Speech Separation based on pre-trained model and Deep Modularization

Anonymous authors
Paper under double-blind review

Abstract

Deep neural networks (DNN) have been used extensively to achieve impressive results in speech separation. Most of the DNNs implementations to speech separation relies on supervised learning which is data hungry, and success is pegged on availability of large-scale parallel clean-mixed speech pair. This kind of data is always not available since it is difficult to create hence limiting the implementation of supervised learning. Moreover, the implementation of supervised learning in speech separation requires that systems deal with the permutation problem (permutation ambiguity). This places an upper limit of the quality of separated speech that a tool can attain. To avoid the problem of permutation ambiguity, speech separation based on clustering has been proposed by some recent works. However, these clustering techniques still rely on supervised learning and therefore still require quality paired data. To deal with the problem permutation ambiguity and eliminate need for paired training dataset, we propose a fully unsupervised speech separation technique based on clustering of spectrogram points or raw speech blocks. Our technique couples the traditional graph clustering objectives and deep neural networks to achieve speech separation. We start by establishing features of spectrogram points or raw speech blocks using a pre-trained model and consequently use the features in a downstream task of clustering using deep modularization. Through this we are able to identify clusters of spectrogram points or raw speech blocks dominated all speakers in a mixed speech. We perform extensive evaluation of the proposed technique and show that it outperforms state of the art tools included in the study.

1 Introduction

Speech separation involves isolating each independent speech composed in a mixture speech. For a mixture $y(n)$ that is composed of C independent speech signals $x_c(n)$ with $c = 1, \cdots, C$, $y(n)$ can be represented as:

$$y(n) = \sum_{c=1}^{C} x_c(n) \tag{1}$$

Where n indexes time. Separating speech from another speech is a daunting task by the virtue that all speakers belong to the same class and share similar characteristics Hershey et al. (2016a). Some tools such as Wang et al. (2017) and Wang et al. (2016) develop techniques that perform speech separation on a mixed speech signal based on gender voices present. They leverage the large discrepancy between male and female voices in terms of vocal track, fundamental frequency contour, timing, rhythm, dynamic range to attain gender-based speech separation. In speech separation tasks where the mixture is composed of speakers of the same gender, the separation task is much difficult since the pitch of the voice is in the same range Hershey et al. (2016a). To perform speech separation in such cases, most tools such as Zeghidour & Grangier (2021a) Huang et al. (2011) Weng et al. (2015) Isik et al. (2016) Hershey et al. (2016a) and Luo & Mesgarani (2019a) cast the task as a multi-class regression problem. In that case, training a speech separation model involves comparing its output to a source speaker and the model will always output a dimension for each target class. When multiple sources of the same type exist, the system needs to select arbitrarily which output

dimension to map to each target and this raises a permutation problem (permutation ambiguity) Hershey et al. (2016a). Therefore, systems that perform speaker separation in this manner have an extra burden of designing mechanisms that are geared towards handling the permutation problem. The standout technique for tackling permutation ambiguity problem is the permutation invariant training (PIT) technique Yu et al. (2017) and Kolbæk et al. (2017). The key strategy in PIT is to determine the best output-target pairing and perform error minimization based on this pairing. While effective, the technique has been criticised for having a high computation complexity of $O(S!)$ which is computationally expensive when the number of sources S is high Tachibana (2021) Dovrat et al. (2021). PIT is also unable to handle the output dimension mismatch problem where there is a mismatch on the number of speakers between training and inference Jiang & Duan (2020). To evade the permutation ambiguity problem, works in Hershey et al. (2016b) Byun & Shin (2021) Isik et al. (2016) Qin et al. (2020) and Lee et al. (2022) propose a clustering technique that seeks to identify the multiple speakers present in a mixed speech signal. In these models, the deep neural network (DNN) f_θ takes as its input a whole mixed speech spectrogram X and generates a d dimension embedding vector V i.e., $V = f_\theta(X) \in R^{N\times d}$. Here, the embedding V learns the features of the spectrogram X and is considered a permutation-and-cardinality-independent encoding of the network's estimate of the signal partition. For the network f_θ to be learn how to generate an embedding vector V given the input X, it is trained to minimize the cost function.

$$C_Y(V) = ||VV^T - YY^T||_F^2 = \sum_{ij}(< v_i, v_j > - < y_i, y_j >)^2 \tag{2}$$

Here, $Y = \{y_{i,c}\}$ represents the target partition that maps the spectrogram S_i to each of the C clusters such that $y_{i,c} = 1$ if element i is in cluster c . YY^T is taken here as a binary affinity matrix that represents the cluster assignment in a partition-independent way. The goal in equation 2 is to minimise the distance between the network estimated affinity matrix VV^T and the true affinity matrix YY^T. The minimization is done over the training examples. $||A||_F^2$ is the squared Frobenius norm. Once V has been established, its rows v_i of V are clustered using K-means clustering algorithm. The resulting clusters of V are then used to construct a binary mask which is applied to the mixed spectrogram X to separate the sources. Despite the success of this technique, the existing clustering technique employ supervised training which require a costly process of data labelling. Furthermore, these methods require that the number of speakers to be known before execution which may not be practical in some cases. Our work proposes a speech separation technique based on clustering known as deep modularization that requires no parallel dataset and prior knowledge of number of sources present in the mixture. The clustering is done on the spectrogram points or raw speech blocks. Deep modularization bridges the gap between traditional graph clustering objectives and deep neural networks. We start by establishing features of spectrogram points or raw speech blocks using a pre-trained model and use the features in downstream task of clustering using deep modularization. Through this we achieve speech separation in a purely unsupervised manner. Our contributions are as follows:

1. We propose an unsupervised clustering technique of speech separation based on optimization of cluster assignments of spectrogram points or raw speech blocks in an end-to-end differentiable manner.
2. We perform evaluation comparing the performance of the proposed technique when raw speech and Fourier transformed speech is used as input.
3. We perform extensive evaluation of the proposed technique on different datasets to evaluate its suitability in speech separation in different scenarios.
4. We perform ablation comparing the proposed technique of deep modularization and the classical k-means clustering technique.

2 Related work

Speech enhancement and separation tools can be categorised into two broad categories based on the type of input features, i.e., those using Fourier spectrum features as input and those using time domain features. Fourier spectrum-based features tools do not work directly on the raw signal (i.e., signal in the time domain)

rather they incorporate the discrete Fourier transform (DFT) in their signal processing pipeline mostly as the first step to transform a time domain signal into frequency domain. These models recognise that speech signals are highly non-stationary, and their features vary in both time and frequency. These features include Log-power spectrum features (Fu et al., 2017) (Du & Huo, 2008) (Xu et al., 2015) (Du et al., 2014), Mel-frequency spectrum features (Liu et al., 2022) (Ueda et al., 2016) (Du et al., 2020) (Fu et al., 2018) (Weninger et al., 2014) (Donahue et al., 2018), DFT magnitude features (Nossier et al., 2020) (Fu et al., 2018) Grais & Plumbley (2018) Fu et al. (2019) Jansson et al. (2017) Kim & Smaragdis (2015) and Complex DFT features (Fu et al., 2017) (Williamson & Wang, 2017) (Kothapally & Hansen, 2022a) (Kothapally & Hansen, 2022b). The assumption made by most DNN models that use Fourier spectrum features is that phase information is not crucial for human auditory. Therefore, they exploit only the magnitude or power of the input speech to train the DNN models to learn the magnitude spectrum of the clean signal and the factor in the phase during the reconstruction of the signal (Xu et al., 2014) (Kumar & Florencio, 2016) (Du & Huo, 2008) (Tu et al., 2014) (Li et al., 2017). The use of the phase from the noisy signal to estimate the clean signal is based on works such as (Ephraim & Malah, 1984) that demonstrated that the optimal estimator of the clean signal is the phase of the noisy signal. However, recent research Paliwal et al. (2011) have demonstrated through experiments that further improvements in quality of estimated clean speech can be attained by processing both the short-time phase and magnitude spectra. Due to phase challenge while working with Fourier spectrum features different tool such as (Luo & Mesgarani, 2018) (Luo et al., 2020) (Luo & Mesgarani, 2019a) (Subakan et al., 2021a) explore the idea of designing a deep learning model for speech separation that accepts speech signal in the time-domain. The fundamental concept of these models is to replace the DFT-based input with a data-driven representation that is jointly learnt during model training. The models therefore accept as their input the mixed raw waveform sound and then generate either the estimated clean sources or masks that are applied on the noisy waveform to generate clean sources. By working on the raw waveform, these models address the key limitation of DFT-based models, since the models are designed to fully learn the magnitude and phase information of the input signal during training Luo et al. (2020). The DNN models for speech separation can also be categorised based on how they were trained, that is, spectral mapping techniques (Grais & Plumbley, 2018) (Fu et al., 2016) (Gao et al., 2016), spectral masking techniques Wang & Wang (2013) Isik et al. (2016) Weninger et al. (2014) Fu et al. (2016) Chen et al. (2015) (Huang et al., 2015) Zhang & Wang (2016) Liu & Wang (2019a) and generative modelling(Donahue et al., 2018)(Fu et al., 2019). The use of pre-trained models in speech separation and enhancement has been explored in (Huang et al., 2022).

3 Deep modularization speech separation model

3.1 Model pretraining

3.1.1 Using spectrogram points as input

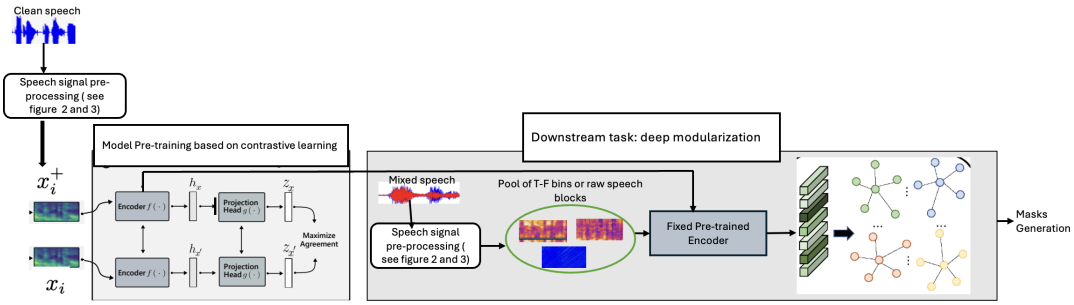

Figure 1: Overall overview of the proposed speech separation based on deep modularization

Our model first establishes a pre-trained model f that learns to generate embeddings of spectrogram points (Time-Frequency bins (T-F bins)). We implement the contrastive learning similar to the one proposed in Saeed et al. (2021) but using spectrogram points (T-F bins) as the inputs. The goal of contrastive self-supervised learning is to establish a representation function $f : x \mapsto R^d$ that maps augmentations to a

$d-$dimensional vectors by ensuring that similar view of augmentations are closer to each other as compared to those of random ones. The practice is to pick augmentations (x, x^+) that are obtained by passing a given input through two different augmentation functions. Ideal augmentations of inputs are those that retain features of the inputs that are crucial in the intended task (e.g., classification) but modify the features that are less important for that task. Unlike work in Saeed et al. (2021) which does not use explicit augmentations, our work generates positive pairs by injecting both noise and reverberation to the input. Hence, given a clean speech signal from a given speaker in the time domain $x \in R^T$, we create its first augmented version in the time domain by adding randomly sampled excerpt from noise recorded in various urban setting from (Wichern et al., 2019). Its second augmented version is created by adding reverberation to the first augmented version using edited scripts from (Maciejewski et al., 2020)(see Figure 2). Each of the two versions is then downsampled to 8kHz and T-F bins generated from the magnitude spectrum by applying short-time Fourier transform (STFT) using 32 ms Hamming window and 8 ms shift. Two T-F bins that are extracted from both augmented sets belonging to a given speaker are considered a positive pair. It is important to note that since we are working at the T-F bins level, it is impractical to generate augmentations by adding add noise or reverberation to the T-F bins directly hence the indirect approach. We then design a function $f : S \mapsto R^d$ that maps the T-F bins to d-dimensional vectors by encouraging the representations of pairs of T-F bins from a positive pair to be closer to each other than the representations of T-F bins from another random speaker as demonstrated in figure 1. Therefore, given n clean speeches from n different speakers, we generate two sets $\bar{X}_A$ and $\bar{X}_B$ containing noise and noise+reverberation T-F bin augmentations as described above. We then exploit a function $\mathcal{S}(.,. \mid \bar{X}_A, \bar{X}_B)$ that generates an augmentation pair (x_i, x_i^+) by selecting T-F bins from both $\bar{X}_A$ and $\bar{X}_B$ i.e.

$$(x_i, x_i^+) \sim \mathcal{D}_{pos} \equiv (x_i, x_i^+) \sim \text{i.i.d } \mathcal{S}(.,. \mid \bar{X}_A, \bar{X}_B) \tag{3}$$

Here the pair (x_i, x_i^+) is the positive pair with distribution $\mathcal{D}_{pos}$. Given a batch of size b, for a positive pair (x_i, x_i^+), we consider all the other $b-2$ to be the negative examples with a distribution of $\mathcal{D}_{neg}$. To train the model to fit the function f, we use adopt simCLR contrastive loss Chen et al. (2020).

$$\mathbb{E}_{x,x^+\sim D_{pos}, x^-_{i:n-2}\sim D_{neg}}[-\log(\frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^{n-2} e^{f(x)^T f(x_i^-)}})] \tag{4}$$

The loss function seeks to make the similarity $f(x)f(x^+)$ larger as compared to $f(x)f(x^-)$. Once the model is trained to establish T-F bin level features, we exploit the trained model in the downstream task.

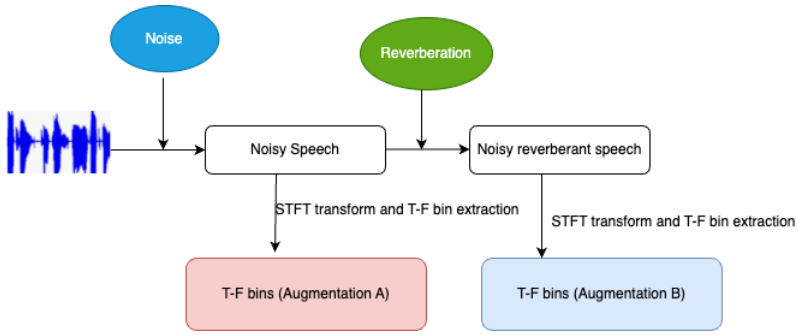

Figure 2: Creating T-F bins augmentations from a speech by first adding noise then reverberation.

3.1.2 Using time domain blocks as input

Given clean speech signal from a given speaker in the time domain $x \in R^T$, similar to discussion in section 3.1.1, we create its first augmented version in the time domain by adding randomly sampled excerpt from noise. The second augmented version is created by adding reverberation to the first augmented version. Each of the augmented speech version in the time-domain $x \in R^T$ is then processed by an encoder similar

to the one proposed in Subakan et al. (2021b) to generate STFT-like representation $h \in R^{F \times T}$. This is then chunked into blocks of size 250 with 125 overlaps between two subsequent blocks along the time axis to generate a set of blocks $L \in R^{F \times S \times N}$ (see figure 3). Therefore, given n clean speeches from n different

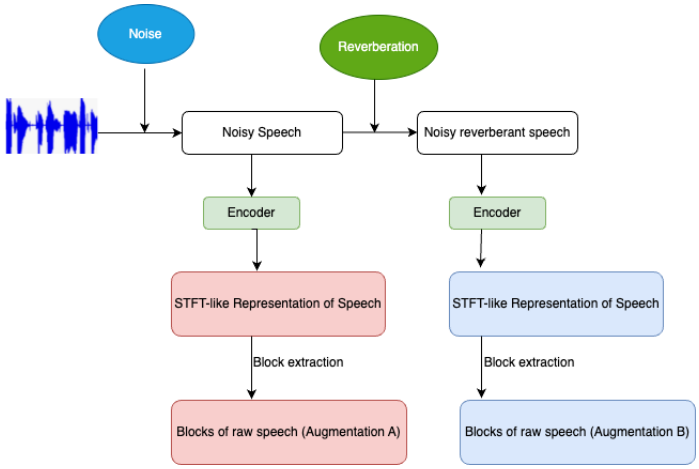

Figure 3: Creating T-F bins augmentations from a speech by first adding noise then reverberation.

speakers, we generate two sets $\bar{X}_A$ and $\bar{X}_B$ containing noise and noise+reverberation time domain speech blocks. We then exploit a function $\mathcal{S}(.,. \mid \bar{X}_A, \bar{X}_B)$ that generates an augmentation pair (x_i, x_i^+) by selecting blocks from both $\bar{X}_A$ and $\bar{X}_B$ belonging to the same speaker. For a batch size of b, a pre-trained model based on raw speech blocks is trained similar to that described in section 3.1.1 and demonstrated in figure 1.

3.1.3 Pre-trained model

For the pre-trained models, we used EfficientNet-B0 (Tan & Le, 2019) as the encoder. EfficientNet-B0 is a lightweight-highly scalable convolutional neural network which was designed to accept 2D image inputs. Since the inputs i.e. T-F bins and raw speech blocks are in 2D, we did not make any modification to the EfficientNet-B0 and used it as originally proposed. In the last layer, we implemented a global max pooling to get an output embedding of $h \in R^{1280}$. When pre-training, we further processed h using a projection head which is a fully connected feed forward layer with 512 units followed by a Layer Normalization and a tanh activation. We discarded the projection head during the downstream task.

3.2 Downstream task

The goal here is to use deep modularization technique to cluster T-F bins or raw speech blocks such that those dominated by a given speaker are clustered together. Deep modularization is a technique that seeks to intersect graph clustering objective and DNN. We begin by defining a graph $G(V, E)$ where $V = (v_1, v_2, \cdots, v_n)$, $|V| = n$ is the set of all T-F bins or raw speech blocks generated from a mixed speech signal as described in section 3.1.1 and 3.1.2. $E \subset V \times V$, $|E| = m$ is the set of all edges of connecting the generated T-F bins or raw speech blocks (subsequently referred to as blocks). We denote the adjacency matrix of G by A where $A_{ij} = 1$ if $\{v_i, v_j\} \in E$ and 0 otherwise. The degree of v_i is defined as $d_i = \sum_j^n A_{ij}$, we are interested in generating a graph partition function $\mathcal{F} : V \rightarrow \{1, \cdots, k\}$ that splits the set of T-F bins or blocks V into k partitions $v_i = \{v_j : \mathcal{F}(v_j) = i\}$ given the T-F bins or blocks attributes $\bar{F} \in R^{n \times d}$ generated by the pre-trained model. In order to partition the vertices, we explore the statistical approach of vertices partitioning known as modularity (Q) (Newman, 2006). Modularity involves comparing the number of edges within partitions and some equivalent randomized partitions (null network) in which edges are placed without regard to relationships that exist in the network. Modularity is, therefore, defined as:

$$Q = \text{Number of edges within partitions} - \text{expected number of such edges} \tag{5}$$

A high value of Q, indicates closer similarities among members belonging to a given partition. Therefore, the goal is to maximise Q. Modularity (Q) is derived in Newman (2006) as:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - P_{ij}) \delta(g_i, g_j) \tag{6}$$

where $\delta(g_i, g_j)$ is 1 if vertex i and j belong to the same partition and 0 otherwise. P_{ij} is the expected number of edges between i and j while A_{ij} is the actual number of edges between i and j. If vertex i and j have degrees d_i and d_j, respectively, then the expected degree of vertex i can be defined as $\sum_j P_{ij} = d_i$. Based on this, vertex i and j are connected with probability $P_{ij} = \frac{d_i d_j}{2m}$ (see (Newman, 2006)). Hence equation 6 is modified to:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{d_i d_j}{2m}) \delta(g_i, g_j) \tag{7}$$

The problem of maximizing Q is NP-Hard (Brandes et al., 2006), however, if we seek to generate k non-overlapping partitions, a partition assignment matrix $S \in R^{n \times k}$ (n represents number of vertices) is defined (Newman, 2006). Each column of S indexes a partition, that is, $S = \{s_1 \mid s_2 \mid, \cdots, \mid s_k|\}$. The columns are vectors of (0,1) elements such that $S_{ij} = 1$ if vertex i belongs to partition j and 0 otherwise. Based on this setup the columns of S are mutually orthogonal since each row of the matrix sums to 1. S therefore satisfies the normalization $Tr(S^T S) = n$ where $Tr(.)$ is the matrix trace. Based on the definition of S, $\delta(g_i, g_j) = \sum_{k=1}^{k} S_{ik} S_{jk}$ and hence

$$Q = \frac{1}{2m} \sum_{ij=1} \sum_{n=1}^{k} (A_{ij} - P_{ij}) S_{ik} S_{jk} = \frac{1}{2m} Tr(S^T B S) \tag{8}$$

where B is the modularity matrix such that $B_{ij} = A_{ij} - P_{ij}$. By relaxing $S \in R^{n \times k}$, the optimal S that maximizes Q is the top k eigenvectors of matrix B. In our case, we seek to optimize Q (learn and optimize cluster assignment matrix S), by modularizing the T-F bin or block features $\bar{F} \in R^{n \times d}$ learned via the pre-trained model. To optimize the cluster assignment, we adapt the deep neural network graph partition technique proposed in Bianchi et al. (2020) and Müller (2023). They partition nodes of a graph by the following formulation:

$$\bar{F} = \text{GNN}(\tilde{A}, X, \theta_{GNN}) \tag{9}$$

$$S = \text{softmax}(\bar{F}) \tag{10}$$

Where $\tilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$, X are the input features, D is the diagonal matrix with the degrees $d_1, \cdots, d_n$ on the diagonal and A is the adjacency matrix. In equation 9, node features $\bar{F}$ are learned via graph neural network (GNN) and the assignment matrix S is established via SoftMax activation function. In (Bianchi et al., 2020), the assignment matrix S is established by multilayer perception (MLP) with SoftMax on the output layer. In our case, we formulate the problem as:

$$\bar{F} = \text{Con}(X, \theta_{con}) \tag{11}$$

$$S = \text{SepFormer}(\bar{F}, \theta_{trans}) \tag{12}$$

Where the T-F bin or block feature matrix $\bar{F}$ is established via a pre-trained model (Con). The partition assignment of a T-F bin or block is established using a SepFormer proposed in (Subakan et al., 2021b). SepFormer is a transformer-based model that is able to learn both short-term and long-term dependencies that exist within a T-F bin or a block. The output of the SepFormer is passed through a feedforward network equipped with a softmax that assigns a T-F bin or a block to a given partition (see figure 4). This essentially maps each T-F bin or block's feature $\bar{f}_i \in \bar{F}$(with $1 \leq i \leq n$) to the j column of the cluster assignment matrix S. To optimise the assignment S, we use the loss function in Equation 13 (Müller, 2023). By training the SepFormer using the objective in equation 13, we train a DNN model using a graph clustering objective. The loss in equation 13 is composed of a modularity (derived in Equation 8) term and a collapse regularizer. The collapse regularizer is crucial to avoid S generating trivial partitions (Müller, 2023). Furthermore, it

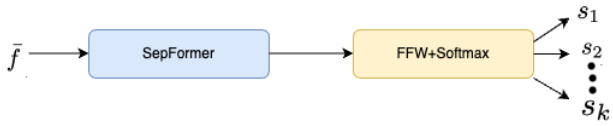

Figure 4: How SepFormer is utilized to perform T-F bin or block partitioning.

has been shown in Müller (2023) that the loss function in equation 13 maintains consistency of community detection as the number of nodes increases.

$$L(S) = -\frac{1}{2m}Tr(S^T BS) + \frac{\sqrt{k}}{n} \mid\mid \sum_i S_i^T \mid\mid_F -1 \tag{13}$$

Here, $||.||_F$ is the Frobenius norm. The complexity of the modularity term $Tr(S^T BS)$ is $\mathcal{O}(n^2)$ per update of $L(S)$ which makes the training process computationally costly. Therefore, to efficiently update $L(S)$, Müller (2023) proposes to decompose $Tr(S^T BS)$ into sum of sparse matrix-matrix multiplication and rank one degree normalization $Tr(S^T AS - Sd^T dS)$. This reduces the complexity to $(O)(d^2 n)$ for every update of the loss function.

$$L(S) = -\frac{1}{2m}Tr(S^T AS - Sd^T dS) + \frac{\sqrt{k}}{n} \mid\mid \sum_i S_i^T \mid\mid_F -1 \tag{14}$$

During implementation our model uses the loss in equation 14.

3.3 Adjacency matrix

To construct the adjacency matrix A, for each T-F bin or block i we compute its similarity with all other nodes using inner product i.e.

$$e_{ij} = \bar{f}_i^T \bar{f}_j \tag{15}$$

where $j = 1, 2, \cdots, n$ and $\bar{f}_i$ and $\bar{f}_j \in \bar{F}$. We then select a threshold θ such that if $e_{ij} < \theta$, we remove an edge between i and j then the adjacency matrix is defined as

$$A_{ij} = \begin{cases} 1, & \text{if there is an edge between } i \text{ and j} \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

Optimum θ is established experimentally (explained in section 5.6).

4 Clean source signal estimation

From the established partitions k, we generate k masks in the range $[0, 1]$, where 0 indicates that a given T-F bin or block in the input mixed signal is missing in that cluster, while 1 signifies the presence of a given T-F bin or block. The mask-based separation of sources is predicated on the assumption of sparsity and orthogonality of the sources in the mixed signal in the domain in which masks are computed. Based on this assumption, the dominant signal at a given range is taken to be only signal at that range. Therefore, the generation of clusters through modularization is used to estimate the dominant signals in a given range. Once the masks have been established, they are applied to the input mixed signal to generate k estimated clean signals. For the input speech signal that has been transformed to STFT, the mask is applied to the input STFT spectrogram to obtain the estimated spectrograms of clean speech signals. The inverse STFT is then used to estimate a clean speech signal. In case of a time domain signal, the mask is applied to the STFT-like transformation generated by the encoder. The decoder (transposed encoder) is the used to generate estimated signal. For STFT phase reconstruction, we use the technique proposed in Wang et al. (2018) which jointly reconstructs the phase of all sources in each mixture by exploiting their estimated magnitudes and the noisy phase using the multiple input spectrogram inversion (MISI) algorithm (Gunawan & Sen, 2010).

5 Experimental setup

Pre-training dataset: To pre-train the two model variants (i.e. one trained using T-F bins as input and another with raw speech blocks), we used the popular Wall Street Journal (WSJ0) corpus (Paul & Baker, 1992). The dataset was recorded using a close-talk microphone hence free from reverberation and noise. We used 30 hours of speeches from *si_tr_s* to train the models. 30 hours length of speeches generate over 10 million T-F bins or blocks input dataset.
Pre-training configuration: To train the two variants of pre-trained models, we used the Adam optimiser and the cyclical learning rate (Smith, 2017) with a minimum learning rate of $1e-4$ and a maximum of $1e-1$. Each model was trained with a single NVDIA V100 GPU for 2M steps with a batch size of 512 T-F bins or blocks.
Speech separation dataset: To evaluate the quality of separated speech resulting from the proposed technique, we used wsj0-2mix, wsj0-3mix Hershey et al. (2016a),wsj0-4mix, ws0-5mix (Nachmani et al., 2020), Libri5Mix, Libri10Mix (Dovrat et al., 2021). The wsj0-2mix, wsj0-3mix, wsj0-4mix, and ws0-5mix datasets are made of 2, 3, 4, 5 speaker mixtures, respectively, created from the WSJ0 corpus. The datasets were created by exploiting randomly selected gains in order to achieve relative levels between 0 and 5 dB between the 2, 3, 4, 5 speech signals. The datasets are composed of 30 h training, 10 h validation, and 5 h test sets. The training and validation sets share common speakers, which is not the case for test set. Libri5Mix and Libri10Mix are speech mixture composed of 5 and 10 different speakers respectively. The dataset was created from the LibriMix dataset (Cosentino et al., 2020), which was created from LibriSpeech Panayotov et al. (2015). The mixtures were created in Dovrat et al. (2021)from clean utterances with no noise with the resulting mixtures having an SNRs that are normally distributed with a mean of 0 dB and a standard deviation of 4.1 dB. The datasets wsj0-2mix, wsj0-3mix, wsj0-4mix, ws0-5mix, Libri5Mix and Libri10Mix assume that speech separation will happen in anechoic environment which is an unrealistic assumption for speech separation since mixed speech is always recorded by a distant microphone hence it is always riddled with noise and reverberation. To demonstrate the performance of the proposed technique to separate mixed speech which contain noise and reverberation, we use WHAM ! (Wichern et al., 2019) and WHARM ! (Maciejewski et al., 2020) datasets. These datasets were derived from wsj0-2mix dataset by adding environmental noise and noise+reverberation respectively. For all of these datasets, we used the test dataset for speech separation. The T-F bins or blocks generated from the speeches in the test dataset were processed by the relevant pre-trained model for embedding generation, e.g., if a pre-trained model was trained using time domain blocks it processes time-domain blocks to generate embeddings.
Downstream training: During feature clustering we used two configurations: First, we trained a SepFormer on top of the frozen encoder to partition T-F bins or blocks. In the second configuration, the pre-trained encoder is fine-tuned . The SepFormer used in the downstream task of clustering spectrogram points or raw speech blocks is optimized in an end-to-end differentiable manner using equation 14 as the loss function. Based on this, we trained four different SepFormer configurations:

1. SepFormer clustering of T-F bins based on frozen encoder trained using T-F bins (Sep1).
2. SepFormer clustering of raw speech blocks based frozen encoder trained using raw speech blocks (Sep2).
3. SepFormer clustering of T-F bins based on fine-tuning an encoder trained using T-F bins (Sep3).
4. SepFormer clustering of raw speech blocks based on fine-tuning an encoder trained using raw speech blocks(Sep4).

We used the Adam optimiser and the cyclical learning rate (Smith, 2017) with a minimum learning rate of $1e-4$ and a maximum of $1e-1$. The models were trained with a single NVDIA V100 GPU for 2M steps with a batch size of 216 T-F bins or blocks. We optimized the downstream model according to equation 14 to generate partitions. We set the maximum number of clusters $k = 20$.
Evaluation metrics: We used objective metrics of Short-time objective intelligibility (STOI)(Taal et al., 2011), perceptual evaluation of speech quality (PESQ) algorithm (Rix et al., 2001), SI-SNR improvement (SI-SNRi), Signal-to-Distortion Ratio improvement (SDRi), Deep Noise Suppression MOS (DNSMOS) which

is a reference-free metric that evaluates perceptual speech quality Reddy et al. (2021). It is a DNN based model trained on human ratings obtained by using an online framework for listening experiments based on ITU-T P.808. We also use SIG, BAK, OVRL: The non-intrusive speech quality assessment model DNSMOS P.835 (Reddy et al., 2022).

6 Results

6.1 Quality of clusters

To begin our experiments, we first evaluate which of the four model configurations (Sep1, Sep2, Sep3, Sep4) results in better cluster (partition) generation. We do this by measuring the quality of clusters generated by each. To evaluate how good the clusters are, we use the graph-based cluster measurement metrics proposed in (Yang & Leskovec, 2012). We are particularly interested in metrics that capture how well a given partition is separated from the rest i.e., quantifying the number of edges pointing from a given partition to other partitions. A good partition should have few edges pointing outwards. The most relevant metrics for our study being graph modularity and conductance.

Cluster conductance (C)= $\frac{c_s}{2m_s+c_s}$, if S is a partition, the function C measures how similar the nodes of S are where m_s is the number of edges in S i.e., $m_s = \{(u,v) \in E, u \in S, v \in S\}$ and c_s is the number of edges in the boundary of S i.e. $c_s = \{(u,v) \in E : u \in S, v \notin S\}$. Conductance quantifies the fraction of edges pointing outside a given partition. Quality partitions should have a small conductance value.

Graph modularity ($\mathcal{Q}$)= $\frac{1}{4}(m_s - E(m_s))$ where $E(m_s)$ is the expected m_s. Quality partitions should have high modularity.

The results of our evaluation are reported in table 1. When raw speech block features are used to train the encoder, the pre-trained model generates more quality embeddings that lead to quality clusters in the downstream task as compared to when T-F bins are used as input. The features from pre-trained model with raw speech block as input generates better clusters in all the three test datasets used in evaluation. This is an indication that for this setup, the use of raw speech features captures more speech features as compared to the STFT transformed features. We also note that fine-tuning the encoder is beneficial to the downstream task as compared to the use of frozen encoder. Note that quality of cluster evaluation is done prior to speech reconstruction hence the drop in cluster quality when using STFT transformed features cannot be attributed to phase handling issues.

Table 1: Results of conductance C and modularity Q when using different input configurations and different mixtures. Here the values of C and Q have been multiplied by 100.

WSJ0-3mix test-dataset		
Model Configuration	**C(↓)**	**$\mathcal{Q}$(↑)**
Sep1	14.9	88.1
Sep2	14.4	86.7
Sep3	14.1	88.5
Sep4	13.9	88.9
WSJ0-4mix test-dataset		
Sep1	16.2	85.3
Sep2	15.8	86.9
Sep3	15.1	87.2
Sep4	14.6	87.6
WSJ0-5mix test-dataset		
Sep1	19.0	76.6
Sep2	18.4	78.5
Sep1	18.0	78.8
Sep2	17.7	79.6

6.2 Performance on speech separation

We evaluated the quality of separated speeches by applying masks generated by Sep1, Sep2, Sep3 and Sep4 on wsj0-2mix, wsj0-3mix wsj0-4mix, ws0-5mix test datasets. Table 2 reports the results based on the evaluation

metrics. In all the four datasets, the quality of separated speeches is highest on Sep4 based speech separation. This followed by that of Sep2. This trend does not follow the evaluation on cluster quality which indicated that Sep4 produced the best quality clusters followed by those of Sep3. We attribute the drop of quality in Sep3 to the phase reconstruction associated with STFT transformation. The results also show that when raw blocks are used in both pre-training and downstream task, the quality of separated speech is superior.

Table 2: Speech separation results when the two variants of inputs are used

WSJ0-2mix test-dataset								
Model	**SDRi(↑)**	**SI-SNRi(↑)**	**STOI(↑)**	**PESQ (↑)**	**DNSMOS (↑)**	**SIG (↑)**	**BAK (↑)**	**OVRL (↑)**
Sep1	19.6	19.4	0.8969	3.93	3.98	3.96	4.11	4.01
Sep2	21.9	21.6	0.9146	4.04	4.14	4.09	4.17	4.16
Sep3	20.6	20.2	0.9054	3.98	4.02	4.00	4.13	4.07
Sep4	**22.6**	**22.3**	**0.9346**	**4.08**	**4.17**	**4.11**	**4.23**	**4.20**
WSJ0-3mix test-dataset								
Sep1	17.9	17.3	0.8702	3.76	3.86	3.88	3.91	3.89
Sep2	19.7	19.5	0.9051	4.01	4.05	4.01	4.08	4.11
Sep3	18.6	17.9	0.8837	3.86	3.98	3.91	4.07	4.01
Sep4	**21.8**	**21.3**	**0.9150**	**4.04**	**4.10**	**4.07**	**4.14**	**4.09**
WSJ0-4mix test-dataset								
Sep1	15.8	15.5	0.8414	3.52	3.67	3.75	3.81	3.78
Sep2	17.4	17.0	0.8896	3.92	3.97	3.94	4.01	4.03
Sep3	16.3	16.01	0.8522	3.70	3.84	3.87	4.03	3.98
Sep4	**20.9**	**20.5**	**0.9011**	**3.98**	**4.02**	**4.00**	**4.08**	**4.04**

6.3 Comparison with other speech separation tools in few ($n \leq 3$) source mixtures.

Here, we compare how the proposed technique of speech separation performs as compared to other state of the art speech separation tools. The results are reported in table 3. In wsj0-2mix, Sep4 based speech separation improves SI-SNRi and SDRi of the best performing tool SepFormer+DM by 0.3 and SDRi by 0.5 respectively . In wsj0-3mix, the scalability of the proposed technique to high source mixtures is evidence as compared to the other tools. While the performance of SepFormer + DM with regard to SI-SNRi and SDRi drops by 2.8 and 2.7 respectively in the wsj0-3mix dataset when compared to its performance in wsj0-2mix, the performance of Sep4 based speech separation drops marginally by only 0.8 and 1.2 respectively in the two metrics. This may signal the ability of deep modularity technique to generalize to mixture with high sources.

Table 3: Comparing the results of the proposed technique with other state of the art speech separation tools.

WSJ0-2mix test-dataset		
Model	**SI-SNRi**	**SDRi**
SepFormer Subakan et al. (2021b)	20.4	20.5
SepFormer+DM Subakan et al. (2021b)	22.3	22.4
Wavesplit Zeghidour & Grangier (2021b)	21.0	21.2
Wavesplit+DM Zeghidour & Grangier (2021b)	22.2	22.3
DeepCASA Liu & Wang (2019b)	17.7	18.0
ConvTasnet Luo & Mesgarani (2019b)	15.3	15.6
Sep1	19.6	19.8
Sep2	21.9	22.1
Sep3	20.6	21.0
Sep4	**22.6**	**22.9**
WSJ0-3mix test-dataset		
SepFormer	17.6	17.9
SepFormer+DM	19.5	19.7
Wavesplit	17.3	17.6
Wavesplit+DM	17.8	18.1
ConvTasnet	12.7	13.1
Sep1	17.9	18.1
Sep2	19.7	19.7
Sep3	18.6	19.0
Sep4	**21.8**	**21.7**

6.4 Comparison with other speech separation tools in high($n \geq 5$) source mixtures.

Here, we evaluate the performance of the proposed technique in mixtures with many sources. The results are shown in table 4. Sep4 based speech separation outperforms the existing tools by 0.2, 0.3 and 2.0 when evaluated on wsj0-5mix, Libri5Mix and Libri10Mix dataset on SDRi metric. This shows the proposed technique can scale to high source mixtures and generate quality estimated sources.

Table 4: Performance of the proposed technique on high source mixtures as compared to other tools that can perform high source mixtures separation.

WSJ0-5mix test-dataset	
Model	**SDRi**
ConvTasNet Luo & Mesgarani (2019a)	6.8
DPRNN Luo et al. (2020)	8.6
MulCat Nachmani et al. (2020)	10.6
Hungarian Dovrat et al. (2021)	13.2
Sep1	10.4
Sep2	12.3
Sep3	10.9
Sep4	**13.4**
Libri5Mix test-dataset	
SinkPIT Tachibana (2021)	9.4
MulCat Nachmani et al. (2020)	10.8
Hungarian Dovrat et al. (2021)	12.7
Sep1	9.6
Sep2	11.5
Sep3	10.1
Sep4	**13.0**
Libri10Mix test-dataset	
SinkPIT Tachibana (2021)	6.8
MulCat Nachmani et al. (2020)	4.8
Hungarian Dovrat et al. (2021)	7.8
Sep1	7.9
Sep2	8.1
Sep3	9.3
Sep4	**9.8**

6.5 Results on WHAM! and WHAMR! datasets

We also evaluated the proposed technique in dataset that contains noise and reverberation. The results of the performance of the proposed technique on WHAM and WHARM datasets are shown in table 5. For WHAM dataset, the model performs denoising in addition to speech separation. In that case it must add a partition for noise. For WHARM dataset, the model performs denoising, dereverberation and speech separation. The results on the WHAM! and WHAMR! datasets are compared to the other state of the art tools. Speech separation based on Sep4 registers the best results in both datasets as compared to other state of the art results. This shows that the proposed technique is able to deal with mixtures with noise and reverberation.

6.6 Selecting similarity threshold θ

Selecting the ideal threshold (θ) when creating adjacency matrix is not trivial. If θ is high, we risk losing important relationships between frames. On the other hand, selecting low θ results in a large graph dominated by uninformative edges and increases the clustering time. To select the optimum θ we conducted experiments with different datasets where we varied the value of θ and recorded modularity and the number of clusters generated. The graph showing how modularity and number of clusters generated vary when using ws0-5mix test dataset is shown in figure 5. The modularity values in figure 5 have been normalised by multiplying by 100, and values of number of clusters have been normalised by multiplying by 10 for easy visualisation. As can be seen in figure 5, as the similarity increases, modularity increases at the risk of generating a singleton partition. Decreasing the similarity lowers modularity and the risk of generating extra partitions increases. In our case we selected a $\theta = 0.3$. The same threshold is used in all the experiments.

Table 5: Comparing the results of the proposed technique with other state of the art speech separation tools on WHAM and WHARM test-dataset .

WHAM test-dataset		
Model	**SI-SNRi**	**SDRi**
SepFormer Subakan et al. (2021b)	14.7	15.1
SepFormer+DM Subakan et al. (2021b)	16.4	16.7
Wavesplit+DM Zeghidour & Grangier (2021b)	16.0	16.5
ConvTasnet Luo & Mesgarani (2019b)	12.7	-
Sep1	15.6	15.9
Sep2	16.4	16.8
Sep3	15.8	16.0
Sep3	17.2	17.4
WHARM test-dataset		
SepFormer	11.4	10.4
SepFormer+DM	14.0	13.0
Wavesplit+DM	13.2	12.2
ConvTasnet	8.3	-
BiLSTM Tasnet	9.2	-
Sep1	13.4	14.0
Sep2	14.1	14.4
Sep3	13.6	14.3
Sep3	15.0	15.4

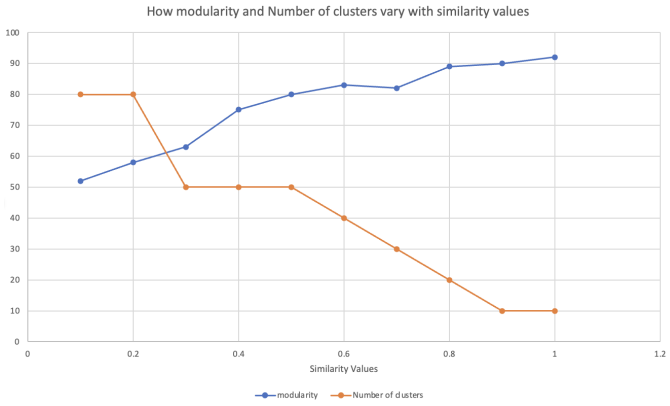

Figure 5: Graph showing how modularity and number of clusters vary as we change the similarity threshold

7 Ablation

Here, instead of using deep modularization to generate partitions, we replace it with the classical k-means where we set $k = 20$. Concretely, after generating T-F bins or raw speech blocks features using a frozen pre-trained encoder, we exploit k-means to generate partitions which are then used for mask generation. We exploit k-means for the two versions:

1. k-means (raw speech blocks): Here k-mean is applied to cluster features of raw speech blocks.
2. k-means (T-F bins): Here k-mean is applied to cluster features of T-F bins.

The mask generation and speech reconstruction remains the same as described in section 4. We compare the results of quality of speech separated based on k-means and that of deep modularization. Since in k-means we use frozen pre-trained encoder to generate features of the input, we compare its results with that of Sep1 and Sep2 which also employ frozen encoder. The results are shown in table 6. The results show that deep modularization achieves superior results as compared to that of k-means. This shows that deep modularization is better at capturing relationship between T-F bins or raw speech blocks as compared to k-means.

Table 6: Comparing the results of the proposed technique with other state of the art speech separation tools.

WSJ0-2mix test-dataset		
Model	**SI-SNRi**	**SDRi**
Sep1	19.6	19.8
Sep2	21.9	22.1
k-means(raw speech blocks)	17.5	17.8
k-means(T-F bin)	17.3	17.4
WSJ0-3mix test-dataset		
Sep1	17.9	18.1
Sep2	19.7	19.7
k-means(raw speech blocks)	15.5	15.9
k-means(T-F bin)	15.1	15.7

8 Conclusion

This work proposes an unsupervised technique of speech separation. The technique relies on a pre-trained model to generate input features which are then used downstream. In the downstream task, we combine deep neural network and graph clustering objectives to create clusters of spectrogram points or raw speech blocks features. The clusters are created such that T-F bins or raw speech blocks dominated by a given speaker are clustered together. We conduct extensive experiments with a number of dataset and establish that the proposed tool achieves state of the art results.

References

Filippo Maria Bianchi, Daniele Grattarola, and Cesare Alippi. Spectral clustering with graph neural networks for graph pooling. In *International conference on machine learning*, pp. 874–883. PMLR, 2020.

Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Görke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner. Maximizing modularity is hard. *arXiv preprint physics/0608255*, 2006.

Jaeuk Byun and Jong Won Shin. Monaural speech separation using speaker embedding from preliminary separation. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 29:2753–2763, 2021. ISSN 23299304. doi: 10.1109/TASLP.2021.3101617.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Zhuo Chen, Shinji Watanabe, Hakan Erdogan, and John R. Hershey. Speech enhancement and recognition using multi-task learning of long short-term memory recurrent neural networks. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2015-Janua, pp. 3274–3278, 2015. doi: 10.21437/interspeech.2015-659.

Joris Cosentino, Manuel Pariente, Samuele Cornell, Antoine Deleforge, and Emmanuel Vincent. Librimix: An open-source dataset for generalizable speech separation. *arXiv preprint arXiv:2005.11262*, 2020.

Chris Donahue, Bo Li, and Rohit Prabhavalkar. Exploring speech enhancement with generative adversarial networks for robust speech recognition. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2018-April(Figure 1):5024–5028, 2018. ISSN 15206149. doi: 10.1109/ICASSP.2018.8462581.

Shaked Dovrat, Eliya Nachmani, and Lior Wolf. Many-speakers single channel speech separation with optimal permutation training. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 4:2408–2412, 2021. ISSN 19909772. doi: 10.21437/Interspeech.2021-493.

Jun Du and Qiang Huo. A speech enhancement approach using piecewise linear approximation of an explicit model of environmental distortions. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pp. 569–572, 2008. ISSN 19909772. doi: 10.21437/interspeech.2008-168.

Jun Du, Yanhui Tu, Yong Xu, Lirong Dai, and Chin Hui Lee. Speech separation of a target speaker based on deep neural networks. *International Conference on Signal Processing Proceedings, ICSP*, 2015-January (October):473–477, 2014. doi: 10.1109/ICOSP.2014.7015050.

Zhihao Du, Xueliang Zhang, and Jiqing Han. A Joint Framework of Denoising Autoencoder and Generative Vocoder for Monaural Speech Enhancement. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 28:1493–1505, 2020. ISSN 23299304. doi: 10.1109/TASLP.2020.2991537.

Yariv Ephraim and David Malah. Speech Enhancement Using a Minimum Mean-Square Error Short-Time Spectral Amplitude Estimator. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(6): 1109–1121, 1984. ISSN 00963518. doi: 10.1109/TASSP.1984.1164453.

Szu Wei Fu, Yu Tsao, and Xugang Lu. SNR-aware convolutional neural network modeling for speech enhancement. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 08-12-Sept:3768–3772, 2016. ISSN 19909772. doi: 10.21437/Interspeech.2016-211.

Szu Wei Fu, Ting Yao Hu, Yu Tsao, and Xugang Lu. Complex spectrogram enhancement by convolutional neural network with multi-metrics learning. *IEEE International Workshop on Machine Learning for Signal Processing, MLSP*, 2017-September:1–6, 2017. ISSN 21610371. doi: 10.1109/MLSP.2017.8168119.

Szu Wei Fu, Tao Wei Wang, Yu Tsao, Xugang Lu, Hisashi Kawai, Daniel Stoller, Sebastian Ewert, Simon Dixon, Xugang Lu, Yu Tsao, Shigeki Matsuda, Chiori Hori, Yong Xu, Jun Du, Li Rong Dai, Chin Hui Lee, Tian Gao, Jun Du, Li Rong Dai, Chin Hui Lee, Szu Wei Fu, Yu Tsao, Xugang Lu, Felix Weninger, John R. Hershey, Jonathan Le Roux, Bjorn Schuller, Yong Xu, Jun Du, Li Rong Dai, Chin Hui Lee, Francesc Lluís, Jordi Pons, and Xavier Serra. Speech enhancement based on deep denoising autoencoder. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 08-12-Sept(1):7–19, 2018. ISSN 19909772. doi: 10.21437/interspeech.2013-130.

Szu Wei Fu, Chien Feng Liao, Yu Tsao, and Shou De Lin. MetricGAN: Generative adversarial networks based black-box metric scores optimization for speech enhancement. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:3566–3576, 2019.

Tian Gao, Jun Du, Li Rong Dai, and Chin Hui Lee. SNR-based progressive learning of deep neural network for speech enhancement. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 08-12-Sept:3713–3717, 2016. ISSN 19909772. doi: 10.21437/Interspeech.2016-224.

Emad M. Grais and Mark D. Plumbley. Single channel audio source separation using convolutional denoising autoencoders. *2017 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2017 - Proceedings*, 2018-Janua:1265–1269, 2018. doi: 10.1109/GlobalSIP.2017.8309164.

David Gunawan and D. Sen. Iterative phase estimation for the synthesis of separated sources from single-channel mixtures. *IEEE Signal Processing Letters*, 17(5):421–424, 2010. ISSN 10709908. doi: 10.1109/LSP.2010.2042530.

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)*, pp. 31–35. IEEE, 2016a.

John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. Deep Clustering : Discriminative Embeddings For Segmentation And Separation. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 31–35, 2016b.

Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Deep Learning For Monaural Speech Separation. *Acta Physica Polonica B*, 42(1):33–44, 2011. ISSN 05874254.

Po Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Joint Optimization of Masks and Deep Recurrent Neural Networks for Monaural Source Separation. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 23(12):2136–2147, 2015. ISSN 23299290. doi: 10.1109/TASLP.2015.2468583.

Zili Huang, Shinji Watanabe, Shu Wen Yang, Paola García, and Sanjeev Khudanpur. Investigating Self-Supervised Learning for Speech Enhancement and Separation. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2022-May:6837–6841, 2022. ISSN 15206149. doi: 10.1109/ICASSP43922.2022.9746303.

Yusuf Isik, Jonathan Le Roux, Zhuo Chen, Shinji Watanabe, and John R. Hershey. Single-channel multi-speaker separation using deep clustering. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 08-12-Sept:545–549, 2016. ISSN 19909772. doi: 10.21437/Interspeech.2016-1176.

Andreas Jansson, Eric Humphrey, Nicola Montecchio, Rachel Bittner, Aparna Kumar, and Tillman Weyde. Singing voice separation with deep U-Net convolutional networks. *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017*, pp. 745–751, 2017.

Fei Jiang and Zhiyao Duan. Speaker attractor network: Generalizing speech separation to unseen numbers of sources. *IEEE Signal Processing Letters*, 27:1859–1863, 2020. ISSN 15582361. doi: 10.1109/LSP.2020.3029704.

Minje Kim and Paris Smaragdis. Adaptive denoising autoencoders: A fine-tuning scheme to learn from test mixtures. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9237:100–107, 2015. ISSN 16113349. doi: 10.1007/978-3-319-22482-4_12.

Morten Kolbæk, Dong Yu, Zheng-Hua Tan, and Jesper Jensen. Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(10):1901–1913, 2017.

Vinay Kothapally and John HL Hansen. Complex-valued time-frequency self-attention for speech dereverberation. *arXiv preprint arXiv:2211.12632*, 2022a.

Vinay Kothapally and John HL Hansen. Skipconvgan: Monaural speech dereverberation using generative adversarial networks via complex time-frequency masking. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:1600–1613, 2022b.

Anurag Kumar and Dinei Florencio. Speech enhancement in multiple-noise conditions using deep neural networks. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 08-12-September-2016:3738–3742, 2016. ISSN 19909772. doi: 10.21437/Interspeech.2016-88.

Joo Hyun Lee, Joon Hyuk Chang, Jae Mo Yang, and Han Gil Moon. NAS-TasNet: Neural Architecture Search for Time-Domain Speech Separation. *IEEE Access*, 10:56031–56043, 2022. ISSN 21693536. doi: 10.1109/ACCESS.2022.3176003.

Hao Li, Xueliang Zhang, Hui Zhang, and Guanglai Gao. Integrated speech enhancement method based on weighted prediction error and dnn for dereverberation and denoising. *arXiv preprint arXiv:1708.08251*, 2017.

Haohe Liu, Xubo Liu, Qiuqiang Kong, Qiao Tian, Yan Zhao, DeLiang Wang, Chuanzeng Huang, and Yuxuan Wang. Voicefixer: A unified framework for high-fidelity speech restoration. *arXiv preprint arXiv:2204.05841*, 2022.

Yuzhou Liu and Deliang Wang. Divide and Conquer: A Deep CASA Approach to Talker-Independent Monaural Speaker Separation. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 27(12):2092–2102, 2019a. ISSN 23299304. doi: 10.1109/TASLP.2019.2941148.

Yuzhou Liu and DeLiang Wang. Divide and conquer: A deep casa approach to talker-independent monaural speaker separation. *IEEE/ACM Transactions on audio, speech, and language processing*, 27(12):2092–2102, 2019b.

Yi Luo and Nima Mesgarani. TaSNet: Time-Domain Audio Separation Network for Real-Time, Single-Channel Speech Separation. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2018-April:696–700, 2018. ISSN 15206149. doi: 10.1109/ICASSP.2018.8462116.

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, 2019a. ISSN 23299304. doi: 10.1109/TASLP.2019.2915167.

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, 2019b.

Yi Luo, Zhuo Chen, and Takuya Yoshioka. Dual-Path RNN: Efficient Long Sequence Modeling for Time-Domain Single-Channel Speech Separation. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2020-May:46–50, 2020. ISSN 15206149. doi: 10.1109/ICASSP40776.2020.9054266.

Matthew Maciejewski, Gordon Wichern, Emmett McQuinn, and Jonathan Le Roux. Whamr!: Noisy and reverberant single-channel speech separation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 696–700. IEEE, 2020.

Emmanuel Müller. Graph clustering with graph neural networks. *Journal of Machine Learning Research*, 24:1–21, 2023.

Eliya Nachmani, Yossi Adi, and Lior Wolf. Voice separation with an unknown number of multiple speakers. *37th International Conference on Machine Learning, ICML 2020*, PartF16814:7121–7132, 2020.

Mark EJ Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical review E*, 74(3):036104, 2006.

Soha A Nossier, Julie Wall, Mansour Moniri, Cornelius Glackin, and Nigel Cannings. A Comparative Study of Time and Frequency Domain Approaches to Deep Learning based Speech Enhancement. In *Proceedings of the International Joint Conference on Neural Networks*, 2020. ISBN 9781728169262. doi: 10.1109/IJCNN48605.2020.9206928.

Kuldip Paliwal, Kamil Wójcicki, and Benjamin Shannon. The importance of phase in speech enhancement. *Speech Communication*, 53(4):465–494, 2011. ISSN 01676393. doi: 10.1016/j.specom.2010.12.003.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 5206–5210. IEEE, 2015.

Douglas B Paul and Janet Baker. The design for the wall street journal-based csr corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992.

Shan Qin, Ting Jiang, Sheng Wu, Ning Wang, and Xinran Zhao. Graph convolution-based deep clustering for speech separation. *IEEE Access*, 8:82571–82580, 2020. ISSN 21693536. doi: 10.1109/ACCESS.2020.2989833.

Chandan KA Reddy, Vishak Gopal, and Ross Cutler. Dnsmos: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6493–6497. IEEE, 2021.

Chandan KA Reddy, Vishak Gopal, and Ross Cutler. Dnsmos p. 835: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 886–890. IEEE, 2022.

A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra. Perceptual evaluation of speech quality (PESQ) - A new method for speech quality assessment of telephone networks and codecs. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2:749–752, 2001. ISSN 15206149. doi: 10.1109/icassp.2001.941023.

Aaqib Saeed, David Grangier, and Neil Zeghidour. Contrastive learning of general-purpose audio representations. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3875–3879. IEEE, 2021.

Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pp. 464–472. IEEE, 2017.

Cem Subakan, Mirco Ravanelli, Samuele Cornell, Mirko Bronzi, and Jianyuan Zhong. Attention is all you need in speech separation. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2021-June:21–25, 2021a. ISSN 15206149. doi: 10.1109/ICASSP39728.2021.9413901.

Cem Subakan, Mirco Ravanelli, Samuele Cornell, Mirko Bronzi, and Jianyuan Zhong. Attention is all you need in speech separation. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 21–25. IEEE, 2021b.

Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time-frequency weighted noisy speech. *IEEE Transactions on Audio, Speech and Language Processing*, 19(7):2125–2136, 2011. ISSN 15587916. doi: 10.1109/TASL.2011.2114881.

Hideyuki Tachibana. Towards listening to 10 people simultaneously: An efficient permutation invariant training of audio source separation using Sinkhorn's algorithm. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2021-June:491–495, 2021. ISSN 15206149. doi: 10.1109/ICASSP39728.2021.9414508.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pp. 6105–6114. PMLR, 2019.

Yanhui Tu, Jun Du, Yong Xu, Lirong Dai, and Chin Hui Lee. Deep neural network based speech separation for robust speech recognition. *International Conference on Signal Processing Proceedings, ICSP*, 2015-January(October):532–536, 2014. doi: 10.1109/ICOSP.2014.7015061.

Yuma Ueda, Longbiao Wang, Atsuhiko Kai, Xiong Xiao, Eng Siong Chng, and Haizhou Li. Single-channel Dereverberation for Distant-Talking Speech Recognition by Combining Denoising Autoencoder and Temporal Structure Normalization. *Journal of Signal Processing Systems*, 82(2):151–161, 2016. ISSN 19398115. doi: 10.1007/s11265-015-1007-3.

Yannan Wang, Jun Du, Li-Rong Dai, and Chin-Hui Lee. Unsupervised single-channel speech separation via deep neural network for different gender mixtures. In *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1–4. IEEE, 2016.

Yannan Wang, Jun Du, Li Rong Dai, and Chin Hui Lee. A gender mixture detection approach to unsupervised single-channel speech separation based on deep neural networks. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 25(7):1535–1546, 2017. ISSN 23299290. doi: 10.1109/TASLP.2017.2700540.

Yuxuan Wang and De Liang Wang. Towards scaling up classification-based speech separation. *IEEE Transactions on Audio, Speech and Language Processing*, 21(7):1381–1390, 2013. ISSN 15587916. doi: 10.1109/TASL.2013.2250961.

Zhong Qiu Wang, Jonathan Le Roux, and John R. Hershey. Alternative objective functions for deep clustering. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2018-April:686–690, 2018. ISSN 15206149. doi: 10.1109/ICASSP.2018.8462507.

Chao Weng, Dong Yu, Michael L. Seltzer, and Jasha Droppo. Deep neural networks for single-channel multi-talker speech recognition. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 23 (10):1670–1679, 2015. ISSN 23299290. doi: 10.1109/TASLP.2015.2444659.

Felix Weninger, John R. Hershey, Jonathan Le Roux, and Bjorn Schuller. Discriminatively trained recurrent neural networks for single-channel speech separation. *2014 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2014*, pp. 577–581, 2014. doi: 10.1109/GlobalSIP.2014.7032183.

Gordon Wichern, Joe Antognini, Michael Flynn, Licheng Richard Zhu, Emmett McQuinn, Dwight Crow, Ethan Manilow, and Jonathan Le Roux. Wham!: Extending speech separation to noisy environments. *arXiv preprint arXiv:1907.01160*, 2019.

Donald S Williamson and DeLiang Wang. Time-frequency masking in the complex domain for speech dereverberation and denoising. *IEEE/ACM transactions on audio, speech, and language processing*, 25(7): 1492–1501, 2017.

Yong Xu, Jun Du, Li Rong Dai, and Chin Hui Lee. An experimental study on speech enhancement based on deep neural networks. *IEEE Signal Processing Letters*, 21(1):65–68, 2014. ISSN 10709908. doi: 10.1109/LSP.2013.2291240.

Yong Xu, Jun Du, Li Rong Dai, and Chin Hui Lee. A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 23(1):7–19, 2015. ISSN 23299290. doi: 10.1109/TASLP.2014.2364452.

Jaewon Yang and Jure Leskovec. Defining and evaluating network communities based on ground-truth. In *Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics*, pp. 1–8, 2012.

Dong Yu, Morten Kolbaek, Zheng Hua Tan, and Jesper Jensen. Permutation invariant training of deep models for speaker-independent multi-talker speech separation. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 241–245, 2017. ISSN 15206149. doi: 10.1109/ICASSP.2017.7952154.

Neil Zeghidour and David Grangier. Wavesplit: End-to-End Speech Separation by Speaker Clustering. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 29(iv):2840–2849, 2021a. ISSN 23299304. doi: 10.1109/TASLP.2021.3099291.

Neil Zeghidour and David Grangier. Wavesplit: End-to-end speech separation by speaker clustering. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2840–2849, 2021b.

Xiao Lei Zhang and Deliang Wang. A deep ensemble learning method for monaural speech separation. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 24(5):967–977, 2016. ISSN 23299290. doi: 10.1109/TASLP.2016.2536478.

Impact Statement

"This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

A Appendix