DEEP ATTRACTOR NETWORKS FOR SPEAKER RE-IDENTIFICATION AND BLIND SOURCE SEPARATION

Lukas Drude, Thilo von Neumann, Reinhold Haeb-Umbach

Paderborn University, Department of Communications Engineering, Paderborn, Germany

ABSTRACT

Deep clustering (DC) and deep attractor networks (DANs) are a data-driven way to monaural blind source separation. Both approaches provide astonishing single channel performance but have not yet been generalized to block-online processing. When separating speech in a continuous stream with a block-online algorithm, it needs to be determined in each block which of the output streams belongs to whom. In this contribution we solve this block permutation problem by introducing an additional speaker identification embedding to the DAN model structure. We motivate this model decision by analyzing the embedding topology of DC and DANs and show, that DC and DANs themselves are not sufficient for speaker identification. This model structure (a) improves the signal to distortion ratio (SDR) over a DAN baseline and (b) provides up to $61\,\%$ and up to $34\,\%$ relative reduction in permutation error rate and re-identification error rate compared to an i-vector baseline, respectively.

Index Terms— Speech separation, monaural, speaker identification, multi-talker, embedding

1. INTRODUCTION

Recently, a variety of deep learning based monaural source separation systems have been proposed, which neither make strong assumptions on the number of sources, nor rely on training on the same set of speakers as used for testing.

DC [1] is a pioneering work using bidirectional long short term memory networks (BLSTMs) [2] as an encoder network to provide embeddings for source separation. Based on a matrix similarity loss, embeddings of the same speaker are encouraged to move closer together while embeddings of different speakers move further apart during the training process. The embedding vectors are then clustered using k-means to obtain binary masks for source separation. Subsequently, it was shown that the separation performance can be greatly improved by adding a separate network which takes the initial DC hard masks and provides masks which are then used for source separation [3]. This turned out to boost separation performance sufficiently to allow speech recognition on the separated streams. Although the DC embedding topology has not been investigated yet, it is assumed that the rank of the correlation matrix of the embedding vectors is related to the number of sources in the mixture [4].

A major change to the DC approach was put forward by DANs [5]: DANs allow to train the encoder network with a speech reconstruction criterion. This enabled end-to-end training while still requiring k-means at test time. In [5] the topology of the embedding space was analyzed using a PCA projection. The topology of the DAN projection suggested, that the attractor vectors (k-means centroids) can be kept fixed at test time, since the topology of any two-speaker mixture is fairly stable. Consequently, [6] evaluates, if fixed attractors can be used to decode two and three speaker mixtures.

Since DC and DANs are based on BLSTMs they are inherently not ready for online processing applications. Even if one resorts to block-online processing, it is not guaranteed, that a speaker always occurs on the same output channel: A block permutation problem arises [7]. Since the encoder network of DC and DANs produces an independent embedding space for each mixture, the attractors (k-means centroids) can not be used to trace the speakers. This problem can be addressed by using spatial information in a multi-channel setup. In a monaural scenario, one has to resort to speaker identification techniques. A traditional method is to use i-vectors [8] for matching speech parts of the same speaker.

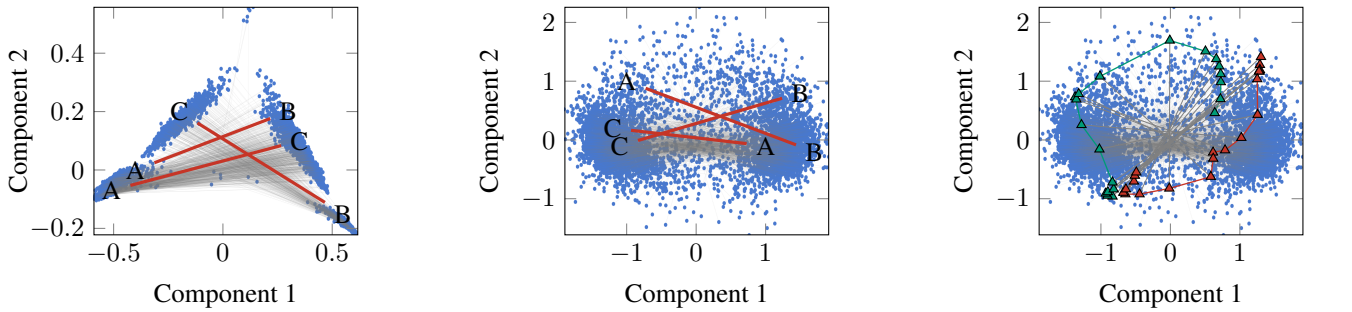
In this contribution, we will first analyze the embedding topology in Sec. 5 to better understand why speaker tracing is necessary. We will then gradually develop a DAN-based system in Sec. 6, which allows to re-identify speakers in consecutive blocks and even within other unrelated mixtures. Subsequently, we evaluate the proposed model and compare it with two baseline systems in Sec. 7.

2. SIGNAL MODEL

A single channel speech mixture y_{tf} is observed in the short time Fourier transform (STFT) domain, where t is the time frame index and f is the frequency bin index. It is assumed that the mixture is composed of K clean speech signals x_{ktf}:

$$y_{tf} = \sum_k x_{ktf} + n_{tf}, \tag{1}$$

where k is the source index. For simplicity and to comply with [1, 3, 5], the noise signal n_{tf} is assumed to be zero.



(a) DC: The location of highlighted speakers change from mixture to mixture.

(b) DAN: The location of highlighted speakers change from mixture to mixture.

(c) DAN: Attractor locations change depending on the mixing proportions.

Fig. 1: Principle component analysis (PCA) projection of the embedding space. Each dot represents a k-means centroid of all two speaker test mixtures. The thin gray lines indicate (for a few examples) which attractors belong to the same mixture.

3. DEEP CLUSTERING

In DC an encoder network consists of BLSTM layers and a linear layer. It consumes the log power spectrum of a mixture signal and produces an E dimensional embedding $\mathbf{e}_{tf}$ for each time frequency point. The encoder network is trained to minimize the Frobenius norm of the difference between the estimated and true affinity matrix [1]. At test time, a k-means algorithm clusters the embedding vectors into K binary masks, which can be used to extract each source signal.

4. DEEP ATTRACTOR NETWORK

The encoder network is conceptually equal to the DC encoder network. Attractors are calculated by a weighted sum of all embedding vectors in the mixture:

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{tf} M^{\text{oracle}}_{ktf} \mathbf{e}_{tf}, \quad N_k = \sum_{tf} M^{\text{oracle}}_{ktf}, \tag{2}$$

where M^{oracle}_{ktf} is the supervision ideal binary mask at training time. The separation mask is then obtained by calculating the inner product between the attractor $\boldsymbol{\mu}_k$ and the embedding vector $\mathbf{e}_{tf}$. Instead of using the DC loss a mean squared error (MSE) signal reconstruction loss can be used to train the encoder network:

$$\ell_{\text{MSE}} = \underset{ktf}{\text{MSE}} \left(\hat{x}_{ktf}, x_{ktf} \right), \tag{3}$$

$$\hat{x}_{ktf} = \hat{M}_{ktf} y_{tf}, \quad \hat{M}_{ktf} = \underset{k}{\text{softmax}}\, \boldsymbol{\mu}_k^{\mathsf{H}} \mathbf{e}_{tf}. \tag{4}$$

5. INSIGHTS INTO THE EMBEDDING TOPOLOGY

The topology analysis is based on networks trained on the MERL database [1]. It consists of mixture file lists referencing clean WSJ utterances [9]. More details regarding the evaluation can be found in Sec. 7. The E dimensional embedding space can be visualized by a projection method. We opted for a PCA projection since it offered easier interpretability than a t-SNE projection [10] in the given context.

Fig. 1a shows the k-means centroids for DC embeddings of all test mixtures in the MERL database. It can be observed that the centroids themselves form four distinguishable clusters, although the network was trained on two speaker mixtures and the training database contains 101 different speakers. At least judged from the first principal components the location of the cluster centroids may only remotely relate to the true speaker label.

Fig. 1b and Fig. 1c show the attractors (k-means centroids) of a DAN trained on two speaker mixtures. Roughly two independent clusters of attractors can be identified. Fig. 1b highlights the attractor location of three speakers in three different mixtures. If speaker A, B and C can be found in two speaker mixtures, at least one of the speaker attractors must be found in both clusters. Fig. 1c emphasizes the attractor locations of selected two speaker mixtures. Locations belonging to the same speaker are connected with line segments indicating that mixture weight changes result in small changes in the embedding space. The mixing weights of a given mixture were changed from $-10\,\text{dB}$ to $10\,\text{dB}$. One can observe that the location of the attractor vectors highly depend on the mixture weight. In some cases, a change of the mixture weight even changes the cluster an attractor is associated to. Again this leads to the conclusion that the k-means centroids (or attractors) can not be easily used for speaker identification or tracing.

Moreover, the topology of the DAN attractors exhibits, why fixed attractors can be used during test time: Since the attractors form two major attractor-clusters, the attractor-cluster centers can then be used as fixed attractors and provide a fairly good guess for unseen mixtures. This is, however only possible, if the number of speakers is known at training time, neutralizing one of the benefits of DANs. In [6] the idea of more fixed attractors than speakers is evaluated, allowing applicability to a different number of speakers at test time than at training time.

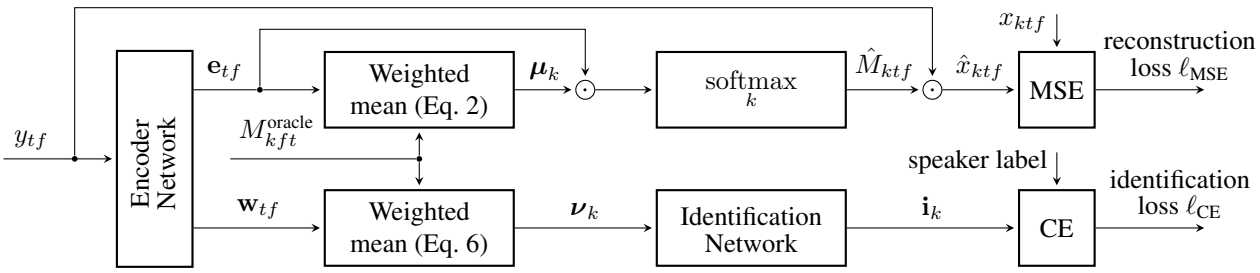


Fig. 2: Training procedure of a DAN with additional speaker identification embedding $\mathbf{w}_{tf}$ and speaker identification loss. At test time, the oracle mask M^{oracle}_{kft} is replaced by a mask obtained using k-means clustering of the embedding $\mathbf{e}_{tf}$.

6. PROPOSED METHOD

We propose a method to solve the block permutation problem and find a previously found speaker in an open test set. We explore two approaches: First, we add an additional speaker identification loss. Second, we model speaker identity with additional speaker embeddings. The reconstruction loss and separation procedure of DANs remains unchanged.

6.1. Speaker identification loss

At training time a shallow feed-forward network with a softmax output nonlinearity can be used to predict speaker labels from the attractor vectors. That way, the attractors tend to capture speaker information which can be used at test time to re-identify speakers even in an open speaker test set. An additional cross-entropy loss ℓ_{CE} is then used for training:

$$\ell_{\text{total}} = \ell_{\text{MSE}} + \alpha\, \ell_{\text{CE}}, \tag{5}$$

where α is a weighting factor to adjust the trade-off between both losses. In general, multi-task training has proven to be advantageous [11, 12]. Adding an auxiliary loss to the main loss of interest often improved the performance of the primary output. In case of a related music separation task with DC, the overall separation performance improved due to the additional loss [13]. However, an auxiliary loss may also deteriorate performance, if separation and identification information encoded into the same embedding vector is contradictory.

6.2. Speaker identification embedding

Instead of producing just the embedding vectors $\mathbf{e}_{tf}$ the encoder network can be modified to output a separate set of embedding vectors $\mathbf{w}_{tf}$ for the purpose of speaker identification. To do so, the number of units in the output layer of the encoder network is doubled. This is a marginal increase in network complexity, since the main time is spend in the BLSTM layers. At training time, speaker identification attractors can then be calculated similar to Eq. 2:

$$\boldsymbol{\nu}_k = \frac{1}{N_k} \sum_{tf} M^{\text{oracle}}_{ktf} \mathbf{w}_{tf}, \quad N_k = \sum_{tf} M^{\text{oracle}}_{ktf}, \tag{6}$$

The encoder network is then trained with a reconstruction loss on the separation embeddings (top path in Fig. 2) and a speaker identification loss on top of a shallow network on the speaker identification attractors (bottom path in Fig. 2) in contrast to Sec. 6.1.

6.3. Solving the permutation problem

The block permutation problem arises in block-online processing. The speaker index may change from block to block rendering it necessary to trace a speaker in consecutive blocks – similar as argued for frames in [14]. If speaker identification attractors $\boldsymbol{\nu}_k$ are available (Sec. 6.2), the permutation of a block can be obtained by selecting the permutation with the minimal mean Euclidean distance to the previous $\boldsymbol{\nu}_k$.

6.4. Speaker identification in unseen mixtures

In contrast to the permutation problem as in Sec. 6.3 another task is to identify a target speaker in a mixture with other unknown speakers. This is useful, if you already found a speaker and you are trying to extract the same speaker from other meetings. To identify which output stream belongs to the target speaker, attractors (or speaker attractors) are extracted from a given mixture and compared with the reference attractor from a previously decoded test mixture: The output stream with the minimal Euclidean distance to the reference is selected.

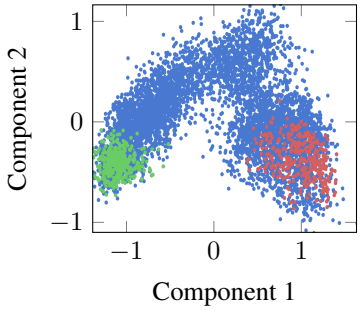


Fig. 3: Speaker identification attractors $\boldsymbol{\nu}_k$ of the proposed DAN variant. Two speakers are highlighted in green and red.

7. EVALUATION

Based on the file lists provided with [1] 20 k, 5 k and 3 k single channel mixtures of 8 kHz WSJ [9] utterances were created for training, cross-validation and test. Sets contained 101, 101 and 18 speakers, respectively. All presented results are on the test set with open speaker conditions (speakers not seen during training).

Features are extracted using an STFT with size and shift of 512 and 128, respectively. The encoding network consists of two BLSTM layers with 600 forward and 600 backward units followed by a linear layer to map to a $E = 20$ dimensional embedding space ($E = 40$ for DC) for each time frequency slot. Forward and backward streams are concatenated for the next layer. In case of the DAN with speaker attractors, the output dimension is $2E$.

Tbl. 1 shows the source separation performance in terms of SDR, signal to interference ratio (SIR) and signal to artifact ratio (SAR) values [15, 16]. The SDR for the DAN matches the results in the reference paper [5]. An additional speaker identification loss (Sec. 6.1) improves separation performance for moderately high loss weight α, indicating that multi-task learning is helpful here. Similarly, when additional speaker identification attractors $\boldsymbol{\nu}_k$ are used (Sec. 6.2), the additional loss improves the source separation slightly, too. Fig. 3 shows a PCA projection of all identification attractors $\boldsymbol{\nu}_k$ of all test mixtures. The speaker attractors of two speakers are highlighted and form a fairly dense cluster, each. This indicates, that the identification attractors are much more informative of speaker identity than the attractors as in Fig. 1b.

To investigate how well the proposed identification attractors $\boldsymbol{\nu}_k$ can be used to re-identify a speaker, we analyze two different tasks in Tbl. 2. The permutation accuracy shows how well two output channels can be matched to the correct speakers based on a reference mixture as in Sec. 6.3. It can be seen that DC and DAN already show some stability regarding the attractor location $\boldsymbol{\mu}_k$. Therefore, DC and DAN already work fairly well on the permutation task. However, the proposed model with separate speaker identification attractors $\boldsymbol{\nu}_k$ outperforms all other systems.

The identification accuracy measures, how well a single speaker of a test mixture can be found in all other test mixtures by finding the minimum distance (see Sec. 6.4). Now, the error rate of the vanilla DAN is much higher, which is supported by the highlighted speaker example in Fig. 1b. The proposed DAN with separate speaker identification attractors $\boldsymbol{\nu}_k$ provides the lowest error rate of all tested systems.

Commonly, i-vectors are used for speaker identification. Thus, we trained an i-vector extractor using Kaldi [17] and used the cosine distance for all experiments. If it is trained on WSJ clean speech, it performs poorly on blindly separated test data. However, if the i-vector extractor is trained on the output channels separated by a DAN (i.e. when the extractor is aware of artifacts), a much lower error rate is reached at the cost of a more complicated pipeline. If an additional voice activity detection (VAD) to guide the i-vector extractor is used, an error rate not much worse than the proposed system is possible as reported in Tbl. 2.

Table 1: Influence of an additional speaker identification loss weight α on the source separation performance.

	α	SDR/dB	SIR/dB	SAR/dB
DAN		9.4	16.7	10.8
DAN + ID loss	0.001	10.1	17.4	11.4
	0.01	9.9	17.2	11.2
	0.1	9.9	17.2	11.2
	1	9.7	17.1	11.0
	10	8.9	16.0	10.3
DAN + ID emb.	0.001	9.9	17.3	11.2
	0.01	9.8	16.8	11.3
	0.1	10.0	17.4	11.3
	1	10.1	17.1	11.5
	10	9.2	16.4	10.6

8. CONCLUSIONS

This contribution demonstrates, how speaker information can be extracted with a DAN to allow speaker tracing for a block-online processing setup and identify speakers in unseen test mixtures. Additional speaker identification embeddings can be extracted with a minimal increase of computational complexity. In comparison to an i-vector baseline a 61 % relative permutation error rate reduction and a 34 % relative re-identification error rate reduction is achieved, while improving separation performance up to 0.7 dB.

Table 2: Permutation error rate to correctly trace speakers in block-online processing. Identification error rate to find a speaker in an unrelated mixture. Baseline systems are in gray.

Error Rate / %:	α	Permutation	Identification
Chance level		50.0	50.0
i-vector with VAD		8.0	9.7
DC		7.3	33.4
DAN		5.8	31.5
DAN + ID loss	0.001	6.7	32.7
	0.01	6.0	31.1
	1	5.0	20.1
	10	4.0	9.3
DAN + ID emb.	0.001	4.7	9.9
	0.01	3.7	7.7
	1	4.2	8.5
	10	3.1	6.4

9. REFERENCES

[1] J. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep Clustering: Discriminative embeddings for segmentation and separation," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[2] M. Schuster and K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[3] Y. Isik, J. Le Roux, Z. Chen, S. Watanabe, and J. Hershey, "Single-channel multi-speaker separation using Deep Clustering," *Interspeech*, 2016.

[4] T. Higuchi, K. Kinoshita, M. Delcroix, K. Zmolikova, and T. Nakatani, "Deep Clustering-based beamforming for separation with unknown number of sources," *Interspeech*, 2017.

[5] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[6] Z. Chen, Y. Luo, and N. Mesgarani, "Speaker-independent speech separation with deep attractor network," *arXiv preprint arXiv:1707.03634*, 2017.

[7] Y. Shao and D. Wang, "Model-based sequential organization in cochannel speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 289–298, 2006.

[8] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[9] J. Garofolo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) Complete LDC93S6A," *Philadelphia: Linguistic Data Consortium*, 1993.

[10] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[11] R. Caruana, "Multitask learning," *Learning to learn*, pp. 95–133, 1998.

[12] S. Thrun, "Is learning the n-th thing any easier than learning the first?," *Advances in neural information processing systems*, pp. 640–646, 1996.

[13] Y. Luo, Z. Chen, J. Hershey, J. Le Roux, and N. Mesgarani, "Deep Clustering and conventional networks for music separation: Stronger together," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[14] M. Kolbæk, D. Yu, Z. Tan, and J. Jensen, "Multitalker 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.

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

[16] C. Raffel, B. McFee, E. Humphrey, J. Salamon, O. Nieto, D. Liang, and D. Ellis, "mir_eval: A transparent implementation of common MIR metrics," *International Society for Music Information Retrieval Conference, (ISMIR)*, 2014.

[17] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi Speech Recognition Toolkit," *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2011.