VOCAL TRACK EXTRACTION USING NEURAL NETWORKS

Chenyu Xi[1]

***Abstract*— Deep neural network has been widely used in recognition and classification tasks. In this paper, we discuss the possibility to implement neural networks for vocal extraction tasks. In our report, we test neural networks such as deep clustering model based on RNN structure and U-net model based on CNN structure for vocal track extraction task, and we propose a new model which takes pretrained RNN model as reference to improve the model performance of the U-net model.**

I. INTRODUCTION

Vocal track extraction aims to separate vocal track from audio files, which is a sub-task of Music Information Retrieval (MIR). Vocal track extraction has been largely used in many fields, such as singer identication or lyrics transcription. People have been putting much effort on this problem and there are many meaningful solutions for the task.

One method for vocal track extraction is implement same strategies used for semantic segmentation, which describes the process of associating each pixel of an image with a class label. Common models for semantic segmentation have been transferred for vocal extraction usage, one famous example is U-net model. This method generated vocal track mask directly from audio feature map.

Another method is applying deep clustering model. Instead of generating vocal track mask, deep clustering model outputs embedding vector of T-F bins in mel-spectrogram. On the second stage, unsupervised method, such as k-means, is implemented to separate vocal T-F bins and background T-F bins.

In our project, we firstly implement models addressed in [1] and [2], then we try to design a hybrid model and compare the model performance with traditional models.

II. RELATED WORK

A. Convolutional Neural Network and Semantic Segmentation

Convolutional neural network(CNN) was firstly used as image feature extractor, and has been explored for other feature recognition tasks, such as semantic parsing and audio feature extraction. CNN extract hierarchical features of input signal with convolutional kernels.

The first attempt for applying DNN in semantic segmentation tasks is Fully Convolutional Networks (FCN). FCN extracts the features of input figures using traditional CNN networks such as VGG16. In order to recover the segmentation result from the small size feature map, FCN implements upsampling on the feature map output to gain segmentation mask with the same size as input, and thus convert the segmentation problem into a pixel-scale classification problem. In order to maintain more information during upsampling stage, FCN also combines the feature map with pool4 or pool3 layer output[3]. FCN is the first model structure that achieves the end-to-end segmentation; However, this model is not a good candidate for music source separation as it has a bad performance in keeping details near the separation boundary.

U-net is a famous segmentation model based on FCN models, and mainly used for bio-medical image segmentation. One improvement of U-net compared with FCN networks is increasing the channel number of upsampling convolutional layers in order to propagate context information to high resolution layers, and it also assign higher loss weights for boundary pixels to make the mask more precise. U-net model is proved to have good performance in small dataset.

B. Recurrent Neural Network

Recurrent neural network(RNN) is designed to solve sequential input problems. RNN models are frequently used in the field of natural language processing and audio signal modeling. Gated recurrent unit(GRU) is a gating mechanism in RNN, which is similar with LSTM while has fewer parameters by eliminating the output gate.

Deep clustering models use four-layer Bi-LSTM as the feature extractor and embedding the input mel-spectrogram with high dimension vectors. In [2], deep clustering model is implemented along with conventional network to improve the model performance. In our works, we reproduce this model while replace the Bi-LSTM layers with Bi-GRU layers to simplify the training stage.

III. DATA PREPOSSESSING

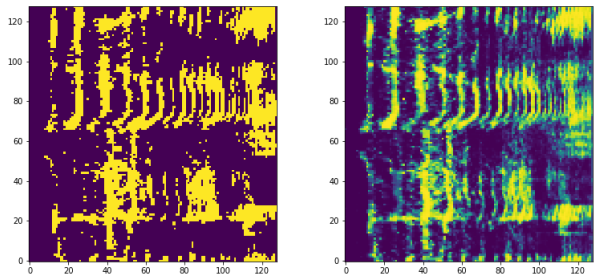
A. DSD100 Dataset

DSD100 is a famous dataset used for MIR tasks. This dataset contains 100 full lengths music tracks along with isolated tracks. In order to enrich our training data, besides keeping original music tracks, we also mix vocal tracks in DSD100 dataset with other instrumental tracks randomly to create more audio files.

B. Multiframe

Multiframe strategy[4] is adopted to augment the training data and speed up the training process. We firstly cut off the silence part in vocal track as well as the corresponding part in mixed music track. Then we convert mixed music tracks and ground truth voice tracks into log scale mel-spectrogram with 128 mel bands and 16000 sample rate, and divide the

[1]cx2219@columbia.edu, Department of electrical engineering, Columbia University



(a) binary mask (b) proportion mask

Fig. 1. Mask for vocal track

original spectrogram into smaller chunks to create 128×128 feature maps. Each chunk stands for approximately one second length audio sequence. Notice that the x-axis in mask denotes features and the y-axis in mask denotes the time as all spectrogram are transposed.

To extract vocal track out of music files, we convert the log scale mel-spectrogram into power scale and multiply the spectrogram with a filter mask generated by models, and then convert the vocal track spectrogram back into audio files. For each music-vocal chunk pair, we design two kinds of training target mask. The first one is binary vocal mask, which is generate by comparing power between vocal spectrogram and background spectrogram, we marked T-F bins either belong to vocal source or background source. This mask is used for training clustering models. Another target mask is proportion mask, which stands for how much vocal source dominated for each T-F bins in the mel-spectrogram, and we use this proportion mask when training U-net model and hybrid U-net model. Sample masks are shown in Fig 1.

C. Vocal Track Recovering

As there is no phase information kept when we process audio data, we use the Griffin Lim algorithm[5] to recover the vocal track from mel-spectrogram. Griffin Lim algorithm simulate the phase information during iterations, and we use that for audio reconstruction.

IV. MODEL DESCRIPTION

A. Deep Clustering Model

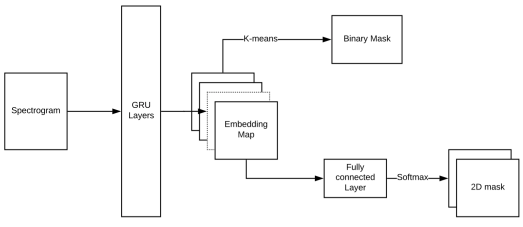


Fig. 2. The hybrid clustering model

The architecture of deep clustering model is similar with works in [2]. We reduce the input mel-spectrogram size from 150×150 to 128×128 and replace the LSTM layer with GRU layer. The four-layer Bi-GRU assigned D-dimensional feature vector to each T-F bins. Where D is set to 20 following the suggestion in [2].

We also implement the hybrid network described in [2], this structure keeps the basic architecture of deep clustering model and add a new head which outputs a mask with softmax activation. For each T-F bin in model output, we use a 2D vector to represent how much vocal source and background source contributes to the total power.

B. U-net based model

On the first stage, we reproduce the traditional U-net work described in [3]. The U-net model consists of four down-sampling layers and four upsampling layers. Though the original model servers as a semantic segmentation model for image processing task, we found that the model also works when we directly put it into vocal extraction problem.

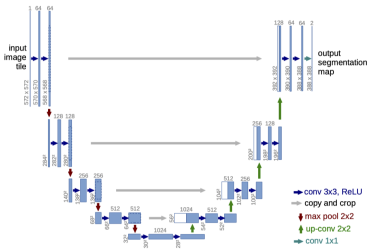


Fig. 3. U-net model[3]

However, compared with deep clustering model, the U-net model takes much more time in training stage and easily overfits on current training data. We make an attempt to combine deep clustering model with U-net model to see if it can make a difference. For the original U-net model, the input channel is set to 1 for log mel-spectrogram. In our modified U-net model, we train U-net model together with deep clustering model. We combine the original spectrogram with the output softmax mask of clustering model to create a three channel feature map for U-net block input. The hybrid U-net model is proved to have better performance over the old model. Both models takes 128×128 mel-spectrogram as input and out put a proportion mask to represent how much the vocal source dominate T-F bins in the feature map.

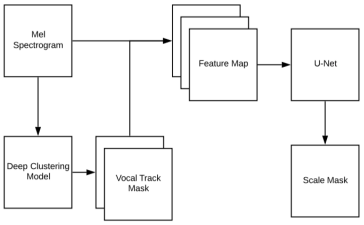


Fig. 4. The hybrid U-net model

V. EXPERIMENT

A. Training

1) Deep Clustering Models: The deep clustering model is trained with a batch size of 32 and we use rmsprop algorithm as the optimizer. We start with training the model merely on the clustering embedding part. The training target is to make the embedding output generated by the model has similar affinity matrix with the binary mask. The binary mask is resized to a matrix $Y \in \mathbb{R}^{TF\times 1}$, where T represents the number of frames and F represents the feature dimensions in the feature map. Meanwhile, the output of clustering model is a matrix $V \in \mathbb{R}^{TF\times D}$, where D is the embedding dimension. The loss function is denoted as $L = ||VV^T - YY^T||_F^2$. However, this operation involves large matrices multiplication and requires large GPU memory. To simplify the calculation, loss function is transformed to $L = ||V^TV||_F^2 + ||Y^TY||_F^2 - 2||V^TY||_F^2$ [6]. The deep clustering model is learned in a way such that the afnity matrix can be approximated from the deep clustering model by minimizing the objective function.

For the proportion-mask head of the hybrid deep clustering model, the training model is trained to generate a vocal source proportion mask with the same size as input. We use binary cross-entropy(BCE) loss for objective function.

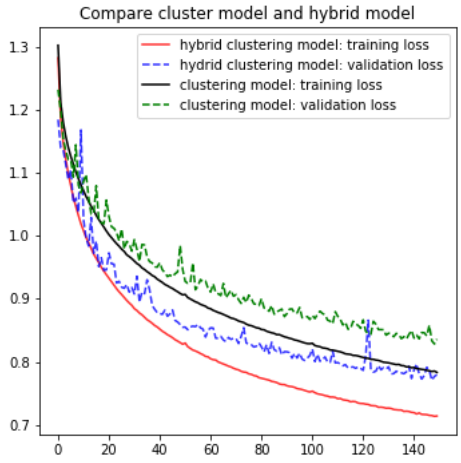


Fig. 5. Loss vs Epoch for clustering models

As shown in the Fig 5, the hybrid clustering model with an extra head has better performance. The addition of proportion-mask head speeds up the training process and also decreases the model's overfitting.

2) U-net based Models: The U-net models use rmsprop algorithm as the optimizer, and the batch size is also set to 32. We use the proportion mask as the training target, and use BCE Loss for backpropagation. We add dropout layers and set the drop out rate to 0.5 to prevent overfitting.

In Fig 4, we compare traditional U-net model with the deep clustering model's proportion-mask head. It's explicit that the U-net model has a overall better performance over the deep cluster model. 50 epoch is enough for training U-net model with same learning rate as deep clustering model. However, the U-net model suffers from overfitting as

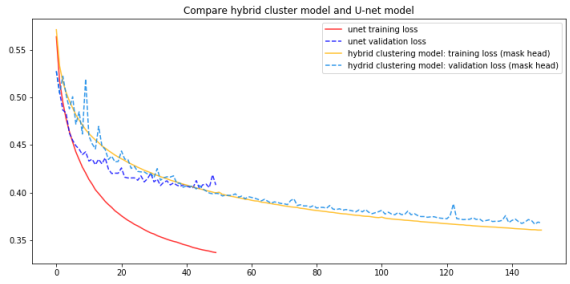


Fig. 6. Loss vs Epoch for U-net/Clustering models

traditional U-net model cannot offer high enough resolution needed to separate vocal source and background voice in feature map. The power difference between voice and background source is not fully represented in this output.

In order to avoid the overfitting issues and also keep the advantage of U-net, we jointly train U-net model with deep clustering model. We name this hybrid architecture UH-net. The loss vs epoch plot of U-net and UH-net is shown in Fig 5. We can tell from the figure that the addition of deep clustering part significantly improve the performance of the traditional U-net model. The model has less loss on validation set and converges faster.

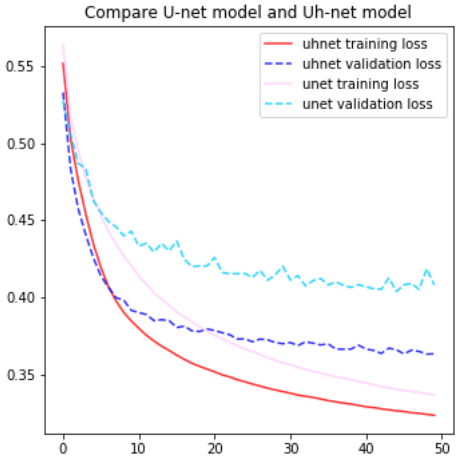
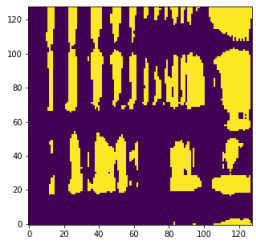


Fig. 7. Loss vs Epoch for U-net/UH-net models

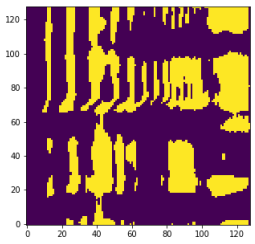
B. Results

1) Deep Clustering Models: There are two kinds of model output in our works. The first one is the embedding matrix. In order to recovering binary mask from embedding matrix, we firstly group the T-F bins using k-means and then create the binary mask with regard to it. The output result of these two clustering models are illustrated in Fig 8. compared with the simple model without extra output head, the hybrid clustering model has a better performance on capturing the detail information near the segmentation boundary, and also makes less mistakes.

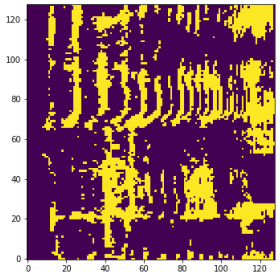
One criterion for binary mask prediction is using IoU (intersection over union) to estimate the prediction score.



(a) Clustering Model

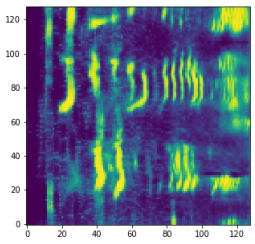


(b) Hybrid Clustering Model

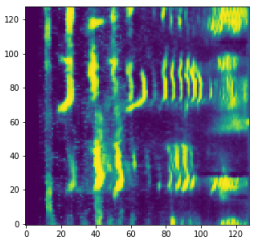


(c) Ground Truth

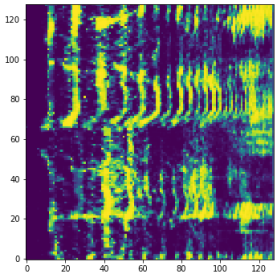
Fig. 8. Binary mask for vocal track



(a) U-net Model



(b) UH-net Model



(c) Ground Truth

Fig. 9. Proportion mask for vocal track

We calculate the IoU score on test set for both models. The IoU is calculated by the following equation:

$$IoU = \frac{Area\ of\ overlap}{Area\ of\ union} \tag{1}$$

For the basic model, the IoU accuracy is 74.78% and the IoU of hybrid model is 79.44% where we can see a explicit improvement.

2) U-net Based Models: The U-net based models generate vocal proportion masks. Sample outputs are listed on Fig 9, which illustrates that the UH-net model maintains more details and generate more explicit segmentation boundaries.

To evaluate how well the predicted mask matches ground truth, we calculate the prediction score using Peak Signal to Noise Ratio (PSNR) algorithm. This algorithm is mainly used to calculate the similarity between different images, but also is proper for evaluating our model performance. Here is the equation we used for calculate PSNR:

$$PSNR = 10\log_{10}(\frac{R^2}{MSE})$$

The UH-net has a PSNR score for 17.12, while the U-net model's baseline score is 16.24. We take these scores as a quantitative metric of performance.

3) Music Recover Result: In order to make comparison between models with different output format, we convert model output from mask to vocal track by multiple mask with original power-scale mel-spectrogram and then convert the new spectrogram to audio sequence with Griffin Lim algorithm. When we test model on full lengths songs, we first divided the song into small frames and do vocal extraction for each frame chunk, then we combine all results together to get the pure vocal track for the whole song.

Fig 10 shows the power-scale spectrogram generated by different models for a sample one second length frame, along with the original mix and vocal tracks. One thing needs to be mentioned here is that since k-means cannot specify which T-F bin group is vocal-dominated when we implement deep clustering models, we are using the proportion-mask head output in hybrid deep clustering model to select the right group.

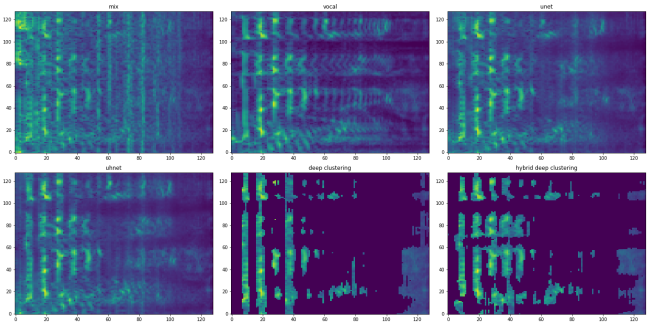


Fig. 10. Spectrogram

We calculate source to distortion (SDR) as the metric for vocal track separation model, the result is shown in the table. We test models on DSD100 test set.

TABLE I
SDR SCORE FOR EACH MODEL

UH-net	U-net	HDC
5.92	4.34	1.56

This clear that UH-net model has the best overall performance. However, the vocal extraction's baseline in [2] is 6.3, which means our models are not fully trained and need to be further modification for improvement.

During the test stage, we also found that the deep clustering model might fail to deal with music without vocal components or has week vocal components. This case is shown in Fig 11. We can see that the clustering models cannot correctly group the T-F bins. U-net based models won't suffer from this situation.

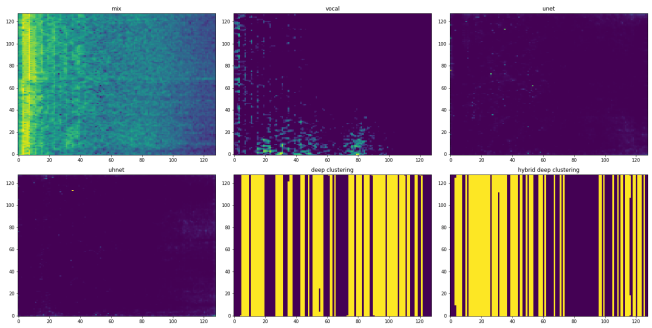


Fig. 11. Mask

VI. Conclusion

In this paper, we discussed four different models for vocal track extraction task. These models are mainly based on two theories. The first one is deep clustering which using embedding and unsupervised learning technologies. The other one is based on semantic segmentation theory which treats the music source separation same as image processing problems.

The UH-net model has overall better performance. By introducing an addition of clustering head, we speed up the training stage of U-net model and also increase the model accuracy. However, from the quantitative perspective of model accuracy, our models still have space for improvement. For example, we could use larger dataset with high resolution input feature map and make further modifications on current model structures.

APPENDIX

The project's code part and sample vocal extraction results are uploaded to GitHub, the repository's link is https://github.com/XiplusChenyu/vocal-track-extraction-using-deep-learning

ACKNOWLEDGMENT

We would like to express our gratitude here to Prof. Mesgarani and TA Luo Yi in the Electrical Engineering Department of Columbia University, who provided essential suggestions and guidance for us.

References

[1] Jansson, Andreas, et al. "Singing voice separation with deep U-Net convolutional networks." (2017).
[2] Luo, Yi, et al. "Deep clustering and conventional networks for music separation: Stronger together." 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017.
[3] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.
[4] https://github.com/XiplusChenyu/Musical-Genre-Classification
[5] Perraudin, Nathanal, Peter Balazs, and Peter L. Sndergaard. "A fast Griffin-Lim algorithm." 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. IEEE, 2013.
[6] Wang, Zhong-Qiu, Jonathan Le Roux, and John R. Hershey. "Alternative objective functions for deep clustering." 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018.