
Flavors of Moonshine: Tiny Specialized ASR Models for Edge Devices

Evan King* Adam Sabra* Manjunath Kudlur* James Wang Pete Warden

Moonshine AI

Abstract

We present the Flavors of Moonshine, a suite of tiny automatic speech recognition (ASR) models specialized for a range of underrepresented languages. Prevailing wisdom suggests that multilingual ASR models outperform monolingual counterparts by exploiting cross-lingual phonetic similarities. We challenge this assumption, showing that for sufficiently small models (27M parameters), training monolingual systems on a carefully balanced mix of high-quality human-labeled, pseudo-labeled, and synthetic data yields substantially superior performance. On average, our models achieve error rates 48% lower than the comparably sized Whisper Tiny model, outperform the 9x larger Whisper Small model, and in most cases match or outperform the 28x larger Whisper Medium model. These results advance the state of the art for models of this size, enabling accurate on-device ASR for languages that previously had limited support. We release Arabic, Chinese, Japanese, Korean, Ukrainian, and Vietnamese Moonshine models under a permissive open-source license.

1. Introduction

Automatic Speech Recognition (ASR) has seen growing interest in recent years, driven by new opportunities for speech-driven human-machine interaction (Namvarpour & Razi, 2025). Low latency ASR models that are small enough to deploy on-device can empower a new generation of voice-driven applications, from real-time translation devices to intelligent conversational interfaces in automobiles (Rege et al., 2024) and smart appliances (King et al., 2025). While resource-intensive ASR models like Whisper Large (Radford et al., 2023) and NVIDIA Canary (Puvvada et al., 2024), or multimodal models like Phi-4 (Abouelenin et al., 2025)

can be deployed in the cloud, they incur substantial infrastructure costs, rely on internet connectivity, and raise user privacy concerns. These limitations have spurred the development of ASR models that are compact enough to run efficiently on the edge (Jeffries et al., 2024).

To date, most lightweight ASR research has focused on English or on multilingual models that underperform on non-English languages. Whisper Tiny is a multilingual model that is small enough to run on-device, and provides a good example: while it achieves a strong 12% error rate on English, the model is less-than-stellar for Vietnamese ASR, where it has a 60% error rate (Radford et al., 2023). In other words, while Whisper Tiny *technically* supports Vietnamese, its usability in real-world on-device applications is limited. This performance gap motivates the development of tiny ASR models that better support non-English languages.

Prior work suggests that multilingual models can leverage cross-lingual similarities, allowing knowledge from high-resource languages to transfer and improve recognition in lower-resource ones (Cho et al., 2018; Toshniwal et al., 2018; Ardila et al., 2019; Pratap et al., 2020). However, training models that are lightweight enough for edge devices requires reducing the number of learnable parameters, meaning that convergence on a truly high-performance multilingual model is challenging. In this paper, we focus our efforts on training lightweight (27M parameter) *monolingual* models for edge devices, exploiting efficiencies in training data that we achieve from a combination of open human-labeled datasets, high-quality pseudo-labels, and synthetic utterances. We choose six languages—Arabic, Chinese, Japanese, Korean, Ukrainian, and Vietnamese—with varying levels of training data availability. Our models achieve word error and character error rates (WER/CER) that are on-average 48% lower than the comparably sized Whisper Tiny model, and are in most cases on-par or better than the 28x larger Whisper Medium model. Additionally, by exploiting the architectural benefits of the Moonshine model architecture, our models run between 5x-15x faster than Whisper in on-device applications.

In summary, this paper introduces the following:

*Equal contribution. Correspondence to: Pete Warden <pete@moonshine.ai>.

- We present new Moonshine ASR models for Arabic, Chinese, Japanese, Korean, Ukrainian, and Vietnamese. We show that the models achieve an average of 48% lower WER/CER than Whisper Tiny, making them significantly better-suited to edge ASR applications.
- We show that in all cases, the models outperform the 9x larger Whisper Small model. We also show that in most cases, the models are on par or 5-10% better than Whisper Medium, which is 28x larger.
- We release the models under a permissive open-source license.

The remainder of this paper is structured as follows. Section 2 describes our approach to training data collection, preprocessing, and model training. Section 3 summarizes the results of our evaluations. Section 4 concludes the paper. Appendices A, B, and C include information about public datasets, evaluation procedure, and detailed results.

Language	Open	Internal	Synthetic	Total
Arabic	4.6	10.0	0.9	15.5
Chinese	50.9	19.0	0.0	69.8
Japanese	36.9	17.0	0.0	53.9
Korean	27.6	44.4	0.0	72.0
Ukrainian	1.7	12.9	5.1	19.6
Vietnamese	8.4	85.8	0.0	94.2

Table 1. Training data hours (in thousands) by language and source. We supplement existing publicly-released datasets with internally collected datasets and, in some cases, synthesize utterances for lower-resource languages.

2. Approach

This section describes the Moonshine model architecture before detailing our data preparation and model training process.

2.1. Architecture

We leverage the Moonshine model architecture (Jeffries et al., 2024), an encoder-decoder transformer that applies rotary position embeddings (RoPE) in its encoder and decoder layers (Su et al., 2024). Compared to Whisper, Moonshine is especially well-suited for edge applications because its inference cost—in terms of FLOPs and corresponding latency—scales with the duration of the input audio. In contrast, Whisper pads all inputs to 30 seconds, regardless of length, leading to unnecessary computation. We adopt the Moonshine Tiny variant (27M parameters), which is small enough to be deployed in resource-constrained environments. Table 2 provides a comparison between the

Moonshine Tiny architecture and the similarly-sized Whisper Tiny model.

Parameter	Moonshine	Whisper
Dimension	288	384
Encoder layers	6	4
Decoder layers	6	4
Attention heads	8	6
Encoder FFN activation	GELU	
Decoder FFN activation	SwiGLU	GELU
Parameters (M)	27.1	37.8
FLOPs vs. Whisper Tiny	0.7x	1.0x

Table 2. Tiny model architectures and inference FLOPs

2.2. Data collection, preprocessing, & synthesis

The amount of data used as input for training a transformer-based ASR model is loosely predictive of model performance, with a minimum of between 10^4 to 10^5 hours required for usable results (Radford et al., 2023). Some languages are considered “higher-resource” than others—that is, there is more recorded audio available that can be used for training. As we are focused on a range of mid to low-resource languages, the availability of data varies for each. With this in mind, we approached data collection, preprocessing, and, in some cases, synthesis, using a three-stage strategy applied to each language:

1. **Aggregate publicly-available ASR datasets.** We leveraged ASR datasets from prior work. This allowed us to establish a baseline for model performance—effectively the best results we could achieve without performing raw data collection and labeling.
2. **Collect and pseudo-label publicly-available data.** Because existing public datasets lack the volume needed to train high-performing models, we gathered and pseudo-labeled a large collection of raw audio from publicly available sources such as podcasts and radio streams.
3. **Synthesize labeled data from text-only datasets.** In cases where enough raw data was not widely available, we used high-quality text-to-speech models to synthesize diverse utterances from traditionally text-only datasets.

Table 1 provides an overview of the sources and sizes of training datasets for each language. Appendix A provides a detailed breakdown of the public datasets for each language.

Since we are targeting lower-resource languages, publicly available datasets are insufficient to achieve high performance. Our data collection target was to exceed the amount

(in hours) used to train the initial Whisper models, based on the intuitions that (1) monolingual models do not benefit from transfer learning, and thus need larger datasets and (2) model performance loosely scales with dataset size. We therefore collected and labeled a large amount of raw, in-the-wild speech data available on the open internet. Leveraging WhisperX (Bain et al., 2023) modified to run in a custom framework for distributed data processing, we prepared around 173,000 hours of internal datasets across languages. In all cases except Chinese¹, our monolingual datasets exceeded the size of those used to train the original Whisper by an order of magnitude.

In the cases of Arabic and Ukrainian, raw data sources on the open internet were insufficient to meet the threshold for dataset hours. To fill the gap, we leveraged text-only datasets for each language to create uniform distributions of utterance lengths, which we then input to high-quality, high-diversity text-to-speech models to generate fully synthetic utterances. Employing style interpolation between speaker embeddings helped us ensure a diverse set of synthetic speakers (Dumoulin et al., 2016).

2.3. Training

We train all models using a schedule-free AdamW optimizer (Defazio et al., 2024) with a learning rate of $2e^{-5}$ for 8 epochs, with a batch size of 32. All models were trained via DDP on 8xH100 GPUs until completion. Training took between 1 to 3 days depending on the amount of data in our language-specific corpus.

3. Evaluations

Our evaluations measure the performance of Moonshine Tiny models against several multilingual Whisper variants of increasing size—Tiny, Base, Small, and Medium. We choose these Whisper variants since they are small enough to be deployed in the same on-device applications we target with Moonshine. We use a beam size of 1 in evaluations with Whisper, which is functionally equivalent to the greedy decoding used by Moonshine².

Our primary metric is error rate. Both word error rate (WER) and character error rate (CER) are commonly used in evaluating ASR systems; however, there is some subjectivity as to the best metric for certain languages. For completeness, we measure both WER and CER in every evaluation (except Japanese and Chinese, which lack word boundaries) and

¹Since Whisper’s release, the amount of publicly-available Chinese data has grown substantially, which reduces the need for internal collection.

²In the original Whisper paper, the authors use a beam size of 5. This produces slightly different results than the ones we report in this paper.

report the exhaustive results in Appendix C. The summarized results in this section only report one or the other as is appropriate for the language.

We rely on two multilingual test sets for evaluating every language: Common Voice 17 (Ardila et al., 2019) and Fleurs (Conneau et al., 2023). For some languages, we also evaluate using language-specific test sets. Arabic uses the test sets proposed by the Open Universal Arabic ASR Leaderboard (Wang et al., 2024), Japanese uses Reazon Speech (Fujimoto, 2016), Korean uses Zeroth-Korean (Jo & Lee, 2022), and Ukrainian uses Eurospeech (disco-eth, 2025).

To ensure consistent evaluations across datasets, models, and languages, we normalize both the reference and predicted transcription prior to calculating the error rate.

3.1. Results

Moonshine outperforms Whisper models between 1x and 28x its size. We first compare Moonshine Tiny with the similarly-sized Whisper Tiny model. Table 3 shows that the monolingual Moonshine Tiny models significantly outperform Whisper Tiny on all tests. Table 4 further illustrates the difference in error rate between Moonshine and progressively larger Whisper models (averaged across all evaluations for a language). These results show that Moonshine outperforms the 9x larger Whisper Small model, attaining performance that is on-par or better than the 28x larger Whisper Medium model.

There is an inherent tradeoff between model size and performance: more compute resources allow for larger models, which have a higher upper bound of performance. We compare this tradeoff using a measure of models’ “unit accuracy”, i.e., the inverse of error rate divided by model size $(100 - \text{Error})/(\# \text{ of Parameters})$. Figure 1 visualizes this tradeoff for Moonshine and Whisper, showing that Moonshine Tiny generally offers a superior tradeoff between compute requirements and accuracy than Whisper models its size and larger.

Performance scales loosely with dataset size. Figure 3 compares two checkpoints of Moonshine Tiny training (“Baseline” and “Final”) with increasing hours of training data. Model performance improves as the size of the training set increases, but the rate of increase is not consistent across languages. Vietnamese, for instance, has a large increase in data and a correspondingly large increase in performance; Korean, on the other hand, has an equally large increase in data, but the performance benefits are less significant.

Performance degrades gracefully with reduced gain and increased noise. Figure 2 depicts the error rate of each model when evaluated on Fleurs with varying linear gain and signal-to-noise ratio. This allows us to assess the ef-

	Model	Arabic	Chinese	Japanese	Korean	Ukrainian	Vietnamese
CV17	whisper-tiny	88.9	66.0	96.1	37.3	67.1	100.9
	moonshine-tiny	36.6	36.1	18.3	14.9	26.1	18.8
Fleurs	whisper-tiny	66.0	71.1	47.2	15.8	63.8	91.9
	moonshine-tiny	20.8	29.4	17.9	8.9	18.2	13.0

Table 3. Error rates of Moonshine Tiny and Whisper models across languages. Chinese, Japanese, and Korean results are CER; other languages are WER. Specialized Moonshine Tiny models outperform multilingual Whisper Tiny on all languages.

Comparison	Relative Size	Arabic	Chinese	Japanese	Korean	Ukrainian	Vietnamese
vs. whisper-tiny	1.4x	-42.4	-35.7	-81.0	-14.1	-47.1	-80.5
vs. whisper-base	2.7x	-36.4	-26.4	-57.0	-6.1	-28.9	-36.9
vs. whisper-small	9.0x	-12.1	-14.0	-25.3	-0.0	-6.2	-10.5
vs. whisper-medium	28.5x	1.0	-7.6	-12.2	2.2	3.2	-2.6

Table 4. Error rate difference between Moonshine Tiny and all Whisper models; lower is better. Chinese, Japanese, and Korean results are CER; other languages are WER. Moonshine Tiny outperforms Whisper Small, which is 9x larger; in some cases, Moonshine Tiny outperforms the 28x larger Whisper Medium model.

fect of input audio gain (quiet to loud) and audio noise on model performance. Performance trends lower as gain decreases and noise increases, though the models are robust to around -20 dB gain and 20 dB SNR, respectively. Interestingly, Ukrainian performance does not degrade more substantially than other languages despite being trained on a relatively larger amount of clean audio from synthetic speakers. End-to-end applications can leverage methods for input gain normalization and noise suppression to provide added robustness beyond the model’s baseline capabilities.

4. Discussion & Conclusion

This section discusses limitations and future work before concluding the paper.

Low and ultra-low resource languages. We aim to further expand our scope to low and ultra-low resource languages, leveraging similar data collection and pseudo-labeling techniques outlined in the paper. More extensive use of synthesis, as well as introduction of data augmentation techniques may be necessary to fill the gaps in raw data available.

Language-specific tokenizers. Moonshine currently relies on GPT-2’s multilingual tokenizer, which has a larger vocabulary size than necessary for individual languages. We expect that reducing the vocabulary size for each model will simplify the next-token prediction task, with potential benefits to accuracy and latency.

In summary, we introduce a family of 27M parameter Moonshine Tiny models that match or outperform the 28x larger Whisper Medium model across 6 languages. Our experiments show that small, monolingual, and specialized ASR

models are better suited for on-device tasks in underrepresented languages. To contribute to the broader research effort around these tasks, we release the open weights to all the models discussed in the paper with a permissive license.

References

- Abouelenin, A., Ashfaq, A., Atkinson, A., Awadalla, H., Bach, N., Bao, J., Benhaim, A., Cai, M., Chaudhary, V., Chen, C., et al. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*, 2025.
- Alharbi, S., Alowisheq, A., Tüske, Z., Darwish, K., Alrajeh, A., Alrowithi, A., Tamran, A. B., Ibrahim, A., Aloraini, R., Alnajim, R., et al. Sada: Saudi audio dataset for arabic. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 10286–10290. IEEE, 2024.
- Alnwsany, Y. M. Masrispeech-full: Large-scale egyptian arabic speech corpus, 2025. URL <https://huggingface.co/datasets/NightPrince/MasriSpeech-Full>.
- Ardila, R., Branson, M., Davis, K., Henretty, M., Kohler, M., Meyer, J., Morais, R., Saunders, L., Tyers, F. M., and Weber, G. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*, 2019.
- Bain, M., Huh, J., Han, T., and Zisserman, A. Whisperx: Time-accurate speech transcription of long-form audio. *INTERSPEECH 2023*, 2023.

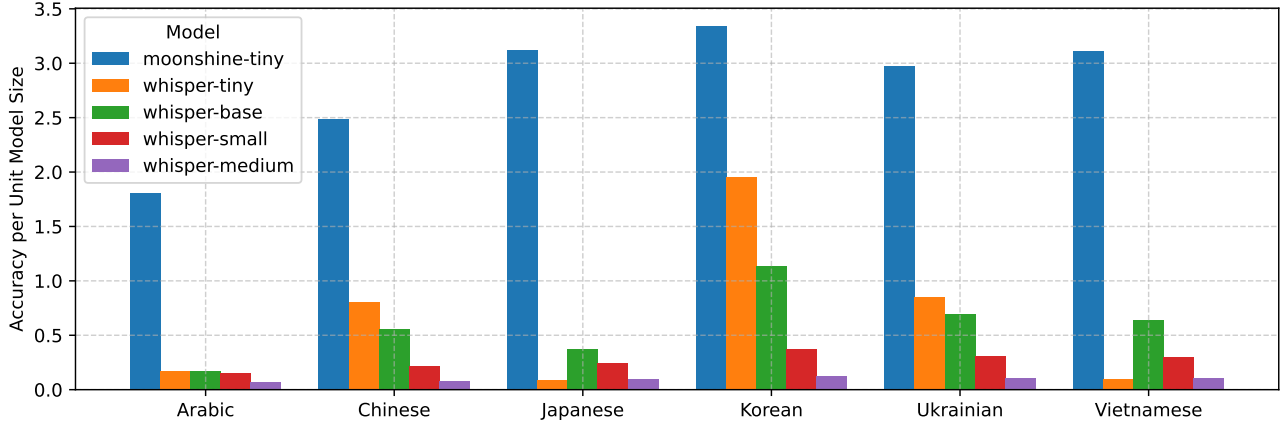


Figure 1. Difference in accuracy per model size between Moonshine and Whisper models. Model size places an upper bound on performance, and some models manage this tradeoff better than others. Moonshine offers a superior tradeoff between performance and size than Whisper models its size and larger.

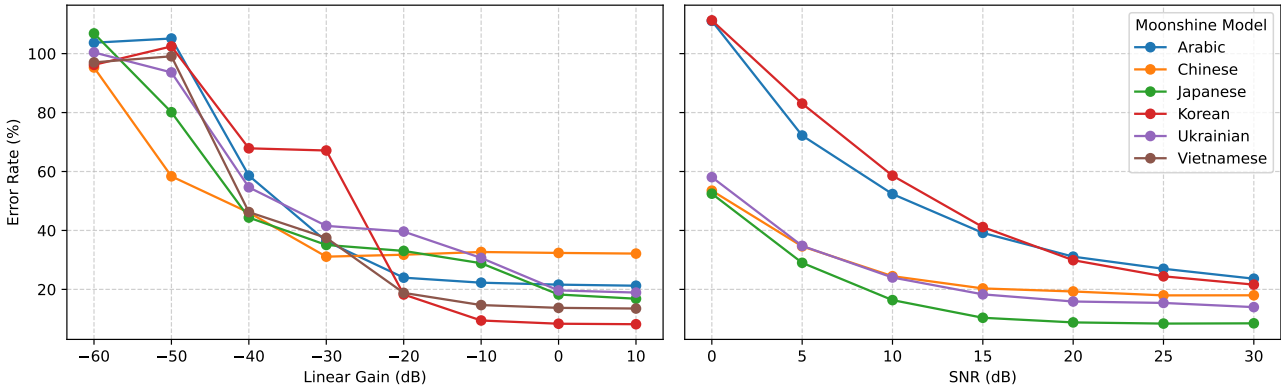


Figure 2. Effect of input audio gain and signal-to-noise ratio (SNR) on Moonshine Tiny error rate.

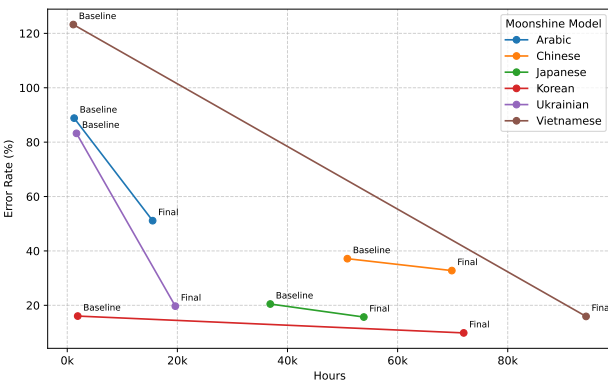


Figure 3. Error rate vs. hours of training data. Model performance scales loosely with training data hours, dependent on data quality.

Butt, S. A. Arabic tts wav 24k dataset. <https://huggingface.co/datasets/NeoBoy/arabic-tts-wav-24k>, 2025.

Cho, J., Baskar, M. K., Li, R., Wiesner, M., Mallidi, S. H., Yalta, N., Karafiat, M., Watanabe, S., and Hori, T. Multilingual sequence-to-sequence speech recognition: architecture, transfer learning, and language modeling. In *2018 IEEE SLT Workshop*, pp. 521–527. IEEE, 2018.

Chowdhury, S. A., Hussein, A., Abdelali, A., and Ali, A. Towards one model to rule all: Multilingual strategy for dialectal code-switching arabic asr, 2021.

Conneau, A., Ma, M., Khanuja, S., Zhang, Y., Axelrod, V., Dalmia, S., Riesa, J., Rivera, C., and Bapna, A. Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE SLT Workshop*. IEEE, 2023.

- Defazio, A., Yang, X., Mehta, H., Mishchenko, K., Khaled, A., and Cutkosky, A. The road less scheduled, 2024.
- disco-eth. EuroSpeech: A large-scale multilingual parliamentary speech corpus. Hugging Face Datasets, 2025. URL <https://huggingface.co/datasets/disco-eth/EuroSpeech>.
- Dumoulin, V., Shlens, J., and Kudlur, M. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016.
- Fujimoto, Y. Reazonspeech: A free and massive corpus for japanese asr. 2016.
- He, H., Shang, Z., Wang, C., Li, X., Gu, Y., Hua, H., Liu, L., Yang, C., Li, J., Shi, P., et al. Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation. In *2024 IEEE SLT Workshop*, pp. 885–890. IEEE, 2024.
- Jeffries, N., King, E., Kudlur, M., Nicholson, G., Wang, J., and Warden, P. Moonshine: Speech recognition for live transcription and voice commands. *arXiv preprint arXiv:2410.15608*, 2024.
- Jo, L. and Lee, W. Zeroth-Korean: Korean Open-source Speech Corpus for Speech Recognition (SLR40). Open Speech and Language Resources (OpenSLR), 2022. Available at <https://openslr.org/40/>.
- King, E., Yu, H., Vartak, S., Jacob, J., Lee, S., and Julien, C. Teaching things to think: Bootstrapping local reasoning for smart (er) devices. In *2025 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 78–88. IEEE, 2025.
- Kulkarni, A., Kulkarni, A., Shatnawi, S. A. M., and Aldarmaki, H. Clartts: An open-source classical arabic text-to-speech corpus. In *2023 INTERSPEECH*, pp. 5511–5515, 2023. doi: 10.21437/Interspeech.2023-2224.
- Le, T.-T., Nguyen, L. T., and Nguyen, D. Q. Phowhisper: Automatic speech recognition for vietnamese. *arXiv preprint arXiv:2406.02555*, 2024.
- Liao, H., Ni, Q., Wang, Y., Lu, Y., Zhan, H., Xie, P., Zhang, Q., and Wu, Z. Nvspeech: An integrated and scalable pipeline for human-like speech modeling w/ paralinguistic vocalizations. *arXiv preprint arXiv:2508.04195*, 2025.
- Luong, H.-T. and Vu, H.-Q. A non-expert Kaldi recipe for Vietnamese speech recognition system. In *Proceedings of the Third International WLSI / OIAF 4 HLT 2016*, pp. 51–55, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.
- Namvarpour, M. and Razi, A. The art of talking machines: A comprehensive literature review of conversational user interfaces. In *Proceedings of the 7th ACM Conference on CUI*, pp. 1–18, 2025.
- Nhut, P. Q., Anh, D. P. H., and Tiep, N. V. Vietspeech: Vietnamese social voice dataset, 2024. URL <https://github.com/NhutP/VietSpeech>.
- Obeid, O., Zalmout, N., Khalifa, S., Taji, D., Oudah, M., Alhafni, B., Inoue, G., Eryani, F., Erdmann, A., and Habash, N. CAMEL tools: An open source python toolkit for Arabic natural language processing. In *Proceedings of the 12th LREC*, May 2020.
- Pratap, V., Sriram, A., Tomasello, P., Hannun, A., Liptchinsky, V., Synnaeve, G., and Collobert, R. Massively multilingual asr: 50 languages, 1 model, 1 billion parameters. *arXiv preprint arXiv:2007.03001*, 2020.
- Puvvada, K. C., Żelasko, P., Huang, H., Hrinchuk, O., Koluguri, N. R., Dhawan, K., Majumdar, S., Rastorgueva, E., Chen, Z., Lavrukhin, V., et al. Less is more: Accurate speech recognition & translation without web-scale data. *arXiv preprint arXiv:2406.19674*, 2024.
- Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. Robust speech recognition via large-scale weak supervision. In *ICML*. PMLR, 2023.
- Rege, A., Currano, R., Sirkin, D., and Kim, E. ‘talking with your car’: Design of human-centered conversational ai in autonomous vehicles. In *Proceedings of the 16th International Conference on AutoUI*, pp. 338–349, 2024.
- Smoliakov. opentts-uk (revision 32abc9c), 2025. URL <https://huggingface.co/datasets/Yehor/opentts-uk>.
- Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.
- Toshniwal, S., Sainath, T. N., Weiss, R. J., Li, B., Moreno, P., Weinstein, E., and Rao, K. Multilingual speech recognition with a single end-to-end model. In *2018 IEEE ICASSP*, pp. 4904–4908. IEEE, 2018.
- Wang, Y., Alhmoud, A., and Alqurishi, M. Open Universal Arabic ASR Leaderboard. *arXiv preprint arXiv:2412.13788*, 2024.
- Yang, Y., Song, Z., Zhuo, J., Cui, M., Li, J., Yang, B., Du, Y., Ma, Z., Liu, X., Wang, Z., Li, K., Fan, S., Yu, K., Zhang, W.-Q., Chen, G., and Chen, X. Gigaspeech 2: An evolving, large-scale and multi-domain asr corpus for low-resource languages with automated crawling, transcription and refinement. *arXiv preprint arXiv:2406.11546*, 2024.

A. Public Datasets

Language	Dataset (Hugging Face)	Hours
Arabic	NeoBoy/arabic-tts-wav-24k (Kulkarni et al., 2023; Butt, 2025)	12
	nadsoft/arabic-98	163
	MAde1121/arabic-egy-cleaned	72
	Sabri12blm/Arabic-Quran-ASR-dataset	360
	MohamedRashad/SADA22 (Alharbi et al., 2024)	647
	NightPrince/MasriSpeech-Full (Alnwsany, 2025)	3100
	xmodar/commonvoice-12.0-arabic-voice-converted (Ardila et al., 2019)	148
	google/fleurs (Conneau et al., 2023)	116
Chinese	Hannie0813/NVSpeech170k (Liao et al., 2025)	573
	amphion/Emilia-Dataset (He et al., 2024)	50300
Japanese	reason-research/reazonspeech (Fujimoto, 2016)	35000
	kadarnar/Combined-Japanese-TTS	995
	mozilla-foundation/common_voice_17_0 (Ardila et al., 2019)	610
	amphion/Emilia-Dataset (He et al., 2024)	266
Korean	brainer/korean-telemedicine-speech	1120
	imTak/korean-audio-text-economy	39
	Junhoe/STT_Korean_Dataset	267
	JaepaX/korean_dataset	424
	Bingsu/zeroth-korean (Jo & Lee, 2022)	52
	jp1924/GyeongsangSpeech	2546
	amphion/Emilia-Dataset (He et al., 2024)	7500
	jp1924/KoreanUniversityLectureData	4000
	jp1924/KoreaSpeech	3630
	jp1924/KconfSpeech	2970
	jp1924/KrespSpeech	2907
	jp1924/JeollaSpeech	2121
Ukrainian	disco-eth/EuroSpeech (disco-eth, 2025)	1287
	speech-uk/voice-of-america (Smoliakov, 2025)	391
Vietnamese	NhutP/VietSpeech (Nhut et al., 2024)	1100
	speechcolab/gigaspeech2 (Yang et al., 2024)	7324

Table 5. Publicly-available training datasets. We include citations for datasets that have an associated paper, or that have citation instructions on the repo at time of writing.

B. Normalization Steps

This section outlines the normalization steps for each language.

To normalize Arabic, we leveraged the normalization code provided by (Chowdhury et al., 2021; Wang et al., 2024). To be specific, we removed punctuation and diacritics, and converted Eastern Arabic numerals to Western Arabic numerals. For Korean, we leveraged the normalizing steps proposed by Zeroth³. To normalize Korean, we removed all punctuation (including brackets) and converted Western Arabic numbers to their phonetic spellings. For Japanese, we followed the

³<https://github.com/goodatlas/zeroth>

neologism dictionary for MeCab ⁴. Specifically, we leverage the default normalizer ⁵. For Ukrainian, Vietnamese, and Chinese, we leverage the basic text normalizer provided by Whisper.

C. Complete Results

Model	WER				CER			
	CV17	Casablanca	Fleurs	SADA22	CV17	Casablanca	Fleurs	SADA22
whisper-medium	36.09	76.2	18.79	69.48	12.4	39.16	5.82	44.39
whisper-small	48.02	87.22	29.09	88.55	17.39	46.19	9.1	57.08
whisper-base	80.35	108.33	50.15	111.39	38.21	63.16	17.7	73.92
whisper-tiny	88.89	109.32	66.01	109.76	44.02	66.98	25.7	72.93
moonshine-tiny (Baseline)	77.54	97.8	79.16	100.82	40.41	60.75	40.97	61.5
moonshine-tiny (Final)	36.55	80.62	20.76	66.55	12.94	43.1	7.62	35.38

Table 6. Arabic

Model	CER	
	CV17	Fleurs
whisper-medium	28.06	52.75
whisper-small	35.46	58.06
whisper-base	54.11	64.14
whisper-tiny	65.92	71.1
moonshine-tiny (Baseline)	22.17	52.13
moonshine-tiny (Final)	36.1	29.44

Table 7. Chinese

Model	CER		
	CV17	Fleurs	Reason Speech
whisper-medium	29.09	11.5	43.06
whisper-small	37.69	16.76	68.37
whisper-base	71.13	27.13	119.81
whisper-tiny	96.11	47.2	146.81
moonshine-tiny (Baseline)	30.99	19.16	11.28
moonshine-tiny (Final)	18.3	17.87	10.89

Table 8. Japanese

⁴<https://github.com/neologd/mecab-ipadic-neologd>

⁵<https://github.com/ikegami-yukino/neologdn>

Model	WER			CER		
	CV17	Fleurs	Zeroth	CV17	Fleurs	Zeroth
whisper-medium	26.93	16.18	21.83	9.38	6.99	6.66
whisper-small	33.38	19.56	27.95	12.32	8.47	8.82
whisper-base	45.45	27.16	43.56	19.42	11.95	16.42
whisper-tiny	63.79	35.44	47.84	37.27	15.83	18.66
moonshine-tiny (Baseline)	55.46	29.1	12.48	28.89	13.52	5.72
moonshine-tiny (Final)	35.22	20	16.38	14.94	8.9	5.72

Table 9. Korean

Model	WER		CER	
	CV17	Fleurs	CV17	Fleurs
whisper-medium	23.54	13.45	12.28	6.78
whisper-small	30.31	22.61	15.19	11.09
whisper-base	59.54	46.03	35.54	22.79
whisper-tiny	100.9	91.89	72.46	61.18
moonshine-tiny (Baseline)	149.15	97.33	115.51	82.25
moonshine-tiny (Final)	18.83	13.01	10.26	6.74

Table 10. Vietnamese

Model	WER			CER		
	CV17	Eurospeech	Fleurs	CV17	Eurospeech	Fleurs
whisper-medium	20.9	17.01	11.62	5.93	8.99	3.68
whisper-small	32.44	24.94	20.41	9.03	11.1	5.33
whisper-base	55.55	46.06	44.08	18.4	18.6	14.51
whisper-tiny	67.07	69.41	63.83	24.03	28.93	20.69
moonshine-tiny (Baseline)	92.49	64.93	92.28	63.49	44.21	68.12
moonshine-tiny (Final)	26.11	14.73	18.25	8.57	7.81	6.43

Table 11. Ukrainian