Audio Separation Using Deep Neural Networks
ECE 685 Fall 2020

Jiajun Song
jiajun.song@duke.edu

Heather Qiu
zhi.qiu731@duke.edu

Abstract

Source separation has always been an important topic in audio processing. Apart from the traditional methods such as ICA, some nonlinear techniques such as deep-learning-based have been proposed and proved to be effective in audio sources separation from the mixed signal. In this project, we test the effectiveness of both a traditional linear method (ICA) and deep learning nonlinear methods (Open-unmix and Demucs) on a benchmark data set (MUSDB18) with the reference implementation Open-Unmix.

1 Introduction

Throughout the years, audio separation has become an important topic in the filed of audio signal processing. As stated in [1], "this problem has been extended through other disciplines, such as image processing, digital communications, medical imagining, among many others, with similar goals", making it a very practical and significant topic worth exploring. Among all the sub-fields, this paper will mainly focus on source separation or more technically called as the Blind Audio Source Separation (BASS), which basically means separating different source signals from a set of mixed signals. Many original deep learning solutions for this problem are proposed and developed in recent years, but not all of them have been well tested for effectiveness [12]. Therefore, the main interest of this paper it to explore the effectiveness of some popular deep learning models (e.g., Open-unmix [11] and Demucs [3]) and compare their performance with some of the traditional methods (e.g., ICA).

As such, the paper is organized as the following: Firstly we would give a short review of the theoretical background of ICA, Open-unmix and Demucs in Section II. Then we introduce project details in Section III about data collection (MUSDB18), pre-processing and application of these algorithms to our prepared data set with Open-Unmix, which is a deep neural network reference implementation for music source separation [11]. Next, metric selection and model evaluation are included in Section IV. Lastly, conclusion, future work as well as possible improvement would be discussed in Section V.

2 Related Work

The idea of audio source separation was first proposed in [2], where Cherry formalized the problem as the "cocktail party effect". As indicated by the name, in a situation of cocktail party people are continuously chatting everywhere in a room and human brain is smart enough to distinguish a single conversation from all the other surrounding noise. This raised researchers' interest in mimicking this effect and extend its application to the general audio source separation problem. Correspondingly, many research has been done since then and multiple statistical methods such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), and Non-negative matrix factorization (NMF) have been proved effective in many cases for audio source separation before the start of the 21st century.

2.1 Linear - ICA

The mathematical formulation of ICA is shown as below [5]:

$$\boldsymbol{x}^{(i)} = \boldsymbol{A}\boldsymbol{s}^{(i)} \tag{1}$$

where $\boldsymbol{x}^{(i)}$ denotes the observed signals of one mixed data sample $\boldsymbol{x}$ at time i, $\boldsymbol{s}^{(i)} \in \mathcal{R}^n$ represents the hidden components (the original sources) at time i with an independent assumption. $\boldsymbol{A}$ is an unknown square matrix called the mixing matrix.

Then observed signal j at time i is given by

$$x_j^{(i)} = \sum_{k=1}^{n} a_{jk} s_k^{(i)} \tag{2}$$

where n is the number of observed signals, a_{jk} is the j,k-th element of the mixing matrix $\boldsymbol{A}$, and $s_k^{(i)}$ represents source k at time i. As we can see, this is a linear representation and we can multiply the observed signals $\boldsymbol{x}$ by the demixing matrix $\boldsymbol{W} = \boldsymbol{A}^{-1}$ to recover the original source signals $\boldsymbol{s}$ that we wish to isolate. The $\boldsymbol{w}$ vectors in $\boldsymbol{W}$ are adaptively calculated by using a cost function which either maximizes the statistical independence of the source signals or minimizes the mutual information [5].

In summary, ICA algorithm is based on a linear combination of statistically independent sources and tries to maximize the independence of the output signals [7]. In contrast to PCA where signals are decorrelated in transformation, ICA conducted more reduction in higher order statistical dependencies and is very useful for high-dimensional data sets. However, such linearly independent assumption is quite restrictive and rarely holds in reality. Therefore, the basic linear model is usually too simple to fully represent the true data.

Here comes one obvious draw back of these linear transformations or algorithms. They mostly focus on “a distinct variational problem" and used different model constraints to over-simplify the modeling problem [1]. Such constraints undermine these algorithms’ generalization ability and make them quite sensitive to noise in data.

2.2 Nonlinear - Open-unmix and Demucs

In order to relax those constraints in linear based algorithms, nonlinear deep learning methods have been proposed in recent years. We select Open-unmix[11] and Demucs[3] to show how nolinear deep learning methods bypass these constraints and have already achieved advanced progress.

Most deep-learning-based methods in audio source separation can be categorized into the spectrogram-domain-based or waveform-domain-based for transformation of data. Regarding the spectrogram-based models, the features are firstly generated by the Short-Time Fourier Transform (STFT). Then the downstream models are generally related to Convolution Neural Networks (CNN) Layer and Long Short-Term Memory (LSTM) layer because these two layers are especially suitable to tackle problems with sequences in high dimensions.

For example, in the case of Open-unmix [11], the model architecture is based on a three-layer bidirectional LSTM and the details are shown in 1. We choose Open-unmix as our reference implementation and are going to discuss over more details in later sections.

As for the waveform-based models, they directly operate on the raw input waveform and output the waveform for each separated source [8]. Although the waveform-based models do not require transformation on the inputs, their model architecture is typically more well-designed to fit different datasets, in other words more advanced techniques such as skip-connection and U-net architecture are applied. For example, Demucs is based on U-Net convolutional architecture inspired by Wave-U-Net [10] and SING [4], with GLUs, a bidirectional LSTM between the encoder and decoder, specific initialization of weights and transposed convolutions in the decoder.

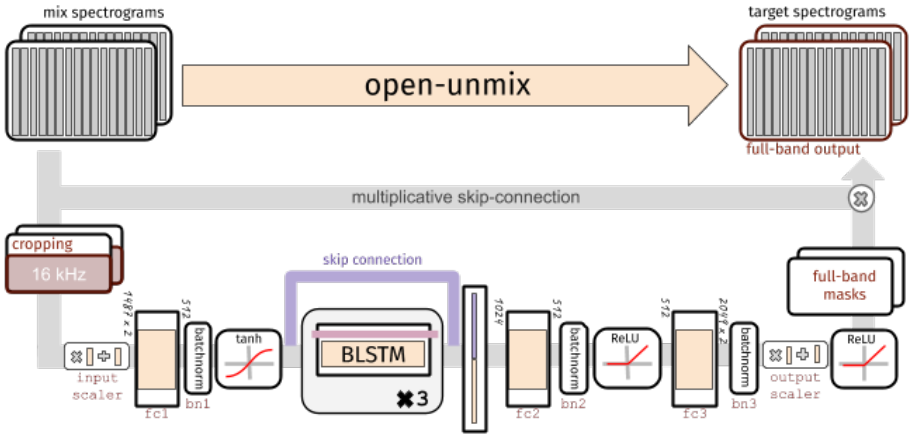

Figure 1: Network architecture of Open-Unmix based on a three-layer bidirectional deep LSTM. The model learns to predict the magnitude spectrogram of a target, like vocals, from the magnitude spectrogram of a mixture input. (Image source: https://github.com/sigsep/open-unmix-pytorch)

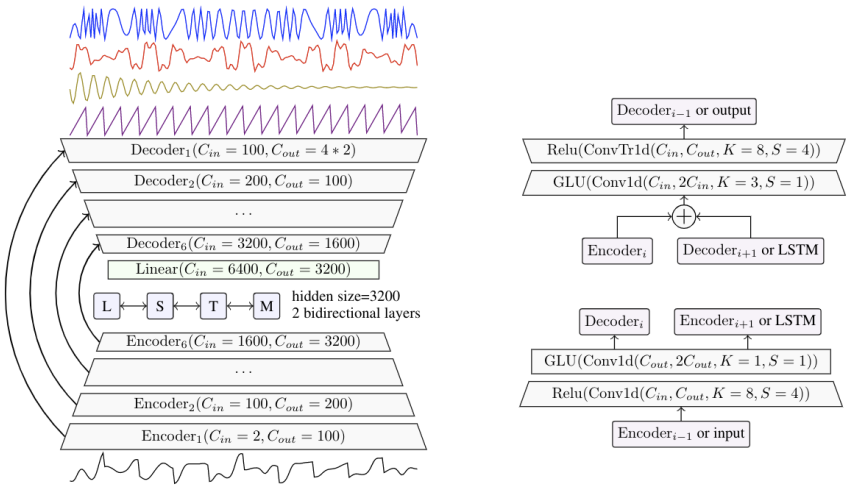

Figure 2: Demucs architecture with the mixture waveform as input and the four sources estimates as output. Arrows represents U-Net connections. (Image source:https://github.com/facebookresearch/demucs)

3 Dataset and Methods

3.1 Dataset

In this paper, models will be evaluated and compared on the benchmark MUSDB18 data set as proposed in [9]. This data set is especially tailored for audio source separation, in which the music audio is recorded as a mixture of several individual instruments such as drum, bass, vocals, and other stems. Our audio source separation objective for MUSDB18 data set is to isolate the vocal source from other accompaniments and to identify the stems further.

Overall, this data set contains 150 full-length music tracks with a total duration of around 10 hours along with the genres. Note that the scenario in MUSDB18 is more challenging than the original "cocktail party problem". According to [10], "there is not a single source of interest to differentiate from an unrelated background noise, but instead a wide variety of tones and timbres playing in a coordinated way". Accordingly, the more coordinated the audio signal behaves, the more difficult the problem is for an algorithm to isolate different sources.

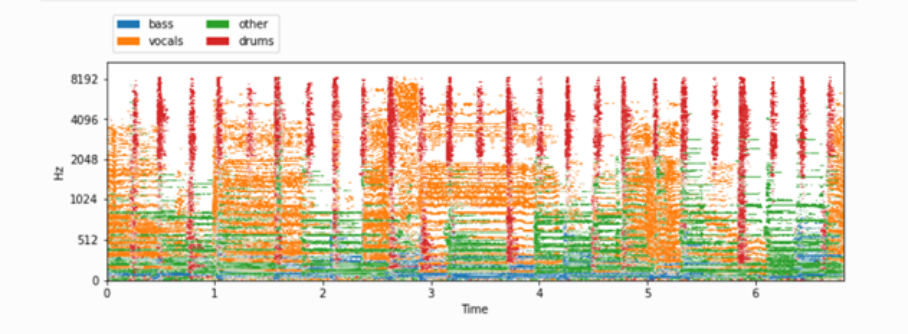

Figure 3: Spectrogram of four sources: drums, bass, vocals, and others in a randomly picked track sample in MUSDB18. (Image source: https://nussl.github.io/docs/tutorials, Data source: Rafii, 2017)

3.2 Pre-processing

Multiple data pre-processing transformations on MUSDB18 are applied to enhance the robustness of the training dataset and function as data augmentation as well. These data pre-processing comes as defaults in the open source implementation of Open-unmix [11] and are tuned to boost the performance.

- **Chunking**: instead of processing the full audio tracks into open-unmix, the dataloader chunk the audio into 6s excerpts. The start time is also randomized as a part of the data augmentation.
- **Source Augmentation**: random gains between 0.25 and 1.25 are applied to all sources before mixing. Furthermore, to fulfill the data augmentation the channels the input mixture are swapped.
- **Random Track Mixing**: for a given target a random track is selected with replacement. To yield a mixture the interfering sources are drawed from different tracks with replacement to increase generalization of the model.
- **Fixed Validation Split**: tracks is validated in full length instead of using chunking to have evaluation as close as possible to the actual test data.

3.3 Methods

As shown in Figure 1, Open-Unmix operates in the time-frequency domain to perform its prediction. The model learns to predict the magnitude spectrogram of a target, like vocals, from the magnitude spectrogram of a mixture input. Due to the recurrent nature of LSTM, the model can be trained and evaluated on arbitrary length of audio signals. Since the model takes information from past and future simultaneously, the model can capture temporal information from both directions but cannot be used in an online fashion. Internally, the prediction is obtained by applying a mask on the input. The model is optimized in the time-frequency domain using mean squared error and the actual separation is done in a post-processing step involving a multichannel wiener filter [6] implemented using norbert. To perform separation into multiple sources, multiple models are trained for each particular target. While this makes the training less comfortable, it allows great flexibility to customize the training data for each target source.

4 Experimental Results

4.1 Metric Selection

According to Vincent (2006), a robust performance measure and ubiquitous test grounds have been missing for a long time in the research of blind audio source separation, even in the easiest case of linear source mixtures without phase delay. Considering such a lack of evaluation, it is quite difficult to compare individually experimented models for source separation.

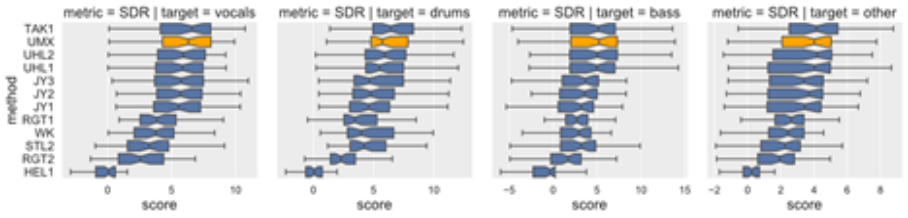

Figure 4: Comparisons of multiple methods on SDR of four target isolation tasks: vocals, drums, bass, and others. Violin plots are shown to demonstrate the distribution of evaluation metrics. Open-unmix (UMX) is highlighted in yellow. (Image source: https://github.com/sigsep/open-unmix-pytorch)

Luckily, recently several performance measures are proposed to be quite appropriate for evaluation of audio source separation performance in various scenarios. Most of these evaluation metrics are proposed and summarized in Vincent, 2006. Source-to-distortion ratio (SDR), Source-to-interface ratio (SIR), Source-to-artifact ratio (SAR), and Source-to-Spatial-Distortion Image (ISR) are some of the evaluation metrics proposed by Vincent for blind source separation. The main foucus of SDR is alignment between the estimates and ground truth, while SIR measures the degree to which the estimates discriminate against other confounding sources. SAR is especially useful in music source separation as it detects the amount of random noise and music in the estimates. Furthermore, ISR aims at the spatial distribution and checks if the algorithm retrieves the audio signal by keeping it in the same spatial location as the ground truth source. There are many other more robust versions of SDR, SIR, SAR, and ISR proposed and developed in other research, but the main idea stays the same.

In addition to these metrics, traditional metrics in classification such as precision or recall can also be applied in blind source separation. Precision and recall scores could be calculated on the binary masks, and the corresponding results will evaluate the difference between the actual mask and estimated mask in terms of the time-frequency bin. Nonetheless, one disadvantage of this method is that a specific threshold must be chosen beforehand so as to obtain the binary mask, and there is little guidance on the selection of such a threshold.

4.2 Model Evaluation

Figure 4 shows the comparison of different neural methods including Open-Unmix (UMX) based on Source-to-distortion (SDR) of four target isolation: vocals, drums, bass, and other. Violin plots are shown to demonstrate the distribution of evaluation metrics. Open-unmix (UMX) is highlighted yellow in the figure. In general, UMX has achieved comparable state-of-the-art results in terms of SDR, with the median SDR of UMX ranking second in two out of four targets isolation tasks. There is an asymptotic tradeoff between variance and bias in these methods. The higher rank methods such as TAK1 and UMX perform well in bias but suffer a little from large variance while methods such as RGT1 and HEL1 come with higher efficiency but their bias is large. Note that although SDR gives straightforward comparisons between multiple methods, there is no sufficient evidence to say that one model is better than the others. When evaluating algorithms in practice one should always listen to the separated results in addition to the evaluation metrics report.

5 Conclusion and Future Work

Historic methods and more recent deep-learning-based algorithms are introduced and discussed in this paper. The dataset MUSDB18 is described in detail as the benchmark for comparisons between models. Evaluation metrics including SDR, Precision, and Recall are discussed. The performance of multiple models including Open-unmix is compared on the MUSDB18 dataset in terms of SDR. However, we note that the performance of Open-unmix decrease greatly as the length of audio gets longer. Future work will foucs on how to address such issue and more research on the waveform-based deep neural network models.

6 Contributions

Through a set of different tasks, both members have made efforts and contributions to the project. Jiajun contributed to literature review, model research, dataset preparation, all models' training and evaluation. Heather contributed to literature review, model research, dataset research and ICA model exploration. Both members contributed to write-up and slides preparation. This paper could be reproduced via codes in the **project's Github Repository** (`https://www.github.com/JiajunSong629/Audio_Separation`).

References

[1] P Chandna. *Audio source separation using deep neural networks*. PhD thesis, Master Thesis, Universitat Pompeu Fabra, 2016.

[2] E Colin Cherry. Some experiments on the recognition of speech, with one and with two ears. *The Journal of the acoustical society of America*, 25(5):975–979, 1953.

[3] Alexandre Défossez, Nicolas Usunier, Léon Bottou, and Francis Bach. Demucs: Deep extractor for music sources with extra unlabeled data remixed. *arXiv preprint arXiv:1909.01174*, 2019.

[4] Alexandre Défossez, Neil Zeghidour, Nicolas Usunier, Léon Bottou, and Francis Bach. Sing: Symbol-to-instrument neural generator. In *Advances in Neural Information Processing Systems*, pages 9041–9051, 2018.

[5] A. Favaro, A. Lewis, and G. Schlesinger. Ica for musical signal separation. 2011.

[6] Peter Hoeher, Stefan Kaiser, and Patrick Robertson. Two-dimensional pilot-symbol-aided channel estimation by wiener filtering. In *1997 IEEE international conference on acoustics, speech, and signal processing*, volume 3, pages 1845–1848. IEEE, 1997.

[7] Te-Won Lee. Nonlinear approaches to independent component analysis. 2000.

[8] Francesc Lluís, Jordi Pons, and Xavier Serra. End-to-end music source separation: is it possible in the waveform domain? *arXiv preprint arXiv:1810.12187*, 2018.

[9] Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, Derry FitzGerald, and Bryan Pardo. An overview of lead and accompaniment separation in music. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(8):1307–1335, 2018.

[10] Daniel Stoller, Sebastian Ewert, and Simon Dixon. Wave-u-net: A multi-scale neural network for end-to-end audio source separation. *arXiv preprint arXiv:1806.03185*, 2018.

[11] Fabian-Robert Stöter, Stefan Uhlich, Antoine Liutkus, and Yuki Mitsufuji. Open-unmix - a reference implementation for music source separation. *Journal of Open Source Software*, 4(41):1667, 2019.

[12] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte. Performance measurement in blind audio source separation. *IEEE transactions on audio, speech, and language processing*, 14(4):1462–1469, 2006.