# Part IV.
# Source Separation

**Reinhold Haeb-Umbach**

# Problem description


- Known as cocktail party problem [Cherry, 1953]
- Distinguishing speech of different speakers is more difficult than separating speech from noise
- Long history of research

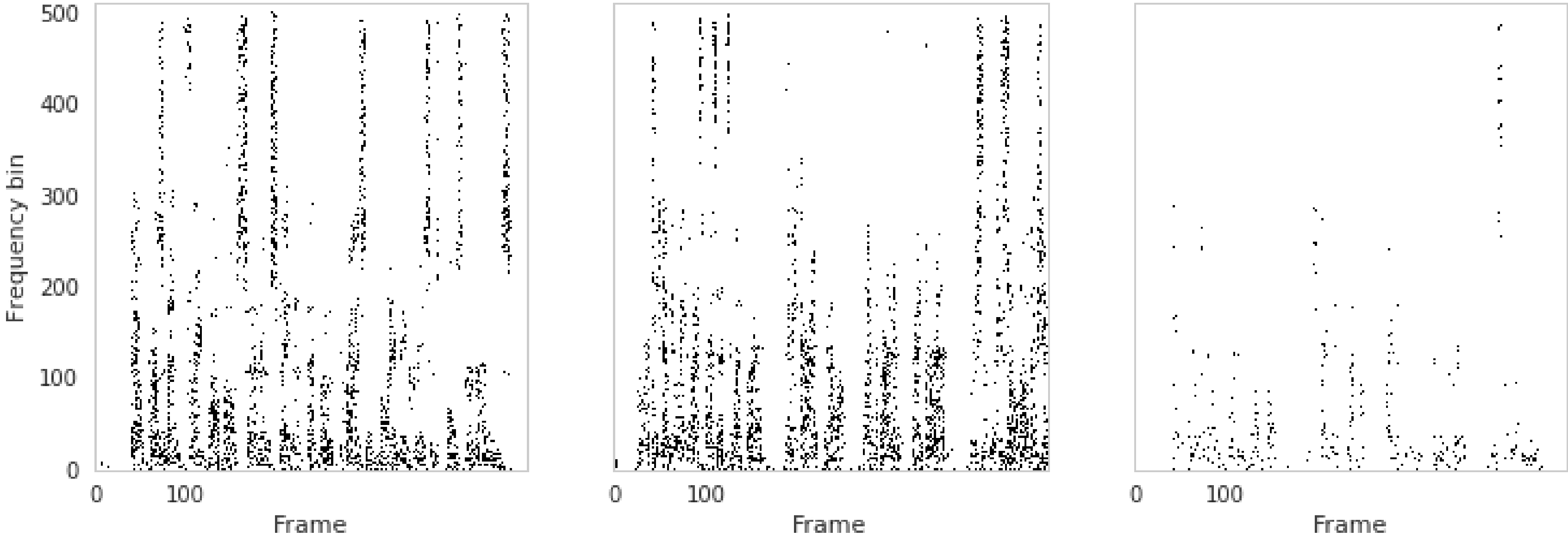
# Table of contents in part IV

# Blind Source Separation: Taxonomy of Approaches

- ICA (Independent Component Analysis) based
  - Assumption: mutual independence of sources and one or more of the following
    - Non-Gaussianity, non-whiteness, non-stationarity
  - Requires #sensors ≥ #sources
- Sparseness based
  - Assumption: in an appropriate domain, each source does not occupy the whole space, e.g, time-frequency sparseness of speech
  - #sensors can be smaller than #sources
- NMF (Non-negative Matrix Factorization) based
  - Assumption: sources are non-negative and mixing system is additive; sources have low rank
  - Originally single-channel approach, has been extended to multi-channel
- And combinations / variants of them: IVA, ILRMA, IDLMA, …

# Here: Blind Speech Separation

- Sparseness based approaches are particularly effective
  - Sparseness of speech in the time-frequency (STFT) domain [Yilmaz and Rickard, 2004]
    - 90% of the speech power is concentrated in 10% of the tf-bins
    - Different speakers populate different tf-bins




Spkr #1 Spkr #2 (Spkr #1) ⊙ (Spkr #2)

# BLIND speech separation

### Supervised / Guided

- Known mixing system
  - Speaker location
  - Array geometry
  - Acoustic transfer function
- Known diarization
  - On/offset times of speakers
- Known speakers

### Blind

- Unknown mixing system
  - Unknown spkr location
  - Unknown array geometry
  - Unknown acoustic transfer function
- Unknown diarization
  - Unknown on/offset times
- Unknown speakers
  - Speaker-independent source separation

# Model in STFT domain

- Narrowband assumption
(length of acoustic impulse response << STFT analysis window):

$$\mathbf{y}_{t,f} = \sum_{i=1}^{I} \mathbf{a}_f^{(i)} s_{t,f}^{(i)} + \mathbf{n}_{t,f} =: \sum_{i=1}^{I} \mathbf{x}_{t,f}^{(i)} + \mathbf{n}_{t,f}$$

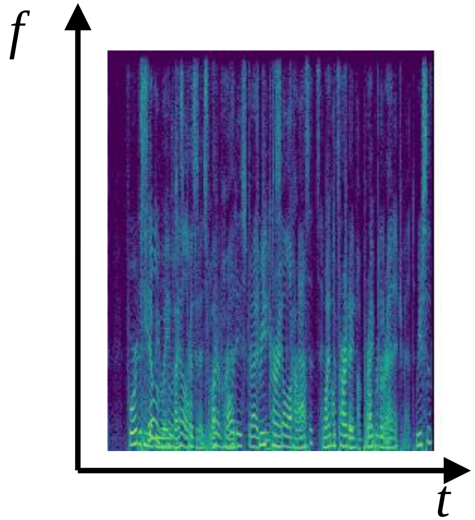
- Often, noise is neglected or treated as an additional source:

$$\mathbf{y}_{t,f} = \sum_{i=1}^{I} \mathbf{x}_{t,f}^{(i)}; \quad \mathbf{y}_{t,f} = \sum_{i=0}^{I} \mathbf{x}_{t,f}^{(i)}$$

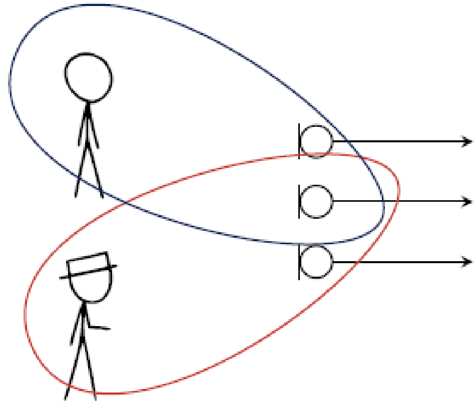
- Our goal is to reconstruct the images of the source signals at a reference microphone (e.g. mic #1):

$$x_{1,t,f}^{(i)};\ i = 1, \ldots I$$

# Separation cues: spectro-temporal vs spatial

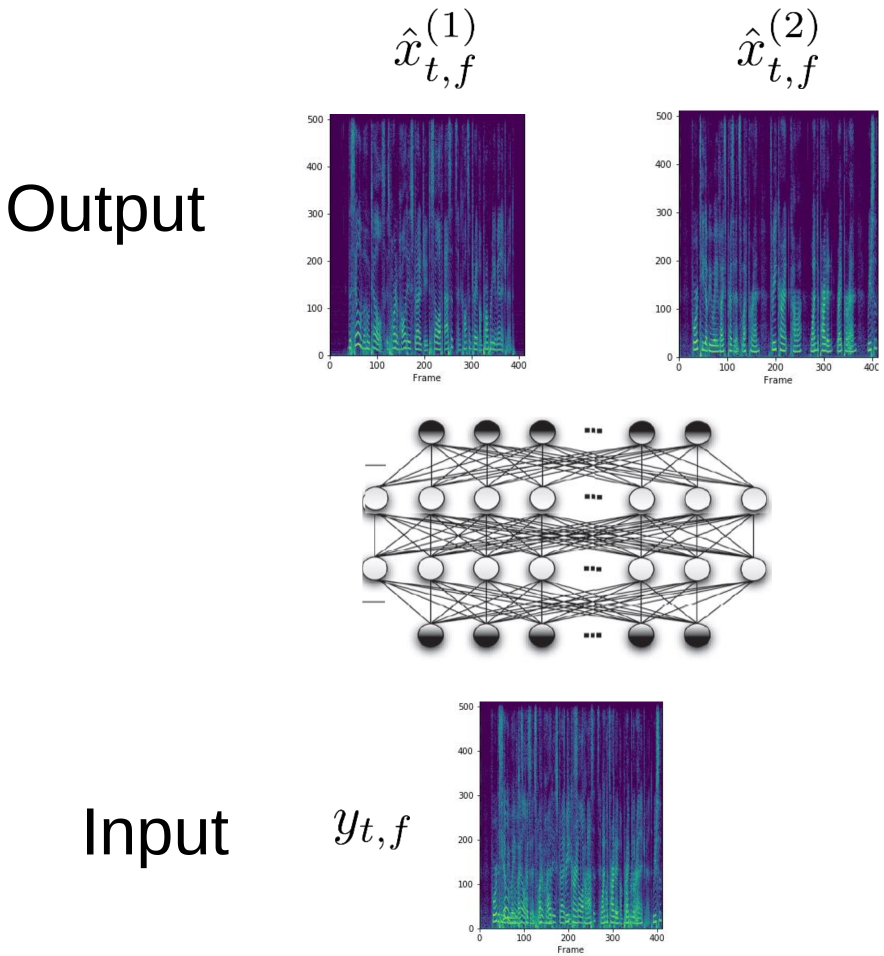


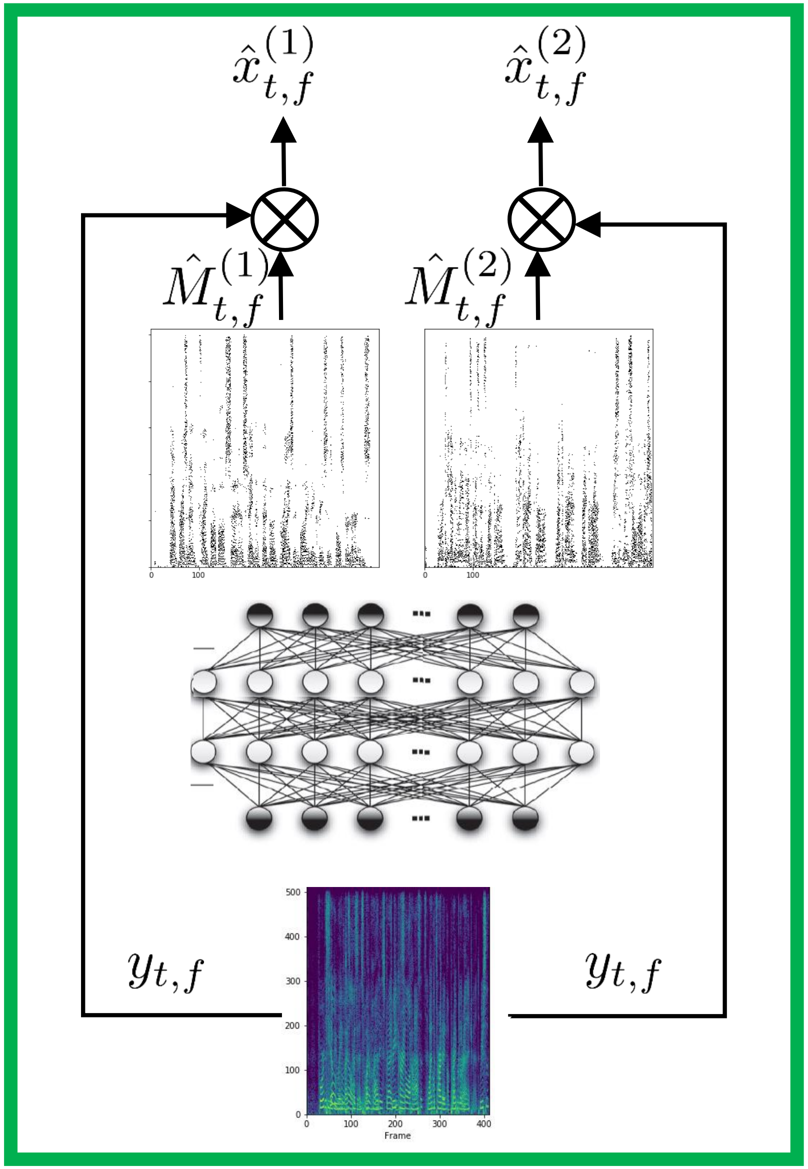

- Spectro-temporal cues
  - Model speech characteristics
  - Can work with **single-channel input**
  - Leverage training data
  - Typically supervised trng
  - DNN based



- Spatial cues
  - Exploits spatial selectivity
  - Requires **multi-channel input**
  - Does not require trng phase
  - Unsupervised learning (EM alg.)
  - Spatial mixture model based

# Spectra vs masks as training targets




Mask based extraction performs better than direct signal estimation

# Mask estimation

- Predict, for each tf-bin, the presence/absence of a target speaker
- Two types of objective functions
  - Mask approximation, e.g., cross entropy between estimated and ground truth mask
    - Appropriate if we do not need a decision for every tf bin
    - See spatial covariance matrix estimation in beamforming section
    - Does not measure reconstruction error
  - Signal approximation:

$$J(\theta) = \sum_{i,t,f} \left| \hat{x}_{t,f}^{(i)}(\theta) - x_{t,f}^{(i)} \right|^2 = \sum_{i,t,f} \left| \hat{M}_{t,f}^{(i)}(\theta) y_{t,f} - x_{t,f}^{(i)} \right|^2$$

    - Now, the training objective is the reconstruction error

Signal approximation performs better than mask approximation

# Masks for signal approximation

$$J(\theta) = \sum_{i,t,f} \left| \hat{x}^{(i)}_{t,f}(\theta) - x^{(i)}_{t,f} \right|^2 = \sum_{i,t,f} \left| \hat{M}^{(i)}_{t,f}(\theta) y_{t,f} - x^{(i)}_{t,f} \right|^2$$

- The optimal mask for the above trng objective is the ideal complex mask $$M^{(i)}_{t,f} = \frac{x^{(i)}_{t,f}}{y_{t,f}}$$
  - But phase estimation is tricky …
- To avoid  phase estimation, use best <u>real-valued</u> approximation to it: *ideal phase-sensitive mask* [Erdogan et al., 2015]

$$M^{(i)}_{t,f} = \Re\left\{ \frac{x^{(i)}_{t,f}}{y_{t,f}} \right\} = \frac{|x^{(i)}_{t,f}|}{|y_{t,f}|} \cos\left[ \varphi^{(x^{(i)})}_{t,f} - \varphi^{(y)}_{t,f} \right]$$

  - Thus trng objective fu:

$$\left| \hat{M}^{(i)}_{t,f} y_{t,f} - x^{(i)}_{t,f} \right|^2 \propto \left( \hat{M}^{(i)}_{t,f} |y_{t,f}| - |x^{(i)}_{t,f}| \cos\left[ \varphi^{(x^{(i)})}_{t,f} - \varphi^{(y)}_{t,f} \right] \right)^2$$

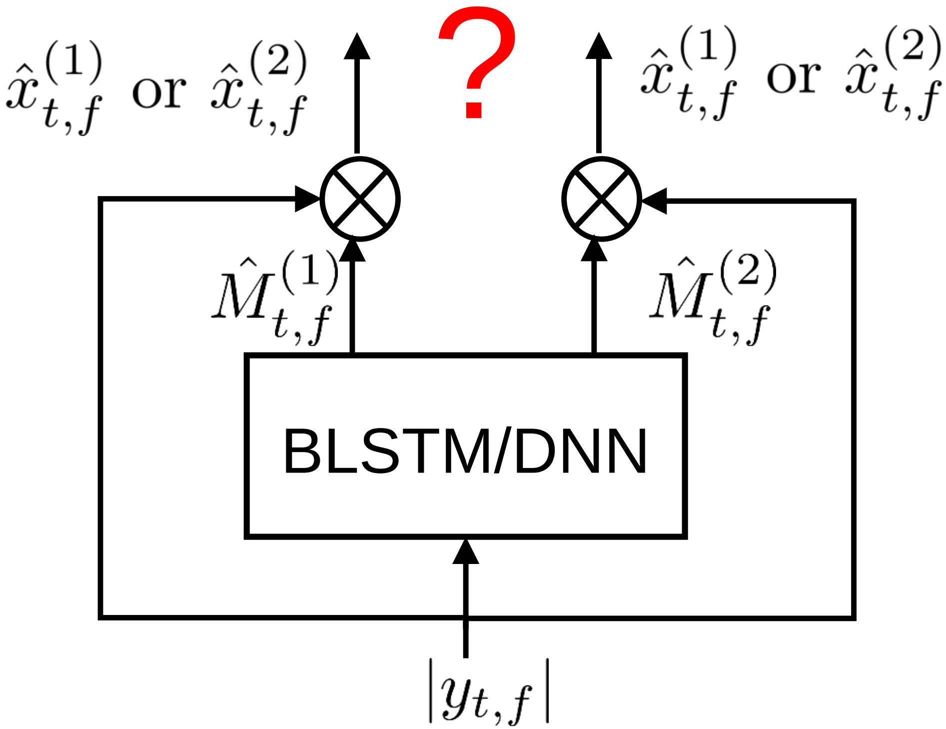
This trng objective has consistently shown better results than Ideal Binary Mask, Ideal Ratio Mask, etc. [Erdogan et al., 2015] [Kolbæk et al., 2017b]

# DNN-based single-channel BSS

- Permutation Invariant Training (PIT)
- Deep Clustering (DC)
- Time Domain Audio Separation Network (Tasnet)

# Utterance-PIT [Kolbæk et al., 2017b]

- Label ambiguity:



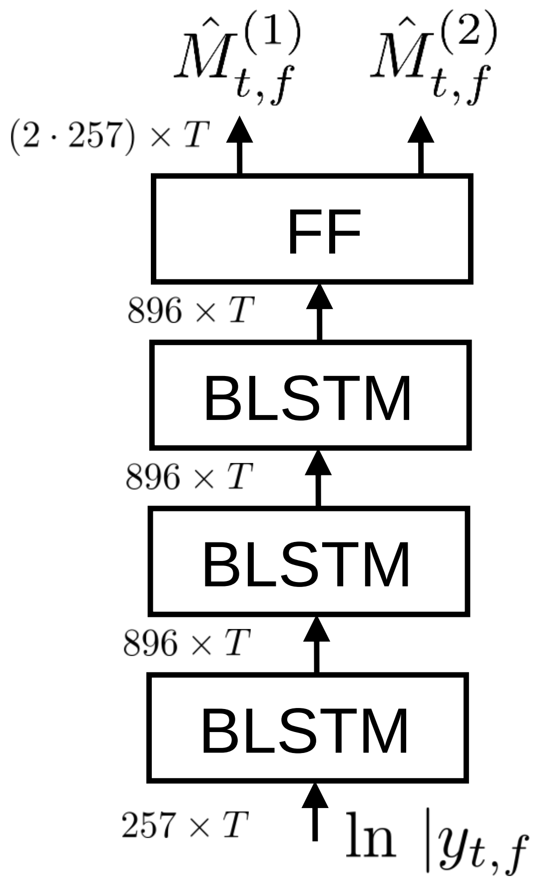

- Compute all permutations between the targets and the estimated sources and find permutation $\phi$ (over whole utterance) which minimizes MSE

$$J = \min_{\phi \in \mathcal{P}} \sum_{i,t,f} \left| \hat{M}^{(i)}_{t,f} y_{t,f} - x^{(\phi(i))}_{t,f} \right|^2$$

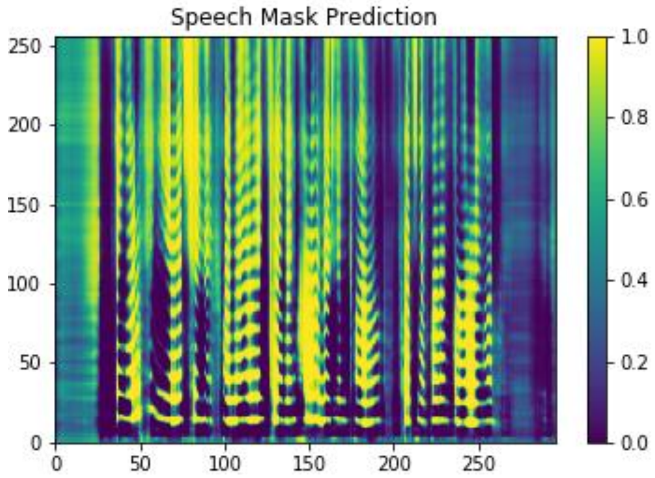
E.g.: $\min \left[ \sum_{t,f} \left\{ \left| \hat{M}^{(1)} y - x^{(1)} \right|^2 + \left| \hat{M}^{(2)} y - x^{(2)} \right|^2 \right\}; \sum_{t,f} \left\{ \left| \hat{M}^{(1)} y - x^{(2)} \right|^2 + \left| \hat{M}^{(2)} y - x^{(1)} \right|^2 \right\} \right]$

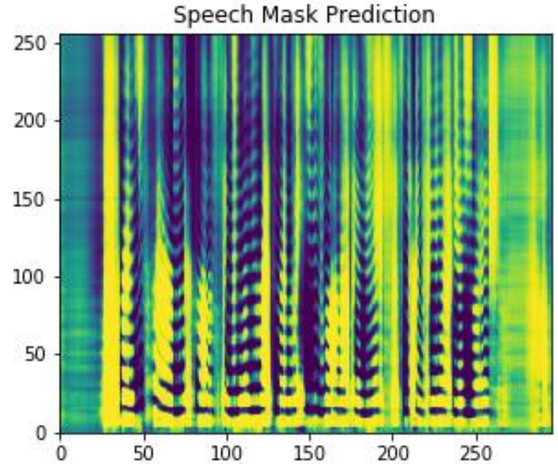
# Example configuration

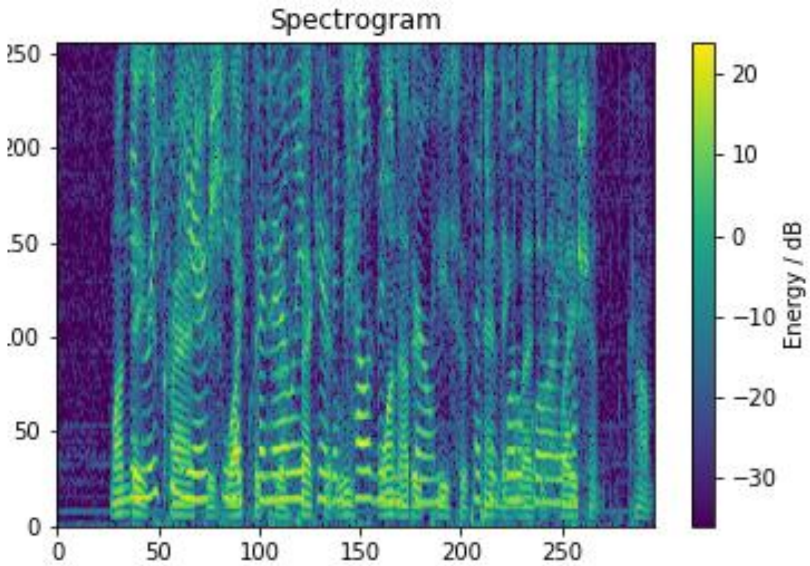
- Example configuration
  - Sampling rate 8 kHz; STFT window size: 64 ms; advance: 16 ms
  - Input: log-spectral magnitude features
  - 3 BLSTM layers with 896 nodes each
  - 1 FF layer with *(I x F)* nodes: I: #spkrs; F: #freq.bins (e.g., *I=2, F=257*); sigmoid output nonlinearity

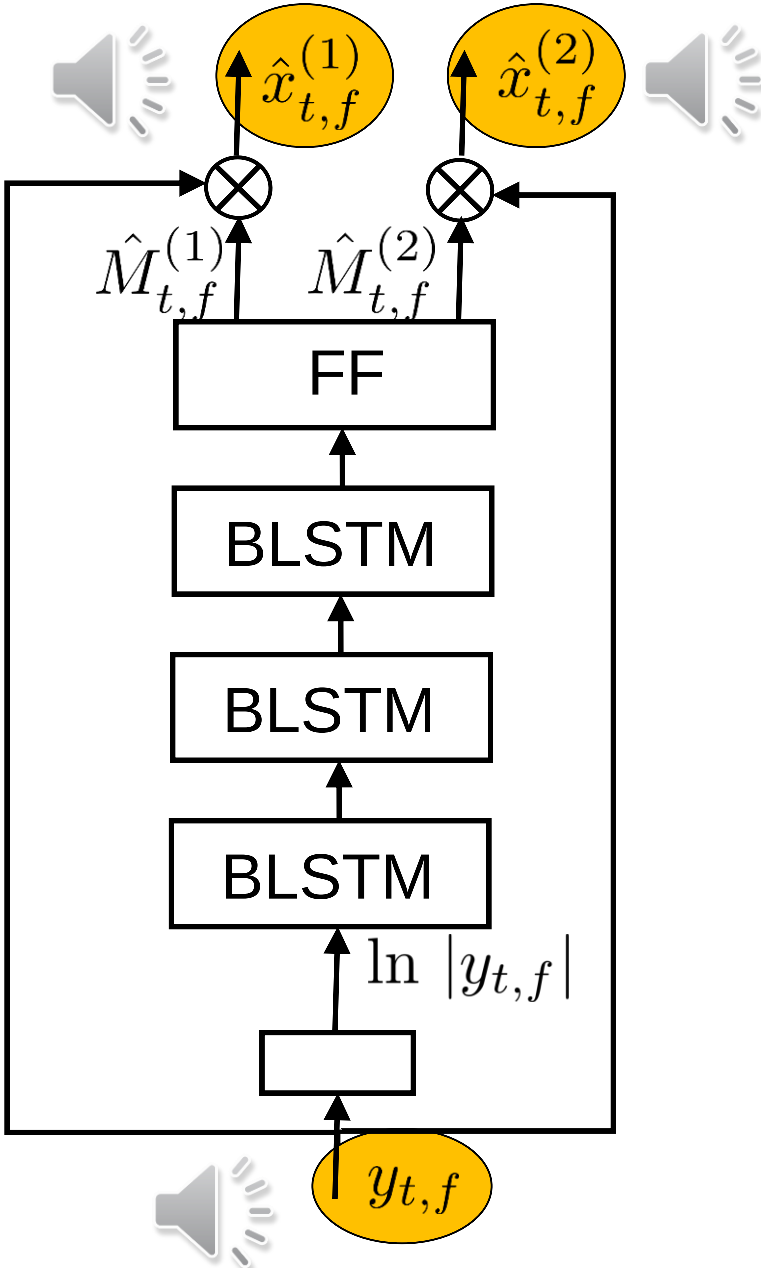


# Demonstration



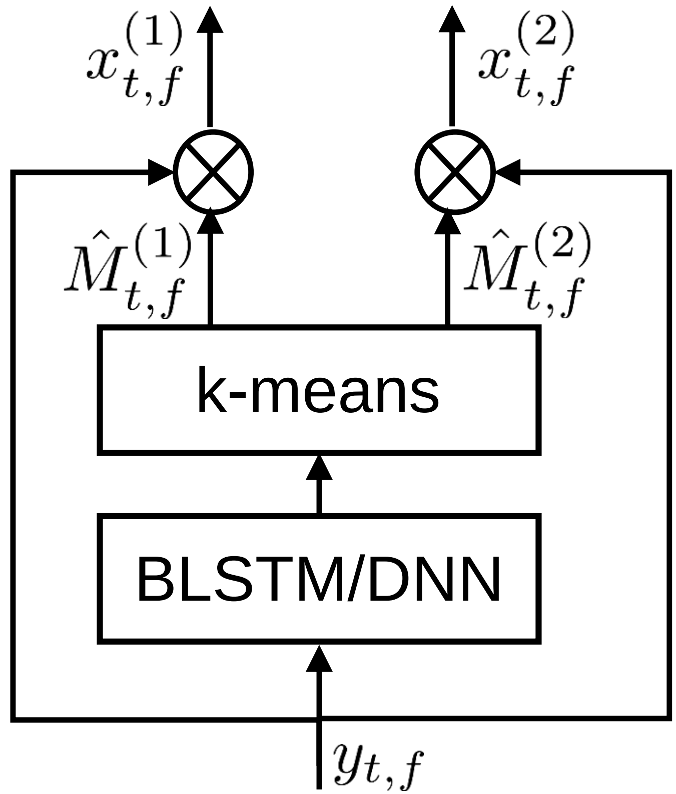



# Deep Clustering [Hershey et al., 2016]

- Map each tf-bin to an embedding vector $\mathbf{e}_{t,f}$ , where $\|\mathbf{e}_{t,f}\| = 1$
- Goal: tf-bins dominated by the same speaker form a cluster
  – Mapping via BLSTM network
- Mask estimation
  – K-means clustering of embedding vectors: hard assignments
  – Alternatively: estimate mixture model on embedding vectors: soft assignments



# Training objective

- Affinity matrix $\mathbf{A}$ of size $(T \cdot F \times T \cdot F)$:
  - $[\mathbf{A}]_{n,n'} = 1$ if $n$-th and $n$'-th tf-bin from same speaker
  - $n$ stands for certain time-frequency bin $(t,f)$
  - E.g, first and third tf-bin occupied by same speaker:

| | | | |
|---|---|---|---|
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 |

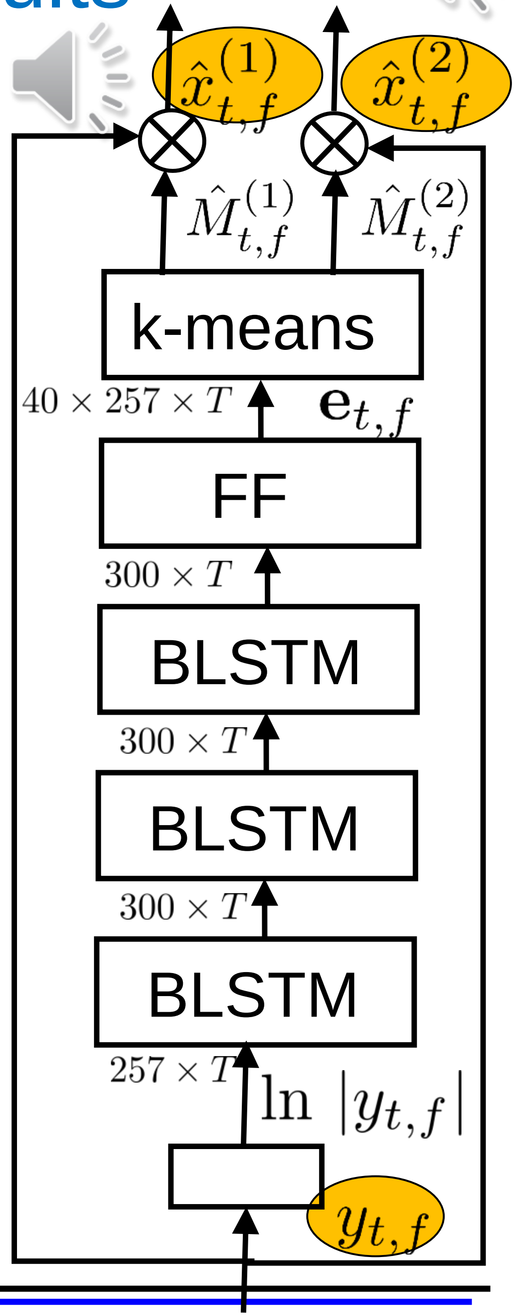
- Training objective: Minimize Frobenius norm of difference between estimated and true *affinity* matrix:

$$J(\theta) = \|\hat{\mathbf{A}}(\theta) - \mathbf{A}\|_{\mathrm{F}}^2$$

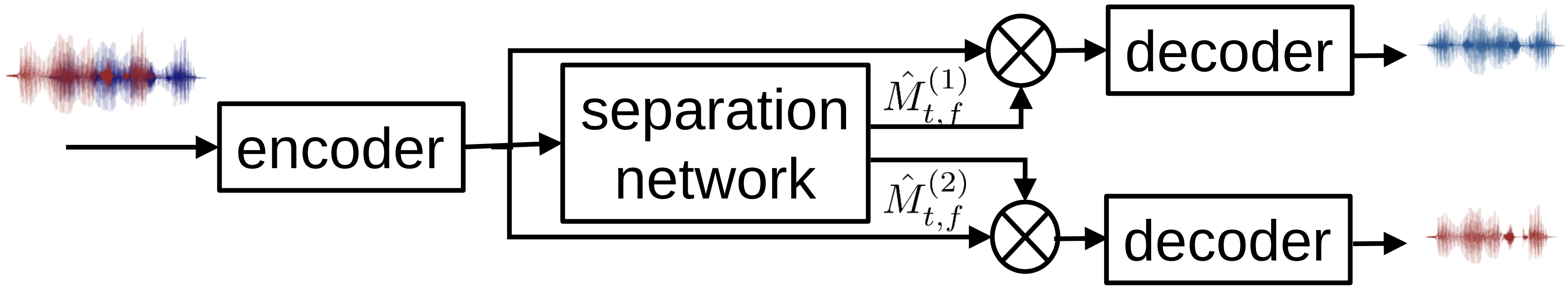
  - Estimated affinity matrix $\hat{\mathbf{A}} = \mathbf{E}\mathbf{E}^\top$, where $\mathbf{E}$ is matrix of embedding vectors $\mathbf{e}_{t,f}$

# Example configuration and results

- Example configuration:
  - Embedding network: 3 BLSTM layers with 300 units in each direction
  - Final linear layer with *(K x F) nodes*: *K*: embedding dimension; *F*: #freq.bins (e.g., *K=40, F=257*)



# TasNet [Luo and Mesgarani, 2018]




- Time-domain source separation
  - STFT replaced by learnt transformation (encoder):
    - Form segments of speech (e.g. 20 samples, i.e., 2.5 ms)

$$\mathbf{y}[tB] = [y[tB], y[tB-1] \ldots, y[tB-L+1]^{\mathsf{T}}$$

    - 1-D convolution layers applied to overlapping segments of speech

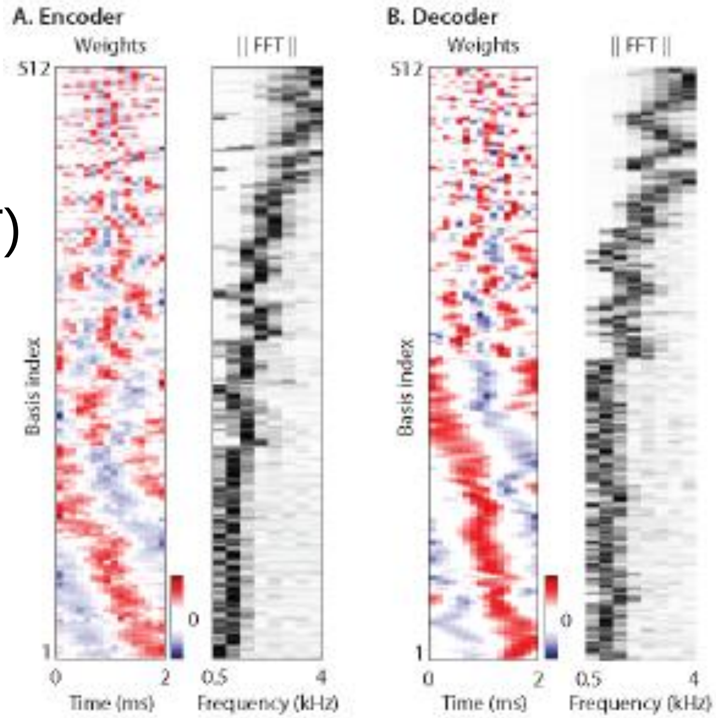
$$\mathbf{w}_t = \mathrm{ReLU}\left(\mathbf{y}[tB] \circledast \mathbf{U}\right); \quad \mathbf{U} \in \mathbb{R}^{N \times L}$$

    - Encoder transforms time-domain signal to nonnegative representation using $N$ encoder basis functions
  - Mask estimation in transform domain
  - Source extraction by masking: $\hat{\mathbf{x}}_t^{(i)} = \mathbf{w}_t \odot \hat{\mathbf{M}}_t^{(i)}$
  - Learned decoder generates waveform: $\hat{\mathbf{x}}^{(i)}[tB] = \hat{\mathbf{x}}_t^{(i)} \circledast \mathbf{V}$

# Learned transformations

- Encoder / Decoder
  - No constraint on orthogonality of bases
  - Non-negativity constraint on encoder output
  - Decoder is not inverse of encoder (as in STFT)

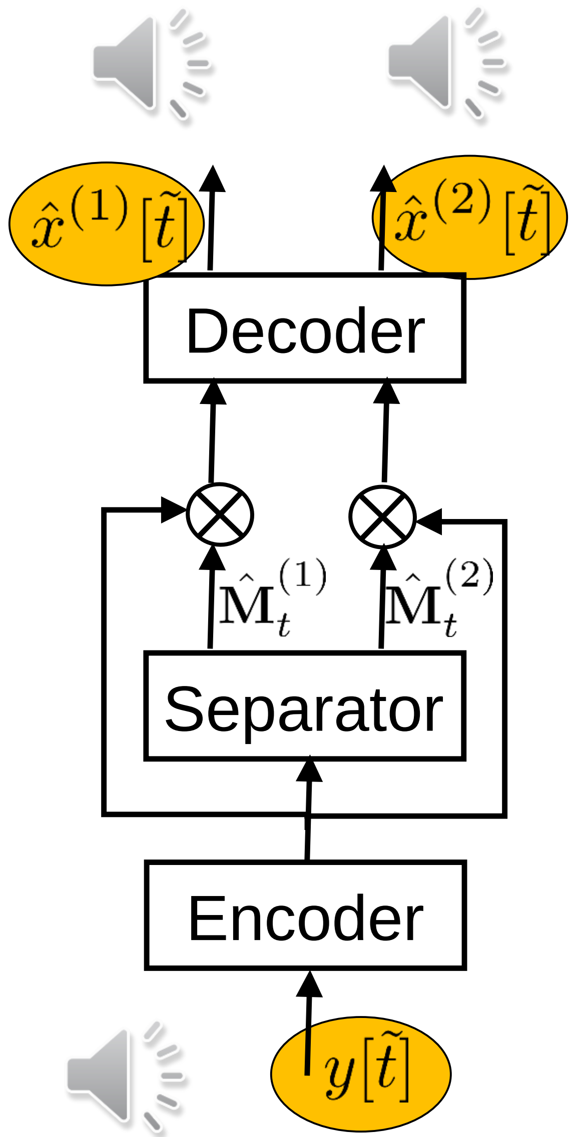
- Can the learned bases be interpreted?
  - Most filters at low frequencies
  - Filters of same frequencies with different phases




Basis functions of encoder/decoder and the magnitudes of their FFT; taken from [Luo and Mesgarani, 2018]

# Example configuration and results

- Example configuration
  - Encoder: sampling rate 8 kHz; 1-D convolution operation with window of L = 20 (2.5ms); N = 256 basis functions
  - Separator:
    - Stacked 1-D dilated convolutional blocks, see [Luo and Mesgarani, 2018]
  - Decoder: 1-D transposed convolution operations



# Discussion

- PIT, DC, TasNet and DAN (Deep Attractor Network) achieve very good speaker independent BSS

Results on wsj0-2mix: [Le Roux et al., 2018b]

| Method | SDR [dB] |
|---|---|
| PIT | (10.0) |
| DC | 10.8 |
| TasNet | 14.6 |

- TasNet naturally incorporates phase restoration, while the others estimate only magnitude spectrum
- TasNet achieves largest SDR improvement
  - Others come close when phase reconstruction component is added
- As a time domain approach TasNet has lowest latency

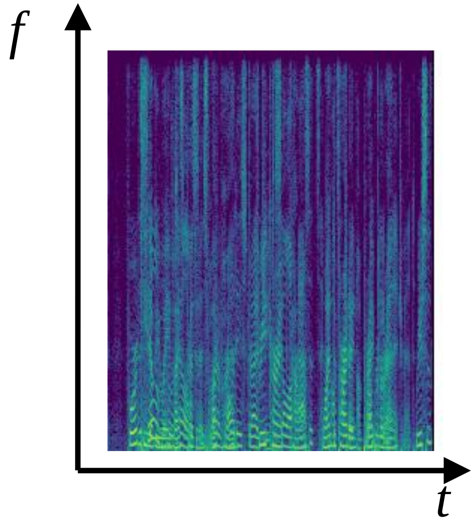
- Number of speakers must be known
  - In PIT, even the network architecture depends on the (max.) no of speakers

# Extensions

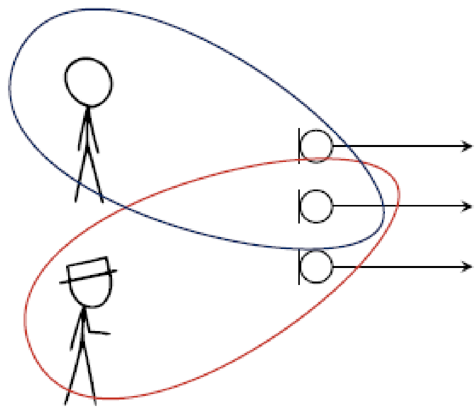
- Combinations of approaches, e.g., PIT network trained with additional DC loss [Wang and Wang, 2019]
- Extension to multi-channel input: use cross-channel features as additional input (e.g. inter-channel phase differences)
- Now that magnitude reconstruction is so good, phase reconstruction has come in the focus of research
  - Time-domain solutions (TasNet)
  - Phase reconstruction at the output of a good magnitude estimation network [Wang et al., 2018b]
  - Estimation of phase masks using discrete representation of phase diff. between noisy and clean phase [Le Roux et al., 2018a]

# Table of contents in part IV

# Separation cues: spectro-temporal vs spatial




- Spectro-temporal cues
  - Model speech characteristics
  - Can work with **single-channel input**
  - Leverage training data
  - Typically supervised trng
  - DNN based



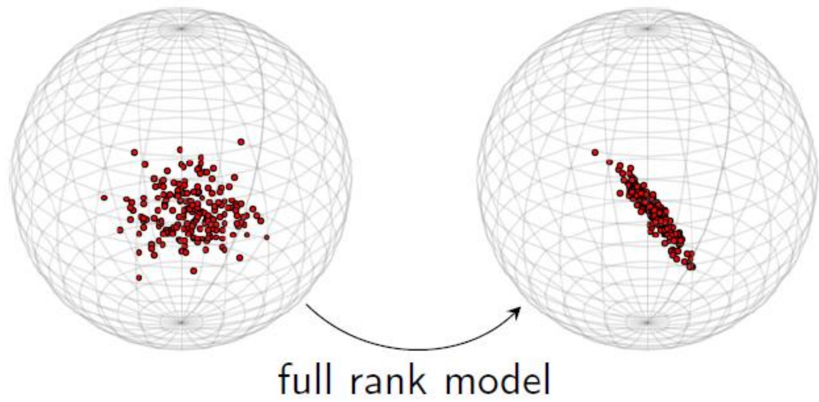
- Spatial cues
  - Exploits spatial selectivity
  - Requires **multi-channel input**
  - Does not require trng phase
  - Unsupervised learning (EM alg.)
  - Spatial mixture model based

# Spatial mixture model

- Straightforward extension of beamforming case

$$p(\mathbf{y}_{t,f}) = \sum_i \Pr(M_{t,f} = i) p(\mathbf{y}_{t,f} | M_{t,f} = i); \; i \in \{0, 1, \ldots, I\}$$

  - E.g., Complex angular central Gaussian Mixture Model with $I+1$ components



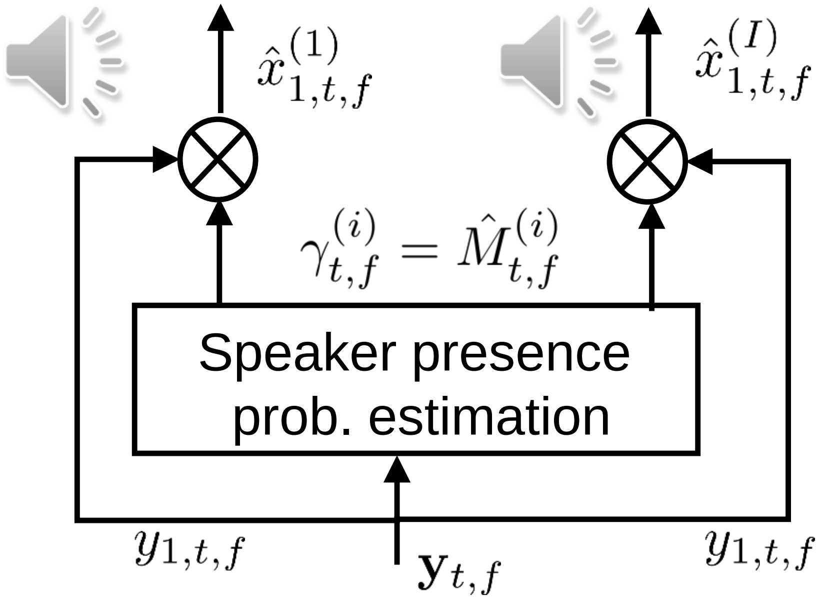

- EM algorithm to estimate speaker presence probabilities

$$\gamma_{t,f}^{(i)} = \hat{\Pr}(M_{t,f} = i | \mathbf{y}_{t,f}) =: \hat{M}_{t,f}^{(i)}$$

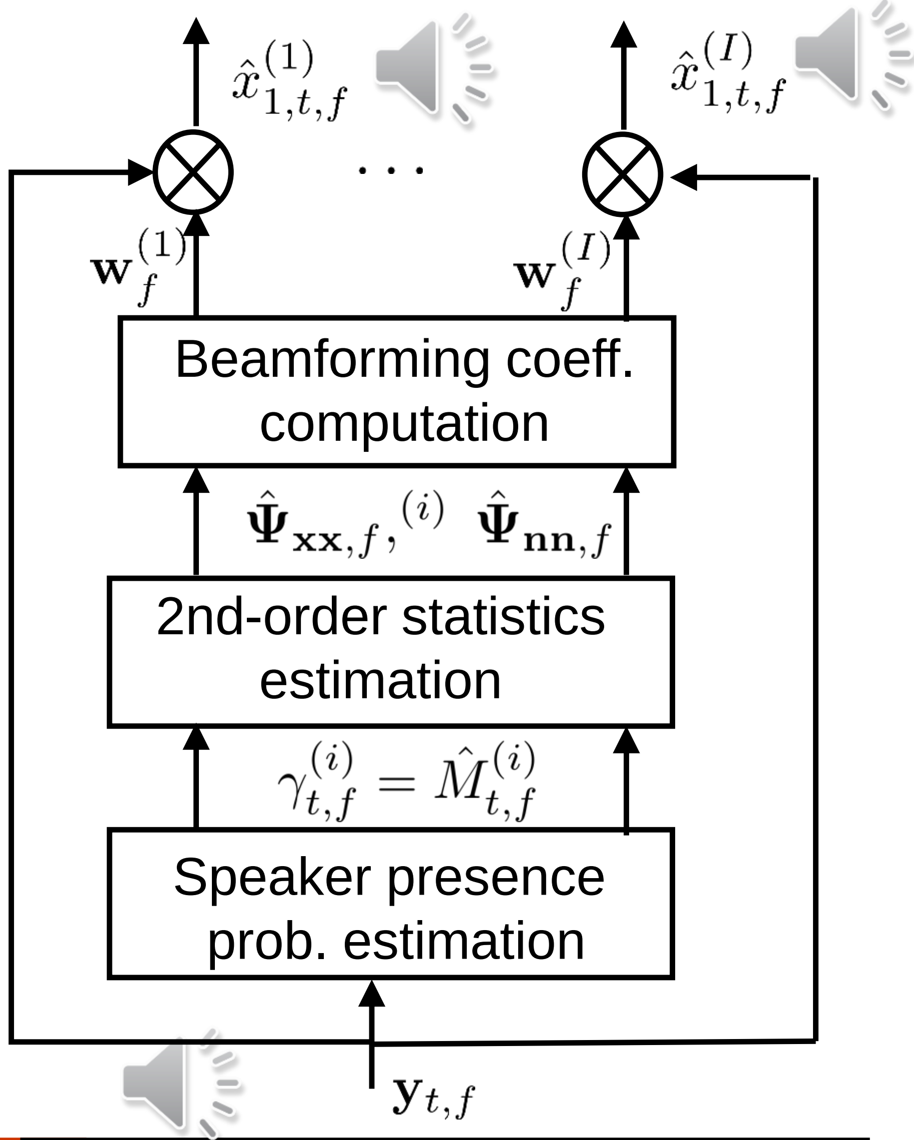
# Source extraction

## by masking

Beamforming achieves better perceptual quality (and WER performance)




## by beamforming

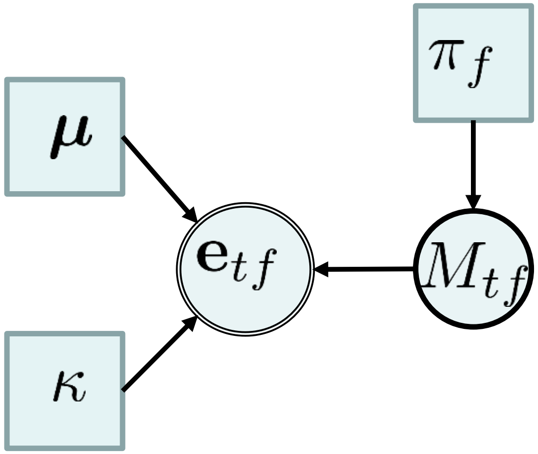


# Table of contents in part IV

# Integration of Deep Clustering and mixture models

- Goal: combine the strengths of both methods
  - Exploit spectral and spatial cues for separation
  - Leverage trng data and do unsupervised learning on test utterance
- Weak integration
  - Use k-means result of DC as initialization of $\gamma_{t,f}^{(i)}$ (speaker presence prob.) of the spatial mixture model and run EM steps on test utterance
- Strong integration
  - Take embedding vectors $\mathbf{e}_{t,f}$ and microphone signals $\mathbf{y}_{t,f}$ as two observations in a mixture model

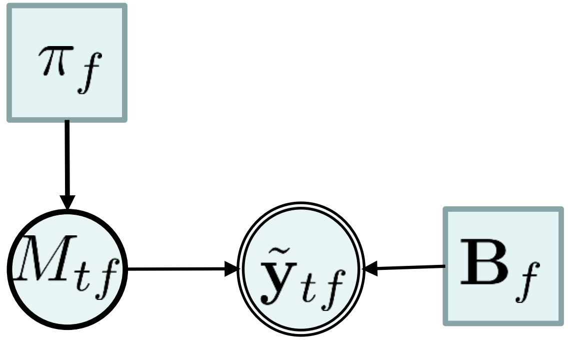
# Mixture model for DC embeddings




- Model embedding vectors as r.v.
  - Mixture of von-Mises Fisher distributions
  - K-means replaced by EM

$$
\begin{aligned}
p(\mathbf{e}_{tf}) &= \sum_i \Pr(M_{t,f} = i) p(\mathbf{e}_{t,f} | M_{tf} = i) \\
&= \sum_i \pi_f^{(i)} \cdot \mathrm{vMF}(\mathbf{e}_{tf}^{(i)}; \boldsymbol{\mu}^{(i)}, \kappa^{(i)})
\end{aligned}
$$

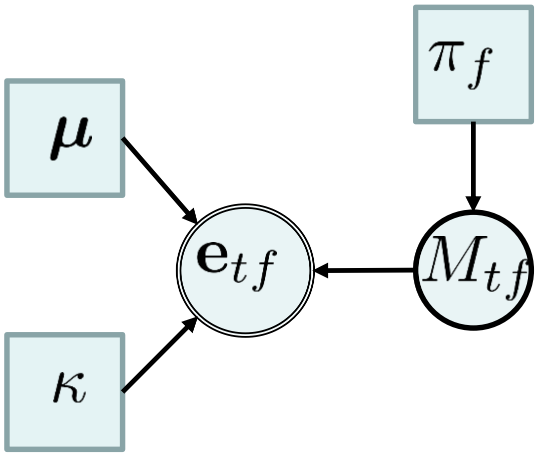
# Recall spatial mixture model

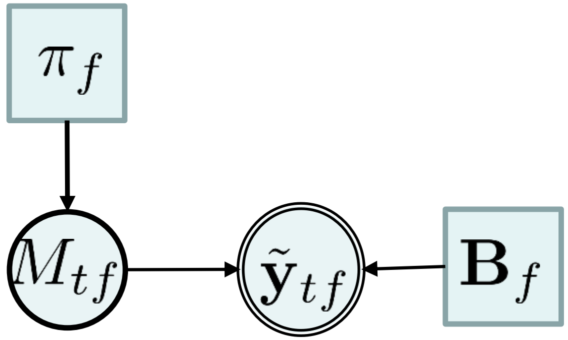



$$p(\tilde{\mathbf{y}}_{t,f}) = \sum_i \Pr(M_{t,f} = i) p(\tilde{\mathbf{y}}_{t,f} | M_{t,f} = i)$$

$$= \sum_i \pi_f^{(i)} \mathrm{cACG}(\tilde{\mathbf{y}}_{t,f}; \mathbf{B}_f^{(i)})$$

# Strong integration

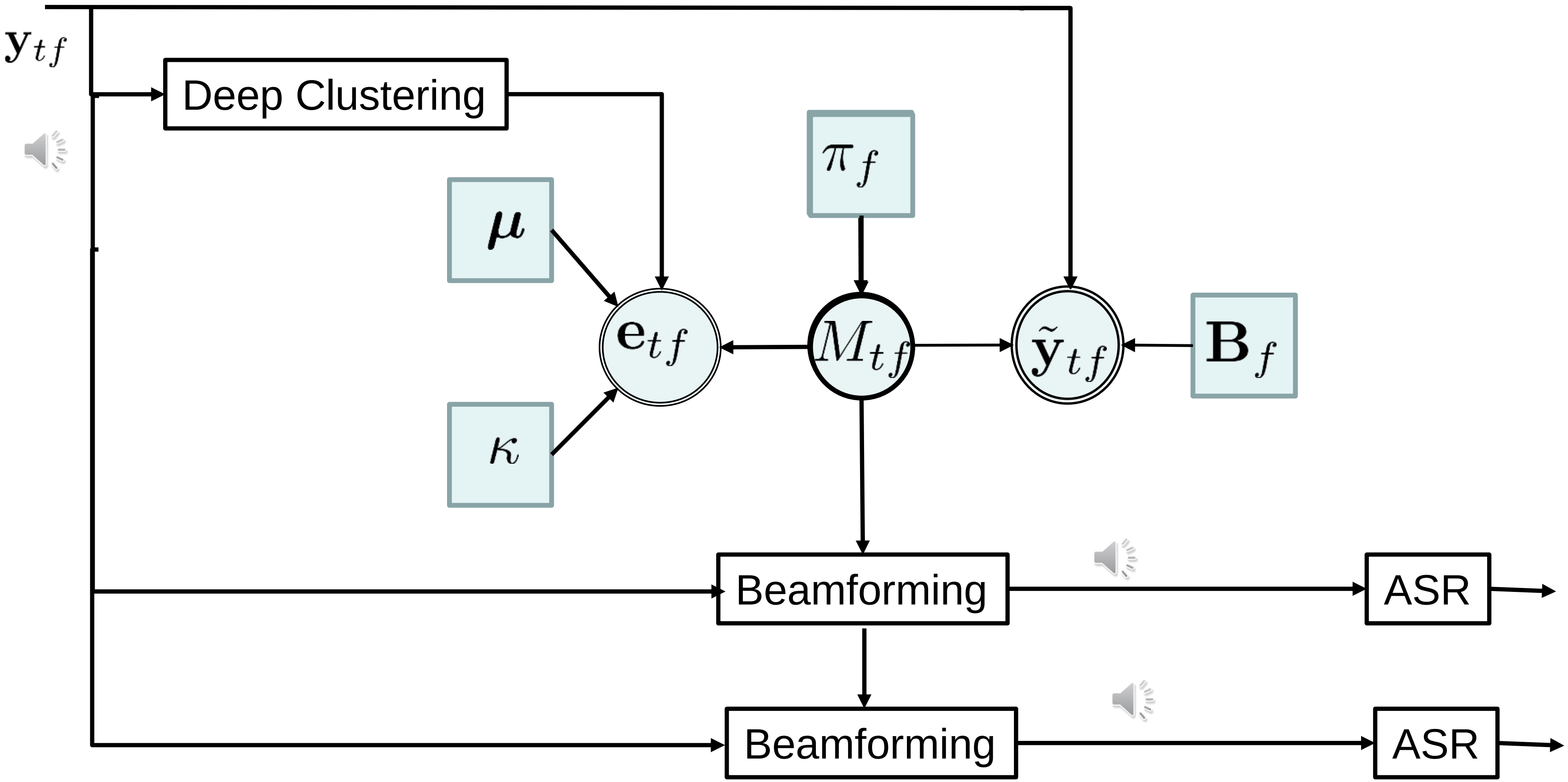



Integrated mixture model

- Coupling via latent class affiliation variable (speaker presence prob.)
- Hypothesis: better estimates when estimated jointly

# Overall system



# Results [Drude and Haeb-Umbach, 2019]

- Database: spatialized multi-channel wsj-2mix
  - Artificial 2-speaker mixtures from WSJ utterances
  - 8 channels
  - $T_{60}$ = 0.2 – 0.6 s
- Acoustic model trained either on **clean** speech or on **image** of clean speech at reference microphone (includes reverb.)

| Model | WER [%] Clean | WER [%] Image |
|---|---|---|
| Spatial mixture model (cACGMM) | 40.9 | 28.2 |
| Deep Clustering (DC) | 42.5 | 26.6 |
| Weak integration | 34.4 | 21.6 |
| Strong integration (DC + cACGMM) | 33.4 | 18.9 |
| oracle | 31.1 | 10.7 |

# Pros and cons of NN and spatial mixture model based BSS

| | Spatial mixture models | Neural networks |
|---|---|---|
| Spatial characteristics modeling | • **Strong** | • Moderate (use of cross-channel features at input) |
| Spectro-temporal characteristics modeling (for speech) | • Weak<br>- Permutation problem<br>• No concept of human speech (pros and cons) | • **Very strong**<br>- Strong speech model based on a priori training |
| #channels required | • Multi-channel | • Single channel |
| Leverage training data | • No training phase | • **Yes**, but parallel data required |
| Adaptation to test condition | • **Strong**<br>- Unsupervised learning applicable | • Weak<br>- Poor generalization<br>- Sensitive to mismatch |

We have seen the same table before

# Software

- Spatial mixture models: https://github.com/fgnt/pb_bss
  - Different spatial mixture models
    - complex angular central Gaussian , complex Watson, von-Mises-Fisher
  - Methods: init, fit, predict
  - Beamformer variants
  - Ref: [Drude and Haeb-Umbach, 2017]

# Summary of part IV

- Speaker-independent single-channel DNN-based BSS is a major improvement over earlier approaches
- Source extraction by beamforming produces less artifacts than by masking
- Both DNN-based and spatial mixture model based BSS achieve comparable results when used with beamformer for source extraction
- DNN based and spatial mixture model based BSS have complementary strengths and can be combined
- Often simplifying assumptions:
  - # active speakers known
  - All speakers speak all the time
  - Most investigations on artificially mixed speech and static scenario
  - offline

Some of those assumptions will be lifted in the next presentation

# Table of contents