INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Lightweight and Efficient Spoken Language Identification of Long-form Audio

Winstead Zhu[⋆†] *Md Iftekhar Tanveer*[⋆†] *Yang Janet Liu*[⋆‡] *Seye Ojumu*[†] *Rosie Jones*[†]

[†]Spotify [‡]Georgetown University

winsteadx@spotify.com, iftekhart@spotify.com, yl879@georgetown.edu,
oluseyeo@spotify.com, rjones@spotify.com

Abstract

State-of-the-art Spoken Language Identification (SLI) systems usually focus on tackling short audio clips, and thus their performance degrade drastically when applied to **long-form audio**, such as podcast, which poses peculiar challenges to existing SLI approaches due to its long duration and diverse content that frequently involves multiple speakers as well as various languages, topics, and speech styles. In this paper, we propose the first system to tackle SLI for **long-form audio** using podcast data by training a lightweight, multi-class feedforward neural classifier using speaker embeddings as input. We demonstrate that our approach can make inference on long audio input efficiently; furthermore, our system can handle long audio files with multiple speakers and can be further extended into utterance-level inference and code-switching detection, which is currently not covered by any existing SLI system.

Index Terms: Spoken language identification, long-form audio, podcast, speaker embedding application

1. Introduction

Spoken Language Identification (SLI) is the task of recognizing spoken languages using audio input only. SLI is an important step in multilingual speech processing tasks such as automatic speech recognition (ASR) and speech translation. For example, Google Cloud's multilingual Speech-to-Text service[1] requires users to submit one primary language code and *at most* three alternative language codes (i.e. four languages in total) and the service uses the language code that produces the best transcription results. This prerequisite indicates that if one does not know the language of the audio beforehand or if there are multiple possible languages, one needs to first perform a manual inspection of the language(s) in the audio before using the transcription service. Similar to the case for Google Cloud's Speech-to-Text service, SLI is critical for many different downstream multilingual speech processing tasks [1, 2] so as to remove dependency on human labeling or metadata for language tagging, which restricts the scalability of multilingual systems. Therefore, an automated SLI approach with high performance is intrinsic to the success of scaling up multilingual speech processing systems.

Long-form audio poses peculiar challenges to existing SLI systems, which so far have only focused on detecting languages from short-form, single-speaker, and monolingual audio. Long-form audio differs significantly from short-form audio, because as the duration of audio increases, the difficulty of SLI increases exponentially due to data sparsity (e.g. there might be long pause or music in long-form audio) and complexity (e.g. long-form audio usually involves multiple speakers as well as diverse speech content and styles). One typical example of long-form audio is podcast. As an increasingly popular media format, podcast data has become a major repertoire of speech content: as of March 2023, there are over 2.5 million podcast shows and more than 70 million episodes in over 100 different languages [3] with over 460 million podcast listeners globally [4]. The rich affordance of speech content from podcast not only opens up new realms for speech content processing and understanding, but also poses new and unique challenges to existing SLI systems: heterogeneous by nature, different podcast shows can have very different duration lengths, speech styles (e.g. scripted vs. spontaneous), number of speakers, and languages (e.g. monolingual vs. multilingual).

In this paper, we propose the first **lightweight and efficient spoken language identification system** for detecting languages from **long-form podcast audio** that can be of arbitrary length and have multiple speakers. We trained a lightweight multi-class feedforward neural network with VGGVox speaker embeddings [5] as input, and we evaluated our system on a human-verified podcast dataset containing five languages (English, Spanish, German, Portuguese, and Swedish) as well as on the same set of languages from the public VoxLingua107 dataset [6], where our model achieved strong results without requiring any audio preprocessing or denoising besides input normalization. We also compared our model against a pre-trained state-of-the-art (SOTA) SLI model i.e. ECAPA-TDNN [7] on the same podcast test set. We observed that ECAPA-TDNN ran into the Out-of-Memory issue when predicting languages for episodes longer than 15 minutes, while our model is able to predict languages for podcast episodes of five hours and longer. The ability of making efficient inference is important, particularly for long-form multilingual audio, because this can avoid biases introduced by audio sampling (e.g. having to sample short clips from long audio) and enable multilingual language detection directly on the original audio without having to split the audio into chunks of shorter lengths or different languages.

2. Previous Work

Existing SLI systems mainly process short-form, monolingual, and single-speaker audio. For instance, Mandal et al. [8] proposed an attention-based convolutional recurrent neural network extracting Mel-frequency Cepstral Coefficients from audio for language identification; their approach, trained on the Indian Language Dataset [9], uses short-form audio samples that are on average less than 30 seconds. Sarthak et al. [10] trained an attention-based SLI model using as input log-Mel spectro-

[⋆]equal contribution; [‡]work done during internship at Spotify.

[1] https://cloud.google.com/speech-to-text/docs/multiple-languages

10.21437/Interspeech.2023-304

gram images on the VoxForge Dataset [11] of short-utterance audio clips in one of the following languages: English, French, German, Spanish, Russian, and Italian. Li et al. [12] used a novel loss function named `tuplemax` loss to replace the commonly used softmax loss function so as to model the prior knowledge of a common speaker who can usually speak a small set of languages instead of a large set of all possible languages; their system only processes short-form audio, using as input utterances truncated to the first 4 seconds.

Existing SLI datasets only contain short-form audio as well, such as the NIST Language Recognition Evaluation (LRE) dataset [13], the Common Language Dataset [14], and the VoxLingua107 dataset [6] etc., of which the audio length varies from a few seconds to a few minutes, which is much shorter than the average duration of commonly seen long-form audio media such as podcast, as over 50% of podcast are between 20 and 60 minutes long according to studies until March, 2023 [15].

As a result, existing SLI systems and datasets are not suited for tackling long-form audio which is more complex due to its heterogeneous characteristics, such as longer duration, multiple speakers, diverse speech styles, multilinguality, and code-switching.

3. Data

A type of long-form audio that involves a lot of variety and complexity is podcast. We use podcast audio to illustrate how our proposed system applies to long-form audio. In this section, we first highlight some unique characteristics of podcast data in general (§3.1) and then present details of the training and test data used in our study (§3.2).

3.1. Overview of Podcast Data

We randomly sampled 30,000 podcast episodes from the Spotify podcast catalogue available online and gathered data of (1) **episode duration** from metadata fetched using Spotify Podcast API [16] and (2) **estimated number of speakers** by applying unsupervised speaker diarization [17] to the podcast audio.

Figure 1 shows the duration distribution of the sampled episodes, where only 22.5% of the episodes have a duration shorter than 5 minutes, and around half of podcast episodes (40.9%) are longer than 30 minutes. Thus, in order to process podcast and other long-form speech media, an SLI system needs to be able to process much longer audio than existing SLI datasets (e.g. utterance-length audio clips that are usually a few seconds long). Figure 2 shows the distribution of estimated number of speakers from the sampled episodes. Even though the majority (71.5%) have only 1 speaker, there are still 28.5% of episodes from the sample set that contain at least 2 speakers. Thus, it is important for a long-form audio SLI system to be able to handle multi-speaker audio data.

Besides processing audio of arbitrary length and with unrestricted number of speakers, our work also aims to make SLI systems aware of underrepresented speaker demographics [18] by steering system development away from adhering closely to standard scripted speech styles that are typically over-represented by existing SLI datasets mentioned in Section 2.

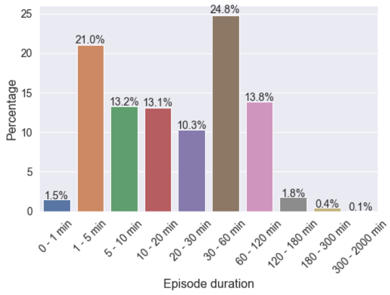


Figure 1: *Duration Distribution of Sampled Podcast Episodes*

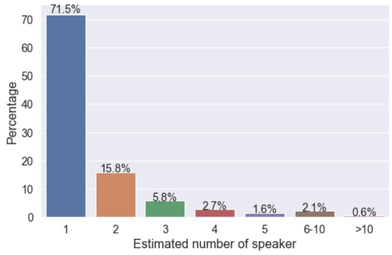


Figure 2: *Estimated Number of Speaker(s) in Sampled Podcast Episodes*

3.2. Training and Test Data

We used Spotify podcast data[2] for constructing both the train and test sets for training and evaluating our long-form audio SLI system. We selected the top 10 languages[3] spoken in the Spotify podcast catalogue based on fetched metadata as our **training languages**. We then randomly sampled 1,000 episodes from different shows for each of the 10 training languages: each sampled episode has a language label from metadata that is directly provided by the creator when uploading the episode [19]. It is worth noting that creator-provided language labels are noisy and can be error-prone due to lack of human verification.

Similar to the training set, for testing, we also sampled from different shows from the Spotify podcast catalogue and constructed a podcast test set that contains **five test languages** (English, Spanish, German, Portuguese, and Swedish), while ensuring that there is no overlap between training and test sets at the show level. Each test language has 1,000 randomly sampled episodes (deduplicated at the show-level from the training set), and the whole test set has been verified by human annotators to ensure that language labels are correct. Specifically, 8,868 episodes were randomly sampled from the Spotify catalogue in English, German, Spanish, Portuguese, and Swedish, with at most one episode per show to increase data diversity, from which 1,000 episodes were then sampled for each test language. An Appen annotation task[4] was created for each 30-second snippet sampled from each episode, for which human annotators were asked to (1) verify the episode language and (2) manually transcribe the sampled 30-second snippet.

Furthermore, in order to benchmark against existing SLI systems to show that our system can also generalize well to short-form audio, we also constructed a second test set from the **VoxLingua107 dev set** [5] containing only short and utterance-level audio clips (i.e. each audio clip on average ranges from 3 to 5 seconds long and contains one utterance), which is largely

[2] `https://open.spotify.com/genre/podcasts-web`

[3] The top 10 languages spoken in the Spotify podcast catalogue, as of 2022 January, are: English, Spanish, Portuguese, German, French, Indonesian, Swedish, Italian, Chinese, and Welsh.

[4] `https://appen.com/`

Language	train set long-form audio (podcast)	test set 1 long-form audio (podcast)	test set 2 short-form audio (VoxLingua107)
	duration	duration	duration
English (en)	1,067 hr	1,103 hr	874 sec
Spanish (es)	1,089 hr	1,120 hr	605 sec
Portuguese (pt)	1,068 hr	1,093 hr	6 sec
German (de)	1,028 hr	1,102 hr	876 sec
Swedish (sv)	1,058 hr	1,067 hr	1,014 sec
French (fr)	1,070 hr	–	–
Indonesian (id)	1,022 hr	–	–
Italian (it)	1,070 hr	–	–
Chinese (zh)	1,035 hr	–	–
Welsh (cy)	1,065 hr	–	–
total duration	10,572 hr	5,485 hr	3,375 sec

Table 1: *Overview of Training and Test Data Composition*

different from the podcast data in terms of both duration and speech style. The inclusion of this second test set serves to benchmark the performance of our system on a public dataset, which can be used to compare against existing SLI systems and show that our proposed system, albeit only trained on long-form podcast audio data, is able to generalize well to short-form audio data and achieve strong performance. Table 1 shows the data composition of our training and test sets.

4. System Design

Our proposed approach to SLI consists of two main steps: (1) Apply unsupervised speaker diarization [17] to generate speaker-level VGGVox speaker embeddings [5]; (2) Feed the averaged speaker embedding into a multi-class feedforward neural network (FNN) for training and inference.

4.1. Generation of Diarized Speaker Embeddings

For both training and inference, given an input podcast audio that is on average longer than 30 minutes (input audio duration of training and test sets follows the same distribution as shown in Figure 1), we extract its raw waveform data and re-sample at 16k Hz. We then perform an unsupervised speaker diarization task using the technique proposed by Tanveer et al. [17] on the re-sampled waveform to generate a series of 512-dimensional VGGVox speaker embedding vectors [5], with each speaker embedding vector corresponding to a segment where a speaker is dominantly speaking. The VGGVox embeddings are trained over a large-scale, wildly collected dataset i.e. VoxCeleb [5] that consists of over a million short utterances from over 7,000 celebrities all over the world. Our choice of VGGVox embeddings as input was motivated by its applicability as an embedding model trained over a large-scale representation of diverse languages and dialects which are available within the VoxCeleb dataset [5].

To convert the series of speaker embeddings as input into an FNN model, we perform z-score normalization across all individual speaker embeddings for the same audio and convert them into an averaged speaker embedding. The averaged VGGVox speaker embedding is then fed to an FNN model for both training and inference. Figure 3 (top) shows the process of speaker diariazation and speaker embedding generation.

4.2. Model Architecture

Figure 3 also provides an overview of the proposed SLI system, VGGVox-FNN, where the averaged VGGVox speaker embedding generated from Step 1 is fed as input into the FNN model as Step 2, a lightweight architecture consisting of two similar blocks where each block has two dense layers, one batch normalization layer and one dropout or softmax layer.

As shown in Figure 3, one advantage of our proposed system is that it is lightweight and efficient: the FNN has only 181,983 trainable parameters and no hyperparameter tuning was conducted during training; once trained, the model is fast to run at inference time, and it does not require GPU or any industry-level compute resources to run. Specifically, at training time, the model training completed within 1 hour on a standard Dataflow `n1-standard-4` machine;[5] at inference time, on the same `n1-standard-4` machine, an audio clip of 1 hour long can be inferred within 2-3 seconds provided that the input speaker embeddings are already generated; adding up the training and inference time leads to the total system runtime, which is very fast and efficient. Furthermore, the maximum memory usage at inference time (using the same `n1-standard-4` machine) is less than 2GB, which is very lightweight. Hence, being lightweight and efficient makes our system readily available for scaling up (e.g. inside a multilingual system such as machine translation) with low cost at runtime, which is critical for systems that need to be served at scale with tight inference budget [20].

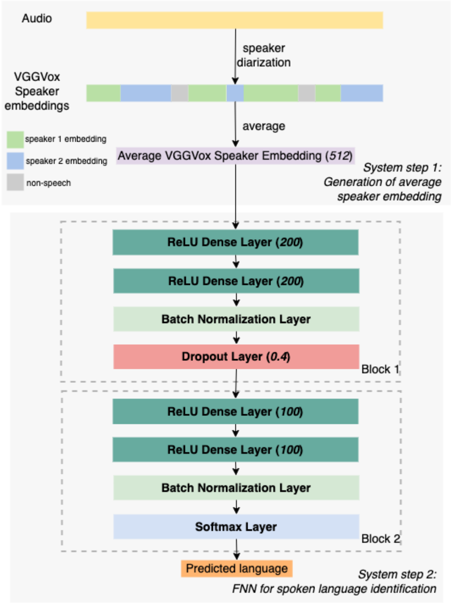


Figure 3: *Proposed SLI System: VGGVox Embedding Input and FNN Classifier*

4.3. Training Process

During training, we applied a 90%/10% train/validation split, and we ensured that every language label was evenly present in each split. We used a batch size of 10 and a dropout rate of 0.4 and trained the model for 500 epochs. We also applied `ReduceLROnPlateau`[6] to reduce learning rate when the loss has stopped improving for over 5 epochs and `EarlyStopping`[7] to stop training when the loss has stopped improving for over 8 epochs.

[5] Dataflow machine types: `https://cloud.google.com/compute/docs/machine-resource`

[6] `https://keras.io/api/callbacks/reduce_lr_on_plateau/`

[7] `https://keras.io/api/callbacks/early_stopping/`

5. Results

After training our proposed system on the podcast `train set` (Table 1), we evaluate it on the two `test sets` (Table 1) described in Section 3.2. We present and discuss the results below.

5.1. Evaluation Results 1: Podcast Test Set

Table 2 shows the evaluation results (averaged over three runs) of our proposed system on the long-form audio podcast test data i.e. `test set 1` in Table 1. Overall, our proposed system achieved an average F1 score of 91.23 across all test languages.

For benchmarking purposes, we also ran the pre-trained ECAPA-TDNN model, a SOTA spoken language recognition model trained on the VoxLingua107 dataset using SpeechBrain [7] [21], on the same podcast test set using the same type of `n1-standard-4` machines on Google Cloud Platform.[8] However, we noticed that at inference time, the ECAPA-TDNN model failed to predict languages for podcast episodes longer than 15 minutes due to Out-of-Memory error; by contrast, our system is able to predict languages for episodes of five hours and longer.

System / Language	VGGVox-FNN			
	Precision	**Recall**	**F1**	**AUC**
English (en)	97.07	88.33	92.50	0.99
Spanish (es)	93.93	87.67	90.69	0.98
German (de)	98.15	88.67	93.17	0.99
Portuguese (pt)	88.46	88.33	86.74	0.95
Swedish (sv)	94.50	91.67	93.06	0.98
mean	**94.42**	**88.93**	**91.23**	**0.98**

Table 2: *Results on the Podcast Test Set (`test set 1`)*

5.2. Evaluation Results 2: VoxLingua107 Dev Set

Table 3 shows the evaluation results (averaged over three runs) of our proposed system on the short-form audio VoxLingua107 dev set i.e. `test set 2` in Table 1. While our approach was trained only on long-form audio podcast data, which is significantly different from the VoxLingua107 data (i.e. shorter utterances rather than long audio, scripted rather than spontaneous, single-speaker rather than multi-speaker etc.) that was unseen during training, our approach still achieved >80% precision and recall on all test languages except for Portuguese, which is largely due to the small data size of Portuguese (there is only one Portuguese audio clip of 6 seconds long in the VoxLingua107 dev set).

The evaluation results on `test set 2` demonstrate that our proposed system, VGGVox-FNN, albeit only trained on long-form audio such as podcast, can generalize well to short-form audio such as utterance clips;[9] by contrast, it is harder for any existing SLI system trained on short-form audio to generalize to long-form audio: one example being the ECAPA-TDNN model which fails to predict languages for audio longer than 15 minutes (Section 5.1).

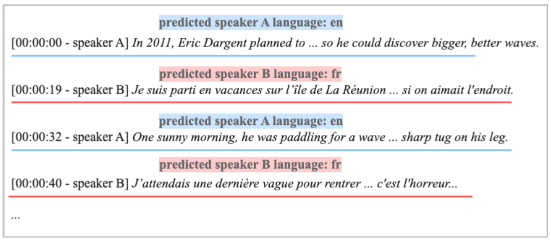
System / Language	VGGVox-FNN			
	Precision	**Recall**	**F1**	**AUC**
English (en)	88.33	85.48	86.89	0.97
Spanish (es)	80.65	83.32	81.97	0.98
German (de)	85.07	83.82	87.69	0.96
Portuguese (pt)	50.00	100.00	66.67	0.99
Swedish (sv)	81.82	88.89	85.21	0.93
mean	**77.17**	**88.30**	**81.69**	**0.97**

Table 3: *Results on the VoxLingua107 Dev Set (`test set 2`) for Selected Languages*

6. Discussion: Speaker-Level SLI for Code-switching

In Section 4 we described our approach that takes as input a VGGVox speaker embedding for SLI, and then in Section 5 we showed that when taking the **average speaker embedding** of a long audio (i.e. podcast) as input, the system predicts a single language spoken in the audio.

For long-form audio, code-switching is rather common, for example, different speakers may speak different languages in the same podcast episode. Our system can be easily adapted to perform **speaker-level SLI** to tackle code-switching audio, in which case, instead of using the average speaker embedding as input, we can simply use the **individual speaker embeddings** as input, and the system will then predict the languages spoken by each individual speaker where the predictions can involve one or multiple languages in case of code-switching. A limitation of this work is that because no annotated data at utterance-level is available, a large-scale speaker-level SLI evaluation is out of scope of the current study. However, we provide an example below which demonstrates the applicability of our system to speaker-level SLI. Figure 4 shows an example of applying the system to *Episode 6: Le surfeur sans limites (The Surfer Without Limits)* from *Duolingo French Podcast* [23]: the audio of this episode contains code-switching, where speaker A speaks in English as a narrator and speaker B speaks in French as an interviewee; the system predicts different languages at speaker-level where speaker A's language is predicted as English ("en") and speaker B's as French ("fr").



predicted speaker A language: en

[00:00:00 - speaker A] *In 2011, Eric Dargent planned to ... so he could discover bigger, better waves.*

predicted speaker B language: fr

[00:00:19 - speaker B] *Je suis parti en vacances sur l'île de La Réunion ... si on aimait l'endroit.*

predicted speaker A language: en

[00:00:32 - speaker A] *One sunny morning, he was paddling for a wave ... sharp tug on his leg.*

predicted speaker B language: fr

[00:00:40 - speaker B] *J'attendais une dernière vague pour rentrer ... c'est l'horreur...*

...

Figure 4: *Example of Speaker-level SLI for Code-switching Audio*

7. Conclusions

We proposed the first spoken language identification (SLI) system that can predict languages for long-form audio, which poses extra challenges to existing SLI systems due to its heterogeneous nature (e.g. diverse speech styles, multiple speakers, varied duration, multilinguality). We use podcast audio as a representation of long-form audio, and we trained and evaluated a system that is lightweight and efficient at runtime, hence easy to scale up and crucial for deployment to production. Our proposed system achieved strong performance on long-form podcast audio, and it was shown to generalize well to short-form audio even if it was unseen during training. Finally, our system can be easily adapted to perform speaker-level SLI to detect multiple languages spoken by different speakers within the same audio, which is important when processing and analyzing audio containing code-switching.

[8] `https://cloud.google.com/compute/docs/machine-resource`

[9] The mean error rate of our proposed system on the five evaluated languages from the VoxLingua107 dev set is 6.1%, on par with the error rates of the SOTA SLI systems presented in Babu et al. [22] which range from 7.2% to 5.7% (though averaged over all 33 languages from the VoxLingua107 dev set).

8. References

[1] C. Zhang, B. Li, T. Sainath, T. Strohman, S. Mavandadi, S.-Y. Chang, and P. Haghani, "Streaming End-to-End Multilingual Speech Recognition with Joint Language Identification," in *Proc. Interspeech 2022*, 2022, pp. 3223–3227.

[2] P. Heracleous, K. Takai, K. Yasuda, Y. F. O. Mohammad, and A. Yoneyama, "Comparative study on spoken language identification based on deep learning," *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 2265–2269, 2018.

[3] N. Schaffer, "The Top 25 Podcast Statistics You Need to Know in 2023)," https://nealschaffer.com/podcast-statistics/, 2023, [Online; accessed 02-Mar-2023].

[4] D. Ruby, "48 Podcast Statistics In 2023 (Listeners, Consumption & Trends)," https://www.demandsage.com/podcast-statistics/, 2023, [Online; accessed 2023].

[5] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep Speaker Recognition," in *Proc. Interspeech 2018*, 2018, pp. 1086–1090.

[6] J. Valk and T. Alumäe, "Voxlingua107: A dataset for spoken language recognition," in *2021 IEEE Spoken Language Technology Workshop (SLT)*, 2021, pp. 652–658.

[7] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in *Proc. Interspeech 2020*, 2020, pp. 3830–3834.

[8] A. Mandal, S. Pal, I. Dutta, M. Bhattacharya, and S. K. Naskar, "Is Attention Always Needed? A Case Study on Language Identification from Speech," *SSRN Electronic Journal*, 2022. [Online]. Available: https://doi.org/10.2139%2Fssrn.4186504

[9] A. Baby, A. L. Thomas, N. Nishanthi, and TTS Consortium, "Resources for Indian Languages," in *Proceedings of Text, Speech and Dialogue*, 2016.

[10] Sarthak, S. Shukla, and G. Mittal, "Spoken Language Identification Using Convnets," in *Ambient Intelligence*, I. Chatzigiannakis, B. De Ruyter, and I. Mavrommati, Eds. Cham: Springer International Publishing, 2019, pp. 252–265.

[11] K. MacLean, "Voxforge," *Ken MacLean.[Online]. Available: http://www.voxforge.org/home.[Acedido em 2012]*, 2018.

[12] L. Wan, P. Sridhar, Y. Yu, Q. Wang, and I. L. Moreno, "Tuplemax loss for language identification," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5976–5980.

[13] S. O. Sadjadi, T. Kheyrkhah, C. Greenberg, E. Singer, D. Reynolds, L. Mason, and J. Hernandez-Cordero, "Performance Analysis of the 2017 NIST Language Recognition Evaluation," in *Proc. Interspeech 2018*, 2018, pp. 1798–1802.

[14] G. Sinisetty, P. Ruban, O. Dymov, and M. Ravanelli, "Commonlanguage," Jun. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.5036977

[15] A. Brooke, "Podcast Statistics and Data," https://www.buzzsprout.com/blog/podcast-statistics, 2023, [Online; accessed Mar-2023].

[16] J. Brown, "Search, browse and follow podcasts using the new Podcast APIs," https://developer.spotify.com/blog/2020-03-20-introducing-podcasts-api, 2020, [Online; accessed 20-Mar-2020].

[17] M. I. Tanveer, D. Casabuena, J. Karlgren, and R. Jones, "Unsupervised speaker diarization that is agnostic to language, overlap-aware, and tuning free," in *Proc. Interspeech 2022*, 2022, pp. 1481–1485.

[18] Z. Mengesha, C. Heldreth, M. Lahav, J. Sublewski, and E. Tuennerman, "I don't think these devices are very culturally sensitive.—Impact of automated speech recognition errors on African Americans," *Frontiers in Artificial Intelligence*, vol. 4, 2021. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frai.2021.725911

[19] Spotify for Developers, "Web API: Retrieve metadata from Spotify content, control playback or get recommendations," https://developer.spotify.com/documentation/web-api.

[20] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "LLaMA: Open and Efficient Foundation Language Models," 2023. [Online]. Available: https://arxiv.org/abs/2302.13971

[21] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, J.-C. Chou, S.-L. Yeh, S.-W. Fu, C.-F. Liao, E. Rastorgueva, F. Grondin, W. Aris, H. Na, Y. Gao, R. D. Mori, and Y. Bengio, "SpeechBrain: A General-Purpose Speech Toolkit," 2021, arXiv:2106.04624.

[22] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino, A. Baevski, A. Conneau, and M. Auli, "XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale," *arXiv*, vol. abs/2111.09296, 2021.

[23] "Duolingo Podcast French." [Online]. Available: https://podcast.duolingo.com/french