	7.94	41.80	6.93	85.55	9.16	56.52	7.96
max	274.68	233.00	345.83	227.65	350.82	236.29	535.00	230.88

Table 1. ASR metrics of the fine-tuned and initial models on the test set

Finally, we evaluate the model on the test set. In Table 1, you can see that fine-tuning leads to improvements in all ASR metrics. Initial wo WER, where “wo” stands for word-only, i.e. with all ORD-specific annotation removed, stands at 49%, which is the same as the value obtained from long-form transcription evaluation on the entirety of ORD by (Sherstinova, Kolobov, and Mikhaylovskiy 2023). We see a reduction of 17 points in that metric, and the final full WER stands at 45.7%, having a lower error

rate on ORD notation than the initial model did on word tokens. We see a reduction in the maximum error rates, but it still remains high at 274.68% WER.

There is a slight, although statistically significant, dependence of the WER on the amount of data in the training set for the informant and the noise level ($p = 0.007$ and $p > 0.001$ accordingly in multiple linear regression).

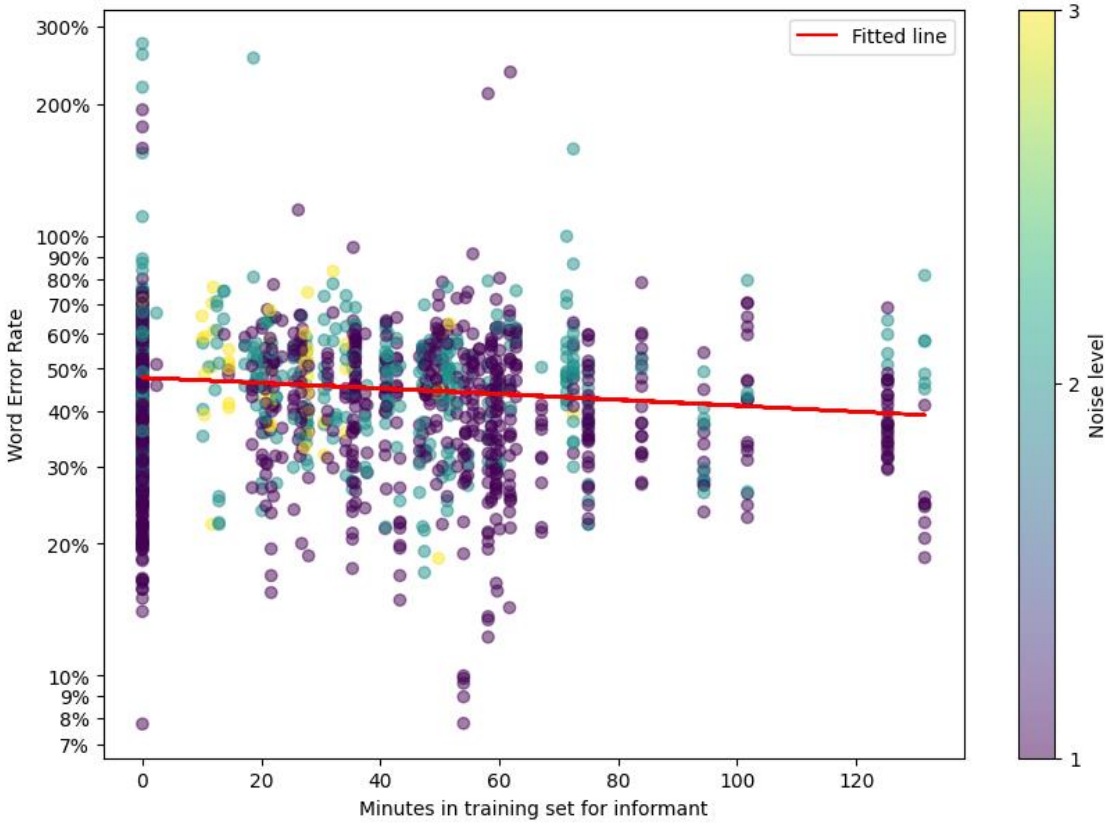

Figure 3. Scatter plot of error rates for different noise levels and amounts of data in the training set

The trained model successfully adopts the new orthography. It drops standard unneeded punctuation, such as commas and periods; it stops capitalizing words sentence-initially, while maintaining the capitalization of proper names. It adds syntagma phrase separator symbols "/" and "//"; it performs ORD-specific speaker diarization by adding the symbols "#" and "@" denoting speech overlap. It transcribes non-speech noise, such as inhalation and coughs, using the correct character sequences.

It recognizes personal names by adding “%” to their end. It transcribes hesitation pauses, both filled and unfilled.

We see improved WER as the model better adapts to noisy, spontaneous and informal speech. Using the final model, we generate ELAN files for new macro-episodes to be part of a spoken corpus. We also make the inference code and model weights available for other contributors to a speech corpus with ORD-style notation.

Chapter 5. Inference

For inference, we merge the DoRa weights into the unquantized Whisper large-v2. We get timed transcriptions through Hugging Face transformers. Pympi is then used to write an ELAN file containing the results of the recognition. We make it available as a Colab notebook, that does not require coding abilities, through this link: colab.research.google.com/drive/1DLBXabtYSREQWdcKllKYDI0MaYLNyvyT

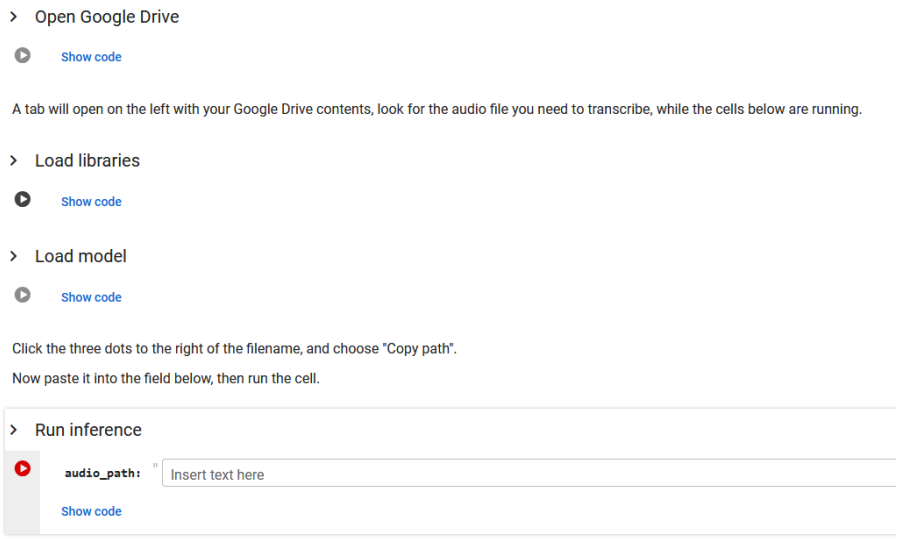

Figure 4. The Colab notebook for ELAN file inference

We also convert the weights into the PyTorch format so that the model can be used in the official openai-whisper library. All weights are available through this link: huggingface.co/mizoru/whisper-large-ru-ORD_0.9_peft_0.2_merged_unquantized

How much time is needed to fully correct the automatically obtained transcriptions is an open question; however, it is the author's opinion that the process will be significantly quicker than transcribing from scratch.

Chapter 6. Limitations

While the original Whisper model sometimes falls victim to hallucination, the fine-tuned model seems to do it more, as it starts filling all seconds with transcription, resulting in repeated notation symbols. The most likely reason for this is the closely packed nature of the produced dataset and the omission of segments containing non-human noise.

This paper is also limited by the computational resources available because, while full fine-tuning and more thorough exploration of different PEFT options would almost surely lead to improvements in performance, we were not able to conduct those. Another omission is pre-training specifically aimed at noise robustness. While our fine-tuning dataset contains noisy audio segments, we did not work to specifically augment the data with noise. No work was put into optimizing inference speed or memory requirements as well.

Chapter 7. Future work

While the manner in which the initial pre-training dataset was cut into segments by OpenAI is not described in sufficient detail (e.g., there is no information on what they do when a segment does not fit into the 30-second window), we expect that brining our dataset closer to the inference inputs by adding more non-speech segments could improve the problem with hallucination.

There are also viable low-effort options for improving the error rates by fine-tuning a different model that already shows better performance, such as large-v3 (which we did not use because of reports of increased hallucination behavior) or whisper-large-v2-ru, available on the Hugging Face Hub, although with missing punctuation.

CONCLUSION

This project streamlines the transcription and annotation of spoken corpora by fine-tuning Whisper on a dataset extracted from the ORD corpus. By adapting Whisper to ORD's unique annotation style, we automate transcription while testing the viability of fine-tuning for corpus scaling.

Our dataset preparation resulted in 82.8 hours of speech data repacked into train, validation, and test splits. The fine-tuned Whisper model successfully adopts ORD orthography, accurately handling syntagma separators, speaker overlap, personal names, and non-speech noise.

This research demonstrates that tailored fine-tuning can significantly improve transcription accuracy, making it a promising strategy for creating and scaling spoken corpora. The final model and inference code are made publicly available and are the most important contribution of this thesis.

REFERENCES

1) Asinovsky, Alexander, Natalia Bogdanova, Marina Rusakova, Anastassia Ryko, Svetlana Stepanova, and Tatiana Sherstinova. 2009. “The ORD Speech Corpus of Russian Everyday Communication ‘One Speaker’s Day’: Creation Principles and Annotation.” In *Text, Speech and Dialogue*, edited by Václav Matoušek and Pavel Mautner, 5729:250–57. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-04208-9_36.
2) Baevski, Alexei, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. “Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations.” arXiv. https://doi.org/10.48550/arXiv.2006.11477.
3) Bain, Max, Jaesung Huh, Tengda Han, and Andrew Zisserman. 2023. “WhisperX: Time-Accurate Speech Transcription of Long-Form Audio.” *INTERSPEECH 2023*.

4) Biewald, Lukas. 2020. “Experiment Tracking with Weights and Biases.” https://www.wandb.com/.
5) Bird, Steven. 2021. “Sparse Transcription.” *Computational Linguistics* 46 (4): 713–44. https://doi.org/10.1162/coli_a_00387.
6) Coats, Steven. 2024. “Noisy Data.” *Linguistics Across Disciplinary Borders: The March of Data*, 17.
7) Dettmers, Tim, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. “QLoRA: Efficient Finetuning of Quantized LLMs.” arXiv. https://doi.org/10.48550/arXiv.2305.14314.
8) Fan, Angela, Edouard Grave, and Armand Joulin. 2019. “Reducing Transformer Depth on Demand with Structured Dropout.” arXiv. https://doi.org/10.48550/arXiv.1909.11556.
9) Gandhi, Sanchit. 2022. “Fine-Tune Whisper For Multilingual ASR with 🤗 Transformers.” November 3, 2022. https://huggingface.co/blog/fine-tune-whisper.
10) Gerganov, Georgi. n.d. “Ggerganov/Whisper.Cpp.” C. Accessed May 24, 2024. https://github.com/ggerganov/whisper.cpp.
11) Hsu, Wei-Ning, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. “HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units.” arXiv. https://doi.org/10.48550/arXiv.2106.07447.
12) Huang, Gao, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Weinberger. 2016. “Deep Networks with Stochastic Depth.” arXiv. https://doi.org/10.48550/arXiv.1603.09382.
13) Liu, Shih-Yang, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. “DoRA: Weight-Decomposed Low-Rank Adaptation.” arXiv. https://doi.org/10.48550/arXiv.2402.09353.
14) Lubbers, Mart, and Francisco Torreira. 2013. “Pympi-Ling: A Python Module for Processing ELANs EAF and Praats TextGrid Annotation Files.” https://pypi.python.org/pypi/pympi-ling.
15) Markl, Nina. 2022. “(Commercial) Automatic Speech Recognition as a Tool in Sociolinguistic Research,” September.
16) Park, Daniel S., William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. “SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition.” In *Interspeech 2019*, 2613–17. https://doi.org/10.21437/Interspeech.2019-2680.
17) Pratap, Vineel, Anuroop Sriram, Paden Tomasello, Awni Hannun, Vitaliy Liptchinsky, Gabriel Synnaeve, and Ronan Collobert. 2020. “Massively Multilingual ASR: 50 Languages, 1 Model, 1 Billion Parameters.” arXiv. https://doi.org/10.48550/arXiv.2007.03001.
18) Provilkov, Ivan, Dmitrii Emelianenko, and Elena Voita. 2020. “BPE-Dropout: Simple and Effective Subword Regularization.” arXiv. https://doi.org/10.48550/arXiv.1910.13267.
19) Radford, Alec, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. “Robust Speech Recognition via Large-Scale Weak Supervision.” arXiv. https://doi.org/10.48550/arXiv.2212.04356.
20) Sherstinova, Tatiana. 2009. “The Structure of the ORD Speech Corpus of Russian Everyday Communication.” In *International Conference on Text, Speech and Dialogue*, 258–65. Springer.
21) Sherstinova, Tatiana, Rostislav Kolobov, and Nikolay Mikhaylovskiy. 2023. “Everyday Conversations: A Comparative Study of Expert Transcriptions and ASR Outputs at a Lexical Level.” In *Speech and Computer*, edited by Alexey Karpov, K. Samudravijaya, K. T. Deepak, Rajesh M. Hegde, Shyam S. Agrawal, and S. R. Mahadeva Prasanna, 43–56. Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-48309-7_4.
22) “SYSTRAN/Faster-Whisper.” n.d. Python. SYSTRAN. Accessed May 24, 2024. https://github.com/SYSTRAN/faster-whisper.

23) Wittenburg, Peter, Hennie Brugman, Albert Russel, Alex Klassmann, and Han Sloetjes. 2006. “ELAN: A Professional Framework for Multimodality Research.” In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC’06)*, edited by Nicoletta Calzolari, Khalid Choukri, Aldo Gangemi, Bente Maegaard, Joseph Mariani, Jan Odijk, and Daniel Tapias. Genoa, Italy: European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2006/pdf/153_pdf.pdf.
24) Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, et al. 2020. “HuggingFace’s Transformers: State-of-the-Art Natural Language Processing.” arXiv. https://doi.org/10.48550/arXiv.1910.03771.
25) Yang, Yao-Yuan, Moto Hira, Zhaoheng Ni, Anjali Chourdia, Artyom Astafurov, Caroline Chen, Ching-Feng Yeh, et al. 2022. “TorchAudio: Building Blocks for Audio and Speech Processing.” arXiv. https://doi.org/10.48550/arXiv.2110.15018.