Interspeech 2022
18-22 September 2022, Incheon, Korea

A Multimodal Strategy for Singing Language Identification

Wo Jae Lee[1], *Emanuele Coviello*[1]

[1]Amazon Music

{leeawo,emacov}@amazon.com

Abstract

Identification of the language of performance of songs is important for applications such as personalized recommendations, discovery, and search. In this paper, we present an automated multimodal approach to identify the singing language of songs that scales to millions of songs. The proposed model uses a variety of song-level features, including a consumption embedding derived from sessions listening data from a music streaming service, segment-level vocals embedding computed from the vocal track of a song, and generic timbral features. Our experimental results show that our approach outperforms benchmark models in the signing-language identification task, and demonstrates the benefit of the multimodal approach through an ablation study. In addition, we present a data augmentation technique to increase the robustness of the model to missing data modalities.

Index Terms: singing language identification, multimodal model, acoustic features, deep learning

1. Introduction

Music streaming services provide a global and diverse user base with access to vast catalogs of several million songs. To create a truly personalized customer experience it is important to accommodate specific use cases such as satisfying the language preferences of users through browsing, discovery, and search. High-quality language metadata for songs is usually not readily available for large portions of music catalogs, while manual curation efforts do not scale up to support a personalized listening experience. Similarly, whereas singing language could be inferred from the text of the lyrics, lyrics files are not easily sourced for large catalogs.

There is a large corpus of research [1, 2, 3, 4] and online services[1] that address the problem of language detection. However, these models are trained on *spoken* speech and human dialog, thus they have lower accuracy when applied to *singing* language in music (see [5] or our results in Section 4.3), due to variable phonation in singing.

In this paper, we present a state-of-the-art solution for the identification of the singing language of songs that is optimized for music and that scales to large catalogs. In particular, we propose a multimodal approach for language tagging based on a variety of music-related features, namely song embeddings derived from a consumption model based on same-sessions listening, vocals embeddings computed from the vocal track automatically extracted from a song, and generic timbral features. Our underlying assumption for the multimodal approach is that the audio-based features and the consumption features are complementary to each other.

Since under several production settings consumption embedding data might not be available (e.g., new releases, long tail of a catalog, or an external facing service API), we propose a simple but effective data-augmentation technique to increase the robustness of the multimodal model to missing consumption embeddings. We present experiments on a dataset comprising music in 10 different languages that demonstrate the validity of our approach and the usefulness of the multimodal approach.

We discuss related work in Section 2, introduce the approach and features in Section 3, and present the dataset and experiments in Section 4.

[1]For Amazon AWS Transcribe or Google Cloud Translate.

2. Related Work

Singing can be considered as a special form of speech (i.e., the musical aspect of voice with some portion of non-lexical vocables). Therefore, similar to a development of the spoken language identification models [6, 7], various acoustic features and machine learning classifiers have been extensively used in the singing language identification models [8, 9]. Experimental results have shown that the classification of singing voices can be done more robustly when separating the singing signals of a song from the background using vocals source separation models [5, 10, 11]. By extracting the vocals sound, a pre-trained spoken language model, which was trained on various speakers and diverse languages data, can be leveraged and customized for a downstream task such as language detection [3, 12]. A popular pre-trained spoken language model consists of a x-vector deep learning model [13] which maps sequences of speech signals to fixed-length embeddings, where the embeddings corresponding to the same language cluster together [3, 12, 14]. Although these pre-trained models have achieved promising results in speech recognition tasks, their performance for the identification of singing language is still left to be explored.

More recently, multimodal approaches have been explored for singing language identification. For example, [15] uses multiple audio-based low-level features, [16] uses a combination of low level generic song features and simple metadata fields (such as album title and artist name), and [17] uses a combination of low level audio features and visual features to detect the language of music videos. [18] explores a combination of non-audio metadata features, including song and album titles, regional popularity, and a song-embedding derived from playlists.

Compared to related work on multimodal approaches, the novelty of our approach is that we use a richer audio representation by introducing vocals embeddings computed from individual voiced segments of songs, and combine them with a high level representation of timbral content (Section 3.1.3) as well as consumption-based song embeddings (Section 3.1.1). Our results confirm the finding of [18] in regards to the predictive strength of consumption based embeddings. However, we also demonstrate that carefully designed audio and vocals features achieve near optimal performance, with a non-negligible contribution over consumption features (see results in Section 4.3.1). In addition, we also propose a simple but effective data augmentation approach to increase the robustness of a multimodal model to missing consumption embeddings.

10.21437/Interspeech.2022-11007

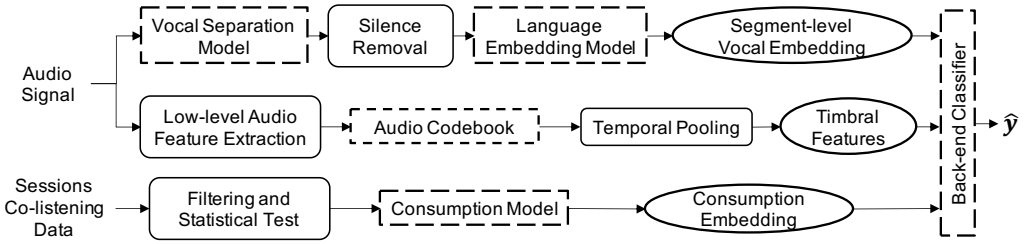

Figure 1: *Block diagram of the proposed method. A dotted rectangular box indicates a learning-based model, a circle indicates the multimodal features, and a rectangular box with rounded edges indicates a mathematical operation*

3. Approach

We formulate signing language identification as a supervised multi-class classification problem where each class corresponds to a language (e.g., English, Italian, Tamil, etc.). Each song is represented by a set of multimodal features $X = [\mathbf{x}_1, \cdots, \mathbf{x}_m]$, representing m different modalities, such as those described in Section 3.1, and is associated with a language class $y \in [1, \ldots, L]$, where L is the number of classes. For songs with multiple languages, we simplify the problem to the identification of the dominant language. The goal of the model is to infer the language of unseen songs.

For this purpose, we train a statistical model to capture the relationship between a song's features and class membership, from a dataset of annotated songs $\mathcal{D} = \{(X_d, y_d)_{d=1}^N\}$ where N is the size of the dataset. We assume the model is a neural network with a softmax output layer, that predicts a vector of posterior probabilities for each class $\hat{y}_i = P(y_i|X)$ on the probability simplex. We present the multimodal features in Section 3.1, and the multimodal model in Section 3.2.

3.1. Song Features

3.1.1. Consumption song embedding

We built the consumption song embeddings as follows. First, from historical user playback data we generate track pairs that are listened in the same session, counting both pair frequency and marginal frequencies. We filter out pairs based on a minimum number of sessions and distinct customers. We then use a statistical test to generate scores on the similarity strength of each pair [19], keep the top-k similarities for each track, and further remove hubness [20] and popularity bias. Finally, to learn track-level embeddings v_i, we define affinity between two tracks as the dot product of their embeddings. We then optimize, using stochastic gradient descent, a weighted cross-entropy between a track source and its top-k similar tracks, of the form $\sum_{i,j} s_{i,j} \log \text{softmax}_j \exp v_i' v_j$ using negative sampling [21], where $s_{i,j}$ is an increasing function of the similarity strength of the pair (i, j). The several hyper-parameters of the approach are optimized based on a track ranking task on a held-out validation test of track pairs.

3.1.2. Segment-level vocals embedding

We represent the vocals of a song as a segment-level vocals embedding by the following steps: (1) vocals separation from instrumentation, (2) silence removal (i.e., we only retain segments with vocals), and (3) embedding over a language space.

For vocals separation we use Spleeter [11], which consists of a pre-trained neural network based on a 12-layer U-nets (encoder/decoder CNN) architecture. Spleeter achieved state-of-the-art performance on a music source separation task on the musdb18 dataset [22], showing consistent performance across a variety of music genres. From the automatically separated vocal track we split it into contiguous intervals of vocals by removing silence intervals, marking as silence those frames below a dB threshold.[2] Removing portions without vocals reduces the computation load in the following step, while using multiple segments increase the robustness of the representation. Finally, for each vocals segment we extract a vocals embedding using a pre-trained spoken language embedding model [3] based on the *x*-vector model [23], and trained on the VoxLingua107 dataset which includes 107 different languages. Note that we obtain a vocals embedding for each of the segments of a song. The spoken language model covers a significantly larger number of languages than those included in our dataset, which intuitively helps increase robustness of end-to-end classifier to false positives from languages not included in the music dataset presented in Section 4.1.

3.1.3. Generic timbral features of a song

To represent the timbral content of a song, we use a *pooled audio codebook histogram* [24, 25], that consists of: (1) low-level audio feature extraction, (2) audio codebook construction, and (3) temporal pooling.

We first represent the acoustic content of a song with Mel-frequency spectral features over half-overlapping windows. We then use a principal component analysis (PCA) transformation to project the spectral features onto the top *P* principal components to retain 95% of variance (we train the PCA model on an unlabeled corpus of songs). To these, we append first and second instantaneous derivatives, which results in a sequence of 3*P*-dimensional low-level audio features, $(\phi_i \in \mathbb{R}^{3P})_{i=1}^T$, where *T* is the number of windows. Next, we encode the low-level audio features using an audio codebook of Gaussian codewords, following the approach of [24]. We build the codebook from the unlabeled corpus of songs, using expectation-maximization [26] to learn a Gaussian Mixture Model (GMM) on the Mel-PCA features. We encode each song as a sequence of audio codebook histograms by mapping each of the audio feature vector to a vector of posterior probabilities of the GMM components. Finally, we use temporal pooling to summarize the sequence of histograms of a song over its entire duration [27]. This has the advantage of transforming variable length sequences of features to compact fixed-length feature vectors, which are invariant to shifts in time and robust to noise. In particular, we concatenate max- and mean- pooling, to capture both average and locally prominent codewords.

[2] We adjusted this threshold in a preliminary experiment based on a qualitative assessment of the silence removal.

Table 1: *Average and per-language performance of variants of our model using different subsets of music features, and spoken language baselines. We report mean average precision (mAP) and annotation precision (P), recall (R) and F1-score (F).*

Language	Timbral				Vocals				Consumption				Audio				Multimodal (Ours)				VoxLingua				CommonLanguage			
	P	R	F	mAP	P	R	F	mAP	P	R	F	mAP	P	R	F	mAP	P	R	F	mAP	P	R	F	mAP	P	R	F	mAP
de	0.574	0.597	0.585	0.636	0.881	0.930	0.905	0.964	0.954	0.974	0.964	0.988	0.852	0.930	0.889	0.982	0.995	0.981	0.988	0.999	0.705	0.543	0.613	0.677	0.831	0.818	0.825	0.893
en	0.496	0.573	0.532	0.569	0.742	0.846	0.790	0.861	0.972	0.934	0.953	0.982	0.738	0.900	0.811	0.938	0.980	0.979	0.979	0.993	0.595	0.519	0.554	0.602	0.607	0.635	0.620	0.674
es	0.769	0.560	0.648	0.708	0.876	0.883	0.879	0.934	0.969	0.959	0.964	0.989	0.886	0.891	0.889	0.946	0.992	0.983	0.987	0.994	0.810	0.563	0.665	0.738	0.729	0.569	0.639	0.707
fr	0.606	0.546	0.574	0.611	0.943	0.925	0.934	0.969	0.980	0.968	0.974	0.994	0.921	0.932	0.926	0.984	0.999	0.982	0.990	0.999	0.865	0.621	0.723	0.767	0.886	0.768	0.823	0.891
hi	0.533	0.598	0.563	0.603	0.786	0.871	0.826	0.886	0.930	0.957	0.943	0.979	0.772	0.892	0.828	0.910	0.954	0.984	0.969	0.995	0.308	0.618	0.412	0.352	NA	NA	NA	NA
it	0.420	0.562	0.481	0.500	0.856	0.863	0.859	0.928	0.987	0.966	0.977	0.993	0.876	0.871	0.873	0.962	0.989	0.993	0.991	0.997	0.817	0.558	0.663	0.703	0.853	0.666	0.748	0.812
ja	0.759	0.684	0.720	0.799	0.699	0.898	0.786	0.887	0.955	0.966	0.961	0.991	0.753	0.844	0.796	0.965	0.988	0.980	0.984	0.989	0.237	0.519	0.325	0.383	0.548	0.513	0.530	0.550
pa	0.783	0.629	0.697	0.773	0.978	0.489	0.652	0.933	0.947	0.927	0.937	0.934	0.943	0.564	0.705	0.942	0.970	0.986	0.978	0.989	0.778	0.682	0.727	0.785	NA	NA	NA	NA
ta	0.549	0.662	0.600	0.620	0.863	0.851	0.857	0.930	0.960	0.976	0.968	0.968	0.862	0.890	0.876	0.951	0.957	0.952	0.955	0.998	0.467	0.678	0.553	0.603	0.509	0.663	0.576	0.606
te	0.375	0.550	0.446	0.402	0.849	0.869	0.859	0.906	0.982	0.968	0.975	0.992	0.869	0.874	0.872	0.908	0.987	0.991	0.989	0.996	0.470	0.426	0.447	0.486	NA	NA	NA	NA
Avg.	0.586	0.596	0.585	0.622	0.847	0.842	0.835	0.920	0.964	0.960	0.962	0.981	0.847	0.859	0.847	0.949	***0.981***	***0.981***	***0.981***	***0.995***	0.605	0.573	0.568	0.610	0.709	0.662	0.680	0.733

3.2. Multimodal Strategy

We present the block diagram of our multimodal approach in Figure 1. We first extract the various features from the audio signal and the consumption data, and then classify these features with a multi-class language prediction model. In this paper we adopted the 1DResNet architecture [28] that consists in stacking tree three residual blocks, followed by a global average pooling layer and a softmax layer. This architecture performed the best against alternatives based on preliminary experiments, and we do not report the results for competing architectures here for brevity. When using the segment-level vocals embedding (extracted on individual segments as discussed in Section 3.1.2), at training time we fan out both all other features and language labels at the segment level, and at inference time we compute song-level predictions by combining the segment level predictions, which is a weighted sum of the segment-level prediction scores based on the duration of each segment.

In our production settings the features based on the audio signals are always available, however the consumption embeddings might be missing. To build robustness to this we experimented with a data augmentation technique where we introduced a replica of each example in the training set but dropped the consumption embedding. Our goal is that when consumption features are missing this model falls back to the classification quality of the audio-level model, which would allow maintaining a single model in a production system.

4. Experiments

4.1. Singing language dataset

We conduct an experiment using an internal music language dataset. After the feature extraction, the dataset consists of 54,170 feature vectors (one per song) covering 10 different target languages: English (en), Hindi (hi), Spanish (es), German (de), Tamil (ta), Telugu (te), Japanese (ja), Punjabi (pa), Italian (it), and French (fr). From the dataset, we observed 25,874 unique artists and 529 unique genre labels. The dataset is sufficiently balanced across the languages (5,000-7,000 songs per language), and each song has a single language label. To increase the model robustness to additional languages that might be encountered in a production setting outside the 10 target languages, we also include a label "other languages" (ot) and sample 7,000 such songs. We split the dataset with a ratio of 70/15/15 between training, validation, and test set. Because a simple random allocation might favor models that memorize an artist's voice, we decided to split the dataset at the artist-level instead of at the track-level. This approach also limits potential information leakage between training, validation and test set (i.e., the consumption embeddings of songs by the same artist are generally closer to each other).

4.2. Models investigated

As a baseline, we compare the proposed model with two state-of-the-art spoken language recognition models – VoxLingua [3] and CommonLanguage [12] (hi, pa, and te are not supported in CommonLanguage). VoxLingua and CommonLanguage cover 107 and 45 languages, respectively. For both models, we used the implemented through SpeechBrain's speech toolkit [29], but we analyzed the automatically extracted vocals, instead of the songs, since it provided better performance. We also include four variants of our model that use a different subset of the multimodal features (see Table 1) presented in Section 3.1. In particular, we consider each feature alone, namely Timbral, Vocals, and Consumption, the combination of timbral and vocals features (Audio), and all the features together (Multimodal).

For training the proposed model, we use mini-batch SGD [30] with a mini-batch size of 64 over 50 epochs, and use adam [31] with a learning rate of 0.001. To combat over-fitting, we include an L2 regularization with a weight decay parameter of 0.001 and incrementally reduce the learning rates across epochs by monitoring a validation loss. We conducted an extensive search for all other hyper-parameter (including the number of channels in the 1DResent model, training optimizer, batch size, classifiers, etc.). Note that we also experimented with different classifiers (e.g., including multi layer perceptrons, CNNs or RNNs), standard low-level audio features as in [16] (e.g., Mel-spectrogram in lieu of the codebook encoding presented in Section 3.1.3), and loss functions, but for brevity we only report results for the best performing architecture.

4.3. Results

4.3.1. Evaluation of the multimodal approach

In Table 1 we report performance for the variants of our model using a different combination of features, in terms of mAP and classification precision (P), recall (R), and F1-score (F).[3]

First, we see that the vocals embedding outperforms the generic timbral features, with a mAP of 0.622 and 0.920, respectively (and on each individual language as well). Intuitively, by focusing only on vocals after eliminating other confounding sounds, these features provide a stronger signal for

[3]For language i we rank the order of all tracks based on $P(y_i|X)$ and compute the mAP as the area under the P-R curve. The mAP had the advantage that it provides a summary of a model quality independently of a specific P-R trade-off point. For P, R and F, for each language i we tune a language acceptance threshold τ_i to determine whether the class membership probability $P(y_i|X)$ is high enough to apply the language label, to optimize the F1 score on a validation set.

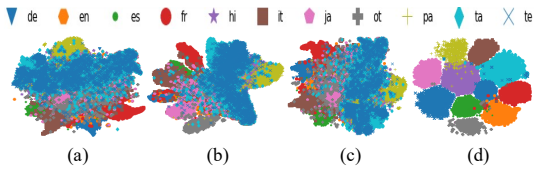

Figure 2: *t-SNE plots of (a) vocals embeddings, (b) output of the last hidden layer in Vocals model, (c) multimodal features, and (d) output of the last hidden layer in Multimodal model.*

the task. Interestingly, including also the timbral features leads to an improvement over the vocals features alone, reaching an mAP of 0.949 (again, the improvement is across all languages). A possible explanation is that the model takes advantage of correlations between certain acoustic characteristics (e.g., genres) and languages.

Second, we see that the consumption features are the single most performing features, generalizing the finding of [18] to an expanded feature set that also includes audio. This is explained by the intuition that same-sessions listening tends to cluster on cohesive themes (e.g., genres or languages), and consequently the session-based consumption features presented in Section 3.1.1 are predictive of language.

Importantly, the best model is the multimodal model that combines both the audio and consumption features, with an mAP of 0.995 (and superior performance on all languages). In Figure 2, we present t-SNE plots [32] for the visualization of vocals embeddings, multimodal features, and features extracted from the vocals model and multimodal model in Table 1. As illustrated in the figure, t-SNE fails to cluster the multimodal features in groups that have a consistent language (sub-figure (c)). After training the back-end classifier (i.e., 1DResNet) with the multimodal features, it demonstrates that the learned embedding represents the structure of languages remarkably well although there are a few deviations (i.e., false positive), probably caused by multilingual lyrics (sub-figure (d)).

4.3.2. *Comparison to spoken language models*

In Table 1 we also compare our models to the state-of-the-art spoken language recognition models. The results show a clear advantage for all variants of our models (except Timbral) based on a different combination of features, compared to directly applying spoken language models to the vocal track of a song. It's important to notice that, despite a spoken language model such as VoxLingua does not provide out-of-the-box a very accurate singing language prediction, it still provides informative features for training a singling language model. In fact, our vocals embedding features are derived from this model (over segments of uninterrupted vocals) and using them as input to a ResNet model (trained on the singing data ground truth) results in a mAP of 0.920. Instead, directly using the language prediction scores from VoxLingua model results in a considerably lower mAP of 0.610. As a corroboration to this, in Figure 2 we see that a t-SNE plot of the vocals embedding (sub-figure (a)) does not result in language clusters that are as well separated in a t-SNE plot on the hidden layer representation learned by the Vocals model trained on these features (sub-figure (b)).

4.3.3. *Robustness to missing consumption embeddings*

In this section we discuss robustness to missing consumption embeddings. This is a realistic scenario for a production system,

Table 2: *mAP for the multimodal model on a test set where we dropped the consumption features, without data augmentation (Baseline) and with data augmentation training.*

Language	Baseline		Data augmentation			
	`Z`	`A`	`Z`	`Z+G`	`A`	`A+G`
de	0.775	0.436	0.961	0.971	0.975	0.959
en	0.393	0.125	0.885	0.898	0.912	0.875
es	0.677	0.347	0.934	0.950	0.956	0.936
fr	0.876	0.780	0.966	0.972	0.976	0.966
hi	0.333	0.326	0.854	0.895	0.899	0.865
it	0.646	0.237	0.925	0.946	0.957	0.925
ja	0.442	0.087	0.917	0.926	0.930	0.896
pa	0.856	0.762	0.922	0.945	0.945	0.926
ta	0.530	0.404	0.899	0.918	0.937	0.894
te	0.669	0.540	0.849	0.888	0.908	0.846
Avg.	0.620	0.404	0.911	0.931	***0.940***	0.909

where for example consumption embeddings may not be available for new recordings, or for the long tail of a catalog. Our goal is to build robustness directly into the multimodal model in such a way that its performance falls back to that of an audio-only model (Audio) when the consumption data is missing.

In order to simulate this scenario, we run experiments where we drop the consumption embeddings from the test set. As a baseline, we first evaluate the mAP on this test set using the multimodal model trained on the full multimodal data, but then in the test set replacing the consumption features with default values of a vector of zeros (`Z`), or the average embedding (`A`). From the results in Table 2 (column "Baseline") we see that the mAP of the best performed model (`Z`) drops to 0.620, which is substantially lower than that for the audio-only model (with an mAP score of 0.949 as reported in Table 1). This suggests that the multimodal model trained on completed multimodal data is not robust to handling missing data modalities.

To increase robustness to missing consumption embeddings, we adopt a simple data augmentation procedure: for each track in the training set we also use a replica where we drop the consumption embeddings and replace them with a default fallback value. In Table 2 we report performance for different options for the fallback value: a vector of zeros (`Z`), the element-wise average of the consumption embeddings in the training set (`A`). We also test a variant where at training time we add Gaussian noise to each of the fallback vectors (`+G`). We see that using the average performs the best across the options we evaluated, with an mAP of 0.940 which is comparable to the performance of the audio-only model (with the advantage that it enables keeping a single model in production).

Note that we could use a similar data augmentation approach to build robustness to missing audio features. However, we do not expand on that here since in our production settings we always have access to the audio of a song.

5. Conclusion

We proposed a multimodal strategy for the identification of singing language based on a variety of music-related features. Our solution effectively predicts a singing language of a song, and the variants of the proposed model demonstrate the benefit of the multimodal approach across all languages. Further, we presented a simple yet effective data-augmentation technique to increase robustness to missing consumption features.

6. References

[1] H. Yu, J. Zhao, S. Yang, Z. Wu, Y. Nie, and W.-Q. Zhang, "Language recognition based on unsupervised pretrained models," *Proc. Interspeech 2021*, pp. 3271–3275, 2021.

[2] Z. Li, M. Zhao, J. Li, L. Li, and Q. Hong, "On the usage of multi-feature integration for speaker verification and language identification." in *INTERSPEECH*, 2020, pp. 457–461.

[3] J. Valk and T. Alumäe, "VoxLingua107: a dataset for spoken language recognition," in *Proc. IEEE SLT Workshop*, 2021.

[4] Z. Fan, M. Li, S. Zhou, and B. Xu, "Exploring wav2vec 2.0 on speaker verification and language identification," *arXiv preprint arXiv:2012.06185*, 2020.

[5] A. Mesaros, "Singing voice identification and lyrics transcription for music information retrieval invited paper," in *2013 7th Conference on Speech Technology and Human-Computer Dialogue (SpeD)*. IEEE, 2013, pp. 1–10.

[6] I. Lopez-Moreno, J. Gonzalez-Dominguez, O. Plchot, D. Martinez, J. Gonzalez-Rodriguez, and P. Moreno, "Automatic language identification using deep neural networks," in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2014, pp. 5337–5341.

[7] J. Gonzalez-Dominguez, I. Lopez-Moreno, H. Sak, J. González-Rodríguez, and P. J. Moreno, "Automatic language identification using long short-term memory recurrent neural networks," in *INTERSPEECH*, 2014.

[8] A. M. Kruspe, J. Abesser, and C. Dittmar, "A gmm approach to singing language identification," in *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio*. Audio Engineering Society, 2014.

[9] W.-H. Tsai and H.-M. Wang, "Automatic identification of the sung language in popular music recordings," *Journal of New Music Research*, vol. 36, no. 2, pp. 105–114, 2007.

[10] D. del Castillo Iglesias, "End-to-end learning for singing-language identification," 2020.

[11] R. Hennequin, A. Khlif, F. Voituret, and M. Moussallam, "Spleeter: a fast and efficient music source separation tool with pre-trained models," *Journal of Open Source Software*, vol. 5, no. 50, p. 2154, 2020, deezer Research. [Online]. Available: https://doi.org/10.21105/joss.02154

[12] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *Interspeech 2020*, Oct 2020. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-2650

[13] D. Snyder, D. Garcia-Romero, A. McCree, G. Sell, D. Povey, and S. Khudanpur, "Spoken language recognition using x-vectors." in *Odyssey*, 2018, pp. 105–111.

[14] R. Duroselle, D. Jouvet, and I. Illina, "Metric learning loss functions to reduce domain mismatch in the x-vector space for language recognition," in *INTERSPEECH 2020*, 2020.

[15] P. G. Shivakumar, S. N. Chakravarthula, and P. G. Georgiou, "Multimodal fusion of multirate acoustic, prosodic, and lexical speaker characteristics for native language identification." in *INTERSPEECH*, 2016, pp. 2408–2412.

[16] K. Choi and Y. Wang, "Listen, read, and identify: Multimodal singing language identification of music," *arXiv preprint arXiv:2103.01893*, 2021.

[17] V. Chandrasekhar, M. E. Sargin, and D. A. Ross, "Automatic language identification in music videos with low level audio and visual features," in *2011 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2011, pp. 5724–5727.

[18] L. Roxbergh, "Language classification of music using metadata," 2019.

[19] G. Linden, B. Smith, and J. York, "Amazon. com recommendations: Item-to-item collaborative filtering," *IEEE Internet computing*, vol. 7, no. 1, pp. 76–80, 2003.

[20] M. Radovanovic, A. Nanopoulos, and M. Ivanovic, "Hubs in space: Popular nearest neighbors in high-dimensional data," *Journal of Machine Learning Research*, vol. 11, no. sept, pp. 2487–2531, 2010.

[21] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[22] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "The MUSDB18 corpus for music separation," Dec. 2017. [Online]. Available: https://doi.org/10.5281/zenodo.1117372

[23] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[24] K. Ellis, E. Coviello, A. B. Chan, and G. Lanckriet, "A bag of systems representation for music auto-tagging," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 12, pp. 2554–2569, 2013.

[25] E. Coviello and G. Lanckriet, "Audio-based annotation of video," Jul. 25 2017, ,U.S. Patent 9,715,902.

[26] J. A. Bilmes *et al.*, "A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," *International Computer Science Institute*, vol. 4, no. 510, p. 126, 1998.

[27] P. Hamel, S. Lemieux, Y. Bengio, and D. Eck, "Temporal pooling and multiscale learning for automatic annotation and ranking of music audio." in *ISMIR*, 2011, pp. 729–734.

[28] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *2017 International joint conference on neural networks (IJCNN)*. IEEE, 2017, pp. 1578–1585.

[29] 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.

[30] H. Robbins and S. Monro, "A stochastic approximation method," *The annals of mathematical statistics*, pp. 400–407, 1951.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[32] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008. [Online]. Available: http://jmlr.org/papers/v9/vandermaaten08a.html