Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 4218–4222
Marseille, 11–16 May 2020

Common Voice: A Massively-Multilingual Speech Corpus

Rosana Ardila,[†] Megan Branson,[†] Kelly Davis,[†] Michael Henretty, Michael Kohler, Josh Meyer,[◦] Reuben Morais,[†] Lindsay Saunders,[†] Francis M. Tyers,[‡] Gregor Weber[†]

[†] Mozilla [‡] Indiana University [◦] Artie, Inc.
Various Cities Bloomington, IN, USA Los Angeles, CA, USA
{rosana, mbranson, kdavis, reuben, lsaunders, gweber}@mozilla.com,
ftyers@iu.edu, michael.henretty@gmail.com, me@michaelkohler.info, josh.meyer@artie.com

Abstract

The Common Voice corpus is a massively-multilingual collection of transcribed speech intended for speech technology research and development. Common Voice is designed for Automatic Speech Recognition purposes but can be useful in other domains (e.g. language identification). To achieve scale and sustainability, the Common Voice project employs crowdsourcing for both data collection and data validation. The most recent release includes 29 languages, and as of November 2019 there are a total of 38 languages collecting data. Over 50,000 individuals have participated so far, resulting in 2,500 hours of collected audio. To our knowledge this is the largest audio corpus in the public domain for speech recognition, both in terms of number of hours and number of languages. As an example use case for Common Voice, we present speech recognition experiments using Mozilla's DeepSpeech Speech-to-Text toolkit. By applying transfer learning from a source English model, we find an average Character Error Rate improvement of 5.99 ± 5.48 for twelve target languages (German, French, Italian, Turkish, Catalan, Slovenian, Welsh, Irish, Breton, Tatar, Chuvash, and Kabyle). For most of these languages, these are the first ever published results on end-to-end Automatic Speech Recognition.
Keywords: spoken corpus, Automatic Speech Recognition, low-resource languages

1. Introduction

The Common Voice project[1] is a response to the current state of affairs in speech technology, in which training data is either prohibitively expensive or unavailable for most languages (Roter, 2019). We believe that speech technology (like all technology) should be open and decentralized, and the Common Voice project achieves this goal via a mix of community building, open source tooling, and a permissive licensing scheme. The corpus is designed to organically scale to new languages as community members use the provided tools to translate the interface, submit text sentences, and finally record and validate voices in their new language [2]. The project was started with an initial focus on English in July 2017 and then in June 2018 was made available for any language.

The remainder of the paper is organized as follows: In Section (2) we motivate Common Voice and review previous multilingual corpora. Next, in Section (3) we describe the recording and validation process used to create the corpus. Next, in Section (4) we describe the current contents of Common Voice, and lastly in Section (5) we show multilingual Automatic Speech Recognition experiments using the corpus.

2. Prior work

Some notable multilingual speech corpora include VoxForge (VoxForge, 2019), Babel (Gales et al., 2014), and M-AILABS (M-AILABS, 2019). Even though the Babel corpus contains high-quality data from 22 minority languages, it is not released under an open license. VoxForge is most similar to Common Voice in that it is community-driven, multilingual (17 languages), and released under an open license (GNU General Public License). However, the VoxForge does not have a sustainable data collection pipeline compared to Common Voice, and there is no data validation step in place. M-AILABS data contains 9 language varieties with a modified BSD 3-Clause License, however there is no community-driven aspect. Common Voice is a sustainable, open alternative to these projects which allows for collection of minority and majority languages alike.

3. Corpus Creation

The data presented in this paper was collected and validated via Mozilla's Common Voice initiative. Using either the Common Voice website or iPhone app, contributors record their voice by reading sentences displayed on the screen (see Figure (1)). The recordings are later verified by other contributors using a simple voting system. Shown in Figure (2), this validation interface has contributors mark <audio,transcript> pairs as being either correct (up-vote) or incorrect (down-vote).

A maximum of three contributors will listen to any audio clip.[3] If an <audio,transcript> pair first receives two up-votes, then the clip is marked as valid. If instead the clip first receives two down-votes, then it is marked as invalid. A contributor may switch between recording and validation as they wish.

Only clips marked as valid are included in the official training, development, and testing sets for each language. Clips which did not receive enough votes to be validated or invalidated by the time of release are released as "other". The train, test, and development sets are bucketed such that

[1] http://voice.mozilla.org

[2] https://discourse.mozilla.org/t/readme-how-to-see-my-language-on-common-voice/31530

[3] In the early days of Common Voice, this voting mechanism contained bugs, and some clips in the official release received over three votes. In these cases we use a simple majority rule. The total number of up and down votes is released with the dataset.

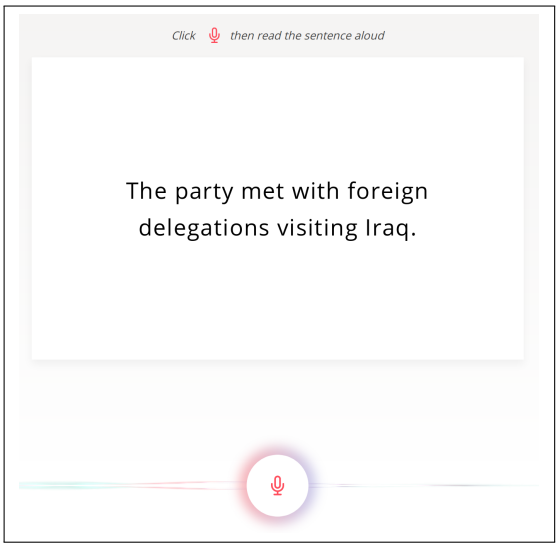

Figure 1: Recording interface for Common Voice. Additionally, it is possible to skip or report as problematic any audio or sentence.

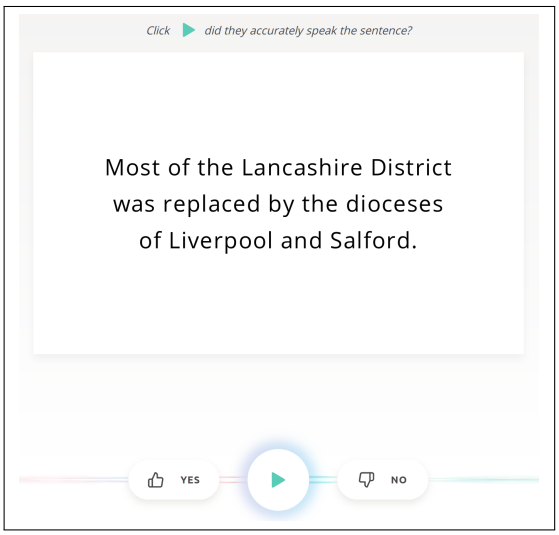

Figure 2: Validation interface for Common Voice. Additionally, it is possible to skip or report as problematic any audio or sentence.

any given speaker may appear in only one. This ensures that contributors seen at train time are not seen at test time, which would skew results. Additionally, repetitions of text sentences are removed from the train, test, and development sets of the corpus.[4]

The number of clips is divided among the three datasets according to statistical power analyses. Given the total number of validated clips in a language, the number of clips in the test set is equal to the number needed to achieve a confidence level of 99% with a margin of error of 1% relative to the number of clips in the training set. The same is true of the development set.[5]

The audio clips are released as mono-channel, 16bit

[4]Repetitions may be found in the other TSV files included in Common Voice.

[5]All code which performs the bucketing can be found here: `https://github.com/mozilla/CorporaCreator`

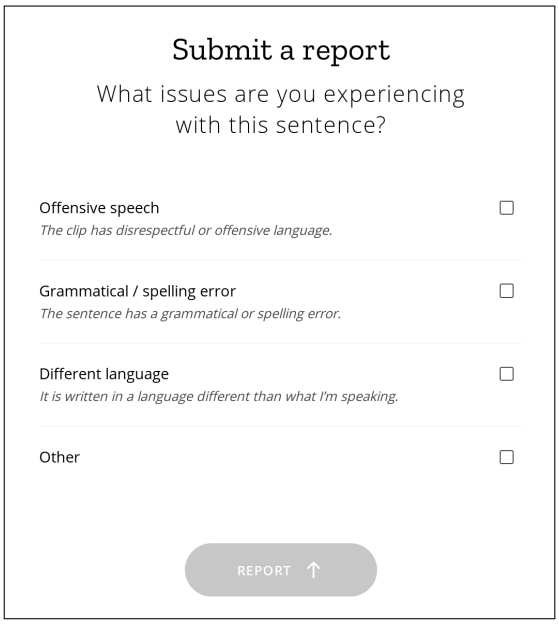

Figure 3: Interface for reporting problematic data. It is possible to report any text or audio as problematic during either recording or validation.

MPEG-3 files with a 48kHz sampling rate. The choice to collect and release MPEG-3 as opposed to a lossless audio format (e.g. WAV) is largely due to the web-based nature of the Common Voice collection platform. MPEG-3 is the most universally supported audio format for the web, and as such is the most reliable recording/playback technique for various devices and browsers. Also practically speaking, the audio quality is appropriate for speech applications.

4. Corpus Contents

4.1. Released Languages

The data presented in Table (1) shows the currently available data. Each of the released languages is available for individual download as a compressed directory from the Mozilla Common Voice website.[6] The directory contains six files with Tab-Separated Values (i.e. TSV files), and a single `clips` sub-directory which contains all of the audio data. Each of the six TSV files represents a different segment of the voice data, with all six having the following column headers: [`client_id`, `path`, `sentence`, `up_votes`, `down_votes`, `age`, `gender`, `accent`]. The first three columns refer to an anonymized ID for the speaker, the location of the audio file, and the text that was read. The next two columns contain information on how listeners judged the <audio,transcript> pair. The last three columns represent demographic data which was optionally self-reported by the speaker of the audio.

4.2. Adding a new Language

In order to add a new language to the Common Voice project, two steps must be completed. First, the web-app user interface must be translated into the target language.

[6]Common Voice Download page: `https://voice.mozilla.org/en/datasets`

Language	Code	Voices	Hours	
			Total	Validated
Abkhaz	ab	3	<1	<1
Arabic	ar	225	15	9
Basque	eu	508	83	46
Breton	br	118	10	3
Catalan	ca	1,834	120	107
Chinese (China)	zh-ZH	288	12	11
Chinese (Taiwan)	zh-TW	949	43	33
Chuvash	cv	38	2	1
Dhivehi	dv	92	8	5
Dutch	nl	502	23	18
English	en	39,577	1,087	780
Esperanto	eo	129	16	13
Estonian	et	225	12	11
French	fr	3,005	184	173
German	de	5,007	340	325
Hakha Chin	cnh	280	4	2
Indonesian	id	54	5	4
Interlingua	ia	11	2	1
Irish	ga	63	3	2
Italian	it	602	40	36
Japanese	ja	48	2	1
Kabyle	kab	584	192	181
Kinyarwanda	rw	32	1	<1
Kyrgyz	ky	97	20	8
Latvian	lv	82	8	6
Mongolian	mn	230	9	8
Persian	fa	1,240	70	67
Portuguese	pr	316	30	27
Russian	ru	64	31	27
Sakha	sah	35	6	3
Slovenian	sl	42	5	2
Spanish	es	611	31	27
Swedish	sv	44	3	3
Tamil	ta	89	5	3
Tatar	tt	132	26	22
Turkish	tr	344	10	9
Votic	vot	2	<1	<1
Welsh	cy	748	48	42
TOTAL		58,250	2,508	2,019

Table 1: Current data statistics for Common Voice. Data in *italics* is as of yet unreleased. Other numbers refer to the data published in the June 12, 2019 release.

For example, the text shown in Figure (3) must be translated. Secondly, text prompts must be gathered in order to be read aloud. These texts are not translated, but gathered from scratch for each language – translation would be very slow and not scalable.

Translation of the interface is managed through the Pontoon platform[7]. Pontoon allows community members to propose translations, and then the moderators for that language approve or decline the proposals. At the time of writing this paper there are 610 text strings used in the Common Voice interface, where each string can range in length from an isolated word to a paragraph of text.

Collecting text for reading aloud is the second step of adding a new language to Common Voice. For languages with more than 500,000 Wikipedia articles, text sentences are extracted from Wikipedia using community provided rule-sets per language[8]. These sentences make up the initial text prompts for the languages in question.

Any language community can gather additional sentences through the Sentence Collector[9] taking advantage of automatic validation mechanisms such as checks for sentence length, foreign alphabets, and numbers. Every sentence submitted through the Sentence Collector needs to be approved by two out of three reviewers, leading to a weekly export of new sentences into the Common Voice database. Once the website is translated and at least 5,000 sentences have been added, the language is enabled for voice recordings.

5. Automatic Speech Recognition Experiments

The following experiments demonstrate the potential to use the Common Voice corpus for multilingual speech research. These results represent work on an internal version of Common Voice from February 2019. The current corpus contains more languages and more data per language.

These experiments use an End-to-End Transfer Learning approach which bypasses the need for linguistic resources or domain expertise (Meyer, 2019). Certain layers are copied from a pre-trained English source model, new layers are initialized for a target language, the old and new layers are stitched together, and all layers are fine-tuned via gradient descent.

5.1. Data

We made dataset splits (c.f. Table (2)) such that one speaker's recordings are only present in one data split. This allows us to make a fair evaluation of speaker generalization, but as a result some training sets have very few speakers, making this an even more challenging scenario. The splits per language were made as close as possible to 80% train, 10% development, and 10% test.[10]

Results from this dataset are interesting because the text and audio are challenging, the range of languages is wider than any openly available speech corpus, and the amount of data per language ranges from very small (less than 1,000 clips for Slovenian) to relatively large (over 65,000 clips for German).

5.2. Model architecture

All reported results were obtained with Mozilla's DeepSpeech v0.3.0 — an open-source implementation of a variation of Baidu's first DeepSpeech paper (Hannun et al., 2014). This architecture is an end-to-end Automatic Speech Recognition (ASR) model trained via stochastic gradient descent with a Connectionist Temporal Classification

[7]Pontoon translation platform: https://pontoon.mozilla.org/projects/common-voice/

[8]Code for Wikipedia extraction can be found here: https://github.com/Common-Voice/common-voice-wiki-scraper

[9]Sentence Collector web app: https://common-voice.github.io/sentence-collector

[10]These experiments were performed before Common Voice was in it's current form. As such, the train, development, and test splits here do not correspond exactly to the official releases.

		Dataset Size					
		Audio Clips			Unique Speakers		
Language	**Code**	Dev	Test	Train	Dev	Test	Train
Slovenian	sl	110	213	728	1	12	3
Irish	ga	181	138	1,001	4	12	6
Chuvash	cv	96	77	1,023	4	12	5
Breton	br	163	170	1,079	3	15	7
Turkish	tr	407	374	3,771	32	89	32
Italian	it	627	734	5,019	29	136	37
Welsh	cy	1,235	1,201	9,547	51	153	75
Tatar	tt	1,811	1,164	11,187	9	64	3
Catalan	ca	5,460	5,037	38,995	286	777	313
French	fr	5,083	4,835	40,907	237	837	249
Kabyle	kab	5,452	4,643	43,223	31	169	63
German	de	7,982	7,897	65,745	247	1,029	318

Table 2: Data used in the experiments, from an earlier multilingual version of Common Voice. Number of audio clips and unique speakers.

(CTC) loss function (Graves et al., 2006). The model is six layers deep: three fully connected layers followed by a unidirectional LSTM layer followed by two more fully connected layers (c.f. Figure (4)). All hidden layers have a dimensionality of 2,048 and a clipped ReLU activation. The output layer has as many dimensions as characters in the alphabet of the target language (including any desired punctuation as well as the blank symbol used for CTC). The input layer accepts a vector of 19 spliced frames (9 past frames + 1 present frame + 9 future frames) with 26 MFCC features each (i.e. a single, 494-dimensional vector).

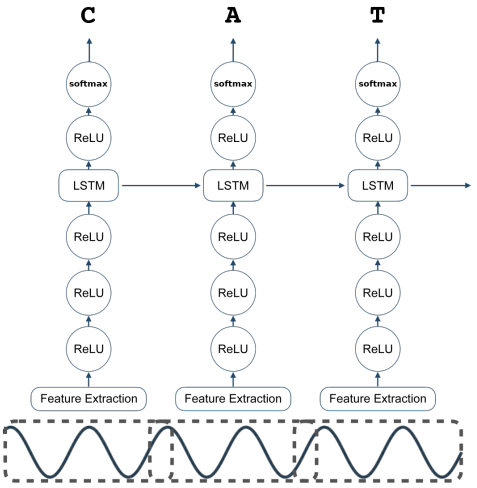

Figure 4: Architecture of Mozilla's DeepSpeech Automatic Speech Recognition model. A six-layer unidirectional CTC model, with one LSTM layer.

All models were trained with the following hyperparameters on a single GPU. We use a batch-size of 24 for train and 48 for development, a dropout rate of 20%, and a learning rate of 0.0001 with the ADAM optimizer.[11] The new, target-language layers were initialized via Xavier initialization (Glorot and Bengio, 2010). After every epoch of back-propagation over the training set, the loss over the entire development set is calculated. This development loss is used to trigger early stopping. Early stopping is triggered when the loss on the held-out development set either (1) increases over a window of five sequential epochs, or (2) the most recent loss over the development set has not improved in a window of five epochs more than a mean loss threshold of 0.5 *and* the window of losses shows a standard deviation of less than 0.5.

[11]For a complete list of ADAM hyperparameters: https://github.com/mozilla/DeepSpeech/blob/v0.3.0/DeepSpeech.py#L79

6. Results

	Character Error Rate					
	Number of Layers Copied from English					
Lang.	None	1	2	3	4	5
sl	23.35	21.65	26.44	19.09	**15.35**	17.96
ga	31.83	31.01	32.2	27.5	25.42	**24.98**
cv	48.1	47.1	44.58	42.75	**27.21**	31.94
br	21.47	19.16	20.01	18.06	**15.99**	18.42
tr	34.66	34.12	34.83	31.79	**27.55**	29.74
it	40.91	42.65	42.82	36.89	**33.63**	35.10
cy	34.15	31.91	33.63	30.13	**28.75**	30.38
tt	32.61	31.43	30.80	27.79	**26.42**	28.63
ca	38.01	35.21	39.02	35.26	**33.83**	36.41
fr	43.33	43.26	43.51	43.24	43.20	**43.19**
kab	25.76	25.5	26.83	25.25	**24.92**	25.28
de	43.76	43.69	43.62	**43.60**	43.76	43.69

Table 3: Fine-Tuned Transfer Learning Character Error Rate for each language, in addition to a baseline trained from scratch on the target language data. Bolded values display best model per language. Shading indicates relative performance per language, with darker indicating better models.

The results from all experiments can be found in Table (3). Each cell in the table contains the Character Error Rate (CER)[12] of the resulting model on the test set, defined as the Levenshtein distance (Fiscus et al., 2006) of the characters between the ground-truth transcript and the decoding result. The results in Table (3) show how the number of layers transfered (columns) influence the performance on individual target languages (rows). Shaded cells indicate relative performance per language, where a darker cell represents a more accurate model. From this table we observe a trend in which four layers copied from pre-trained English DeepSpeech result in the best final model. This trend becomes more obvious in Figure (5), where we average the improvement over all languages relative to the number of layers transfered from a source model.

7. Concluding remarks

We have presented Common Voice: a crowd-sourced, multilingual speech corpus which can scale to any language via community effort. All of the speech data is released under a Creative Commons CC0 license, making Common

[12]We report Character Error Rate as opposed to Word Error Rate (WER) because the former is more language-agnostic. Word Error Rate is most appropriate for languages which exhibit an analytic or isolating morphology and clearly delimit words in their orthography. Many languages do not use whitespace to delimit words, and there is no clear definition of "word" in a multilingual context. In short, both CER and WER make sense for languages like English, but CER is much more appropriate for languages like Mandarin, Turkish, and Chukchi.

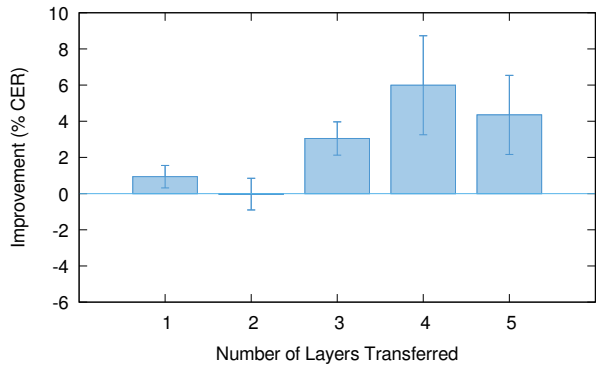

Figure 5: Mean and standard deviation in Character Error Rate improvement over all twelve languages investigated, relative to number of layers transferred from a pre-trained English DeepSpeech v0.3.0 model.

Voice the largest public domain corpus designed for Automatic Speech Recognition. In Section (3) we described the recording and validation process used to create the corpus. In Section (4) we presented the current contents of Common Voice, and lastly in Section (5) we show multilingual Automatic Speech Recognition experiments using the corpus. There are currently 38 language communities collecting data via Common Voice, and we welcome more languages and more volunteers.

Acknowledgments

Common Voice is a living project, and would not be possible without the thousands of hours given by volunteers. We thank all volunteers for their time, and especially the minority language activists who translate, find new texts, and organize Common Voice donation events. We thank George Roter, Gheorghe Railean, Rubén Martín, and Jane Scowcroft for their work on Common Voice, and all members of the Common Voice team, past and present.

This material is based upon work when Josh Meyer was supported by the National Science Foundation under Grant No. (DGE-1746060). Opinions, findings, conclusions, and recommendations are those of the authors and do not necessarily reflect the views of the NSF.

8. References

Fiscus, J. G., Ajot, J., Radde, N., and Laprun, C. (2006). Multiple dimension levenshtein edit distance calculations for evaluating automatic speech recognition systems during simultaneous speech. In *LREC*, pages 803–808. Citeseer.

Gales, M. J., Knill, K. M., Ragni, A., and Rath, S. P. (2014). Speech recognition and keyword spotting for low-resource languages: Babel project research at cued. In *Spoken Language Technologies for Under-Resourced Languages*.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256.

Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. (2006). Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376. ACM.

Hannun, A. Y., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., and Ng, A. Y. (2014). Deep Speech: Scaling up end-to-end speech recognition. *CoRR*, abs/1412.5567.

M-AILABS. (2019). The m-ailabs speech dataset. `https://www.caito.de/2019/01/the-m-ailabs-speech-dataset/`. accessed 11/25/2019.

Meyer, J. (2019). *Multi-Task and Transfer Learning in Low-Resource Speech Recognition*. PhD dissertation, The University of Arizona.

Roter, G. (2019). Sharing our common voices – mozilla releases the largest to-date public domain transcribed voice dataset, Feb. https://blog.mozilla.org/blog/2019/02/28/sharing-our-common-voices-mozilla-releases-the-largest-to-date-public-domain-transcribed-voice-dataset/.

VoxForge. (2019). Voxforge. `http://www.voxforge.org/`. accessed 11/25/2019.