



HEBDB: a Weakly Supervised Dataset for Hebrew Speech Processing

Arnon Turetzky¹, Or Tal¹, Yael Segal-Feldman², Yehoshua Dissen², Ella Zeldes¹, Amit Roth¹, Eyal Cohen², Yosi Shrem², Bronya R. Chernyak², Olga Seleznova², Joseph Keshet², Yossi Adi¹

¹The Hebrew University of Jerusalem, Israel

²Technion, Israel Institute of Technology

arnon.turetzky@mail.huji.ac.il

Abstract

We present HEBDB, a weakly supervised dataset for spoken language processing in the Hebrew language. HEBDB offers roughly 2500 hours of natural and spontaneous speech recordings in the Hebrew language, consisting of a large variety of speakers and topics. We provide raw recordings together with a pre-processed, weakly supervised, and filtered version. The goal of HEBDB is to further enhance research and development of spoken language processing tools for the Hebrew language. Hence, we additionally provide two baseline systems for Automatic Speech Recognition (ASR): (i) a self-supervised model; and (ii) a fully supervised model. We present the performance of these two methods optimized on HEBDB and compare them to current multi-lingual ASR alternatives. Results suggest the proposed method reaches better results than the evaluated baselines considering similar model sizes. Dataset, code, and models are publicly available under <https://pages.cs.huji.ac.il/adiyoss-lab/HebDB/>.

Index Terms: Automatic Speech Recognition, Speech Benchmark, Hebrew Speech Technologies

1. Introduction

Spoken language technologies have seen a great leap in performance following the success of deep neural networks consisting of large-scale models [1, 2, 3] and datasets [4, 5, 6]. This includes Automatic Speech Recognition (ASR) [7, 8, 9], Text-to-Speech (TTS) [10, 11], speech enhancement [12, 13], speaker diarization [14], to name a few.

A fundamental requirement in the success of the aforementioned models is optimization using large-scale datasets [1]. For instance, when considering ASR, Whisper [7] was trained using $\sim 700k$ hours of speech utterances and Google USM was trained over $\sim 12M$ hours of speech recordings. As for TTS, both VALL-E and VoiceBox were trained over 60k hours of speech. As a result, such big performance advancements are mainly kept for high-resource language in which large-scale datasets can be found.

One approach to mitigate the performance gaps between high-resource and low-resource languages is to train speech models considering multi-lingual setups [7, 8]. The authors in [15] empirically demonstrated the benefit of training multi-lingual spoken language processing models, while the authors in [8] specifically demonstrated the benefit of low-resource languages. Although such performance improvements are interesting and important, directly training models over large-scale benchmarks still achieve superior performance [16].

The Hebrew language is among the low-resource languages explored in prior work [7, 8]. The Hebrew language is being spoken by roughly 9 million people worldwide [17]. Besides

the lack of large-scale datasets in Hebrew, the language syntax and structure impose some inheriting challenges, such as: (i) using non-Latin letters, which sets it apart from many languages; (ii) Traditional Hebrew has diacritics (“Nikud”) while modern Hebrew writing rarely uses them. Such discrepancies impose a critical challenge on ASR and TTS systems which need to learn non-trivial pronunciations that directly affect word meanings. Such differences are not presented in writings but sound differently, and can be distinguished mainly based on context. For instance the word “pita” can have two meanings: bread and seduced based on the context; (iii) Hebrew is a morphologically rich language, with common use of prefixes and suffixes to modify words’ meanings and to add prepositions. This property makes tokenization difficult and less efficient, especially under the multi-lingual setup [18].

In this study, we present HEBDB, a weakly supervised spontaneous speech dataset in the Hebrew language. HEBDB is comprised of ~ 2500 hours of in-the-wild natural speech consisting of a numerous number of speakers and diverse topics and vocabulary. We release both the raw recordings together with a pre-processed and weakly transcribed version. We additionally provide a transcription confidence score for each of the data samples, which can be used to develop strategies for fine-tuning considering different supervision qualities. In releasing this dataset, our goal is to advance research and development of Artificial Intelligence (AI) based tools for spoken language processing directly developed for the Hebrew language. To further enhance the development of such tools, we provide two baseline systems: (i) a self-supervised model; and (ii) a fully supervised ASR model. Full dataset, code, and models are publicly available under <https://pages.cs.huji.ac.il/adiyoss-lab/HebDB/>.

The paper is structured as follows. We start by reviewing datasets and speech processing tools directly dedicated to the Hebrew language in Section 2. Next, in Section 3 we provide a detailed description of HEBDB, its curation, statistics, pre-processing, and supervision quality assessment. In Section 4, we describe the baseline systems and compare their performance to current open-source tools. We conclude the paper in Section 5, where we outline future work along this research direction.

2. Related work

Spoken Hebrew benchmarks. As Hebrew is considered a low-resource language, public spoken benchmarks hardly exist. Previous efforts in constructing datasets in Hebrew were either released under a multi-lingual benchmark [19, 20, 8, 21] or relatively small [22, 23, 24, 25]. The authors in [22] established the Corpus of Spoken Israeli Hebrew (CoSIH) with the

Table 1: *Details & Statistics of HEBDB’s raw recordings. We report the list of sources, sampling rates, num of channels, total duration (in hours), and indication of single or multiple speakers in a given source. Both sampling rates and channels are reported percentage.*

Source	Sample rate (kHz)	# Channels	Duration (h)	Single / Multiple speakers
Geekonomy	44.1: 60%; 48: 40%	Stereo: 67% ; Mono: 33%	1146	Multiple speakers
Osim History	44.1: 95%; 48: 3%; 22.05: 2%	Stereo: 93% ; Mono: 7%	270	Mostly single speaker
The Dor Kahn Experience (DKE)	44.1: 100%	Stereo: 100%	218	Multiple speakers
Yo! The podcast	44.1: 25%, 48: 75%	Stereo: 25% ; Mono: 75%	295	Multiple speakers
Good Question	16: 100%	Stereo: 100%	163	Multiple speakers
Yad vashem	48: 100%	Stereo: 100%	492	Multiple speakers

goal of compiling a large database of recordings of spoken Israeli Hebrew in order to facilitate and enhance research in the field. Next, the authors in [23] released The Map Task Corpus (MaTaCOP) of Hebrew dialogues. The authors in [24] collected naturally occurring speech and interaction in Modern Hebrew via telephone conversations during the years 2020–2021 and released the HUJI Corpus of Spoken Hebrew (HUJICorpus). More recently, the authors in [25] released SASPEECH, a high-quality single-speaker Hebrew dataset which is goal is to enhance Hebrew speech synthesis research. Although all of these prior work are important and valuable, the provided benchmarks are relatively small. CoSIH contains ~ 12.3 hours of speech, the MaTaCOP corpus contains ~ 5.3 hours, the HUJICorpus has ~ 3.8 , and SASPEECH which is the largest one contains ~ 30 hours of speech. The most relevant concurrent work to ours is the great work done by [26], which released a dataset denoted as *ivrit.ai*. The authors released ~ 3300 hours of speech from local podcasts and provided the first large-scale dataset in Hebrew. We would like to state that the proposed benchmark is orthogonal to the release of *ivrit.ai*. We believe the community should leverage as many high-quality publicly available datasets as possible to close the gap between low- to high-resource languages. Additionally, unlike *ivrit.ai*, we release two baseline systems (SSL and supervised one) for speech processing and ASR.

Hebrew ASR. With recent advancements in multi-lingual ASR systems, we also observe improvements in Hebrew ASR. The authors in [7] release the Whisper family of models that were trained on $\sim 700k$ hours of labeled data from 100 languages including Hebrew. The authors publicly released models ranging in size from 40M to 1.55B parameters. Later on, the authors in [8] released the *Massively Multi-lingual Speech* (MMS) project which provides speech models for 1107 languages including Hebrew. In this work, we compare the proposed baseline systems trained on HEBDB to both Whisper and MMS.

3. HEBDB dataset

HEBDB contains natural dialogues of spontaneous speech. It is comprised of both testimonies from World War II survivors and five podcasts covering a wide range of subjects and speakers. While the testimonies provide firsthand accounts of historical events, the majority of our dataset consists of podcasts covering diverse topics such as economy, politics, sports, culture, science, history, and music, to name a few. This combination of personal narratives and informative discussions offers a rich and varied resource for analysis and interpretation.

We provide two versions of the dataset: *raw* and *pre-processed*. The raw version contains over 2584 hours of in-the-wild audio in varying sample rates, recorded channels, and number of speakers. Table 1 provides a detailed description of

this version including statistics. We release this version to allow researchers and practitioners to explore different pre-processing alternatives and methods.

The pre-processed version contains roughly 1690 hours of audio, down-sampled, segmented into multiple files, and auto-transcribed. This version is better suited for training acoustic models as is. We optimize and evaluate the proposed baseline systems using the pre-processed version only. In the next subsection, we provide a detailed description of the pre-processing pipeline used.

Both versions of HEBDB corpus are released under the very permissive of CC BY 4.0 license [27].

3.1. Pre-processing

The raw recordings are constructed from full podcast episodes and testimonies and, hence, contain long audio sources and plenty of non-speech segments, e.g. music, environmental sounds, silence, etc. Such in-the-wild conditions make model optimization challenging and require a pre-processing step.

To handle that, we apply the following pre-processing pipeline to the raw version of HEBDB. We first resample all the audio recordings to 16kHz, mono recordings, using `julius`¹ python package. Next, we apply a Voice Activity Detection (VAD) model to partition the waveform to sentences and discard empty and noisy parts. Lastly, we automatically transcribe the segmented speech utterances using a pre-trained ASR model.

Voice activity detection and speech segmentation. We use the `silero-vad` [28] to perform voice activity detection over the 16KHz audio files. Unlike traditional VAD models [29] that are based on Digital Signal Processing heuristics, the ‘`silero-vad`’ is a learning model based on convolutional and LSTM layers². We specifically chose the `silero-vad` as it provides superior performance to other publicly available VAD models and was found beneficial in prior work [26, 25].

In general, VAD relies on frame-wise activity classification. Following that, to properly segment the audio into sentences, we need to calibrate a classification threshold over the model’s frame-wise confidence scores and define a minimal duration of silence between activated segments. We follow this process as we wish to have a minimal number of words in each segment while keeping its length to fit in a processing unit memory.

Specifically, we use a confidence threshold of 0.5 to filter out activated segments with a minimum duration of 1 seconds, separating audio segments by a minimal silence duration of 100ms and padding both sides of the segmented audio with 30ms of silence.

Transcriptions. We provide weak supervision in the form of

¹<https://github.com/adefossez/julius>

²<https://github.com/snakers4/silero-vad>

Table 2: Details of HEBDB’s pre-processed version. We report the total duration (in hours) for each source together with statistics of the processed utterances (in seconds).

Source	Duration (h)	Mean / Min / Max / Med (s)
Geekonomy	998.5	4.7 / 1.0 / 282.1 / 3.2
Osim History	223.9	4.4 / 1.1 / 152.2 / 3.1
DKE	181.4	3.8 / 1.1 / 107.6 / 2.9
Yo! the podcast	142.2	3.3 / 1.1 / 115.7 / 2.5
Good Question	75.0	5.0 / 1.1 / 175.5 / 3.2
Yad Vashem	67.4	5.3 / 1.0 / 361.1 / 4.4

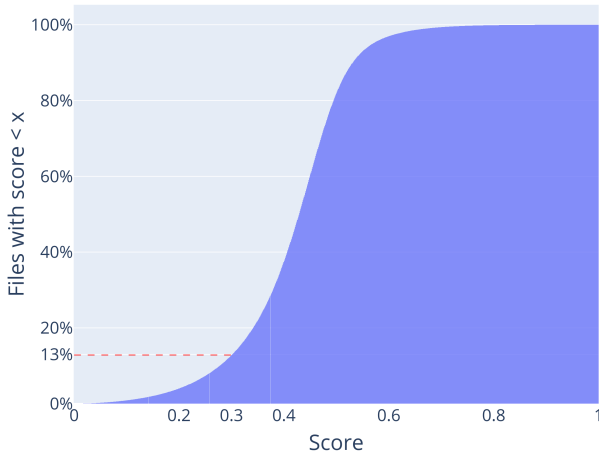


Figure 1: Score level histogram of the data filtering process. We use a threshold of 0.3 which filters roughly 13% of the data.

transcriptions. We leverage the pre-trained Whisper large-v2 (1.55B) version to transcribe all the segmented data. Although Whisper supports transcription in specific languages it might output non-Hebrew characters not limited to Latin. Analyzing the frequencies of character across our train set, most of the non-Hebrew chars were found in 1%, hence we removed those samples from our data. Additionally, Whisper might output an <RTL> token, as this token is not relevant to the acoustics of the speech utterance we simply remove it from the text. For better alignment between acoustics and written text, we converted numbers and dates to words using the num2words package³. Lastly, Hebrew has 5 special letters with final form, we experimented with normalizing the final instances to regular ones and found it to be beneficial. Hence, we adjust the 5-gram LM provided by the MMS to normalize accordingly.

Statistics. After the preprocessing step, we are left with ~ 1690 hours of speech partitioned into varied length segments with the vast majority of the segmented files having less than 10 seconds. Table 2 shows a further subdivision of processed audio with respect to each source separately. Figure 2 depicts a box plot for processed instances quartile distributions over audio duration in seconds and the number of transcribed words with respect to each source, discarding outliers.

Notice, that the pre-processing step did not affect all sources equally. For instance, the Yad vashem source was reduced from 492 hours to 67.4, this is due to bad recording conditions and long silences at the beginning or end of the files.

³<https://pypi.org/project/num2words/>

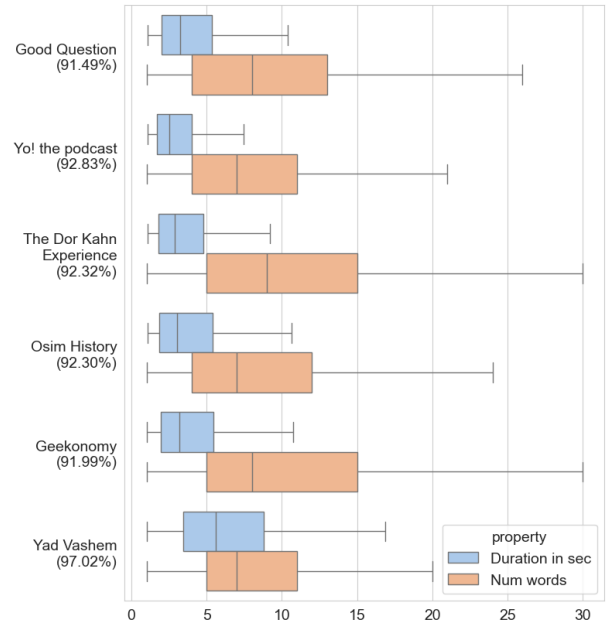


Figure 2: A boxplot of the processed data, percentages denote the corresponding portion of the unfiltered non-outlier instances with respect to each source. We provide statistics for both audio recordings (in hours) and transcriptions (in words count).

3.2. Data filtering

To further enhance the reliability of our transcripts, we employ a forced aligner using an alternative model, specifically the MMS model [30]. This model requires the input text to be in transliterated Latin script for accurate alignment. We achieved this using the Uroman package⁴, which romanizes text from most languages into the Latin alphabet. However, Uroman’s performance dips with non-diacritized text, prompting us to first diacritize all transcriptions using the UNIKUD package⁵, a tool specifically designed to add necessary diacritical marks to Hebrew text, thus ensuring higher transliteration accuracy.

We utilized the forced aligner to generate a confidence score for each utterance, calculated by averaging the confidence scores of individual words. These utterance-level scores were used to filter out lower-quality data, aiming to train our models on high-quality data only. The confidence scores for Hebrew utterances were notably lower on average than for English, with a mean score of 0.417 and a std of 0.11. We set a threshold of 0.3 for the confidence scores to determine the data quality cutoff. Initially, our dataset comprised ~ 1690 hours of speech. After applying the threshold for filtering, we retained 1470 hours of speech with a mean score of 0.447 and a std of 0.08, considered as reliable transcripts for training. Figure 1 presents a histogram of the forced aligner scores.

4. Baseline system

4.1. Implementation details

We provide two baseline systems together with HEBDB. The first one is an SSL model, namely HuBERT [31]. The second

⁴<https://github.com/isi-nlp/uroman>

⁵<https://pypi.org/project/unikud/>

model is a fully supervised one, namely Conformer [32]. Both models were optimized using HEBDB and evaluated on the Hebrew subset from the Fleurs benchmark [21]. Both models were evaluated with and without a language model (LM). We use 5-gram LM provided by the MMS project ⁶.

HuBERT. We train a HuBERT-base with $\sim 95\text{M}$ for two iterations following the standard recipe for 'pretrain' outlined in the fairseq framework [33]⁷. In the first iteration, we utilize 100 clusters generated from 10% of the data using KM-clustering on MFCC features. The model is trained on 4 A5000 GPUs, using gradient accumulation to match the original recipe's specifications of 250k training steps across 32 GPUs. For the second iteration, we increase the number of clusters to 500 and use representations obtained from the 6th layer, still utilizing gradient accumulation but for 400k training steps. After HuBERT pre-training, we employ the connectionist temporal classification (CTC) loss [34] for ASR fine-tuning for 150K steps.

Conformer. The Conformer used is based on the model introduced by Gulati et al [35] trained with the CTC loss and a character tokenizer using 8 A40 GPUs. The model is similar to the large model in [35] with $\sim 100\text{M}$ parameters. The Conformer model's hyper-parameters are as follows: convolution kernel size 31, n-heads 8, hidden-dim 512, 17 layers, and dropout 0.1. The model is fed Mel-spectrograms of 80 filters, with a window size of 25ms and stride of 10ms. We employ time and frequency masking as augmentation techniques during training. We utilize a Noam optimizer [36] with 10,000 warmup steps. Our batch size is measured in audio length, consisting of utterances with lengths ranging from 1 to 30 seconds, cumulatively not exceeding 300 seconds per batch. Finally, we trained for a total of 160k steps for the full training and 140k for the filtered training.

4.2. Results

We compare the previously mentioned baseline systems to both Whisper [7] and MMS [8] family of models. We consider Whisper using 39M, 74M, 244M, 769M, and 1.55B parameter models. For MMS we consider models trained on 61 languages and 1, 107 languages. All MMS models contain 1.5B parameters. Notice, that although Whisper and MMS models are multi-lingual, both were optimized over significantly larger datasets.

Table 3 presents the Word-Error-Rates (WER) results computed over the Fleurs [21] benchmark. When considering comparison to Whisper models the provided baseline systems reach comparable or superior performance up until model size of 769M parameters. When scaling the model size to 1.55B, Whisper provides better performance while being between ~ 15 times bigger. Although providing worse performance, we believe such models and results are interesting and valuable to the community as these could be important for use cases where performance can be compromised over significantly smaller models [37]. When comparing to MMS, the Conformer model trained on HEBDB provides comparable performance while HuBERT was found to be superior. Notice, that both Conformer and HuBERT are significantly smaller than the MMS model ($\sim 15\text{x}$ smaller). When comparing HuBERT and Conformer, results suggest HuBERT provides superior performance (8 – 9 absolute points) with and without LM decoding.

⁶<https://github.com/facebookresearch/fairseq/blob/main/examples/mms/README.md>

⁷<https://github.com/facebookresearch/fairseq/blob/main/examples/hubert/README.md>

Table 3: WER results over the Fleurs [21] benchmark. Results are reported for the provided baseline systems together with Whisper and MMS considering different setups. We provide results with and without LM and data filtering (df) whenever possible. The same LM was used in all of the reported results.

Model	#param	LM	WER
Whisper	39M	-	71.6
Whisper	74M	-	61.7
Whisper	244M	-	44.4
Whisper	769M	-	33.1
Whisper	1.55B	-	27.1
MMS (61 lang)	1.5B	✗	66.6
MMS (61 lang)	1.5B	✓	41.5
MMS (1,107 lang)	1.5B	✗	67.1
MMS (1,107 lang)	1.5B	✓	40.0
HEBDB-HuBERT	95M	✗	41.6
HEBDB-HuBERT	95M	✓	33.9
HEBDB-HuBERT (w. df)	95M	✗	41.0
HEBDB-HuBERT (w. df)	95M	✓	33.6
HEBDB-Conformer	110M	✗	49.5
HEBDB-Conformer	110M	✓	43.0
HEBDB-Conformer (w. df)	110M	✗	48.4
HEBDB-Conformer (w. df)	110M	✓	41.9

Next, we evaluate the effect of the data filtering. We train both HuBERT and Conformer models using the filtered version as presented in Section 3.2. Notice, under this setup, the SSL, and pretraining part of HuBERT were still optimized using the whole dataset, while we modify only the fine-tuning part to use the filtered version only. Results are presented in Table 3 (bottom rows). Results suggest that the data filtering provides a small improvement for both HuBERT and the Conformer models. This suggests that overall in our dataset there is enough signal from the weak supervision to construct a performing acoustic model, however, there is still room for improvement in data quality assessment. We hope the speech community will adopt and develop such techniques in future research.

5. Conclusion & future work

In this work, we present HEBDB, a weakly supervised dataset in the Hebrew language, aimed at improving the development of AI-based speech processing tools directly dedicated to Hebrew. To further enhance the development of speech processing tools for Hebrew, we additionally provide two baseline systems, a self-supervised one and a fully supervised ASR acoustic model. Both HEBDB and the pre-trained models are released under the CC BY 4.0 license. We hope the community will adopt such datasets and baseline systems together with other efforts in the field to advance the automatic development of speech-processing tools in Hebrew.

For future work, we plan to extend this dataset and provide higher-quality annotations in the form of transcriptions, speaker annotations, etc. Additionally, we plan to provide a subset of high-fidelity recordings which will be used to develop systems for generative tasks such as text-to-speech and voice conversion in Hebrew.

Acknowledgements This research work was supported by the Israel Innovation Authority, grant number 78563.

6. References

- [1] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.
- [2] P. K. Rubenstein, C. Asawaroengchai, D. D. Nguyen, A. Bapna, Z. Borsos, F. d. C. Quitry, P. Chen, D. E. Badawy, W. Han, E. Kharitonov *et al.*, “Audiopalm: A large language model that can speak and listen,” *arXiv preprint arXiv:2306.12925*, 2023.
- [3] C. Wang, Y. Wu, Y. Qian, K. Kumatani, S. Liu, F. Wei, M. Zeng, and X. Huang, “Unispeech: Unified speech representation learning with labeled and unlabeled data,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 10937–10947.
- [4] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: an asr corpus based on public domain audio books,” in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.
- [5] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen *et al.*, “Libri-light: A benchmark for asr with limited or no supervision,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7669–7673.
- [6] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, “Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation,” *arXiv preprint arXiv:2101.00390*, 2021.
- [7] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust speech recognition via large-scale weak supervision,” in *International Conference on Machine Learning*. PMLR, 2023, pp. 28 492–28 518.
- [8] V. Pratap, A. Tjandra, B. Shi, P. Tomasello, A. Babu, S. Kundu, A. Elkahky, Z. Ni, A. Vyas, M. Fazel-Zarandi *et al.*, “Scaling speech technology to 1,000+ languages,” *arXiv preprint arXiv:2305.13516*, 2023.
- [9] Y. Zhang, W. Han, J. Qin, Y. Wang, A. Bapna, Z. Chen, N. Chen, B. Li, V. Axelrod, G. Wang *et al.*, “Google usm: Scaling automatic speech recognition beyond 100 languages,” *arXiv preprint arXiv:2303.01037*, 2023.
- [10] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li *et al.*, “Neural codec language models are zero-shot text to speech synthesizers,” *arXiv preprint arXiv:2301.02111*, 2023.
- [11] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, M. Williamson, V. Manohar, Y. Adi, J. Mahadeokar *et al.*, “Voicebox: Text-guided multilingual universal speech generation at scale,” *Advances in neural information processing systems*, vol. 36, 2024.
- [12] C. K. Reddy, H. Dubey, V. Gopal, R. Cutler, S. Braun, H. Gamper, R. Aichner, and S. Srinivasan, “Icassp 2021 deep noise suppression challenge,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6623–6627.
- [13] A. Defossez, G. Synnaeve, and Y. Adi, “Real time speech enhancement in the waveform domain,” *arXiv preprint arXiv:2006.12847*, 2020.
- [14] Y. Dissen, F. Kreuk, and J. Keshet, “Self-supervised Speaker Diarization,” in *Proc. Interspeech 2022*, 2022, pp. 4013–4017.
- [15] H. Yadav and S. Sitaram, “A survey of multilingual models for automatic speech recognition,” *arXiv preprint arXiv:2202.12576*, 2022.
- [16] T. Likhomanenko, Q. Xu, V. Pratap, P. Tomasello, J. Kahn, G. Avidov, R. Collobert, and G. Synnaeve, “Rethinking evaluation in asr: Are our models robust enough?” *arXiv preprint arXiv:2010.11745*, 2020.
- [17] L. Campbell, “Ethnologue: Languages of the world,” 2008.
- [18] A. Petrov, E. La Malfa, P. Torr, and A. Bibi, “Language model tokenizers introduce unfairness between languages,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [19] A. W. Black, “Cmu wilderness multilingual speech dataset,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5971–5975.
- [20] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, “Mls: A large-scale multilingual dataset for speech research,” *arXiv preprint arXiv:2012.03411*, 2020.
- [21] A. Conneau, M. Ma, S. Khanuja, Y. Zhang, V. Axelrod, S. Dalmia, J. Riesa, C. Rivera, and A. Bapna, “Fleurs: Few-shot learning evaluation of universal representations of speech,” in *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2023, pp. 798–805.
- [22] S. Izre’el, B. Hary, and G. Rahav, “Designing cosih: the corpus of spoken israeli hebrew,” *International Journal of Corpus Linguistics*, vol. 6, no. 2, pp. 171–197, 2001.
- [23] J. Azogui, A. Lerner, and V. Silber-Varod, “The open university of israel map task corpus (matacop),” 2016.
- [24] M. Marmorstein and N. Matalon, “The huji corpus of spoken hebrew: An interaction-oriented design of a corpus,” 2022.
- [25] O. Sharoni, R. Shenberg, and E. Cooper, “Saspeech: A hebrew single speaker dataset for text to speech and voice conversion,” in *Proc. Interspeech*, 2023.
- [26] Y. Marmor, K. Misgav, and Y. Lifshitz, “ivrit. ai: A comprehensive dataset of hebrew speech for ai research and development,” *arXiv preprint arXiv:2307.08720*, 2023.
- [27] C. Commons, “Creative commons attribution 4.0 international public license.”
- [28] S. Team, “Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier,” 2021.
- [29] J. Sohn, N. S. Kim, and W. Sung, “A statistical model-based voice activity detection,” *IEEE signal processing letters*, vol. 6, no. 1, pp. 1–3, 1999.
- [30] V. Pratap, A. Tjandra, B. Shi, P. Tomasello, A. Babu, S. Kundu, A. Elkahky, Z. Ni, A. Vyas, M. Fazel-Zarandi *et al.*, “Scaling speech technology to 1,000+ languages,” *arXiv preprint arXiv:2305.13516*, 2023.
- [31] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, “Hubert: Self-supervised speech representation learning by masked prediction of hidden units,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.
- [32] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, “Conformer: Convolution-augmented transformer for speech recognition,” *arXiv preprint arXiv:2005.08100*, 2020.
- [33] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, “fairseq: A fast, extensible toolkit for sequence modeling,” *arXiv preprint arXiv:1904.01038*, 2019.
- [34] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.
- [35] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, “Conformer: Convolution-augmented Transformer for Speech Recognition,” in *Proc. Interspeech 2020*, 2020, pp. 5036–5040.
- [36] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [37] C. Kim, D. Gowda, D. Lee, J. Kim, A. Kumar, S. Kim, A. Garg, and C. Han, “A review of on-device fully neural end-to-end automatic speech recognition algorithms,” in *2020 54th Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2020, pp. 277–283.