CSCI 566 - Project Proposal Template

Theodore Ho
teho@usc.edu

Shuye Huang
shuye.huang@usc.edu

Dongwook Kim
dkim2378@usc.edu

Abstract

Musical Source Separation(MSS) is the process of separating stems from a track. For example, given a music waveform with multiple instruments/audio sources, MSS ouputs just the waveform for one source such as the tracks vocals. MSS has been actively researched for decades with numerous models and different target sources. However, training data for MSS is difficult to obatain as it requires tracks with pre-separated sources. Many of the highest preforming models only train on the MUSDB-18 dataset which only has 150 tracks. We aim to explore data generation and augmentation techniques to create a larger training dataset and investigate the preformance impact on existing models.

1 What is the problem?

Listening to music has become a daily basis lifestyle since the invention of the internet as it provided easy access to a wide variety of music libraries. Alongside the practice of remixing came, where one would mix different tracks of other songs to produce a new song. This is quite straightforward, given that one has access to individual instrumental tracks of the said songs. However, it is more often the case that songs are published as mixed single-track versions hence the attempts to separate music sources from the singular tracks have been made and are called Musical Source Separation(MSS). Key challenges in MSS root from the fact that the final recording mixes not only contain various instruments, but are additionally filtered and non-linearly mastered. Additionally, in stereo mixes, the sources may move or alternate between channels thus making MSS harder. Solving MSS can benefit a wide range of individuals, from content creators such as DJs and samplers, to listeners with an interest to listen to instrumental (vocal-removed) versions of their favorite songs.

2 How is it currently approached?

Deep learning approach for MSS has been developed rapidly in recent years, mainly applied on temporal domain (i.e., waveform) or temporal-frequency domain (i.e., spectrogram). Multiple architectures have been proposed, for instance, waveform U-Net (Défossez et al. [2019]), domain densely connected dilated CNN on spectrogram (Takahashi and Mitsufuji [2020]), spectrogram RNN (Luo and Yu [2023]), local-attention-based dual U-Net on hybrid input (Défossez [2021]), etc. The growth of architecture's sophistication results in requirement of larger dataset for model training.

Unfortunately, limited open dataset of source separated tracks are available, in which case data augmentation is of importance. Methods to enrich an audio dataset include temporal or frequency

Preprint. Under review.

domain masking or warping (Park et al. [2019]), pitch-shifting and spectral transformation (Cohen-Hadria et al. [2019]), remixing channels in batch (Défossez et al. [2019]). It is noticeable that these approaches may not generate realistic music segments, with potential caveats such as beat miss alignment, tempo distortion and break down of melodic structures, leading to possible worse model training. Some previous work has discussed this issue (e.g., Chiu et al. [2021], Défossez [2021]). We here want to systematically evaluate the goodness of these augmentation strategies and their potential implications for different model architectures.

Performance is measured using SDR (signal distortion ratio) between the true source and the models predicted source. To standardize testing of model performance, researchers average the SDR score for all test examples in the MUSDB-18 dataset to compare performance between models.

3 Approach

We will employ similar methods of data generation referenced in the original Demucs paper; mixing stems of different sources to generate new training samples. However, unlike the original Demucs paper we will also use sources from outside the MUSDB-18 by mixing vocals from open-source datasets such as Acapella and open-source instrumental tracks either from a public dataset or scraped using Spotify's API. This is a similar approach to data scraping technique described in the original U-Net paper for music source separation by Spotify. Only instead of scraping pairing vocal and instrumental covers from the same track, we will pair them randomly. We also will experiment with data augmentation techniques. Such as applying a random chromatic shift or BPM change using the Librosa Python library.

- We will select a set of candidate models for music source separation to test our generated dataset. We will select models based on compute requirements, the dataset it was originally trained on (we will only select models that were only trained on the MUSDB-18 dataset), documentation/support for training, and performance (SDR score). Potential models include HDemucs V3, Band-Spit RNN, KUIELAB-MDX-NET.
- The performance of music source separation models is evaluated using the SDR score between ground truth separation and predicted separations. In the literature, all models average this score for all tracks in the MUSDB-18 test set to compare model performance. We will use the same method for measuring performance. We will train two instances of the model. One on the MUSDB-18 dataset that it was originally trained on, and one with our new dataset. We will compare the results. As referenced in the Demucs paper, researchers were able to increase SDR score for vocal separation from 6.84 to 7.29 when trained using their data generation techniques. We will attempt to create a similar or better increase in performance. We will either train the models from scratch, or ontop of the pretrained weights.
- The biggest potential obstacle is compute time. We will need to ensure we select a model that we can train sufficiently with our current compute resources. Another obstacle is ensuring we are training the models correctly and are able to replicate the results in the original papers.

TIMELINE:

Week 5, Sept 24: Begin researching candidate models and data sources (1 week)

Week 6, Oct 1: Implement selected candidate models and set up training infrastructure/pipeline. (1 week)

Week 7 Oct 8: Test implementation by training on MUS-DB18 and verifying we achieve similar performance described in model documentation. (1 week)

Week 8 Oct 15: Precisely define the contents and techniques of our generated dataset(s). As well as our training and benchmark process (1 week)

Week 9 Oct 22: [Break to allow midterm studying]

Week 10 Oct 29: Begin creating dataset(s) (1 week)

Week 11 Nov 12: Training of models using generated dataset. (1 week)

Week 12 - Nov 19: Write-up and buffer-time in-case we need more time. (2 week)

References

C.-Y. Chiu, J. Ching, W.-Y. Hsiao, Y.-H. Chen, A. W.-Y. Su, and Y.-H. Yang. Source separation-based data augmentation for improved joint beat and downbeat tracking. In *2021 29th European Signal Processing Conference (EUSIPCO)*, pages 391–395. IEEE, 2021.

A. Cohen-Hadria, A. Roebel, and G. Peeters. Improving singing voice separation using deep u-net and wave-u-net with data augmentation. In *2019 27th European signal processing conference (EUSIPCO)*, pages 1–5. IEEE, 2019.

A. Défossez. Hybrid spectrogram and waveform source separation. *arXiv preprint arXiv:2111.03600*, 2021.

A. Défossez, N. Usunier, L. Bottou, and F. Bach. Music source separation in the waveform domain. *arXiv preprint arXiv:1911.13254*, 2019.

Y. Luo and J. Yu. Music source separation with band-split rnn. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:1893–1901, 2023.

D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

N. Takahashi and Y. Mitsufuji. D3net: Densely connected multidilated densenet for music source separation. *arXiv preprint arXiv:2010.01733*, 2020.