Second Semester Project Report: Fine-tuning and Hyperparameter Optimization for Hungarian Speech Recognition

Dániel Varga

May 17, 2026

1 Introduction

The objective of this project is to develop a Hungarian speech-to-text system using modern automatic speech recognition (ASR) methods. During the first semester, I focused mainly on understanding the foundations of Whisper-based ASR: log-Mel spectrogram input representations, encoder-decoder transformer models, autoregressive decoding, word error rate (WER), and preliminary robustness experiments. This provided the technical basis for the second semester, where the work shifted from theoretical exploration to building and testing a fine-tuning pipeline for Hungarian speech recognition.

A substantial part of the semester was therefore the implementation of an experimental pipeline around Whisper fine-tuning. The pipeline was designed as a reusable fine-tuning framework rather than a single-purpose experiment script. It supports configuration-driven experiment definition, reproducible artifact management, automatic metric and prediction export, and comparison of multiple training runs. This was important because the aim was not only to evaluate one Whisper-small model, but to build a general infrastructure that can later be reused for larger Whisper variants and more extensive hyperparameter searches under the same controlled experimental setup.

The main experimental question this semester was whether `openai/whisper-small` can be substantially improved on Hungarian speech recognition through supervised fine-tuning, and whether hyperparameter optimization can identify competitive configurations. To answer this, I evaluated the pretrained Whisper-small model on the development split, trained several manually selected fine-tuning configurations, and then ran an Optuna hyperparameter optimization (HPO) study.

The results show a clear improvement from supervised fine-tuning. The *Zero-shot baseline* reached 52.79% WER and 22.52% CER on the development split with 577 examples. The strongest manually selected fine-tuned configuration reached 27.58% WER. The

HPO study identified a short, efficient training configuration; when this configuration was rerun on the full development split as *HPO rerun on full dev*, it reached 27.66% WER and 10.15% CER. These results indicate that fine-tuning reduced the error rate substantially and that HPO found a configuration with the best CER among the main runs.

2 Dataset and Evaluation Setup

Hungarian ASR is challenging because Hungarian is morphologically rich and agglutinative. Word forms often contain several suffixes encoding grammatical information, so even small transcription errors may change the grammatical form of a word. This makes it useful to evaluate the models not only at the word level, but also at the character level.

BEA-Base was designed specifically for ASR training and evaluation on Hungarian speech. The official BEA-Base statistics are summarized in Table 1. The development and evaluation sets are further divided into repeated/read speech and spontaneous speech. This distinction is important because spontaneous speech usually contains hesitations, restarts, less regular sentence structure, and more natural speaking patterns, making it substantially harder than read or repeated sentences.

Subset	Speech type	Duration (h)	Words
Train	mixed training material	71.20	555,322
Dev-repet	repeated/read	0.65	4,110
Dev-spont	spontaneous	4.02	27,939
Eval-repet	repeated/read	0.95	6,229
Eval-spont	spontaneous	4.91	35,178
Development total	repeated/read + spontaneous	4.67	32,049
Evaluation total	repeated/read + spontaneous	5.86	41,407

Table 1: Official BEA-Base subset statistics. The development and evaluation partitions distinguish repeated/read speech from spontaneous speech.

In my experiments, I used the BEA-Base training partition for supervised fine-tuning and the full BEA-Base development set for validation and evaluation. The current development evaluation contains 577 examples. This split is independent from the training split and was used consistently for comparing the baseline, the manual fine-tuning runs, and the HPO-selected configuration.

All reported experiments use the same base model: `openai/whisper-small`. Whisper is an encoder-decoder transformer trained on large-scale multilingual and multitask speech data. This makes it a strong pretrained starting point, but the baseline results show that direct zero-shot use is not sufficient for this Hungarian development set. Model quality

is reported with WER and CER. WER measures word-level edit distance, while CER measures the same type of error at the character level; both are reported as percentages, and lower values are better. The WER formula is: WER $= \frac{S+D+I}{N}$, where S, D, and I are the numbers of substitutions, deletions, and insertions, and N is the number of words in the reference transcript.

3 Baseline and Manual Fine-tuning Experiments

3.1 Zero-shot Whisper-small Baseline

The pretrained `openai/whisper-small` model was evaluated without any fine-tuning to establish a direct reference point. On the 577-example development split, the *Zero-shot baseline* reached: WER $= 52.79\%, \text{CER} = 22.52\%$.

3.2 Manual Fine-tuning Runs

The manual fine-tuning phase tested five main configurations. For readability, I use descriptive names in addition to the original experiment identifiers.

Name	Original ID	LR	Warmup	Steps	Eff. batch	Epochs	WER	CER
Zero-shot baseline	-	-	-	-	-	-	52.79	22.52
Stable long run	exp04	$1.0 \cdot 10^{-4}$	500	7000	16	12.2	**27.58**	10.39
Low-LR run	exp05	$1.0 \cdot 10^{-6}$	500	7000	16	12.2	37.49	15.53
High-LR run	exp06	$1.0 \cdot 10^{-3}$	2000	7000	16	12.2	46.36	21.34
Large-batch long run	exp07	$1.0 \cdot 10^{-4}$	500	5000	64	35.0	28.01	10.36
HPO rerun on full dev	exp08	$1.56 \cdot 10^{-4}$	146	750	128	10.5	27.66	**10.15**

Table 2: Baseline and manual fine-tuning results. WER and CER are reported in percent. The *HPO rerun on full dev* is the full-development-set rerun of the best HPO parameter setting.

Table 2 summarizes the most important runs. The epoch values are approximate training equivalents computed from the number of optimization steps, the global effective batch size, and the 9,157 audio segments in the full training split. I use the formula epochs $\approx$ (steps $\times$ effective batch size)/9157. For all reported fine-tuning runs, the global effective batch size includes two GPUs.

The manual experiments show that learning rate strongly affects performance. The very small learning rate in the *Low-LR run* led to under-adaptation, with WER remaining at 37.49%. The very large learning rate in the *High-LR run* was worse, producing 46.36% WER. The strongest region was around 10^{-4} to $1.6 \cdot 10^{-4}$, as shown by the *Stable long run* and the *HPO rerun on full dev*.

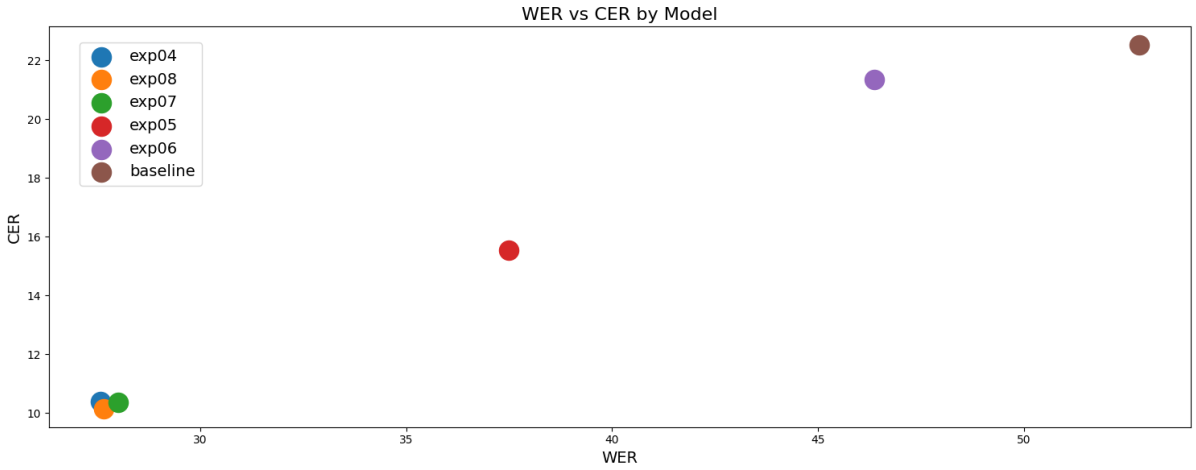

Figure 1: Comparison of zero-shot and fine-tuned Whisper-small results.

The *HPO rerun on full dev* is especially important. During HPO, the in-training evaluations were intentionally performed on an initial slice of the development set. This made the search faster and allowed more configurations to be tested under the available GPU-time constraints. However, because this shortened evaluation protocol was only meant for model selection, the best HPO parameter setting had to be rerun and evaluated on the full development split. This was the purpose of `exp08`. The rerun used the best HPO setting with learning rate $1.56 \cdot 10^{-4}$, 146 warmup steps, effective batch size 128, and 10.5 epochs. It achieved 27.66% WER and the best CER among the main runs, 10.15%. Although its WER is slightly worse than *Stable long run*, its CER result confirms that the HPO search found a genuinely useful configuration.

4 Hyperparameter Optimization with Optuna

Manual experimentation was useful, but it only tested a small number of configurations. Therefore, I used Optuna to run a more systematic HPO study around the best available Whisper-small setup. The goal was not to prove global optimality, but to find a strong configuration under the available GPU and time constraints.

The main HPO search space was:

- learning rate: $5.0 \cdot 10^{-5}$ to $2.5 \cdot 10^{-4}$ on a log scale;
- maximum training steps: 750, 1000, 1250, or 1500;
- warmup ratio: 0.05 to 0.20.

The warmup steps were computed from the sampled warmup ratio and the selected maximum number of steps. The effective batch size was kept fixed at 128, because the purpose of the search was to compare model-quality hyperparameters rather than infrastructure settings. The main HPO study contains 10 trials: 5 completed and 5 pruned.

The pruned trials were stopped because their intermediate results were weaker than the completed trials at comparable evaluation points. Table 3 lists the completed trials.

Trial	LR	Warmup ratio	Warmup	Steps	Eff. batch	Epochs	WER
0	$9.14 \cdot 10^{-5}$	0.160	239	1500	128	21.0	28.39
1	$1.31 \cdot 10^{-4}$	0.073	55	750	128	10.5	27.94
2	$5.49 \cdot 10^{-5}$	0.140	210	1500	128	21.0	29.94
3	$1.56 \cdot 10^{-4}$	0.195	146	750	128	10.5	**27.74**
4	$1.91 \cdot 10^{-4}$	0.077	57	750	128	10.5	27.77

Table 3: Completed trials in the main Optuna HPO study. WER is reported in percent under the HPO evaluation protocol.

The best completed trial was Trial 3, with 27.74% WER under the HPO evaluation protocol. Trial 4 was very close, with 27.77% WER. This suggests that, in the tested search space, the strongest region involved short training, larger effective batch size, and learning rates around $1.5{\cdot}10^{-4}$ to $1.9{\cdot}10^{-4}$. The later full-development-set rerun confirmed that this parameter region was useful, especially in terms of CER.

5 Quantitative Results and Discussion

Table 4 summarizes the main results used for the report narrative.

Name	Type	Steps	Eff. batch	Epochs	WER	CER
Zero-shot baseline	pretrained inference	-	-	-	52.79	22.52
Stable long run	manual fine-tuning	7000	16	12.2	**27.58**	10.39
HPO rerun on full dev	HPO-parameter rerun	750	128	10.5	27.66	**10.15**
HPO best trial	completed HPO trial	750	128	10.5	27.74	10.32

Table 4: Headline results. WER and CER are reported in percent.

The most important conclusion is that fine-tuning was clearly effective. The *Zero-shot baseline* WER was 52.79%, while the strongest fine-tuned runs reached approximately 27-28% WER. The *Stable long run* achieved the lowest final WER, but the *HPO rerun on full dev* is also important because it reached nearly the same WER with 10.5 epochs and the best CER among the main runs.

The HPO study did not beat the best manual WER, but it still provided useful evidence. It identified a short and efficient configuration that was then rerun as *HPO rerun on full dev*. This means that HPO helped narrow the search toward a competitive parameter region. The result should therefore be interpreted as practical optimization under limited resources, not as proof of a globally optimal model.

Training efficiency also matters. The *Stable long run* corresponds to 12.2 epochs, while the *HPO rerun on full dev* corresponds to 10.5 epochs. In epoch terms, the rerun was therefore only moderately shorter, but it used a much larger effective batch size and a higher learning rate. It still achieved 27.66% WER and 10.15% CER.

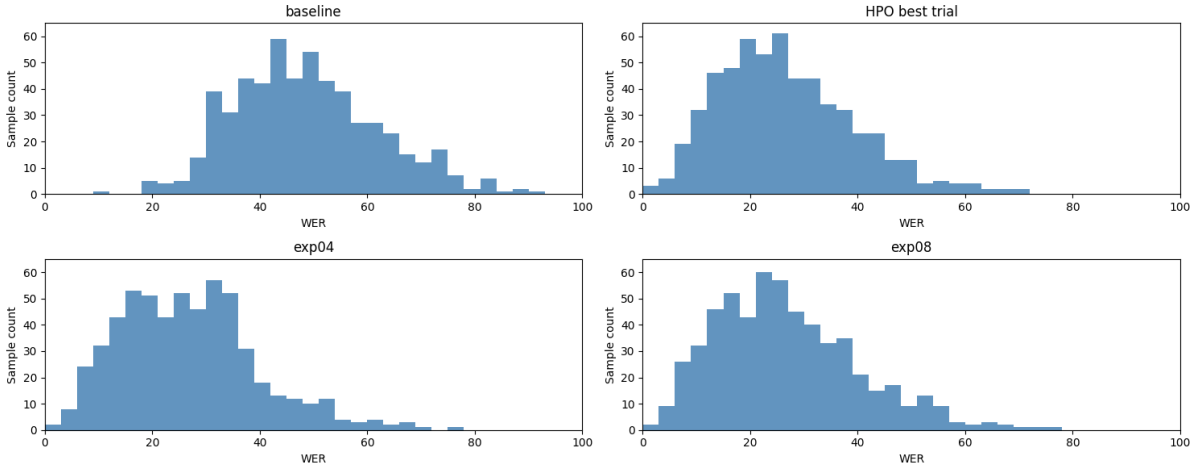

Figure 2: Per-sample WER distributions for the zero-shot baseline and selected fine-tuned Whisper-small models. The x-axes are cut at 100% WER for readability; values above this threshold are omitted from the visualization but not from the metric computation.

Per-sample WER distributions add another useful perspective to the aggregate metrics. The fine-tuned models show a clear leftward shift in the per-sample WER distribution. In *Stable long run* (`exp04`), *HPO best trial*, and *HPO rerun on full dev* (`exp08`), most utterances fall into substantially lower WER ranges than in the zero-shot baseline, with the highest-density regions typically around the 15-35% interval. This indicates that fine-tuning did not merely improve the aggregate score through a few outlier examples; it improved recognition quality across a broad part of the development set. At the same time, the fine-tuned histograms still have a visible right tail, which shows that some utterances remain difficult even after fine-tuning.

6 Limitations and Future Work

The results should be interpreted with appropriate caution. First, the development split is independent from the training split and is suitable for development-time model comparison. However, a final held-out test-set evaluation would still be useful before making a stronger benchmark-style claim about the final model. This is a standard distinction between model selection and final reporting, not a concern about the reliability of the current data. Second, the HPO study was intentionally small. It searched learning rate, maximum steps, and warmup ratio, while keeping other parameters fixed. A larger-volume

HPO run would be the most direct next step, because the completed trials already suggest that the useful region is relatively narrow but not exhausted.

Future work should focus on four concrete directions. First, I would run a larger HPO study with more completed trials and a wider but still realistic search space. Second, I would fine-tune a larger Whisper model, especially Whisper-large-v3. I have high expectations for this direction because Whisper-large-v3 is substantially stronger than Whisper-small, but in the early phase of the project it would have been less time-efficient for testing and stabilizing the fine-tuning pipeline. Starting with Whisper-small made the pipeline faster to debug and allowed more experiments under limited GPU time. Third, I would perform manual error analysis using the saved reference-prediction pairs, with special attention to suffix errors, word boundary errors, named entities, numbers, and hesitation-related mistakes. Fourth, I would experiment with text post-processing in order to produce more reliable subtitles, for example by normalizing punctuation, casing, repeated fragments, and other formatting issues after decoding.

7 Conclusion

During the second semester, the project moved from understanding Whisper-based ASR to building and using a fine-tuning pipeline for measurable Hungarian speech recognition experiments. The *Zero-shot baseline* reached 52.79% WER and 22.52% CER on the development split, while the best fine-tuned configurations reduced WER to approximately 27-28%. This confirms that supervised fine-tuning substantially improved performance in the current Hungarian ASR setting.

The manual experiments showed that learning rate and batch configuration strongly affect model quality. The HPO study then provided a more systematic search and identified a short, efficient configuration. Although this configuration did not surpass the best manual WER, its rerun on the full development split achieved 27.66% WER and the best CER value, 10.15%. Overall, the semester produced both a working experimental pipeline and a set of measurement results that provide a strong basis for continued Hungarian Whisper fine-tuning, larger-model experiments, and detailed error analysis.

References

[1] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. *Robust Speech Recognition via Large-Scale Weak Supervision.* 2022.

[2] Mihajlik, P., Balog, A., Gráczi, T. E., Kohári, A., Tarján, B., & Mády, K. (2022). *BEA-Base: A Benchmark for ASR of Spontaneous Hungarian.* In LREC 2022, Thirteenth International Conference on Language Resources and Evaluation (pp. 1970–1977).