PODCASTMIX: A NOVEL DATASET FOR MUSIC SPEECH SEPARATION

Nicolás Schmidt
Universitat Pompeu Fabra
nicolas.schmidt01
@estudiant.upf.edu

ABSTRACT

Over the last few years, the popularity of podcast shows in streaming services has increased considerably. Licensed music in these shows is frequently used, but the accuracy of song identification services could affected by the speaker's voice in the mix. This represents a major problem both for the musicians, who do not receive their respective royalty payments, and for the broadcasters, who may be fined for non-compliance with international copyright laws. In this research report, a benchmark between two state of the art methods for music source separation is performed against a novel Podcast-like audio dataset called PodcastMix. This new dataset is compound by music from the Jamendo free music streaming service, mixed with a subset of the VCTK speech dataset. The benchmark is performed using the Asteroid toolkit and the evaluation metrics are computed using BSSEval tool in order to measure the quality of the separations.

1. INTRODUCTION

During the last few years, the production and consumption of podcasts has massively increased [1]. The emergence of Podcast as the new on-demand Radio Shows, brings with it technical and legal challenges typical of any broadcasting service. Among the most important issues is compliance with copyright regulations.

With the massification of technologies and new advances in the field of deep learning, new possibilities are opening up to solve this problem. One of the most widely exploited methods for automatic song recognition is the use of acoustic fingerprinting algorithms. This technique consist on the creation of an acoustic summary of a signal, computed from the audio features contained in the signal itself.

Fingerprinting algorithms should work well even in contexts where the song may have background noise, been modified in pitch or even in tempo. However, in podcast contexts, where the background music generally sounds fainter than the speaker's voice and where there are conversations going on at the same time, the effectiveness of song identification algorithms is affected. To allow better identification of the songs source separation techniques can be used.

In order to contribute to the development of new and better models based on supervised learning for source separation, a subset of the Podcastmix dataset is presented. This new dataset is compound by a mix between music licensed under creative commons and the VCTK dataset [5]. The music on the dataset is a subset of the most popular and featured songs from the creative commons music platform Jamendo.

The dataset is presented along with a benchmark of two state-of-the-art methods, the ConvTasNet and the DPTNet. The implementations of these models are based on the Pytorch toolkit Asteroid. The evaluation of the models was performed using the ps_bss_eval module, which allows the computation of the most commonly used metrics in the source separation field.

This document is structured as follows. First, the methodology of this research is presented. Later, the dataset and its most important characteristics are described. The next section presents the two source separation architectures that will be trained over the dataset. The results obtained by computing the BSSEval metrics are also presented. Finally, the conclusions are showed.

2. METHODOLOGY

This research seeks to present a new dataset and its subsequent comparison based on common evaluation metrics in the field of source separation. The source separation architectures used are ConvTasNet and DPTNet, both operating in waveform-domain. These architectures are still a benchmark for comparison with respect to accuracy in any research that seeks to perform the source separation task.

In a first instance, in this research the compilation of the new dataset is performed. This is generated from Jamendo's most popular songs, which are retrieved by using the API REST of the platform, along with a subset of 10 of the speakers of the VCTK dataset. Following, a Dataloader object is created in order to create on-the-fly mixes between the two sources and to provide the models the required data. The models are Pytorch-lightning implementations incorporated into the Asteroid opensource toolkit [4]. Asteroid provides the most relevant architectures as PyTorch-Lightning models in an easy-to-use way. This way researchers and developers can build, train, evaluate and contribute their own architecture implementations.

The metrics used to evaluate and compare the performance of these two architectures trained and evaluated with this novel dataset are the standard statistics used to evaluate source separation tasks. They consists of computing some measures between the estimated sources and the ground truth isolated sources. The used statistics are Source-to-Distortion Ratio (SDR), Source-to-Interference Ratio (SIR), and Source-to-Artifact Ratio (SAR), and they have become a standard in the source separation community [6].

3. DATASET

The dataset, the key piece of this report, is a reduced subset of the dataset that I intend to present as the subject of my thesis: PodcastMix. PodcastMix dataset is constituted by a mixture of two datasets. On one side is the well-known VCTK speech dataset, and on the other, a set of the most popular songs from the creative commons music application, Jamendo.

The VCTK dataset is a set of recordings of English speakers. The performance of 110 English speakers with their respective accents from various English-speaking regions was recorded. They were recorded reciting different phrases of an average length of about 3 seconds. The performances were recorded with two different microphones, the Sennheiser MKH 800 and the DPA 4035 and the recording was carried out in a semi-anechoic chamber located at the University of Edinburgh. The recordings were originally made at 96kHz, 24 bits, and then transformed to 48kHz, 16 bits.

A subset of the VCTK dataset was used for this work, due to its high weight and slow download on Colab Notebooks. The subset is constituted by only 10 speakers. The recordings were downsampled to 44.1kHz, and converted from FLAC to MP3. The reason for this re-sampling process is to have the same sampling frequency for both voice and music as inputs of the models to be trained. The compression to MP3 was performed in order to allow the model to be trained and evaluated faster due to Google Colab's restrictions.

Regarding the music dataset, a set of songs considered the most popular ones uploaded to the creative commons music application Jamendo were used. The reason for wanting to obtain the most popular songs was to be able to have music as similar as possible to music with commercial licences, which is frequently used in radio programs and podcasts. In this sense, the music dataset had to be high quality in both technical and musical terms.

To obtain the most popular songs, the Jamendo API was queried using the filters boost: popularity_total and featured: true. These filters, according to the documentation, allows to obtain the most popular songs of all time, within a subset of songs featured by the Jamendo team members. The concept of popularity for Jamendo, according to its documentation, considers various factors such as the number of downloads, plays, likes, among many others. On the other hand, the factor of a song being featured ensures that the Jamendo team curators have marked it as a song with high musical quality.

	Train	Validation	Test
Speech	5942	743	743
Music	1374	171	173

Table 1. Number of audio files in each one of the subsets of the PodcastMix.

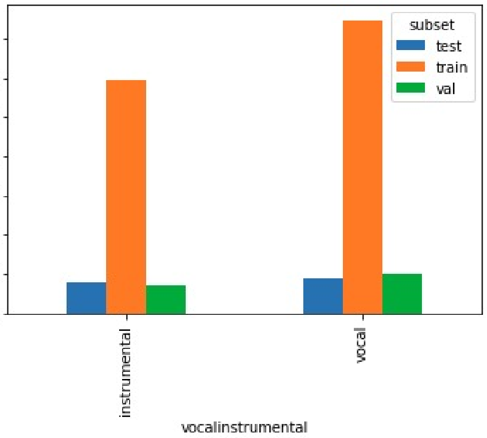

Figure 1. Distribution of instrumental and vocal tracks in the dataset.

Using the Jamendo API, the metadata of the songs was saved in a json dictionary, including licenses, upload date, artist, tags, album and URL for direct download in the selected quality. Using this information, a set of approximately the 1700 most popular Jamendo songs was downloaded.

With the music and speech datasets ready, the process of compiling the PodcastMix started. For this, a ratio of 80% train, 10% validation and 10% testing was used. Table N°2 shows the final number of files per set. Finally, using the metadata files from both the VCTK dataset and the information obtained from the Jamendo API, a dataset was created with the directory template shown in Figure N°1.

Some analysis were done over the speech and music subset of the dataset in order to verify that the training, validation and testing subsets has similar distributions. The results for the speech dataset are quite compelling in terms of age, gender and English accent distributions across the 3 subsets.

	Age		Gender		Accent	
Subset	Mean	SD	M	F	E	S
Train	24.18	4.77	31.2%	68.7%	90.3%	9.6%
Val	24.19	4.72	29.3%	70.6%	91.1%	8.9%
Test	24.05	4.63	31.9%	68.1%	89.6%	10.3%

In the case of the music, the analysis showed that there are more vocal tracks than instrumental ones in the dataset, as shown in Figure 1. Additionally, the dataset showed to have more than twice as much male voices than female (Figure 2), in the case of non-instrumental tracks. Finally, a curious insight is that most of the tracks are tagged as electric, almost three times the amount of acoustic tracks. However, the distributions between the subsets like similar which tell us that the train/val/test partition was done in a healthy manner.

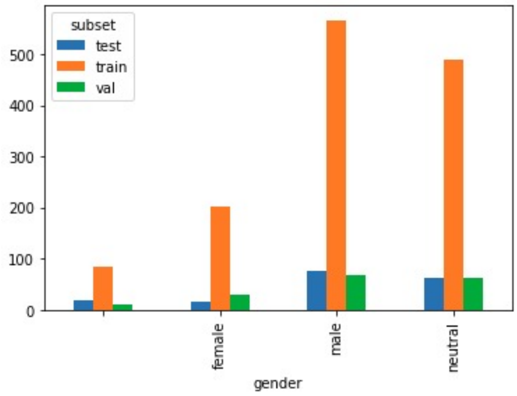

Figure 2. Gender distribution on vocal tracks

Finally, in order to load the data, a Dataloader was created. The main idea here is being able to create on-the-fly a large number of combinations between the music and speech audios. This was achieved by shuffling randomly the audios at the beginning of every epoch, and then combining one speech and one song files to create the podcast-like mix. The dataloaders also receives a segment parameter, which determines the length of the data that will be feed to the system. The portion of the audios that are used to create the podcast-like mix is also chosen randomly. Al the randomnes occurs for both training and validating stages, but the mixes for testing are always the same. The main functions of this dataloader are:

- The loading of the dataset data, for training, validation and testing in batches of a size determined by the user.
- To implement index-based iteration on the dataset.
- Allow a randomization of the dataset at each new training epoch of the system.
- Random selection of segments of a given number of seconds.
- Creation of the mix between background music and speaker's voice, based on the selected normalized audios, linearly summed using a gain factor for the music between 0 and 1.

$$X_{mix} = X_{speech} + G_{music} * X_{music}$$

4. SOURCE SEPARATION ARCHITECTURES

In this section the architectures trained with the presented dataset are presented. These are ConvTasNet and DPTNet.

4.1 ConvTasNet

ConvTasNet is a monaural source separation model designed primarily to separate voices and solve the cocktail party effect. It is an end-to-end model that works in the waveform domain, so it directly estimates isolated sources from masks. The main difference between ConvTasNet and TasNet is that in its predecessor the mixture waveform was modeled by using of a convolutional encoder/decoder with a LSTM network, which consisted of an encoder with non-negativity constraints with respect to its output. At the same time, it used a linear decoder to reverse the encoded signal to reconstitute the waveform. However, the use of the LSTM network in the separation module severely limited the potential applications.

ConvTasNet uses exclusively convolutional layers in all the stages of the input signal processing. Thus, the ConvTasNet architecture consists of 3 fundamental layers:

- Convolutional Encoder: the mixed signal is divided into segments of the same size, which are then transformed to an N-dimensional representation using a 1-D convolution. C Vectors, one for each of the speakers, are then estimated. These are the masks of each of the independent sources and are constrained in their non-negativity. The waveforms of each source are then reconstructed by the decoder. The constraint that all masks must sum to one is added so that the model learns that the architecture of encoders and decoders must perfectly reconstruct the mixed signal.
- Separation: This module is based on a fully-convolutional separation module, which is inspired by a Temporal Convolutional Network (TCN). The implementation of the TCN in the ConvTasNet consists of a stacked 1-D dilated convolutional blocks with exponentially increasing dilatation factors. This is done in order to add large enough temporal context. The output of the TCN is passed then to a convolutional block.
- Decoder: In the decoder, a linear block is added as a bottleneck layer. This block determines the number of channels in the input and residual path of the subsequent convolutional blocks. Finally, the masked encoder features are used to estimated the isolated sources waveforms. This is made using a linear decoder [3].

4.2 DPTNet

The dual-path transformer network is an architecture that incorporates an end-to-end architecture using an improved dual-path transformer to allow the system to learn from the audio context. This makes the model efficient in separating long audio segments.

This model is conceptually based on the ConvTasNet, as its structure follows the same 3-layer approach: encoder, separation layer and decoder. The encoder layer is used to convert the audio segments of the mix into a feature map, which is then input to the separation layer for mask creation. Finally the decoder layer reconstructs the wave signal of each of the isolated source estimations.

The encoder is mainly constituted by a series 1-D convolutions modules, which acts as a filter-bank W of N filters of length L. Thus, for a mixture of speakers $x \in R^{1xT}$, it is subdivided into L-length overlapping vectors $X \in$

R^{LxI}, where I is the number of vectors. Finally the speech signal $X \in R^{NxI}$:

$$X = ReLU(X * W)$$

The separation layer, in turn, consists of three layers: segmentation, dual-path transformer processing and overlap-add. In the segmentation layer X is segmented into overlapping chunks and then all the chunks are concatenated in a 3-D tensor. The dual-path transformer processing also consists of three layers:

- scaled dot-product attention: effective self-attention mechanism for associating different positions of the input and calculating their representations.
- multi-head attention: composed of multiple scaled dot-product attention modules.
- Position-wise feed-forward network: fully connected neural network with two linear transformations and a ReLU activation function between them.

In the third stage of the separation layer, overlap-add, the output of the dual-path transformer layer is used to learn a mask for each of the sources, by using a 2-D convolutional layer. Finally, a transposed convolution module is used in the decoder to reconstruct each signal separately [2].

5. EVALUATION METRICS

To evaluate and compare the performance of the two architectures trained on the PodcastMix dataset, the metrics included in the BSSEval module were used. BSSEval is included within Asteroid and allows the calculation of a number of source separation statistics. The statistics used were Source-to-Distortion Ratio (SDR), Source-to-Interference Ratio (SIR), and Source-to-Artifact Ratio (SAR), and they have become a standard in the source separation community.

Given an estimated source s_i, it can be described as the sum of three elements:

$$s^i = s_{target} + e_{interf} + e_{noise} + e_{artif}$$

where s_{target} is the ground truth isolated source, and e_{interf}, e_{noise}, and e_{artif} are error terms for interference, noise, and added artifacts, respectively. This elements the different statistics can be computed [6]:

- Source-to-Artifact Ratio is interpreted as the amount of unwanted artifacts an estimated source has with relation to the ground true isolated source. It is computed as:

$$SAR = 10 \log_{10} \frac{|s_{target} + e_{interf} + e_{noise}|^2}{|e_{artif}|^2}$$

- Source-to-Interference Ratio is interpreted as the amount of other sources that can be heard in a source estimate. It is computed as:

$$SIR = 10 \log_{10} \frac{|s_{target}|^2}{|e_{interf}|^2}$$

params	ConvTasNet	DPTNet
n_filters	512	64
kernel_size	16	16
stride	8	8
n_blocks	8	N/A
n_repeats	3	2
mask_act	relu	sigmoid
bn_chan	128	N/A
skip_chan	128	N/A
hid_chan	256	N/A
in_chan	N/A	64
out_chan	N/A	64
ff_hid	N/A	256
ff_activation	N/A	relu
norm_type	N/A	gLN
chunk_size	N/A	100
hop_size	N/A	50
bidirectional	N/A	true
sample_rate	16kHz	16kHz
segment	2	2
batch_size	1	1
num_workers	24	24
epochs	50	40

Table 2. parameters used for the ConvTasNet and DPTNet

	si_sdr	sdr	sir	sar	stoi
ConvTasNet	5.9	6.6	10.4	11.9	0.7
DPTNet	0.42	1.81	4.44	13.4	0.66

- Source-to-Distortion Ratio is an overall measure of how good a source sounds. It is usually the most important statistic reported in papers.

$$SDR = 10 \log_{10} \frac{|s_{target}|^2}{|e_{interf} + e_{noise} + e_{artif}|^2}$$

6. RESULTS

After training both models using the PodcastMix dataset, an evaluation was performed using the BSSEval metrics. The summary of the evaluations can be seen in Table 3

The models were trained using the parameters showed in Table 2

As we can see, the DPTNet architecture has a very poor performance. This could be caused by the reduced sample rate used in the experiments. Another reason that could have caused this poor performance is the 2-seconds segment used for both training and evaluation, which are too short to make the dual-path actually learns more features from the context. In the other hand, the ConvTasNet had a great performance comparing with the DPTNet. In fact, it learned really well how to separate both sources even considering the low sample rate used.

As a point of comparison, we know that the performance of Conv-TasNet-gLN on the WSJ0-2mix and LS-2mix datasets with respect to the SDR statistic is 15.6 and 13.5 respectively. On the other hand, DPTNet has values for the SDR of 20.6 and 16.8 respectively. According to these results, we can interpret that the ConvTasNet performs really well even in context of low sample rate. Also, it important to review the mixing strategy between the speech and music, since all the training examples are different even across epochs, there is a possibility that the gradients could be exploding and thus the model is obtaining oscillating loss values.

7. CONCLUSIONS

Throughout this report we have presented the methodology upon which we present to the music information retrieval community a new dataset to perform source separation tasks between background music and foreground speech: the PodcastMix dataset. This dataset presents an opportunity to continue with the implementation of models that perform source separation tasks oriented to radio shows and podcasts.

The dataset was tested and compared under implementations of the ConvTasNet and DPTNet source separation architectures. These models currently are part of the state of the art in terms of source separation in the waveform domain. Separation evaluation values comparable to the same task performed on other known datasets were obtained. In particular, SDRs of 6.6 for ConvTasNet and 1.81 for DPTNet were obtained.

We plan to extend this dataset in the future by improving the format and sampling frequency of the files, together with the preservation of more complete musical metadata, obtained from the Jamendo platform. This work will be carried out in the framework of my Master Thesis.

8. REFERENCES

[1] The infinite dial 2020. `http://www.edisonresearch.com/wp-content/uploads/2020/03/The-Infinite-Dial-2020-U.S.-Edison-Research.pdf`, 2020. Accessed 03-15-2021.

[2] Jingjing Chen, Qirong Mao, and Dong Liu. Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation, 2020.

[3] Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(8):1256–1266, Aug 2019.

[4] Manuel Pariente, Samuele Cornell, Joris Cosentino, Sunit Sivasankaran, Efthymios Tzinis, Jens Heitkaemper, Michel Olvera, Fabian-Robert Stöter, Mathieu Hu, Juan M. Martín-Doñas, David Ditter, Ariel Frank, Antoine Deleforge, and Emmanuel Vincent. Asteroid: the PyTorch-based audio source separation toolkit for researchers. In *Proc. Interspeech*, 2020.

[5] C. Veaux, J. Yamagishi, and Kirsten Macdonald. Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit. 2017.

[6] E. Vincent, R. Gribonval, and C. Fevotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1462–1469, 2006.