MULTI-DECODER DPRNN: HIGH ACCURACY SOURCE COUNTING AND SEPARATION

By

Junzhe Zhu

Senior Thesis in Electrical Engineering

University of Illinois at Urbana-Champaign

Advisor: Mark Hasegawa-Johnson

December 2020

Abstract

We propose an end-to-end trainable approach to single-channel speech separation with unknown number of speakers. Our approach extends the MulCat source separation backbone with additional output heads: a count-head to infer the number of speakers, and decoder-heads for reconstructing the original signals. Beyond the model, we also propose a metric on how to evaluate source separation with variable number of speakers. Specifically, we cleared up the issue on how to evaluate the quality when the ground-truth has *more or less speakers* than the ones predicted by the model. We evaluate our approach on the WSJ0-mix datasets, with mixtures up to five speakers. We demonstrate that our approach outperforms state-of-the-art in counting the number of speakers and remains competitive in quality of reconstructed signals.

Subject Keywords: Source separation

Acknowledgments

Big thanks to Professor Mark Hasegawa-Johnson for his patient and visionary guidance throughout the final year of my undergraduate career. He is an amazing mentor both academically and professionally, and has diligently guided me through challenges faced in research.

Additionally, I would like to thank Raymond Yeh for supervising me closely on the details of the project, as well as providing me with valuable introduction to common practices in software engineering.

Contents

1. Introduction

Source separation is the task of decomposing a mixed signal into the original signals prior to the mixing procedure. This is an important task with many downstream applications, e.g., improve the accuracy of automatic speech recognition with multiple speakers [1], or separating out singing voices and music [2].

2. Literature Review

Due to the recent progress in deep learning, supervised methods have received a lot of interest [3, 4, 5, 6, 7]. These works formulate the source separation as a regression problem, i.e., given the mixed signal regress the individual components. Various specialized deep-net architectures and losses have been proposed. For example, [8] proposed a loss which is permutation invariant in the ordering of the speakers, or [7] presented a dual-path RNN architecture to better capture both short and long-term features. However, these works have focused on the setting where the number of speakers is a priori known. Recently, several works have also considered the case with variable number of speakers. For example, [9] have proposed a method for separating variable number of speakers, where they train a different model for every number of speakers. At test time, they run an activity detector on the largest speaker model to determine the number of speakers and then run the corresponding model for source separation. Another work is [10] where they have proposed to iteratively separate out one speaker at a time. While straightforward, these methods either require training multiple deep-nets or running multiple forward-passes at test-time, both of which scale linearly with the possible number of speakers. To tackle the aforementioned issues, we propose to train a single model with multiple output heads: a count-head to infer the number of speakers, and multiple decoder heads to separate the signals. These output heads share the same backbone feature extractor. Therefore, our method requires a single pass through the network at test time and can be trained from end-to-end. Additionally, we propose a new metric for evaluating the separation of a variable number of speakers. In particular, our metric considers how to evaluate the quality of the reconstruction when the number of speakers differs between prediction and ground-truth. We evaluate our approach on WSJ0-mix dataset, with up to five speaker mixtures. Our approach surpasses all existing approaches in terms of source counting and achieves similar performance to state-of-the-art models in source separation.

3. Approach

We present a single model approach to source separation with a variable number of speakers, illustrated in Fig. 1. In particular, we augment the standard source separation backbone with additional count-head and decoder-heads to support prediction of variable number of speakers in a single pass. In the following, we describe our approach in more detail.

3.1. Problem Formulation

Let **x** denote the mixed input audio, and $\mathcal{Y} = \{\mathbf{y}^n\}$denote the set of separated audios from each speaker. The goal is to learn a parametric function,

$$F_\theta(\mathrm{x}) \mapsto \mathcal{Y}, \qquad \text{(Equation 1)}$$

with trainable parameters θ. One of the challenges is how to construct a model to handle variable number of outputs. For example, a standard deep-net has a fixed number of output dimensions and does not change between examples.

To mitigate this problem, we assume that the maximum number of speakers, $K \geq |\mathcal{Y}| \; \forall(\mathbf{x}, \mathcal{Y}) \in \mathcal{D}$, is known. In this case, we can model a deep-net to count the number of speakers and model a decoder-head for each number of speakers. This allows us to dynamically select which decoder-head to run and output the correct number of speakers.

We propose a single end-to-end trainable deep-net to accomplish this. Our deep-net contains a count-head, which counts the number of speakers in the mixed-audio, and a list of decoder-heads to reconstruct audios for the corresponding number of speakers. These heads share input features extracted from a backbone network [9]. In the remainder of this section, we describe the architecture details and training procedure for our method.

3.2. Model Architecture

Our model contains a mixture encoder to transform waveform into encoding, and a backbone to extract source encoding from mixture encoding following [7] and [9]. Instead of using a single decoder head with a fixed number of output channels, we replaced it with a set of decoder heads, each having a different number of output channels, where one channel contains source from one speaker. We also added a count head that chooses which decoder head to use during inference.

Encoder & Backbone: As in [9], we use convolution with ReLU to encode mixture waveform, then use repeated MulCat blocks as the backbone separation network.

Count-Head: We train a speaker count-head as an additional branch in parallel with the decoder heads. Given the output tensor from the backbone network, we first linearly transform the feature dimension, then apply global average pooling and ReLU. We then linearly project the result to the set of possible decoder choices, and apply softmax to the output.

Decoder-Heads: We use a list of decoders, as in [7] and [9]. For the k^{th} decoder, given an input tensor with feature dimension N, we apply PReLU with a channel-independent parameter, and use 1x1 convolution to project feature dimension to N×k speakers. We then divide the projected tensor into k tensors, each with feature dimension N, and transform the tensor back to audio with overlap-and-add.

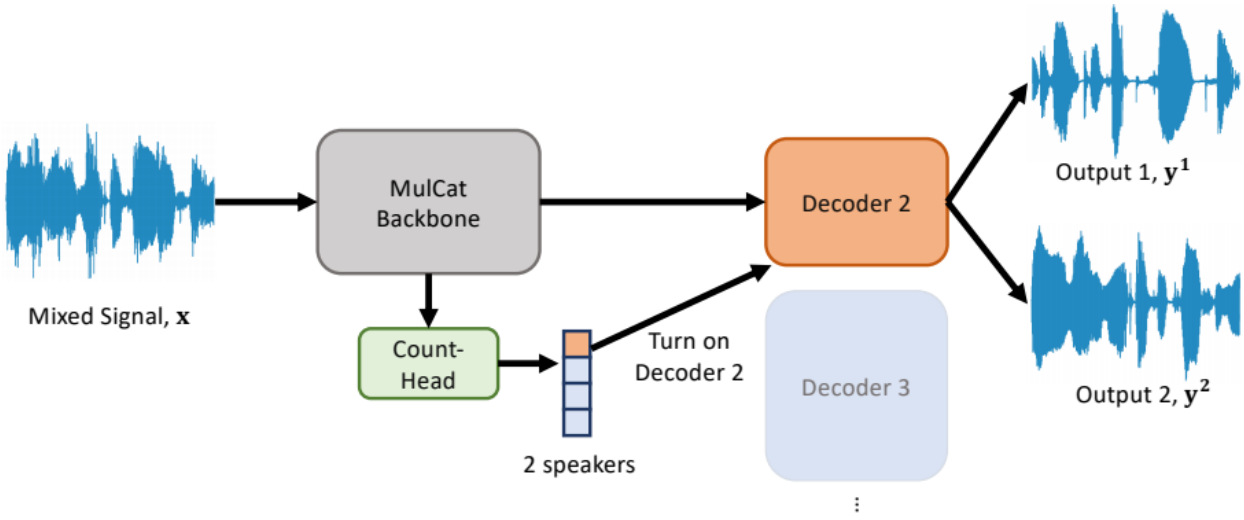

Figure 1. An overview of our proposed approach for handling variable number of speakers for source separation. Given a mixed signal x, our model predicts the number of speakers from the mixed signal and uses the corresponding decoder-head to separate the signal. In this case, decoder 2 is selected, hence a reconstruction of two speakers.

3.3. Training

To train the count-head, we formulate it as a classification task, i.e., we minimize the cross-entropy loss,

$$\mathcal{L}_{\text{count}(\mathbf{x},\mathcal{Y})} = -\sum_{k}^{K} \mathbf{1}_{|\mathcal{Y}|=k} \cdot \log \hat{p}(|\mathcal{Y}| = k)(\mathbf{x}) \qquad \text{(Equation 2)}$$

where 1 denotes the indicator function and $\hat{p}(|\mathcal{Y}| = k)$ denotes the predicted probability that the mixed input audio, **x**, has k speakers. Next, to train the decoder-heads, we utilize the permutation invariant loss, uPit [8], on the decoder-head selected by the ground-truth number of speakers, i.e.,

$$\mathcal{L}_{\text{decoders}}(\mathbf{x}, \mathcal{Y}) = \sum_{k} \mathbf{1}_{|\mathcal{Y}|=k} \cdot \text{uPIT}(\mathcal{Y}, \hat{\mathcal{Y}}_k), \qquad \text{(Equation 3)}$$

where $\hat{\mathcal{Y}}^k$ denotes the output from the k[th] decoder-head and

$$\mathrm{uPIT}(\mathcal{Y}, \hat{\mathcal{Y}}_k) = -\max_{\pi} \sum_{n} \text{SI-SNR}(\mathbf{y}^{\pi(n)}, \hat{\mathbf{y}}_k^n), \qquad \text{(Equation 4)}$$

where π denotes a permutation on the speaker channels, and SI-SNR stands for scale-invariant signal-to-noise ratio, as defined in [11]. Finally, we balance the two losses with a hyper-parameter α, and train over a dataset of paired mixed inputs and separated audio, i.e., $\mathcal{D} = \{(\mathbf{x}, \mathcal{Y})\}$ is as follows,

$$\min_{\theta} \sum_{(\mathbf{x},\mathcal{Y})\in\mathcal{D}} \alpha \cdot \mathcal{L}_{\text{count}}(\mathbf{x}, \mathcal{Y}) + (1-\alpha) \cdot \mathcal{L}_{\text{decoders}}(\mathbf{x}, \mathcal{Y}). \qquad \text{(Equation 5)}$$

3.4. Inference

At test time, the ground-truth number of speakers is not available. In this case, we use the estimated number of speakers from the count-head to select which decoder-head to run, therefore

$$\hat{\mathcal{Y}} = \hat{\mathcal{Y}}_{\hat{c}}, \;\; \hat{c} =_k \hat{p}(|\mathcal{Y}| = k) \qquad \text{(Equation 6)}$$

is the final prediction given **x**.

3.5. Evaluation Metric

Evaluating a system for source separation with variable number of speakers remains an open discussion. It may seem that standard metrics, e.g. SI-SNR, are directly applicable, however these metrics require the number of predicted signals and ground-truth signals to be identical. When the system incorrectly predicts the number of speakers, it is unclear how to compute SI-SNR.

Prior work [9] computes a metric as follows: Let $\hat{S}$ be the number of predicted speakers and S be the ground-truth. In case (a): When $\hat{S} > S$, they compute the correlation between all audio pairs and keep S speakers from the prediction. In case (b): When $\hat{S} < S$, they duplicate $S - \hat{S}$ speakers with the highest correlation to the ground-truth samples. With the speaker number matched, they compute the standard SI-SNR. We note that this choice of duplication / dropping relies on the ground-truth signal. This is not desirable, as a post-processing procedure should not be dependent on the ground-truth. We believe that it is more natural to add "silence" speakers, either ground-truth or the prediction, until the number of speakers between the ground-truth and prediction are identical. Intuitively, a two-speaker mixed signal can be thought of as a three-speaker mixed signal where one of the speakers is silence. However, we run into the issue that SI-SNR is equal to negative infinity if the signal is zero. To avoid this, instead of padding with silence, we choose a negative penalty term $\mathcal{P}_{\text{ref}}$ that would be defined as the approximation to the SI-SNR measured if padded by silence. We name this metric penalized-SI-SNR (P-SI-SNR).

Given dataset $\mathcal{D} = \{(\mathbf{x}, \mathcal{Y})\}$, P-SI-SNR is defined as

$$\frac{1}{|\mathcal{D}|}\sum_{(\mathbf{x},\mathcal{Y})\in\mathcal{D}}\frac{1}{\max(|\mathcal{Y}|,|\hat{\mathcal{Y}}|)}(\mathcal{L}_{\text{match}}+\mathcal{L}_{\text{pad}}), \quad \text{(Equation 7)}$$

where $\hat{\mathcal{Y}}=\{\mathbf{x}^1,...,\mathbf{y}^{\hat{c}}\}$, $\hat{c}$ being the number of predicted sources, and $\mathcal{L}_{\text{match}}$and $\mathcal{L}_{\text{pad}}$ are defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{match}} &= \max_{\pi}\sum_{n=1}^{\min(|\mathcal{Y}|,|\hat{\mathcal{Y}}|)}\text{SI-SNR}(\mathbf{y}^{\pi(n)},\hat{\mathbf{y}}^n) \\ \mathcal{L}_{\text{pad}} &= \mathcal{P}_{\text{ref}}\cdot\Big||\mathcal{Y}|-|\hat{\mathcal{Y}}|\Big|.\end{aligned} \quad \text{(Equation 8)}$$

We believe that our proposed metric is intuitive and naturally balances between the reconstruction quality and mis-classifications in number of speakers. We will next discuss two possible choices of $\mathcal{P}_{\text{ref}}$.

Measuring $\mathcal{P}_{\text{ref}}$ from data: One solution is to choose the “silence” as some zero-mean noise distribution. In this case, we measure the SNR empirically based on samples from WSJ0 recordings. We cut out 0.75 second noise segments from their start, repeat those segments to match the length of recordings, and measure the energy ratio between noise files and recordings. Based on the average of our measurements, we set $\mathcal{P}_{\text{ref}}$ to be -30dB.

Setting $\mathcal{P}_{\text{ref}}$ as average SI-SNR: Another intuitive way to penalize SI-SNR is to have each underestimated or overestimated speaker cancel out the positive contribution to SI-SNR of a correctly predicted speaker. Therefore, we choose $\mathcal{P}_{\text{ref}}$ to be the negative of the average SI-SNR from oracle source counting.

4. Experiments

We first describe our implementation details for dataset preparation, training, testing, model architecture. Next, we provide quantitative comparisons with baselines and demonstrate that our approach achieves state-of-the-art performance in source counting and comparable performance in source separation.

Datasets: We train on WSJ0-2mix and WSJ0-3mix [4], in addition to WSJ0-4mix and WSJ0-5mix [9]. We take 4-second chunks of all audios with 2-second overlap, and pad any chunks at the end that are above 2 seconds. We remove all mixtures below 2s. As mixtures have length equal to the shortest source, those with more speakers are shorter and have fewer chunks. In our training set, 2, 3, 4, 5 speakers all have 20000 audios, and respectively have [24773, 19066, 15986,13809] chunks.

Training Procedure: For each epoch, we use weighted re-sampling with replacement to ensure that chunks for each speaker number are sampled with equal probability. We set probability of choice for each chunk inversely proportional to number of chunks with the same speaker count. We train our model using Adam [13] with learning rate 5e−4, decay of 0.94 every epoch, and batch size of 4. In total, we train our model for 40 epochs, which is much less than the 100 epochs in most previous papers [7, 10].

Testing Procedure: Given an audio signal, we first transform the full audio into chunks. We use the count head to predict which decoder head to use for each chunk and select the decoder head with the highest votes. Using the selected decoder head, we compute separated sources for each chunk, and use the overlap regions to reorder the predicted source channels in a streaming fashion. Lastly, we use overlap-and-add to recover predicted sources for the full audio, and remove the padding at the end chunk.

Architecture Details: For the encoder, we use a convolution kernel size of 8, stride 4, and 256 feature channels. For back-bone, we use LSTM with hidden layer size 256. Similar to[9], we use multi-stage loss, but do not use speaker ID loss for simplicity. During training, we train both the decoders and count-heads with multi-loss, with one set of output after each pair of Mulcat blocks.[1]

Comparisons with Baselines: Many of the systems for variable speaker source separation are not publicly available, therefore we cannot directly compare with them on our proposed P-SI-SNR. To compare, we use the reported numbers from their paper [9][10][12]. Note that since we do not have the exact SI-SNR, in the case of speaker mismatch, we compute an upper bound for the models using their published statistics on oracle SI-SNR and speaker counting accuracy. For computing this upper bound, we assume that each mis-classification of speaker number is overestimated by one, and all the other channels have oracle SI-SNR. This is an upper bound because oracle SNR is always higher than non-oracle SNR, and the

1. See project page for more details: https://junzhejosephzhu.github.io/Multi-Decoder-DPRNN/

ratio of (contribution from correct channels)/penalty is greatest if the error is an overestimated by one channel. For a model with k speakers with oracle SI-SNR x and accuracy a, the upper bound for P-SI-SNR can be computed as

$$\text{P-SI-SNR} \leq a \times x + (1-a) \times \frac{(k \times x + _{\text{ref}})}{k+1} \qquad \text{(Equation 9)}$$

Quantitative Results:

We report quantitative comparisons for source counting performance in Tab. 1, oracle SNR in Tab. 2, and our proposed P-SI-SNR in Tab. 3. We note that models with * are not directly comparable to our model as they train a model for each speaker number, where we have a single model for all speakers. As can be seen from Tab. 1, in the source counting task, our model outperforms all other models, even those with fewer possible choices of speaker counts. Our approach remains competitive in source separation when evaluated using Oracle-SNR, as shown in Tab. 2. Lastly, in Tab. 3, when $\mathcal{P}_{\text{ref}}$ is set to-30 dB, our P-SI-SNR also outperforms all other models in2-speaker and 4-speaker cases, and achieves similar results to best model in the 3-speaker case.

Table 1. Performance of source counting. Each column is recall for corresponding number of speakers. For OR-PIT, only overall accuracy is provided.

Model	**2**	**3**	**4**	**5**
Model-Select(DPRNN) [9]*	81.3	64.4	46.2	85.6
Model-Select(Mulcat) [9]*	84.6	69.0	47.5	92.3
Attractor Network[12]	95.7	97.6	-	-
OR-PIT[10]	95.7		-	-
Ours	99.9	99.2	97.6	97.3

Table 2. Oracle SNR. Each column shows results averaged from all mixtures with corresponding number of speakers. Asterisks indicate models with fixed number of speakers.				
Model	**2**	**3**	**4**	**5**
Conv-Tasnet[6]*	15.3	12.7	-	-
DPRNN[7]*	18.8	-	-	-
DPRNN[9]*	18.21	14.71	10.37	8.65
Mulcat[9]*	20.12	16.85	12.88	10.56
Attractor Network[12]	15.3	14.5	-	-
OR-PIT[10]	14.8	12.6	10.2	-
Ours	19.1	14.1	9.3	5.9

Table 3. P-SI-SNR of each model. For OR-PIT, result is computed by averaging the P-SI-SNR for both 2 and 3 speakers computed with 95.7% recall. Note that models with lower max speaker count generally have higher accuracy, since fewer classes implies a higher P-SI-SNR.				
$\mathcal{P}_{\text{ref}}$=-30dB	**2**	**3**	**4**	**5**
Model-Select(DPRNN) [9]*	15.2	10.7	6.0	7.7
Model-Select(Mulcat) [9]*	17.5	13.21	8.4	10.0
Attractor Network[12]	14.7	14.2	-	-
OR-PIT[10]	13.1		-	-
Ours	19.1	14.0	9.2	5.8
$\mathcal{P}_{\text{ref}}$=-SI-SNR$_{\text{oracle}}$	**2**	**3**	**4**	**5**
Model-Select(DPRNN) [9]*	15.9	12.1	8.1	8.2
Model-Select(Mulcat) [9]*	18.1	14.2	10.2	10.3
Attractor Network[12]	14.9	14.3	-	-
OR-PIT[10]	13.4		-	-
Ours	19.1	14.0	9.3	5.9

5. Conclusion

We present a unified approach to single channel speech separation with an unknown number of speakers. With our proposed multi-decoder architecture and count-head, our model requires a single forward-pass at test-time on a single network. In our experiments, we demonstrate that our model achieves state-of-the-art performance in source counting and competitive source separation quality. Additionally, we propose a new evaluation metric for evaluating source separation with an unknown number of speakers, in which we penalize SI-SNR when the number of sources estimated is incorrect.

References

[1] Amparo Marti, Maximo Cobos, and Jose J Lopez, “Automatic speech recognition in cocktail-party situations: A specific training for separated speech,” The Journal of the Acoustical Society of America, vol. 131, no. 2, pp.1529–1535, 2012.

[2] Po-Sen Huang, Scott Deeann Chen, Paris Smaragdis,and Mark Hasegawa-Johnson, “Singing-voice separation from monaural recordings using robust principal component analysis,” in 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2012, pp. 57–60.

[3] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis,“Deep learning for monaural speech separation,” in 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014, pp. 1562–1566.

[4] John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe, “Deep clustering: Discriminative embeddings for segmentation and separation,” in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016, pp. 31–35.

[5] DeLiang Wang and Jitong Chen, “Supervised speech separation based on deep learning: An overview,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 26, no. 10, pp. 1702–1726, 2018.

[6] Yi Luo and Nima Mesgarani, “Tasnet: time-domain audio separation network for real-time, single-channel speech separation,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 696–700.

[7] Yi Luo, Zhuo Chen, and Takuya Yoshioka, “Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation,” in ICASSP 2020- 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020, pp. 46–50.

[8] Morten Kolbæk, Dong Yu, Zheng-Hua Tan, and Jesper Jensen, “Multi-talker speech separation with utterance-level permutation invariant training of deep recurrent neural networks,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 10, pp.1901–1913, 2017.

[9] Eliya Nachmani, Yossi Adi, and Lior Wolf, “Voice separation with an unknown number of multiple speakers,” in ICML, 2020, pp. 2623–2634.

[10] Naoya Takahashi, Sudarsanam Parthasaarathy, Nabarun Goswami, and Yuki Mitsufuji, “Recursive speech separation for unknown number of speakers,” in INTERSPEECH, 2019.

[11] E. Vincent, R. Gribonval, and C. Fevotte, “Performance measurement in blind audio source separation,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 14, no. 4, pp. 1462–1469, 2006.

[12] Yiming Xiao and Haijian Zhang, “Improved source counting and separation for monaural mixture,” 03 2020, Downloaded 10/21/2020 from https://arxiv.org/abs/2004.00175.

[13] Diederik P. Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” in International Conference for Learning Representations ICLR, 2015, https://arxiv.org/abs/1412.6980.