Source Separation 2

안인규 (Inkyu An)

Speech And Audio Recognition
(오디오 음성인식)

https://mairlab-km.github.io/

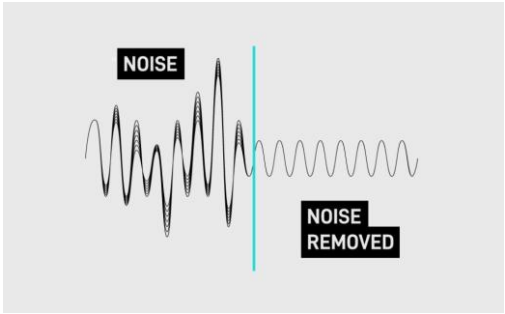
This lecture material refers to
https://github.com/yandexdataschool/speech_course?tab=readme-ov-file and
https://github.com/markovka17/dla

What is Source Separation?

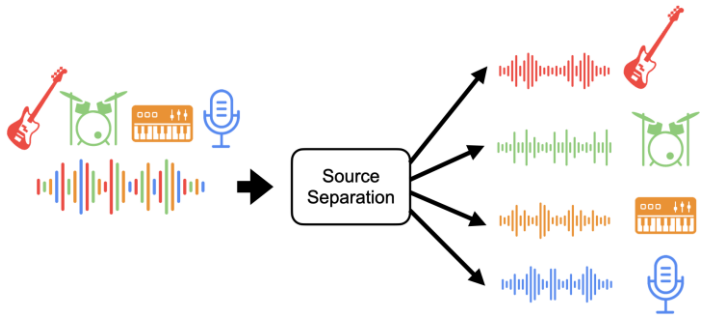
- Source Separation literally means separate any source of particular interest…

Separation?

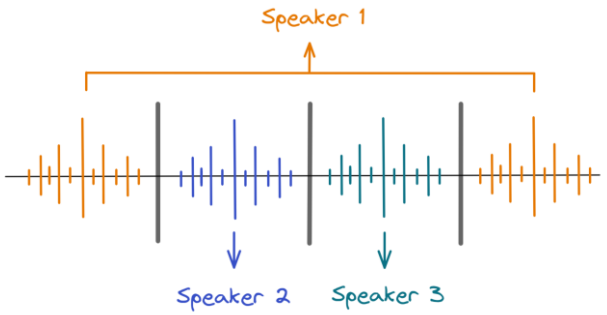
- Separate signal (music, speech, …) from noise → **Denoising**



- Separate audio mix into separate tracks → **Source Separation**



- Guess when specific sound sources are active → **Diarization**

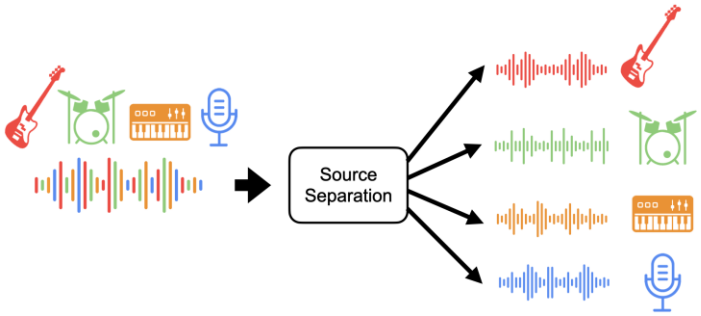


What is Source Separation?

- Source Separation literally means separate any source of particular interest…

Separation?

- Separate signal (music, speech, …) from noise → **Denoising**
- Separate audio mix into separate tracks → **Source Separation**
- Guess when specific sound sources are active → **Diarization**



Blind (any k)

Guided (specifically bass, vocal, guitar)

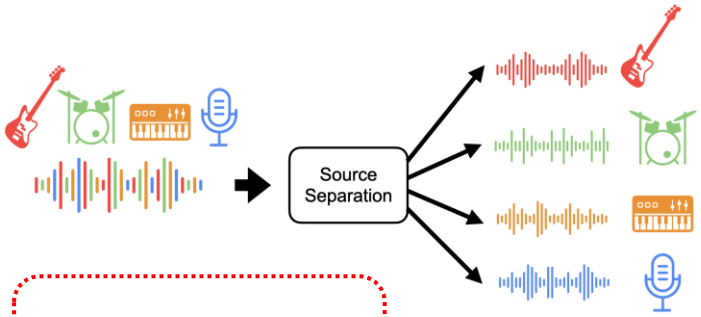
Target (specific source given by reference)

What is Source Separation?

- Source Separation literally means separate any source of particular interest…

Separation?

- Separate signal (music, speech, …) from noise → **Denoising**
- Separate audio mix into separate tracks → **Source Separation**
- Guess when specific sound sources are active → **Diarization**



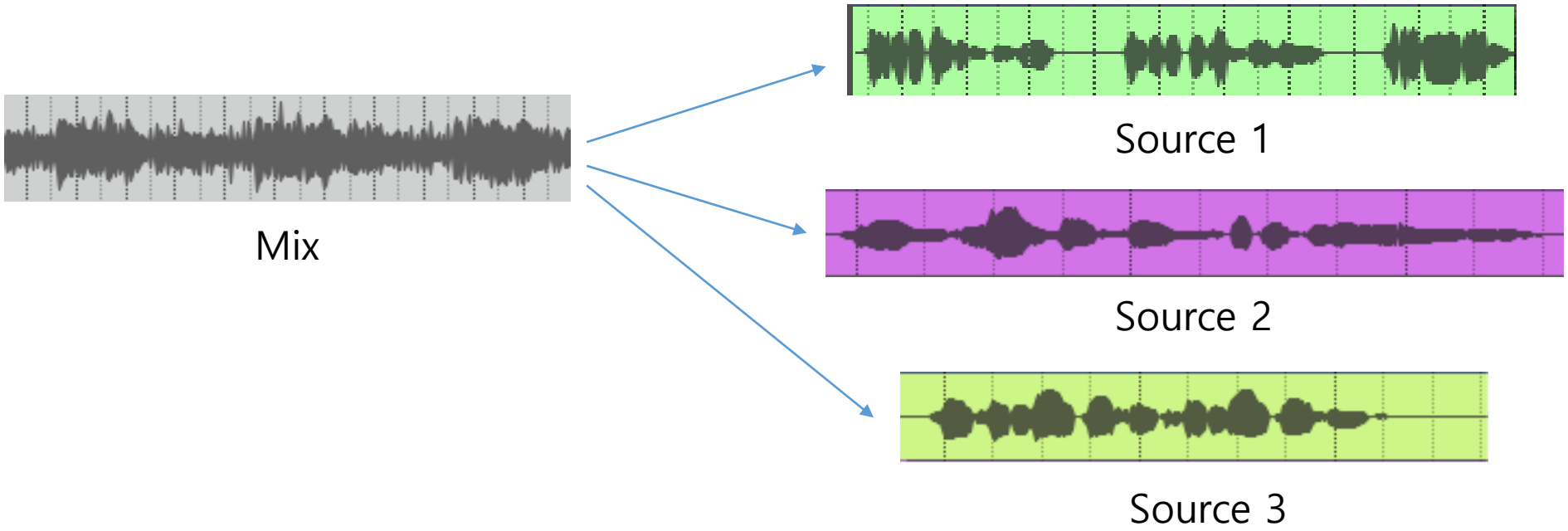
Blind (any k)

Guided (specifically bass, vocal, guitar)

Target (specific source given by reference)

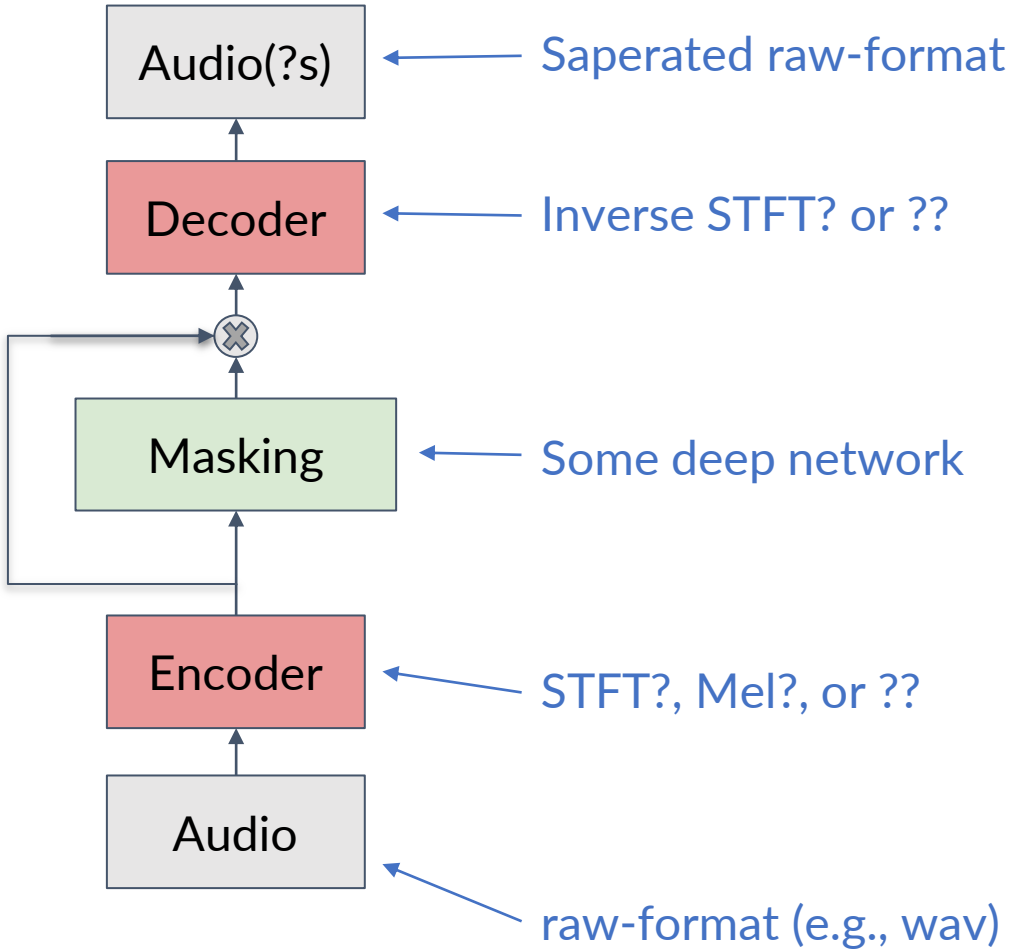
Blind Source Separation

- **Goal**: extract K sources from the noisy mixture **w/o (or with very little) information about the mixing process**

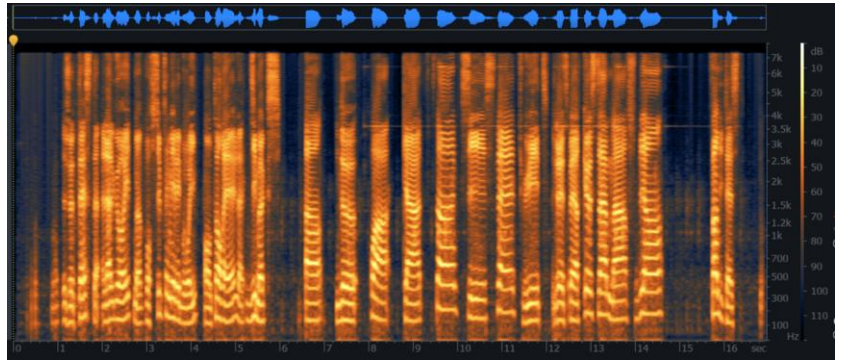


- We know denoising and specific guided separation, how can we apply DL here?

Encoder-Separation-Decoder (ESD)

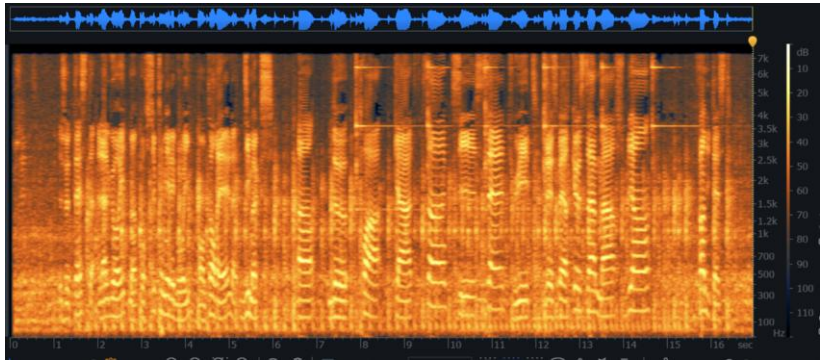


Denoised Alex



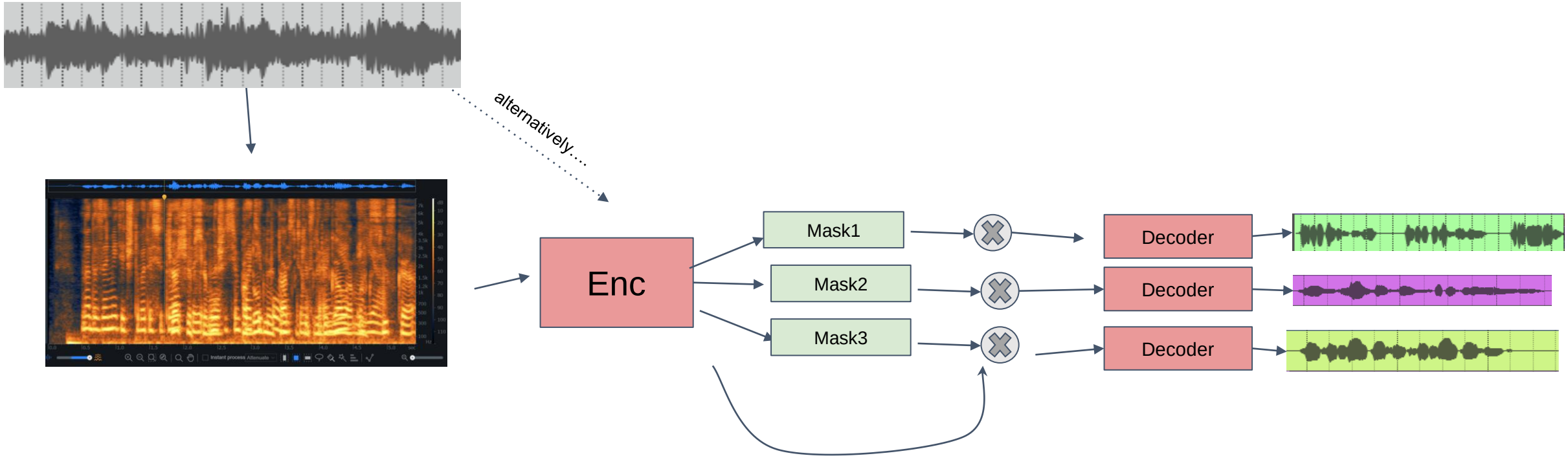
Denoising

Alex & noise



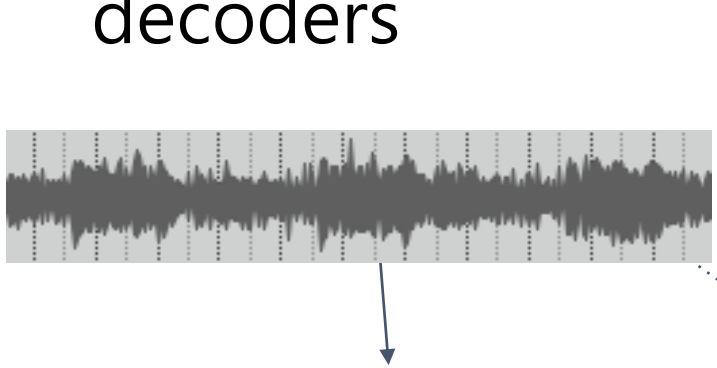
Blind Source Separation – First Idea

- **Goal**: just use Encoder-Separation-Decoder (ESD) with several decoders

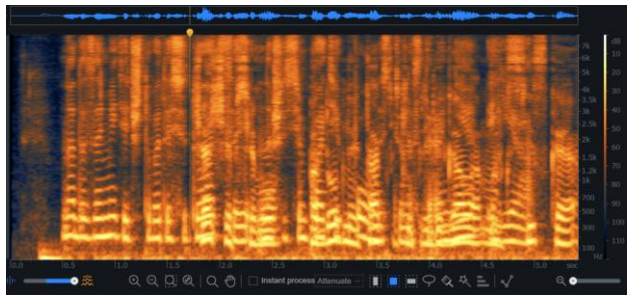


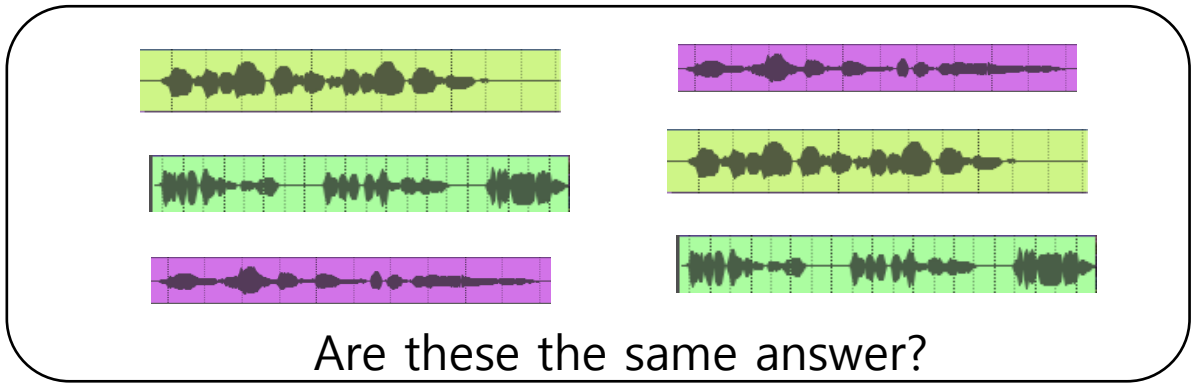
Blind Source Separation – First Idea

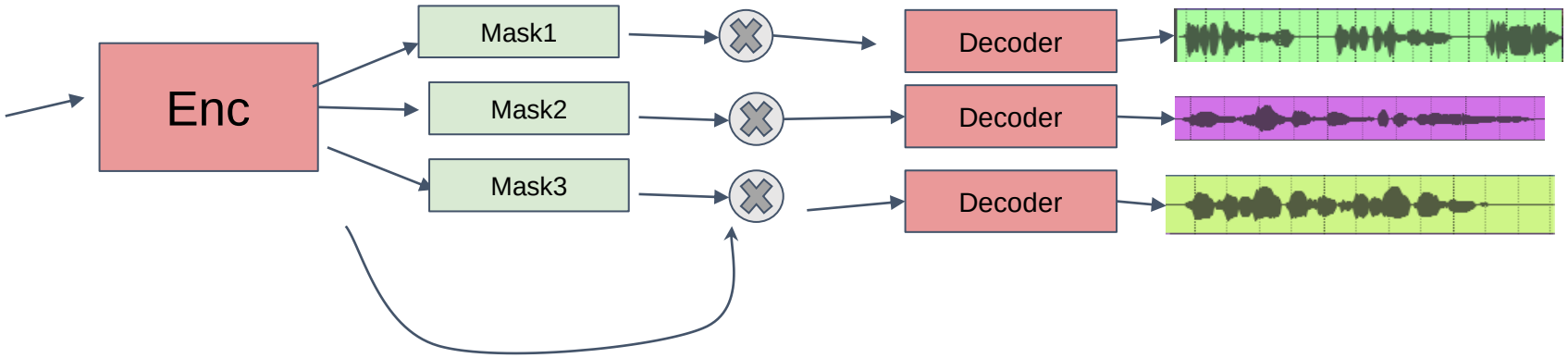
- **Goal**: just use Encoder-Separation-Decoder (ESD) with several decoders



alternatively....

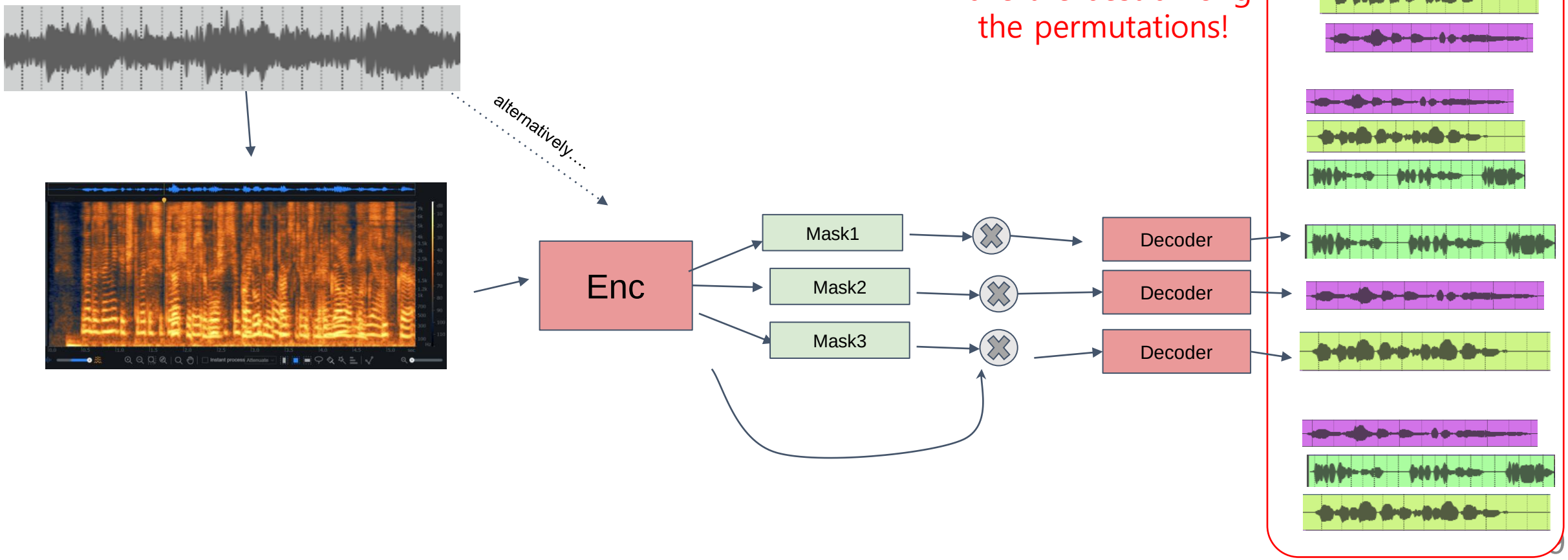






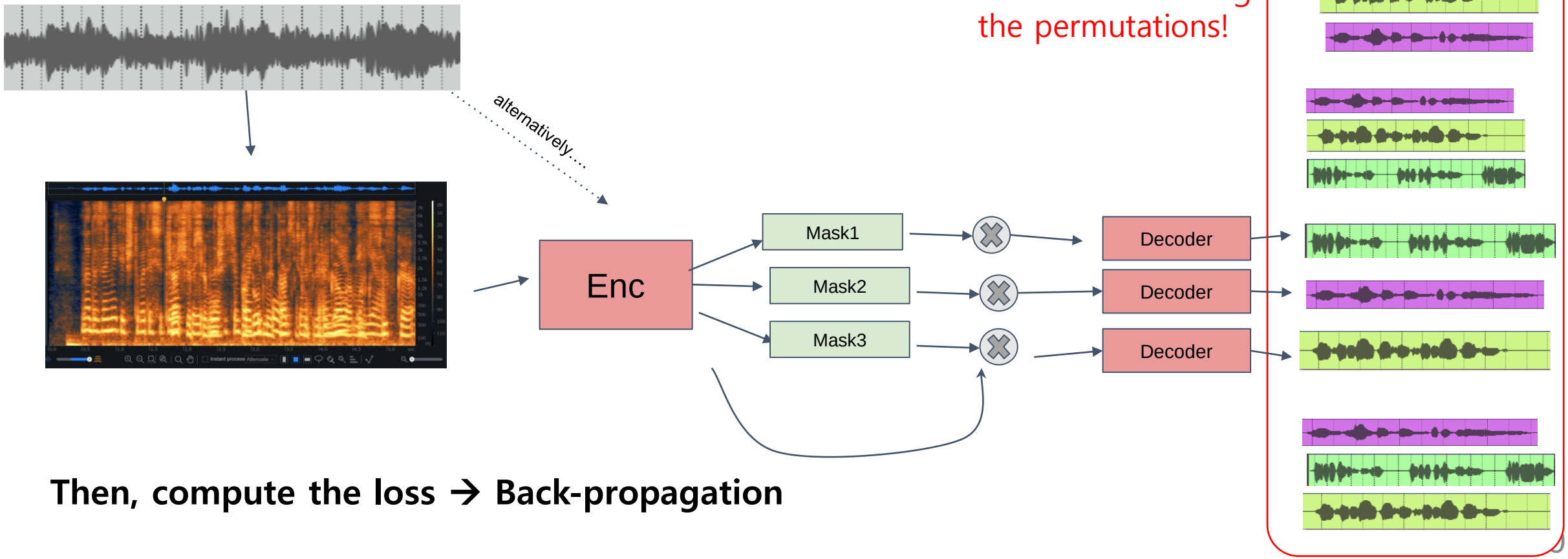
Blind Source Separation – PIT

- **Idea**: Permutation-Invariant Training (PIT), take the best loss of all permutations of predictions



Blind Source Separation – PIT

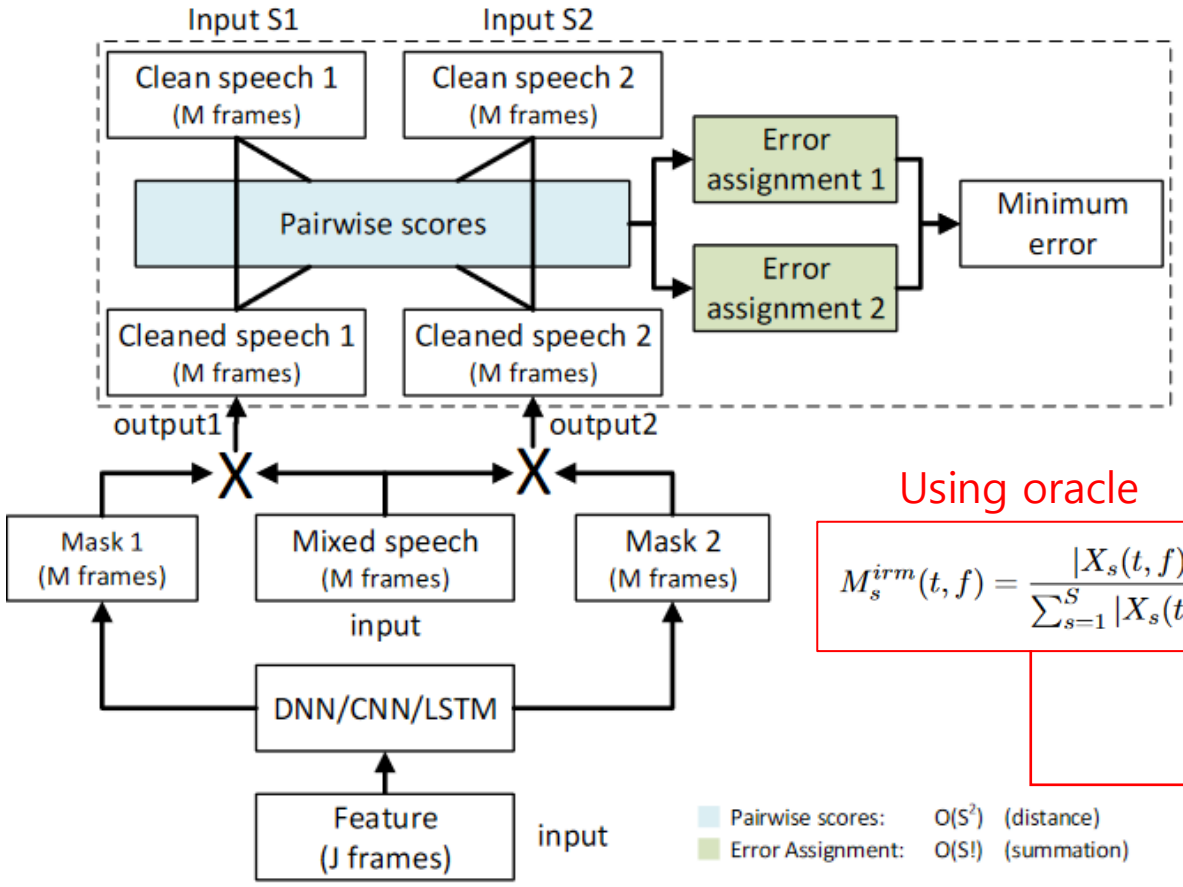
- **Idea**: Permutation-Invariant Training (PIT), take the best loss of all permutations of predictions



Then, compute the loss → Back-propagation

Blind Source Separation – PIT

- **PIT[2017] is the first working DL baseline without clustering**



Using oracle

$$M_s^{irm}(t,f) = \frac{|X_s(t,f)|}{\sum_{s=1}^{S}|X_s(t,f)|}$$

SDR IMPROVEMENTS (DB) FOR DIFFERENT SEPARATION METHODS ON THE WSJ0-2MIX DATASET USING UPIT.

Open condition (본적 없는 화자)

Method	Mask Type	Activation Function	Opt. Assign. CC	Opt. Assign. OC	Def. Assign. CC	Def. Assign. OC
uPIT-BLSTM	AM	softmax	**10.4**	**10.3**	**9.0**	**8.7**
uPIT-BLSTM	AM	sigmoid	8.3	8.3	7.1	7.2
uPIT-BLSTM	AM	ReLU	9.9	9.9	8.7	8.6
uPIT-BLSTM	AM	Tanh	8.5	8.6	7.5	7.5
uPIT-BLSTM	PSM	softmax	10.3	10.2	9.1	9.0
uPIT-BLSTM	PSM	sigmoid	10.5	10.4	9.2	9.1
uPIT-BLSTM	PSM	ReLU	**10.9**	**10.8**	**9.4**	**9.4**
uPIT-BLSTM	PSM	Tanh	10.4	10.3	9.0	8.9
uPIT-BLSTM	NPSM	softmax	8.7	8.6	7.5	7.3
uPIT-BLSTM	NPSM	sigmoid	**10.6**	**10.6**	**9.4**	**9.3**
uPIT-BLSTM	NPSM	ReLU	8.8	8.8	7.6	7.6
uPIT-BLSTM	NPSM	Tanh	10.1	10.0	8.9	8.8
uPIT-LSTM	PSM	ReLU	**9.8**	**9.8**	7.0	**7.0**
uPIT-LSTM	PSM	sigmoid	**9.8**	9.6	**7.1**	6.9
uPIT-LSTM	NPSM	ReLU	**9.8**	**9.8**	**7.1**	**7.0**
uPIT-LSTM	NPSM	sigmoid	9.2	9.2	6.8	6.8
PIT-BLSTM	PSM	ReLU	**11.7**	**11.7**	**-1.7**	-1.9
PIT-BLSTM	PSM	sigmoid	**11.7**	**11.7**	**-1.7**	**-1.7**
PIT-BLSTM	NPSM	ReLU	**11.7**	**11.7**	**-1.7**	-1.8
PIT-BLSTM	NPSM	sigmoid	11.6	11.6	-1.6	**-1.7**
IRM	-	-	12.4	12.7	12.4	12.7
IPSM	-	-	14.9	15.1	14.9	15.1

w/ PIT w/o PIT

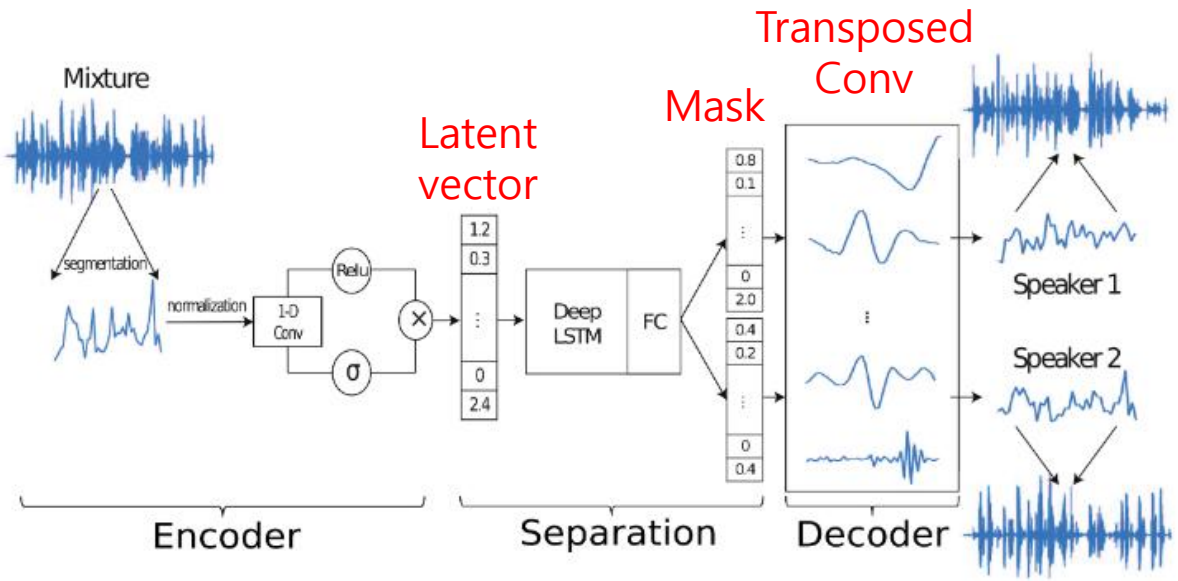
Blind Source Separation – RNN-based

- **TasNet (2017): solve blind speaker separation (2 speakers tried)**

  - **1D CNN** Encoder
  - **Heavy** (1024hid) 4-layer **LSTM**s as Separator
  - **FC**(!) decoders each of 1024 units
  - Short segments of audio (5ms at 8kHz SR) as input
  - **PIT** used

  Trained and evaluated on WSJ0-2mix

  One of the first serious DNN baselines together with PIT



Method	Causal	SI-SNRi	SDRi
uPIT-LSTM [4]	✓	–	7.0
TasNet-LSTM	✓	7.7	**8.0**
DPCL++ [3]	×	**10.8**	–
DANet [5]	×	10.5	–
uPIT-BLSTM-ST [4]	×	–	10.0
TasNet-BLSTM	×	**10.8**	**11.1**

SDRi: improvement over mix, SDR around 7-8(BLSTM)

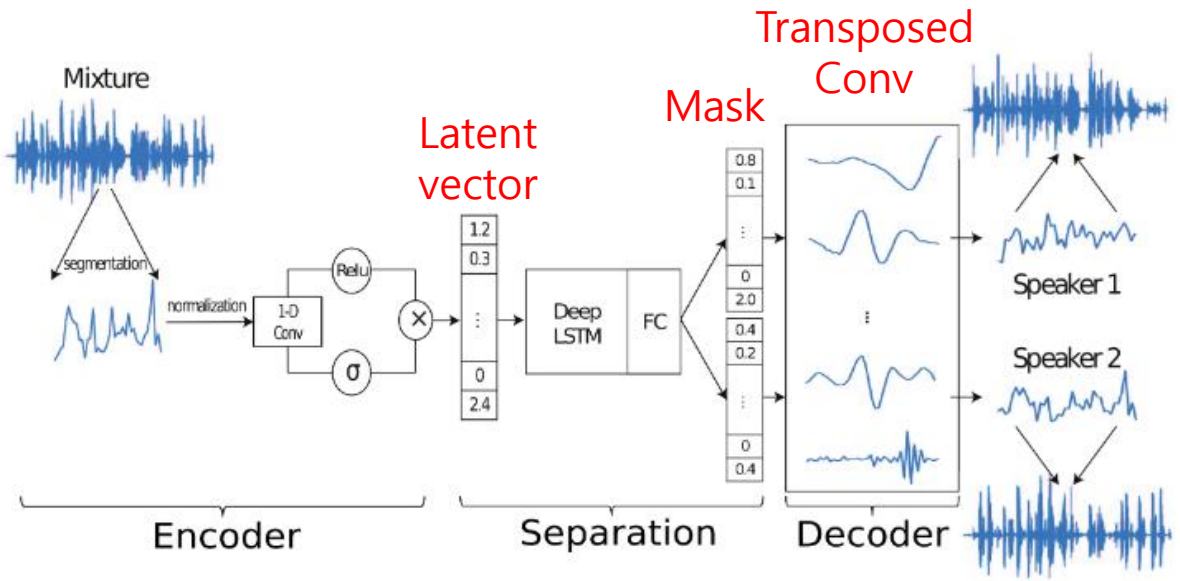
Blind Source Separation – RNN-based

- **TasNet (2017): solve blind speaker separation (2 speakers tried)**

  - **1D CNN** Encoder
  - **Heavy** (1024hid) 4-layer **LSTM**s as Separator
  - **FC**(!) decoders each of 1024 units
  - Short segments of audio (5ms at 8kHz SR) as input
  - **PIT** used

  Trained and evaluated on WSJ0-2mix

  One of the first serious DNN baselines together with PIT



Method	Causal	SI-SNRi	SDRi
uPIT-LSTM [4]	✓	–	7.0
TasNet-LSTM	✓	7.7	**8.0**
DPCL++ [3]	×	**10.8**	–
DANet [5]	×	10.5	–
uPIT-BLSTM-ST [4]	×	–	10.0
TasNet-BLSTM	×	**10.8**	**11.1**

25M parameters

SDRi: improvement over mix, SDR around 7-8(BLSTM)

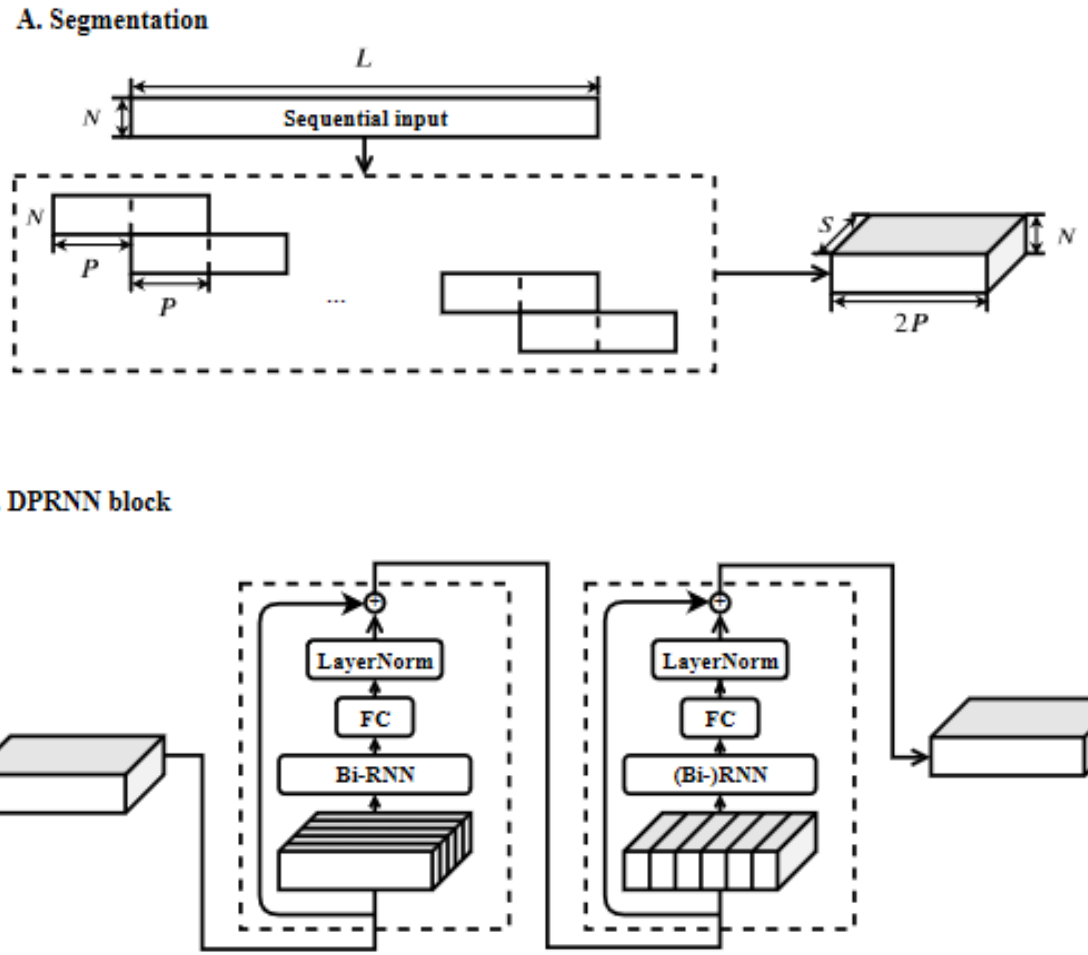
Blind Source Separation – RNN-based

- **Dual-Path RNN (2020):** solve blind speaker separation (2 speakers tried)

  - **Dual-Path RNN**
  - **Light:** 2.6M parameters
  - **Streaming**-ready
  - Short overlapping chunks of audio (2ms at 16kHz SR) as input

  Application: speech enhancement, separation with known number of speakers…

  SI-SNRi: WSJ02-mix ~18



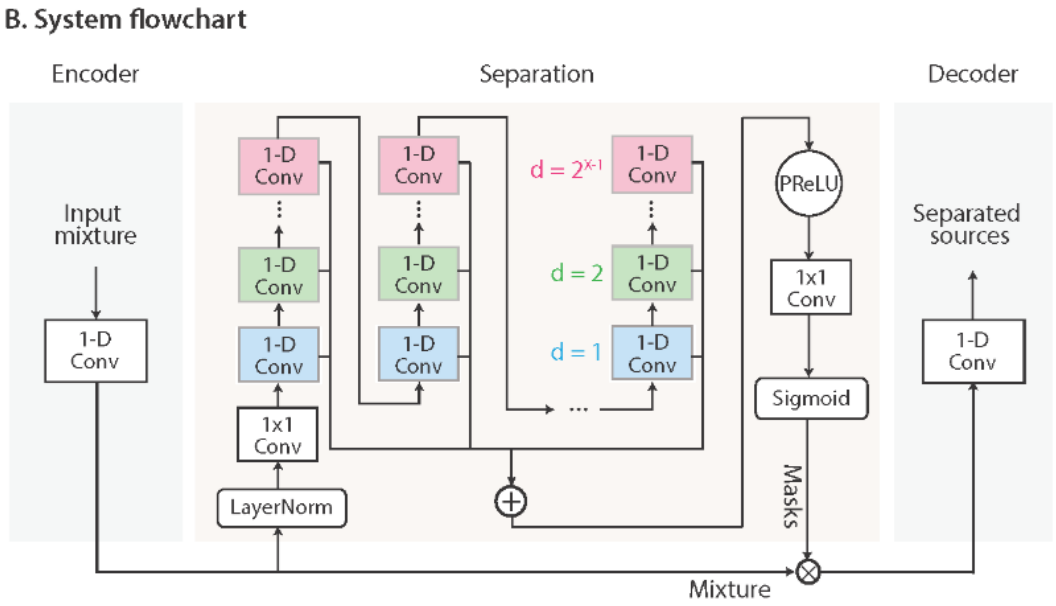
Blind Source Separation – CNN-based

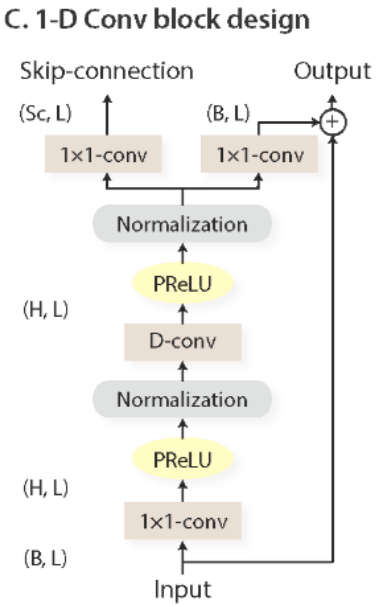
- **ConvTasNet (2018):** solve blind speaker separation (2 speakers tried)

  - **1D CNN** Encoder
  - **TCN (Temporal Convolutional Network) ResNet-like structure** as Separator
  - Arbitrary-length audio as input
  - **PIT** used

Trained and evaluated on WSJ0-2mix

The second serious DNN baseline together with PIT and TasNet





Blind Source Separation – CNN-based

- **ConvTasNet (2018):** solve blind speaker separation (2 speakers tried)

- **1D CNN** Encoder
- **TCN (Temporal Convolutional Network) ResNet-like structure** as Separator
- Arbitrary-length audio as input
- **PIT** used

Trained and evaluated on WSJ0-2mix

The second serious DNN baseline together with PIT and TasNet

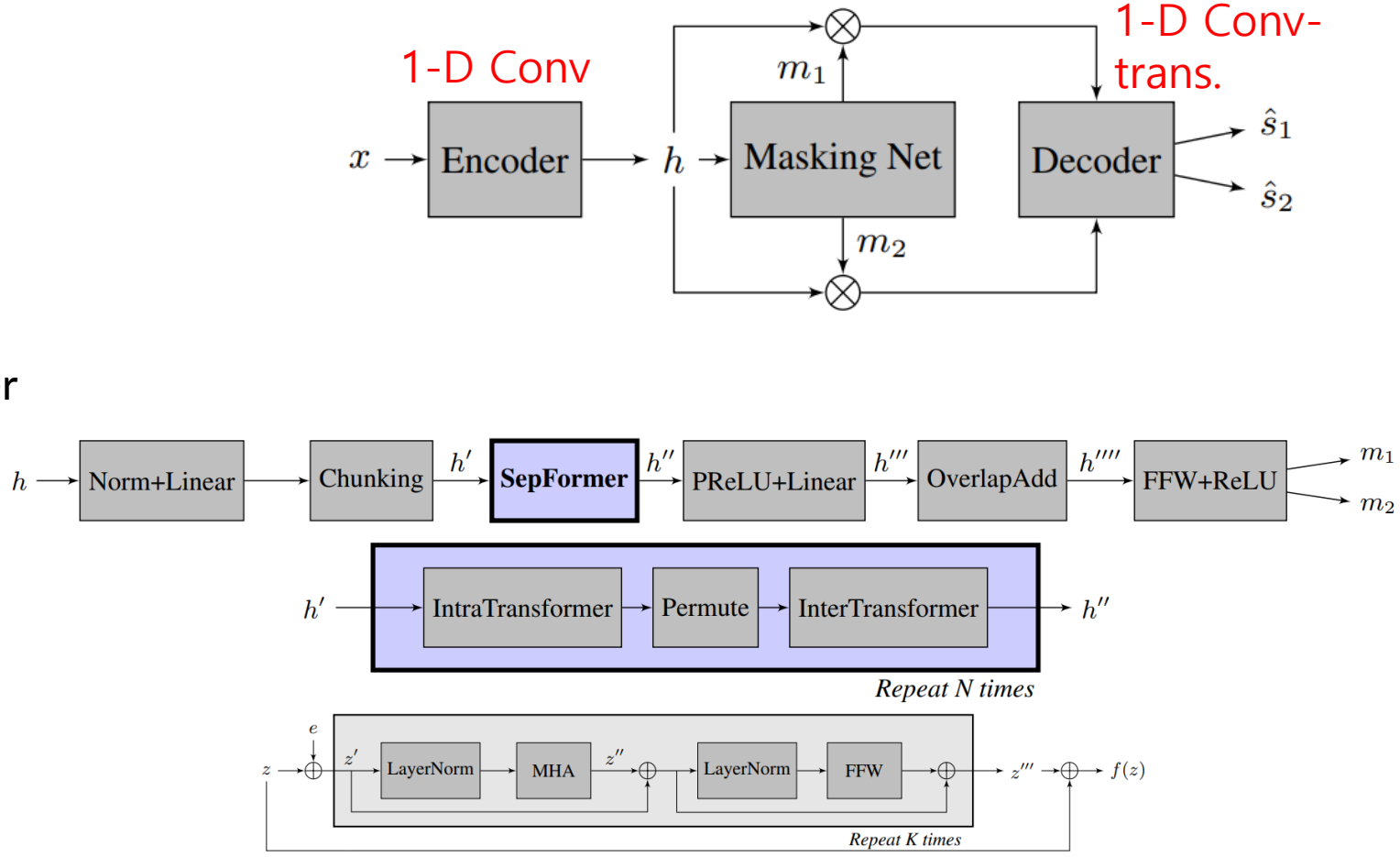
Method	Model size	Causal	SI-SNRi (dB)	SDRi (dB)
DPCL++ [5]	13.6M	×	10.8	–
uPIT-BLSTM-ST [7]	92.7M	×	–	10.0
DANet [8]	9.1M	×	10.5	–
ADANet [9]	9.1M	×	10.4	10.8
cuPIT-Grid-RD [50]	47.2M	×	–	10.2
CBLDNN-GAT [12]	39.5M	×	–	11.0
Chimera++ [10]	32.9M	×	11.5	12.0
WA-MISI-5 [11]	32.9M	×	12.6	13.1
BLSTM-TasNet [26]	23.6M	×	13.2	13.6
Conv-TasNet-gLN	**5.1M**	×	**15.3**	**15.6**
uPIT-LSTM [7]	46.3M	✓	–	7.0
LSTM-TasNet [26]	32.0M	✓	**10.8**	**11.2**
Conv-TasNet-cLN	**5.1M**	✓	10.6	11.0
IRM	–	–	12.2	12.6
IBM	–	–	13.0	13.5
WFM	–	–	13.4	13.8

Blind Source Separation – Transformer-based

- **SepFormer (2021):** solve blind speaker separation (2 speakers tried)

  - **Transformer**-based dual-path architecture
  - **Self-Attention** layers replace TCN blocks in separator
  - Handles **long-term temporal dependencies** efficiently
  - Fully convolutional Encoder/Decoder in time domain
  - **PIT** (Permutation Invariant Training) used

Trained and evaluated on WSJ0-2mix and WHAM! datasets



Blind Source Separation – Transformer-based

- **SepFormer (2021):** solve blind speaker separation (2 speakers tried)

- **Transformer**-based dual-path architecture
- **Self-Attention** layers replace TCN blocks in separator
- Handles **long-term temporal dependencies** efficiently
- Fully convolutional Encoder/Decoder in time domain
- **PIT** (Permutation Invariant Training) used

Trained and evaluated on WSJ0-2mix and WHAM! datasets

Table 1. Best results on the WSJ0-2mix dataset (test-set). DM stands for dynamic mixing.

Model	SI-SNRi	SDRi	# Param	Stride
Tasnet [27]	10.8	11.1	n.a	20
SignPredictionNet [28]	15.3	15.6	55.2M	8
ConvTasnet [15]	15.3	15.6	5.1M	10
Two-Step CTN [29]	16.1	n.a.	8.6M	10
DeepCASA [18]	17.7	18.0	12.8M	1
FurcaNeXt [19]	n.a.	18.4	51.4M	n.a.
DualPathRNN [17]	18.8	19.0	2.6M	1
sudo rm -rf [21]	18.9	n.a.	2.6M	10
VSUNOS [20]	20.1	20.4	7.5M	2
DPTNet* [22]	20.2	20.6	2.6M	1
Wavesplit** [23]	21.0	21.2	29M	1
Wavesplit** + DM [23]	22.2	22.3	29M	1
SepFormer	20.4	20.5	26M	8
SepFormer + DM	22.3	22.4	26M	8