MLS: A Large-Scale Multilingual Dataset for Speech Research

Vineel Pratap[1], *Qiantong Xu*[1], *Anuroop Sriram*[1], *Gabriel Synnaeve*[2], *Ronan Collobert*[1]

[1]Facebook AI Research, Menlo Park
[2]Facebook AI Research, NYC

{vineelkpratap,qiantong,anuroops,gab,locronan}@fb.com

Abstract

This paper introduces Multilingual LibriSpeech (MLS) dataset, a large multilingual corpus suitable for speech research. The dataset is derived from read audiobooks from LibriVox and consists of 8 languages, including about 32K hours of English and a total of 4.5K hours for other languages. We provide baseline Automatic Speech Recognition (ASR) models and Language Models (LM) for all the languages in our dataset. We believe such a large transcribed dataset will open new avenues in ASR and Text-To-Speech (TTS) research. The dataset will be made freely available for anyone at http://www.openslr.org.
Index Terms: speech recognition, multilingual

1. Introduction

The success of LibriSpeech [1] as a standard, freely available, Automatic Speech Recognition (ASR) benchmark is undeniable in the research community. LibriSpeech is English-only, and while benchmarks for other languages are available, there are often low-scale or scattered around different places, and rarely available under an open license. In this paper, we revisit the work which has been done with LibriSpeech but in a multilingual manner and at larger scale, introducing the Multilingual LibriSpeech (MLS) dataset. MLS includes 32K hours of English, and a total of 4.5K hours spread over 7 other languages. As for LibriSpeech, MLS is a read-speech dataset, which leverages LibriVox[1] audiobook data, most of which being based on the Project Gutenberg[2] text data. LibriVox and Project Gutenberg data are released in the public domain, which allows us to release MLS freely to everyone.

In Section 3, we detail how we created the dataset, by (i) training some acoustic models on in-house data, (ii) generating pseudo-labels with these models, and (iii) retrieving the original transcript by matching pseudo-labels to available book transcripts. Section 4 details the statistics of MLS for its different languages. Section 5 introduces languages models we trained for each of language. These languages models are part of the MLS release. Section 6 covers some baseline ASR experiments.

2. Related Work

As for our work, LibriSpeech [1] is derived from the LibriVox data, and is distributed under an open license. It ships with about 1000 hours of labeled audio, obtained by leveraging alignments between textbooks and their read (audio) counterpart. In contrast to our work, it is only mono-lingual (English). A notable multi-lingual ASR dataset was built with the IARPA Babel Program [2]. It collected data for 24 languages, mostly from conversational telephone speech. The dataset is however not released and under an open license, and focused on low-resource languages, with labeled data ranging between 25 to 65 hours per language. On the open license side, two important volunteer-supported multi-lingual speech gathering efforts are being conducted: (i) VoxForge [3] which collected data for about 15 different languages, but remains low-scale (about 300 hours in total). (ii) CommonVoice [4], a more scalable solution, with more than 30 languages available, which keeps growing with 4500 (validated) hours currently available. Other notatable multi-lingual datasets distributed under an open license are the M-AILABS [5] and the CMU Wilderness [6] datasets. M-AILABS is a lower-scale version of our work, with 9 languages collected from LibriVox, for a total of about 1000 hours available. The CMU Wilderness collects readings from the New Testament, with 700 different languages available.

Language	Hours	Books	Speakers
English	71,506.79	12421	4214
German	3,287.48	593	244
Dutch	2,253.68	206	91
Spanish	1,438.41	285	120
French	1,333.35	224	114
Multilingual*	516.82	130	19
Portuguese	284.59	68	31
Italian	279.43	61	28
Russian	172.34	44	29
Latin	138.93	20	16
Polish	137.00	25	16
Church Slavonic	136.42	8	2
Hebrew	125.72	23	13
Japanese	97.67	38	24
Ancient Greek	69.77	43	8

Table 1: *LibriVox Audiobook data statistics for the top 15 languages; * - audio books with mix of multiple languages*

3. Data processing pipeline

This section describes the major steps involved in preparing the MLS dataset.

3.1. Downloading audiobooks

Table 1 shows the LibriVox audiobooks data available for each language that we measured using LibriVox APIs [3]. While English is the most dominant language, we can see that there is a significant amount of audio hours present in languages other than English, making this a valuable source for multilingual dataset preparation.

[1]https://librivox.org
[2]http://www.gutenberg.org
[3]https://librivox.org/api/info

We have selected English, German, Dutch, Spanish, French, Portuguese, Italian, Polish for the MLS dataset preparation. For downloading the LibriVox audiobooks in these languages, we have used data preparation tools available at Libri-Light [4] open source library.

3.2. Audio segmentation and pseudo label generation

Since acoustic model training is usually done on shorter utterances, we have used trained acoustic models in each of the languages to segment the data. The acoustic models are trained using Time-Depth Separable Convolutions with Auto-Segmentation Criterion [7] (ASG) loss. We chose ASG criterion over Connectionist Temporal Classification [8] (CTC) criterion since ASG doesn't exhibit delay in transcriptions compared to CTC [9]. For each language, we train models on in-house datasets consisting of videos publicly shared by users. We use only audio part of the videos and the data is completely de-identified.

Our segmentation process is run in two steps. First, we run inference on the audio and generate viterbi token sequence along with their timestamps. Since the audio files can be very long (>30 minutes), we have used wav2letter@anywhere framework [10] to perform the inference in a streaming fashion. Second, we select the longest silence duration within 10 sec to 20 sec range from the start of an audio and split the audio at the mid point of the silence chunk to create a segment. If no silence frames are found between 10 sec to 20 sec from the starting point, we split the audio at 20 sec mark. Once we generate a segment, we consider the end point of previous segment as the current starting point and repeat the process again till we reach the end of audio. This process guarantees that all the segments are between 10sec and 20sec. A minimum segment duration of 10 sec is kept so that the segments have sufficient number of words spoken which helps with better transcript retrieval (described in Section 3.4).

We generate pseudo labels for the segmented audio samples by performing a beam-search decoding with the same models as mentioned above and using a 4-gram LM trained on the training data used for the models. For English, however, we use pre-trained model from [11] which uses TDS encoder and CTC loss on LibriSpeech [12], LibriVox for generating pseudo labels.

3.3. Downloading text sources for audiobook data

To generate the labels for the audio segments derived from audiobooks, we would need to have the original textbook from which the speaker read the audiobook. For English, we found that ≈60,000 hours of audiobooks is read from four major website domains - `gutenberg.org`, `archive.org`, `ccel.org` and `hathitrust.org`. We wrote parsers to automatically extract the text for each of these domains and downloaded the text sources for all the audiobooks in English.

For other languages, however, we found it more challenging to automatically extract text for the audiobooks because 1. the diversity of domains is large making it impossible to write parsers for each and every domain 2. some of the links were invalid or redirected to an incorrect page. So, we incorporated some manual approaches in our process to cover the audiobook text sources as much as possible. Depending on the language and text source domain, we copied the data directly from the browser or extracted text from .pdf/.epub books using pdftotext[5], or writing HTML parsers to retrieve text data for popular domains in a language.

For the audiobooks with invalid links for text sources, we manually searched online to find alternate sources where the book is available. For example, all the text sources from the `spiegel.de` domain, which accounts of 1/3rd of German audiodata, were being redirected to an invalid page. However, we were able to find the alternate text sources from online websites like `projekt-gutenberg.de`, `zeno.org` for most of these unavailable books from `spiegel.de`.

3.4. Transcript retrieval

The transcript retrieval process involves finding the true target label for the audio segments from the source text of audio. Our procedure closely follows the method described in [13] with few modifications.

We first split the source text into multiple overlapping documents of 1250 words each and striding by 1000 words. We retrieve the documents which best matches with the pseudo label for the audio segments using term-frequency inverse document-frequency (TF-IDF) similarity score on bigrams. We then perform a Smith-Waterman alignment [14] to find the best matching sub-sequence of words. We have used a matching score of 2 and substitution, insertion, deletion score of -1 for the alignment algorithm.

After the above alignment procedure, we generate a candidate target label for each audio segment, which corresponds to the best match of the pseudo label in the source text of audiobook. We filter out all the candidate transcripts generated from the matching algorithm above, if the WER between the candidate transcript and pseudo label generated is >40% .

3.5. Creating validation and test splits

We have used the following principles when splitting the dataset into train, valid and test sets - 1) there is no speaker overlap between the training, development and test sets, 2) speakers are balanced in gender and duration in development and test sets and 3) there are sufficient audio hours and speakers assigned into development and test sets to be able to validate ASR model performance.

First, we select the list of all the books available for a language. We remove all the books with corrupted meta data, such as missing title or information about speakers and authors. Then, to ensure that each recording is unambiguously attributable to a single speaker, we also remove audios with multiple speakers, for example, "Dramatic Reading", which include predominantly multi-reader audio chapters. In addition, we only keep the latest version for books sharing the same authors and title, but different versions. Only the speakers reading the valid books are considered in the transcript retrieval process from section 3.4.

Second, we label the gender of all the speakers with a gender classifier. The classifier is a SVM [15] with RBF kernel trained on 1172 speakers from *train-clean-100* and *train-clean-360* subsets of LibriSpeech . In particular, it consumes 40-dimensional log-filterbank features averaged over time as input features. The test accuracy on the 146 speakers from the joint development and test sets of LibriSpeech is 95%. We use the same gender classifier for other languages as well. We manually checked the quality of this classifier on Dutch and Polish and it's accuracy is 96% and 94% respectively.

[4] `https://github.com/facebookresearch/libri-light`

[5] `https://pypi.org/project/pdftotext/`

Table 2: *Statistics of Train/Dev/Test partitions of each language. Below lists for each partition: the total duration in hours (left), number of speakers in each gender (middle) and the shortest and longest duration in minutes for speakers in dev and test sets (right).*

Language	Hours			Speakers						Min/Speaker	
	Train	Dev	Test	Train (M/F)		Dev (M/F)		Test (M/F)		Min	Max
English	32073	25.5	26.1	2396	2473	34	40	37	39	20	21
German	1430.6	16.1	16.1	56	56	14	14	14	14	20	40
Dutch	1086	3.6	4.1	5	26	5	5	5	5	5	100
Spanish	718.7	8.9	9	36	41	9	9	9	9	20	40
French	637.6	7.6	7.7	62	79	7	7	7	7	20	40
Italian	221	4.8	4.8	22	43	5	5	5	5	20	40
Portuguese	121.4	4.3	4.4	22	12	5	5	5	5	10	30
Polish	58	2.3	2.4	4	2	3	3	3	3	9	40

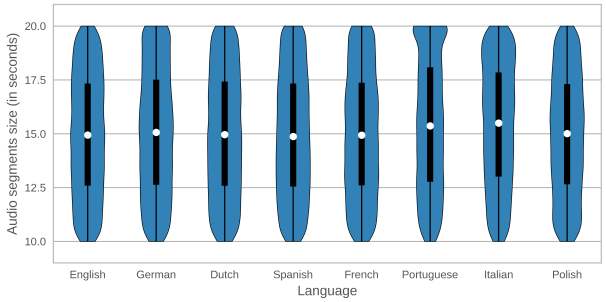

Figure 1: *Violin plots of audio segments duration in the training data for different languages*

Finally, the dataset is split into training, development and test sets as following. We computed the total duration each speaker spends in reading the valid books, and order them by this duration. Speakers with duration shorter than a threshold are assigned into the training set. From the rest speakers, we specify an amount of speakers per gender and select out a series of speakers with the shortest duration in each gender. The selected speakers are then equally split into development and test sets. All the remaining speakers are assigned to training set again. To avoid high speaker imbalance in development and test sets, we further truncate the speakers with high duration by sampling their recordings up to an upper-bound. The detailed statistics can be found in Table 2. For English, however, we make train, valid and test sets a pure superset of LibriSpeech. Specifically, the same procedure for splitting the dataset as above is conducted first on the speaker set that has no overlap with the speakers in LibriSpeech. Then the speakers in LibriSpeech training, development and test sets are assigned into our new splits accordingly. This means that the development and test sets of LibriSpeech are 100% contained in our English development and test sets.

4. Statistics

Table 2 shows the amount of train, valid and test data along with the gender distribution of speakers for the 8 languages that we processed. The training data is lower than the data we presented in Table 1 because of 1) we may not have downloaded the text source for the audiobook, 2) the candidate label is filter during the transcript retrieval stage or 3) filtered while creating valid/test splits.

Figure 1 shows the violin plots of the duration of audio segments in training data for each language. We can see that all the segments are within 10sec to 20 sec range and all the sizes are almost evenly distributed inside the range.

5. Language models

We have trained language models (LM) for all the languages in our dataset. Those LMs are 5-gram models trained on training transcriptions using the KenLM toolkit [16]. The number of words of each LM and their perplexities (excluding out-of-vocabulary words) on the transcriptions in development sets are listed in Table 3. We also release those models, to be potentially used as standard benchmarks when comparing only acoustic models.

Table 3: *Language Models*

Language	Words	Perplexity
English	986491	184.7
German	249440	585.6
Dutch	159067	386.8
Spanish	155627	283.1
French	128938	344.0
Italian	83557	760.4
Portuguese	72856	779.5
Polish	60154	1684.6

6. Experiments and results

6.1. Training setup

All our experiments are run using the wav2letter++ [17] framework. We use 80-dimensional log mel-scale filter banks as input features, with STFTs computed on 25ms Hamming windows strided by 10ms. We use SpecAugment [18] with LibriSpeech Double setting for all the experiments. The AMs take 80-channel log-mel filterbanks as input and are trained end-to-end with Connectionist Temporal Classification (CTC) loss [8].

6.2. Monolingual Baselines

For English, we use the best-performing TDS architecture on LibriSpeech from [19] in our experiments. In particular, the model is mainly built upon Time-Depth Separable Convolution (TDS) [20] blocks. It is composed of one 2-D convolution layer and two fully-connected layers with ReLU, LayerNorm

Table 4: *Baseline WER for different languages.*

Language	No LM		5-gram LM	
	Dev	Test	Dev	Test
English	14.45	15.18	11.90	12.64
German	16.53	17.84	14.37	15.57
Dutch	27.95	30.53	24.28	28.87
Spanish	12.96	12.46	11.40	11.07
French	18.63	19.66	16.58	18.08
Italian	32.77	36.70	24.54	28.19
Portuguese	42.47	44.45	36.88	39.55
Polish	46.92	67.23	43.61	60.32

and residual connections in between. Specifically, the model has 4 groups of TDS blocks with a 1-D convolutions at the beginning of each group as transitions. Similarly, the first 3 convolutions have stride 2 so as to reach the same sub-sampling (striding) rate of 8, thus 80ms. There are 2, 2, 5, and 8 TDS blocks in each group, containing 16, 16, 24, and 32 channels, respectively. Following [19], we also apply a channel increasing factor $F = 2$ in each TDS block.

For the other languages, we experimented with three model architectures of varying capacities, all of which were based on the TDS architecture with CTC loss [8]. The capacity of the models is adjusted by changing the number of TDS blocks. The smallest model architecture (60M parameters) contains two 10-channel, three 14-channel and five 18-channel TDS blocks. In the 100M parameter architecture, we use convolutions of the same width but increase the number of blocks to three, four and eight respectively. Finally, in the largest architecture (200M parameters), we increase the numbers to five, six, and ten respectively. We used dropout and spectral augmentation [18] to regularize the models and we tuned dropout extensively for each language.

We have also experimented with different token sets: graphemes or sub-word tokens generated using the SentencePiece toolkit [21]. For the lowest resource languages (Italian, Portuguese and Polish), we obtained the best results with a 60M parameter model. For these languages, we experimented with graphemes, 300 sentence pieces and 500 sentence pieces as the token set and found graphemes to work best for Italian and Polish and 300 sentence pieces to work best for Portuguese. For the other languages, we only experimented with 5,000 and 10,000 sentence pieces and found 10,000 sentence pieces to work best. We obtained our best results with a 200M parameter model for all languages except Dutch, for which the 100M parameter model outperformed the others.

Finally, we use beam-search decoding in wav2letter++ to integrate external 5-gram language models trained on the training text, together with the AMs. The decoder hyper-parameters are tuned on the validation set.

Table 4 shows the best results obtained for each language on the (averaged) Dev and Test sets with and without a language model. There is a lot of room for improvement upon those baselines, which vary between 11% and 60% WER depending on languages.

7. Conclusions

We have presented the Multilingual LibriSpeech dataset, a large scale multilingual speech dataset with 36.5K hours of training data spread over 8 languages. We believe this dataset will promote open research in large-scale training of ASR systems and in multilingual ASR. This dataset can also be used for Text-to-Speech(TTS) research by extending the LibriTTS [22] dataset, and by creating a larger and multilingual version for TTS Research.

8. Acknowledgements

We would like to thank Steven Garan for help in data preparation and text normalization.

9. References

[1] 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)*, 2015, pp. 5206–5210.

[2] M. P. Harper, "The iarpa babel program," https://www.iarpa.gov/index.php/research-programs/babel.

[3] Voxforge.org, "Free speech... recognition (linux, windows and mac) - voxforge.org," http://www.voxforge.org/, accessed 06/25/2014.

[4] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," 2019.

[5] I. Solak, "The M-AILABS Speech Dataset," https://www.caito.de/2019/01/the-m-ailabs-speech-dataset/, 2019, [Online; accessed 19-April-2020].

[6] A. W. Black, "Cmu wilderness multilingual speech dataset," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5971–5975.

[7] R. Collobert, C. Puhrsch, and G. Synnaeve, "Wav2letter: an end-to-end convnet-based speech recognition system," 2016.

[8] 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.

[9] V. Liptchinsky, G. Synnaeve, and R. Collobert, "Letter-based speech recognition with gated convnets," 2017.

[10] V. Pratap, Q. Xu, J. Kahn, G. Avidov, T. Likhomanenko, A. Hannun, V. Liptchinsky, G. Synnaeve, and R. Collobert, "Scaling up online speech recognition using convnets," 2020.

[11] G. Synnaeve, Q. Xu, J. Kahn, T. Likhomanenko, E. Grave, V. Pratap, A. Sriram, V. Liptchinsky, and R. Collobert, "End-to-end asr: from supervised to semi-supervised learning with modern architectures," 2019.

[12] 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.

[13] V. Manohar, D. Povey, and S. Khudanpur, "Jhu kaldi system for arabic mgb-3 asr challenge using diarization, audio-transcript alignment and transfer learning," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 346–352.

[14] T. Smith and M. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195 – 197, 1981. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022283681900875

[15] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, ser. COLT '92. New York, NY, USA: Association for Computing Machinery, 1992, p. 144–152. [Online]. Available: https://doi.org/10.1145/130385.130401

[16] K. Heafield, "Kenlm: Faster and smaller language model queries," in *Proceedings of the sixth workshop on statistical machine translation*. Association for Computational Linguistics, 2011, pp. 187–197.

[17] V. Pratap, A. Hannun, Q. Xu, J. Cai, J. Kahn, G. Synnaeve, V. Liptchinsky, and R. Collobert, "Wav2letter++: A fast open-source speech recognition system," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6460–6464.

[18] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *Interspeech 2019*, Sep 2019. [Online]. Available: http://dx.doi.org/10.21437/interspeech.2019-2680

[19] G. Synnaeve, Q. Xu, J. Kahn, E. Grave, T. Likhomanenko, V. Pratap, A. Sriram, V. Liptchinsky, and R. Collobert, "End-to-end asr: from supervised to semi-supervised learning with modern architectures," *arXiv preprint arXiv:1911.08460*, 2019.

[20] A. Hannun, A. Lee, Q. Xu, and R. Collobert, "Sequence-to-sequence speech recognition with time-depth separable convolutions," in *INTERSPEECH*, 2019.

[21] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," 2018.

[22] H. Zen, R. Clark, R. J. Weiss, V. Dang, Y. Jia, Y. Wu, Y. Zhang, and Z. Chen, "Libritts: A corpus derived from librispeech for text-to-speech," in *Interspeech*, 2019. [Online]. Available: https://arxiv.org/abs/1904.02882