LISN
LABORATOIRE INTERDISCIPLINAIRE
DES SCIENCES DU NUMÉRIQUE

IRIT
Institut de Recherche
en Informatique de Toulouse
CNRS - INP - UT3 - UT1 - UT2J

université
PARIS-SACLAY

cnrs

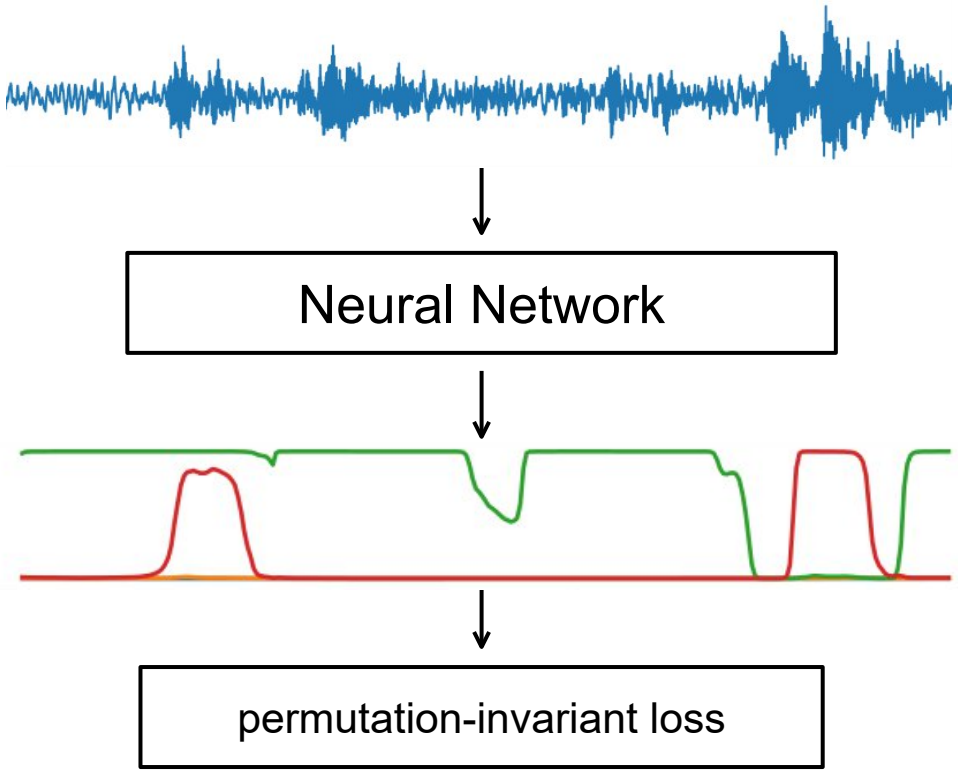
Overlap-aware low-latency online speaker diarization

based on end-to-end local segmentation

Juan M. Coria, Hervé Bredin, Sahar Ghannay, Sophie Rosset

Speaker diarization

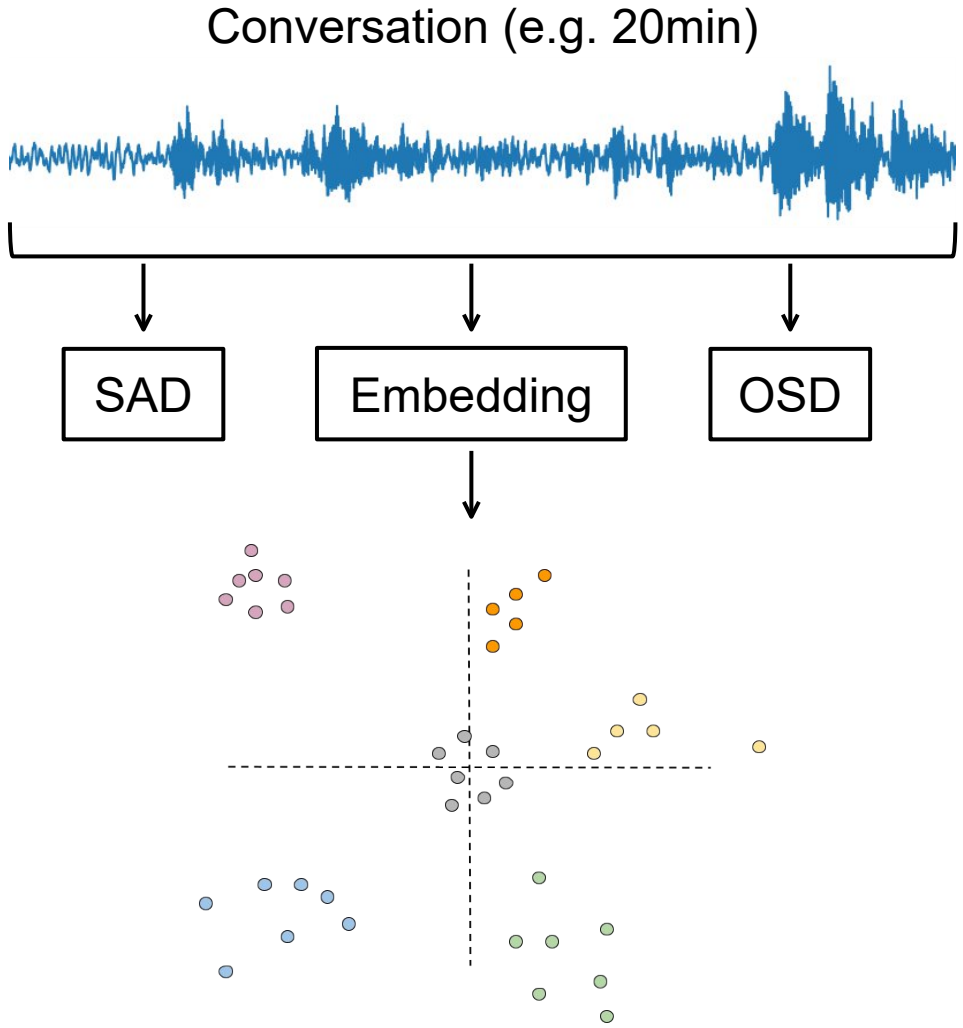
- Who spoke when?
  - Partition a conversation according to speaker identity
  - Specific speaker identity is not important
  - Essentially a clustering task



Speaker diarization

Offline

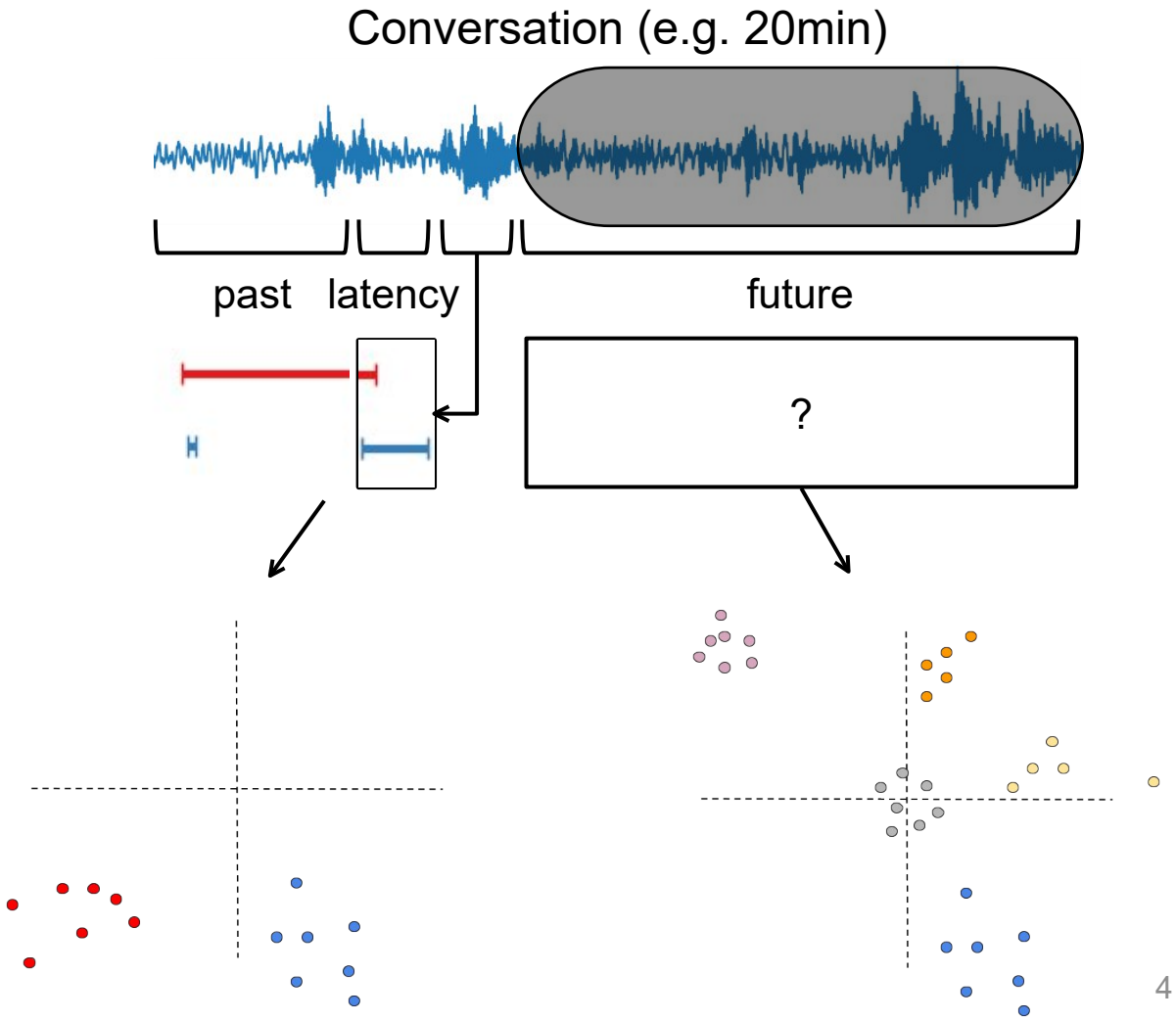
- Entire conversation available from the beginning
- Multiple passes allowed
- All speakers available at once



Speaker diarization

Online

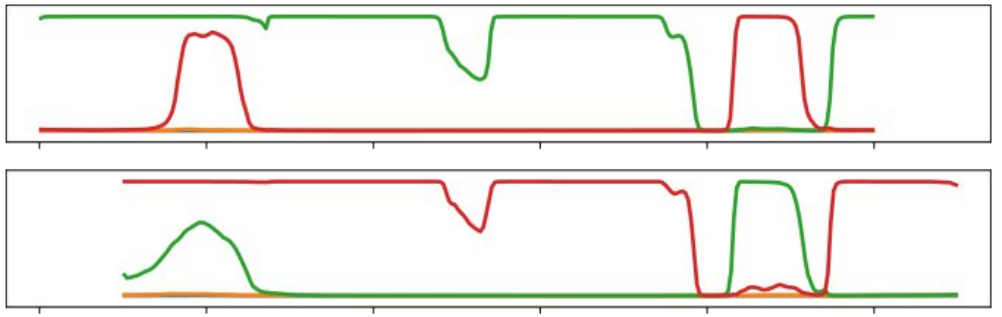
- Conversation as a stream
- Limited context
- Latency
- Detect new speakers as they arrive

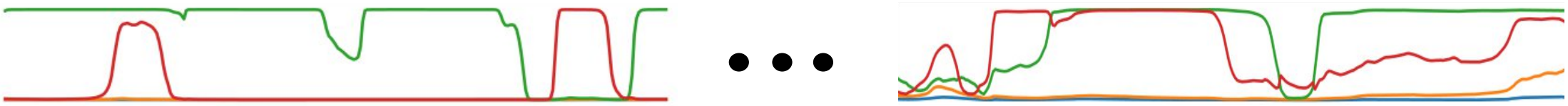


Online end-to-end speaker diarization?

Problems

- Fixed (low) number of speakers
- High latency (e.g. 30s-50s)
- High memory footprint (big input)
- Accidental speaker permutations
- False speaker re-identification





speaker A
speaker B

speaker C
speaker B

Proposed system

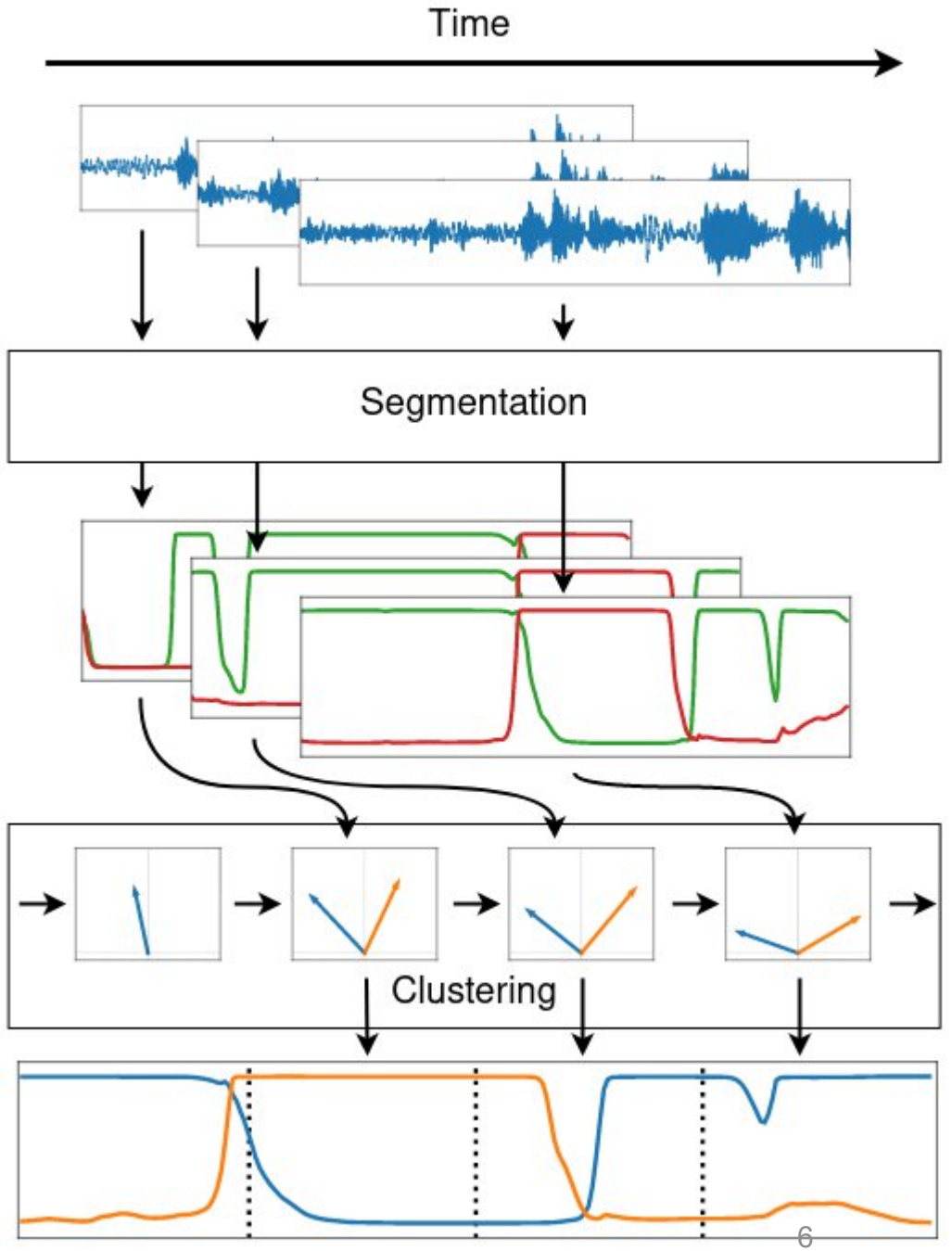
5s buffer updated every 500ms

(Bredin & Laurent, 2021) **Local end-to-end diarization**

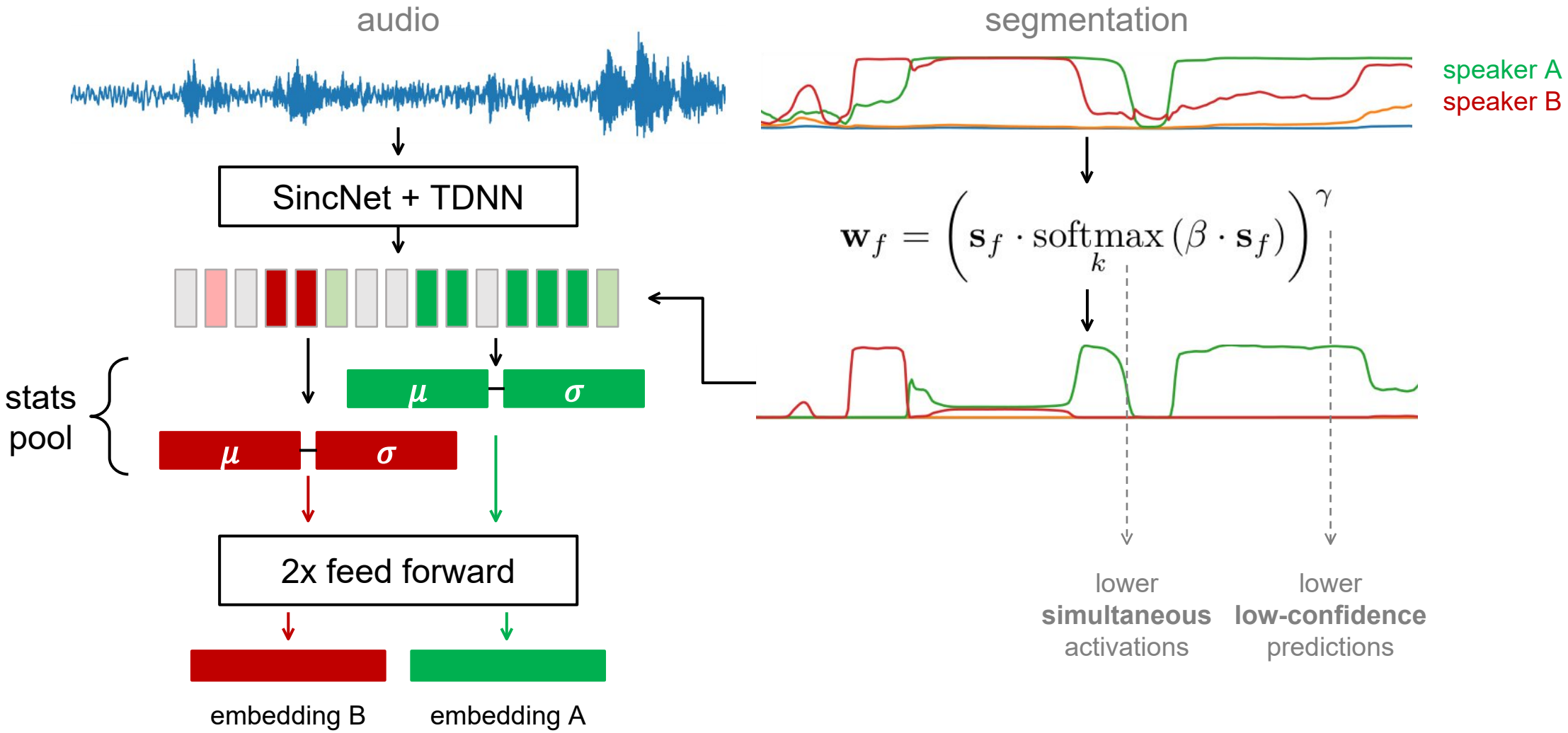
Overlap-aware speaker embedding extraction

Incremental clustering

Output reconstruction



Overlap-aware speaker embedding

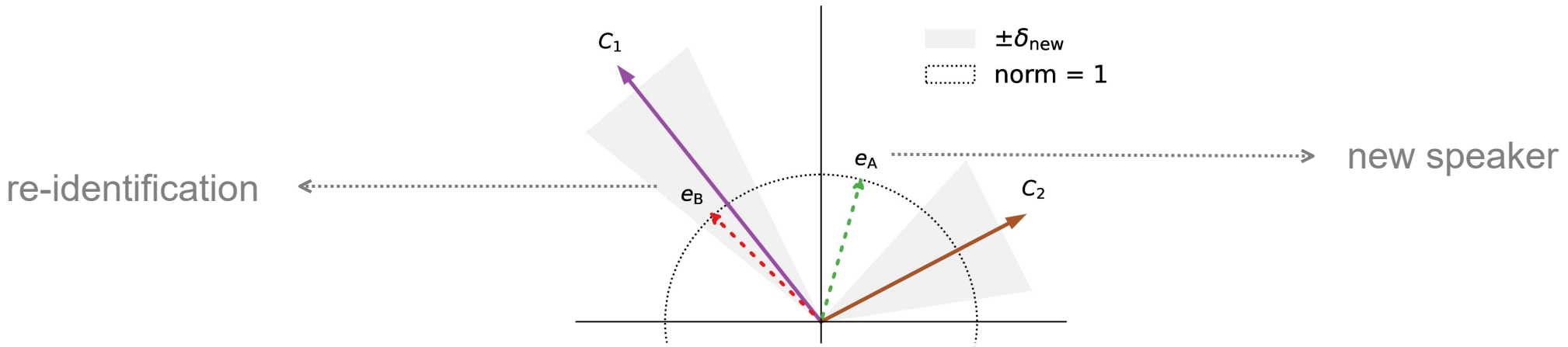


Incremental clustering

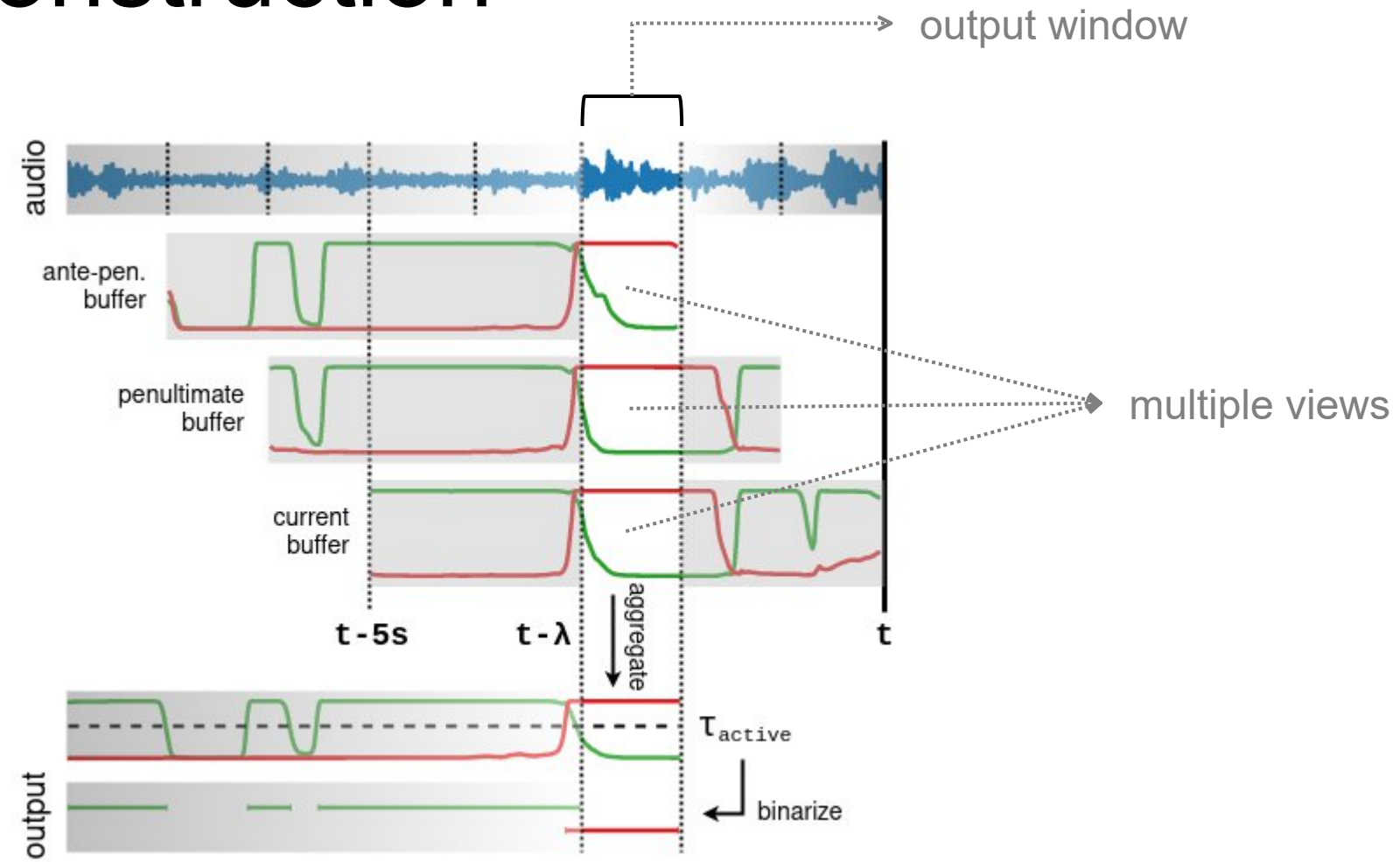
1. Assign each local speaker to the closest centroid
2. Add embeddings to corresponding centroids

Two local speakers cannot be assigned to the same centroid:

$$k \neq k' \implies m(k) \neq m(k')$$



Output reconstruction



Results

Offline

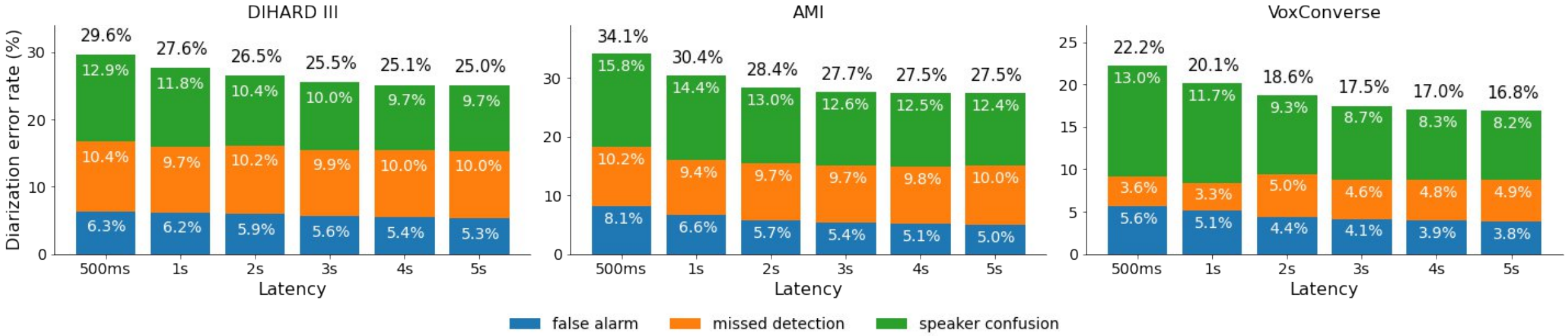
(lower is better)

System	Latency	DIHARD III [18]				AMI [19, 16]				VoxConverse [20]				DIHARD II [21]			
		FA	Miss.	Conf.	DER	FA	Miss.	Conf.	DER	FA	Miss.	Conf.	DER	FA	Miss.	Conf.	DER
VBx [16]	∞	3.6	12.5	6.2	22.3	3.1	17.2	3.8	24.1	3.1	4.6	3.4	11.1	5.0	15.3	7.4	27.7
↪ w/ overlap-aware segmentation [10]	∞	4.7	9.7	4.9	**19.3**	4.3	10.9	4.7	**19.9**	4.6	3.0	3.5	**11.1**	5.6	13.5	7.1	**26.3**
Ours	5s	5.3	10.0	9.7	**25.0**	5.0	10.0	12.4	**27.5**	3.8	4.9	8.2	**16.8**	5.7	14.0	14.4	**34.1**
↪ w/o overlap-aware embedding	5s	4.6	11.3	9.3	25.3	3.0	16.0	11.6	30.5	4.1	5.1	11.2	20.4	5.1	15.5	13.6	34.3
↪ w/ oracle segmentation	5s	2.1	1.4	6.9	10.4	1.0	1.1	15.5	17.7	0.5	0.7	9.1	10.3	2.2	1.6	12.0	15.8
FlexSTB [17]	1s	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	36.0
Ours	1s	6.2	9.7	11.8	27.6	6.6	9.4	14.4	30.4	5.1	3.3	11.7	20.1	5.8•	14.4•	14.9•	**35.1•**

Table 1. Experimental results on test sets. FA, Miss. and Conf. stand for false alarm, missed detection and speaker confusion rates respectively. (• = hyper-parameters optimized with latency $\lambda = 1$s for fair comparison)

- Offline vs online
- Overlap-aware embeddings are better across all datasets
- Better than FlexSTB
  - lower memory footprint and flexible latency

On performance vs latency

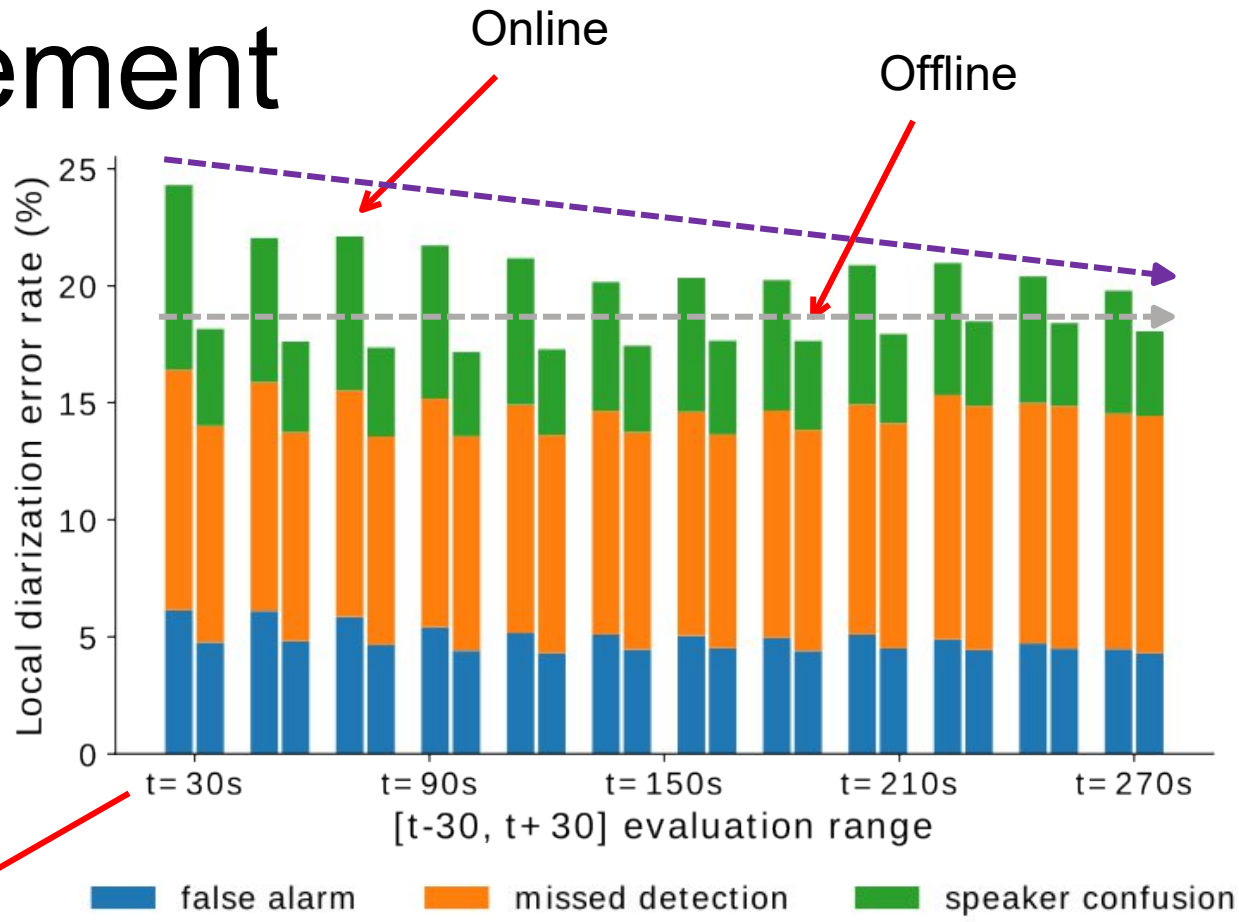


- Hyper-parameters optimized for 5s latency have reasonable performance
- Higher latency leads to lower confusion (aggregation)

On continual improvement

Ours (online) vs topline (offline)

- It almost bridges the gap after 5min
- It can handle daylong streams
- Practically constant memory cost
- Getting better and better

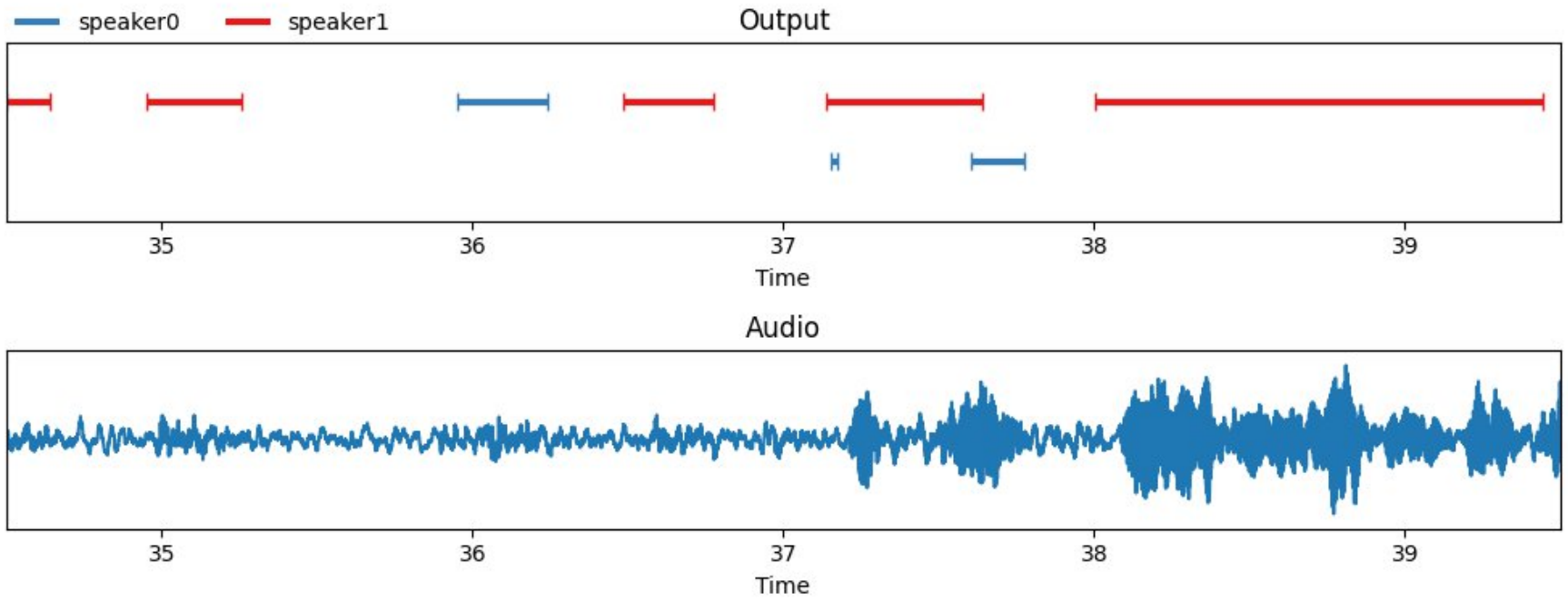


non-overlapping
60s windows

Fig. 6. Evolution of performance as conversations unfold. Left: proposed online approach with 5s latency. Right: offline *topline* [10]. Local diarization error rate computed on the 223 DIHARD III (test) conversations that are longer than 300s.

Demo

- Open source implementation at github.com/juanmc2005/StreamingSpeakerDiarization



microphone
or
streamed file

Thank you