Towards Automatic Speaker Retrieval for Large Multimedia Archives

Marijn Huijbregts and David van Leeuwen
Radboud University Nijmegen

Multimedia retrieval

What was said?

Speaker retrieval

Who said it?

Speaker retrieval task definition

Be able to find *all* speech fragments of *all* speakers in an archive, solely given the *name* of the speaker.

Why speaker retrieval?

marijn.huijbregts@let.ru.nl

Radboud Universiteit Nijmegen

Why speaker retrieval?

Why speaker retrieval?

marijn.huijbregts@let.ru.nl

Radboud Universiteit Nijmegen

Why speaker retrieval?

In combination with other techniques...

"Yes, we can!"

Why speaker retrieval?

In combination with other techniques...

"Yes, we can!"

Why speaker retrieval?

In combination with other techniques...

Can we build it?

"Yes, we can!"

Why speaker retrieval?

- Useful for other speech processing techniques:
  - Automatic speech recognition
  - Summarization
  - Dialog act tagging
- Useful for structuring information
  - visualization
  - user interfaces
- Interesting information on its own
  - When and how often was this politician on TV?
  - Find quotes of a specific person.

Towards speaker retrieval

Speaker retrieval:

Be able to find *all* speech fragments of *all* speakers in an archive, solely given the *name* of the speaker.

Towards speaker retrieval

Speaker diarization:

Be able to find *all* speech fragments of *all* speakers ~~in an archive, solely given the *name* of the speaker~~ in a short recording

Towards speaker retrieval

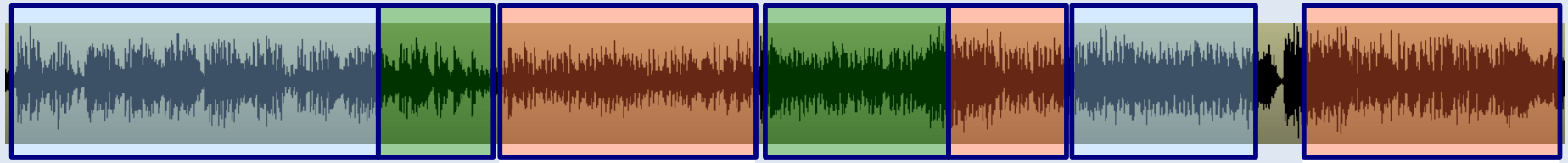
Speaker diarization:

Be able to find *all* speech fragments of *all* speakers ~~in an archive, solely given the *name* of the speaker~~ in a short recording

Towards speaker retrieval

Speaker diarization:

Be able to find *all* speech fragments of *all* speakers ~~in an archive, solely given the *name* of the speaker~~ in a short recording

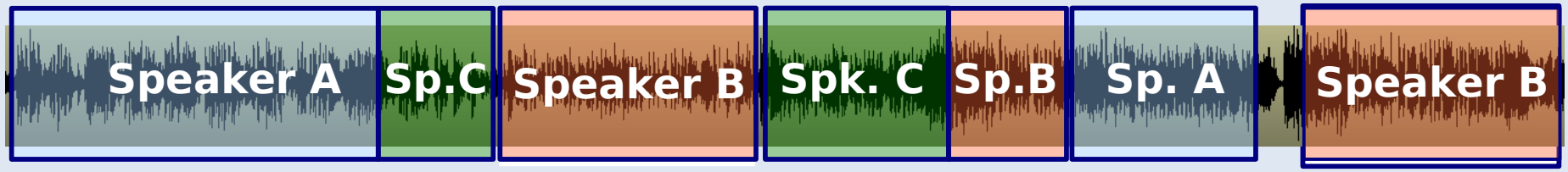


marijn.huijbregts@let.ru.nl

Towards speaker retrieval

Speaker diarization:

Be able to find *all* speech fragments of *all* speakers ~~in an archive, solely given the *name* of the speaker~~ in a short recording



marijn.huijbregts@let.ru.nl

Towards speaker retrieval

Speaker diarization:

Be able to find *all* speech fragments of *all* speakers ~~in an archive, solely given the *name* of the speaker~~ in a short recording

Towards speaker retrieval

Large Scale Speaker diarization:

Be able to find *all* speech fragments of *all* speakers ~~in an archive, solely given the *name* of the speaker~~ in a **long** recording

Towards speaker retrieval

Large Scale Speaker diarization:

Be able to find *all* speech fragments of *all* speakers ~~in an archive, solely given the *name* of the speaker~~ in a **long** recording

- Upscale to long recordings (3 hours)
- Upscale to small collections (15 hours)
- Upscale to large archives (1500 hours and up)

Towards speaker retrieval

Large Scale Speaker diarization:

Be able to find *all* speech fragments of *all* speakers in an archive, ~~solely given the *name* of the speaker~~

- Upscale to long recordings (3 hours)
- Upscale to small collections (15 hours)
- Upscale to large archives (1500 hours and up)

Towards speaker retrieval

Query-by-example speaker retrieval:

Be able to find *all* speech fragments of *all* speakers in an archive, ~~solely given the *name* of the speaker~~

Towards speaker retrieval

Query-by-example speaker retrieval:

Be able to find *all* speech fragments of *all* speakers in an archive, ~~solely given the *name* of the speaker~~

Towards speaker retrieval

Query-by-example speaker retrieval:

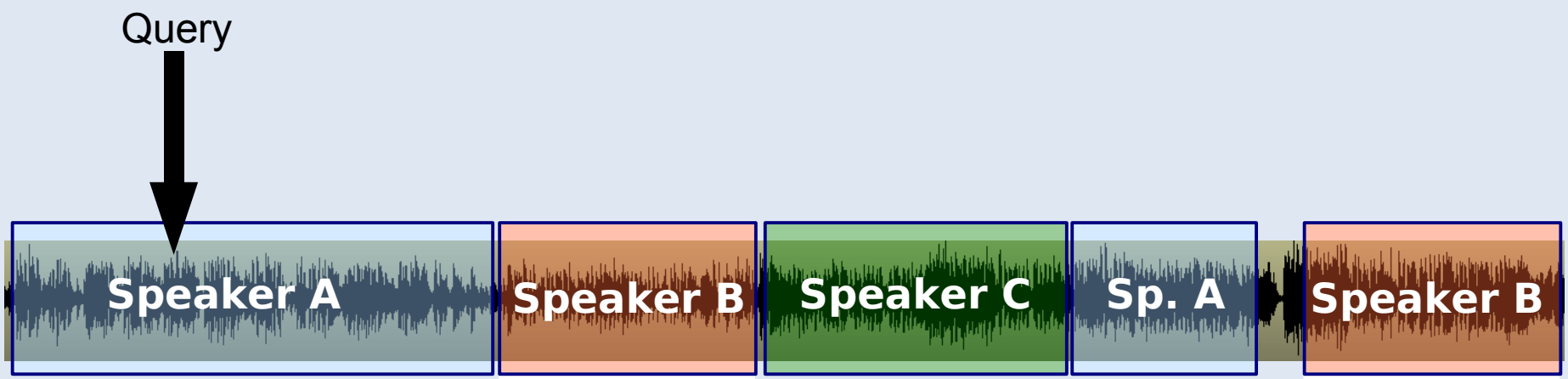
Be able to find *all* speech fragments of *all* speakers in an archive, ~~solely given the *name* of the speaker~~

marijn.huijbregts@let.ru.nl

Towards speaker retrieval

Query-by-example speaker retrieval:

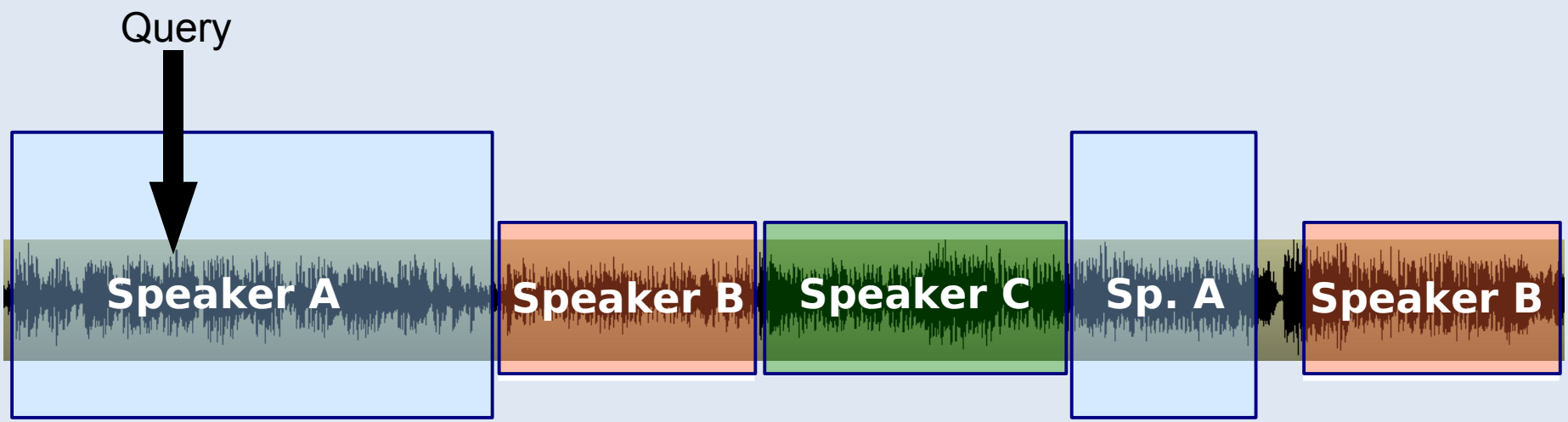
Be able to find *all* speech fragments of *all* speakers in an archive, ~~solely given the *name* of the speaker~~



Towards speaker retrieval

Query-by-example speaker retrieval:

Be able to find *all* speech fragments of *all* speakers in an archive, ~~solely given the *name* of the speaker~~



marijn.huijbregts@let.ru.nl

Towards speaker retrieval

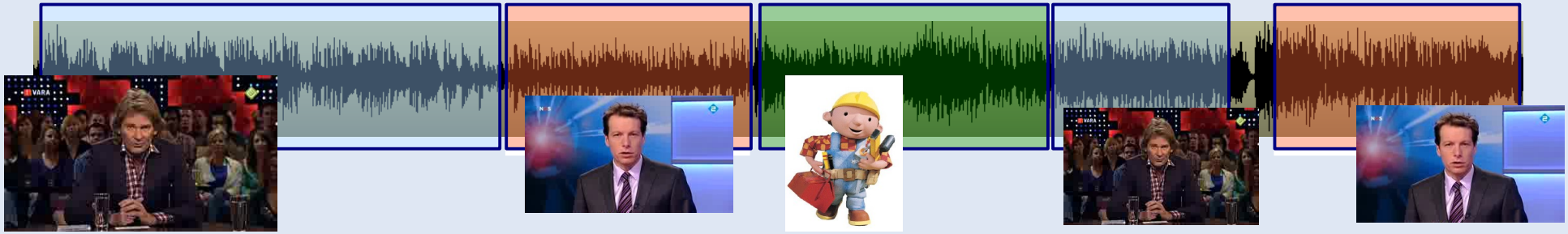
Query-by-example speaker retrieval:

Be able to find *all* speech fragments of *all* speakers in an archive, ~~solely given the *name* of the speaker~~

Towards speaker retrieval

Full speaker retrieval:

Be able to find *all* speech fragments of *all* speakers in an archive, solely given the *name* of the speaker



marijn.huijbregts@let.ru.nl

Towards speaker retrieval

- Speaker diarization
  (20 minutes)
- Large scale speaker diarization
  (3-15 hours)
- Query-by-example speaker retrieval
  (>1500 hours)
- Speaker retrieval
  (identities are known)

Towards speaker retrieval

Scaling challenges:

- Step 1: from 20 minutes to 15 hours
- Step 2: from 15 hours to 1500+ hours

Large scale speaker diarization

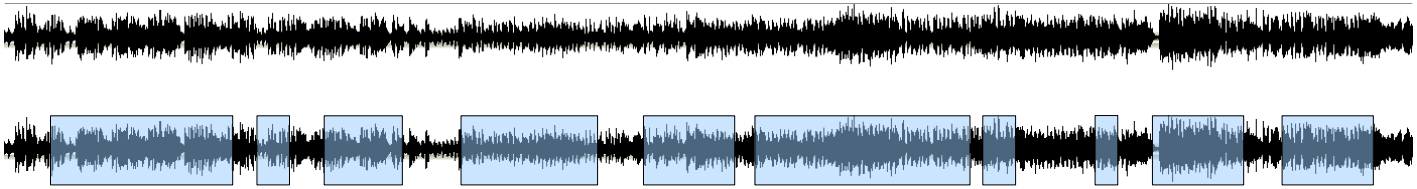
- Speaker recognition
  - Train speech segment: create a model (enrollment)
  - Test speech segment: Is this the same speaker?

- Speaker diarization
  - Segment and cluster
  - Short recordings

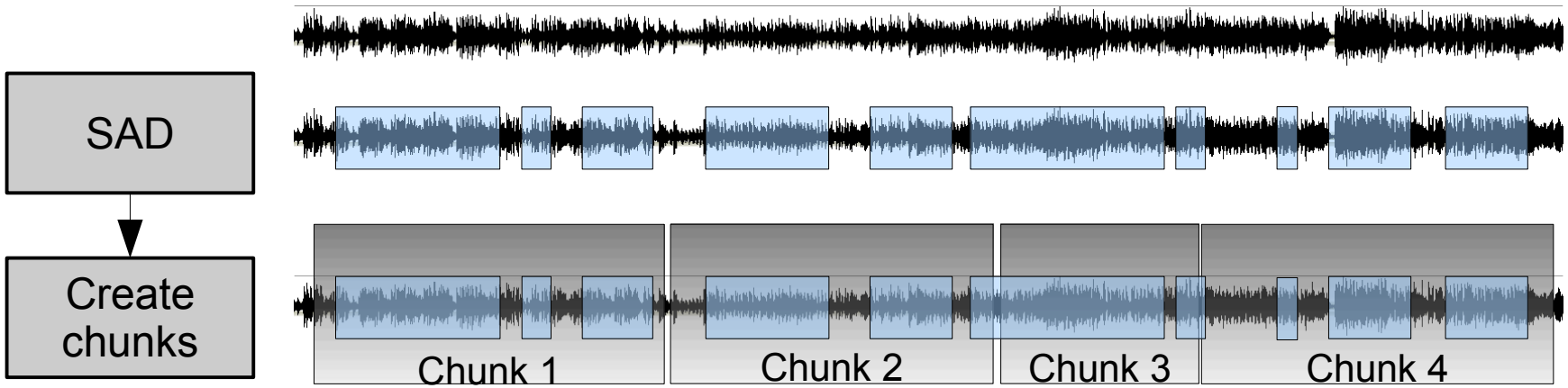
Large scale speaker diarization

Large scale speaker diarization

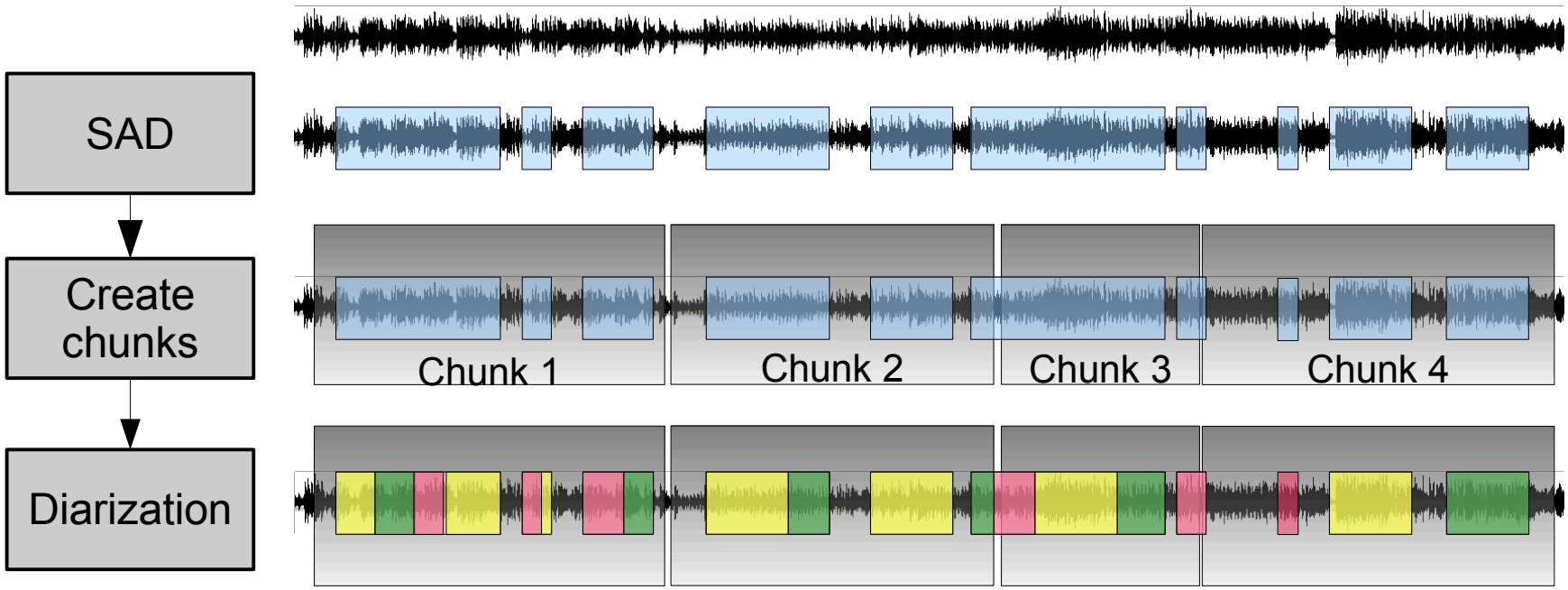
SAD



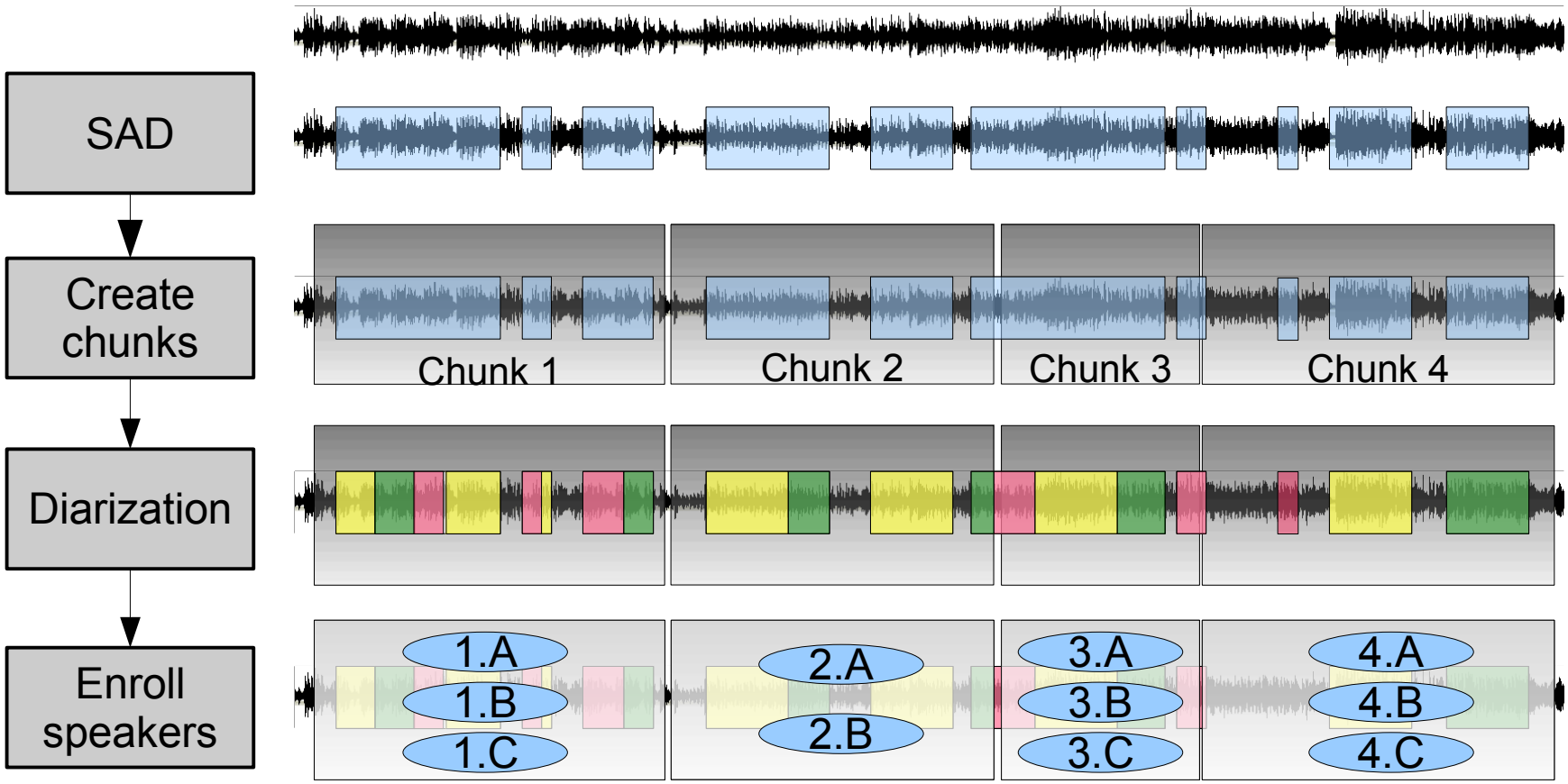
Large scale speaker diarization



Large scale speaker diarization

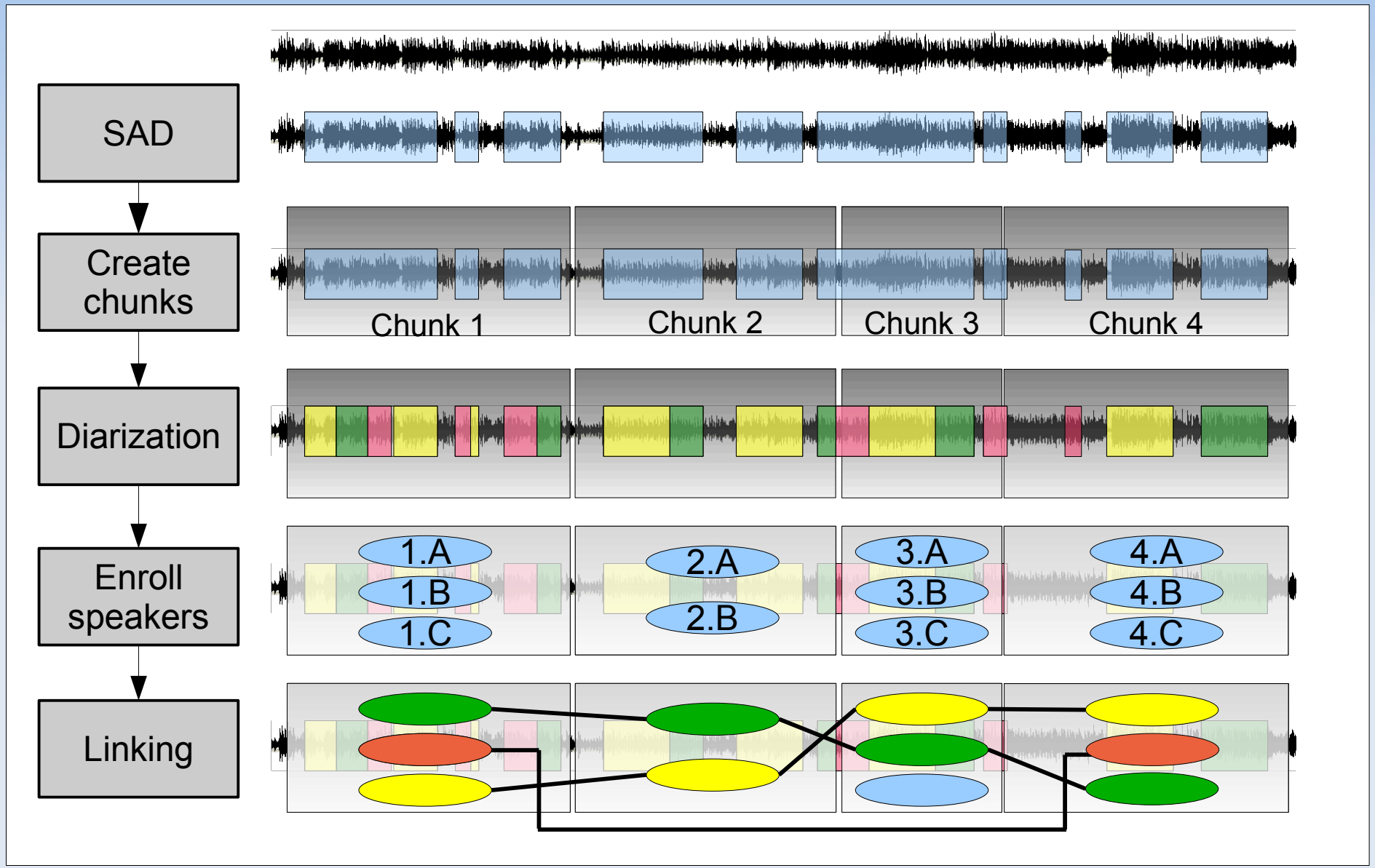


Large scale speaker diarization



marijn.huijbregts@let.ru.nl

Large scale speaker diarization



marijn.huijbregts@let.ru.nl

Large scale speaker diarization

15 ICSI meetings (44-88 minutes each):

- Exp. 1: diarization perspective
  - Evaluate using diarization error rate (DER) on individual recordings and on the entire collection.
- Exp. 2: information retrieval perspective
  - Randomly pick queries (timestamps) and define documents of 2minutes or 12seconds.
  - Evaluate the query-by-example system using Mean Average Precision (MAP) and Precision-at-ten (P@10)

Large scale speaker diarization

15 ICSI meetings (44-88 minutes each):

- Diarization Error Rate per meeting:
  - All-at-once: 14.56%DER - 1936 CPU minutes
  - Chunking: 16.14%DER - 355 CPU minutes
- Diarization Error Rate for entire collection:
  - Collection-wide-chunking, score per meeting: 17.79%DER
  - Collection-wide-chunking, score collection-wide: 25.16%DER

Large scale speaker diarization

15 ICSI meetings (44-88 minutes each):

- Retrieval experiment:

Experiment	MAP	P@10
2min, no normalization	0.57	0.74
2min, normalization	0.37	0.49
12s, no normalization	0.49	0.70
12s, normalization	0.31	0.49

Towards large archives

We have recorded 90 days of Dutch television:

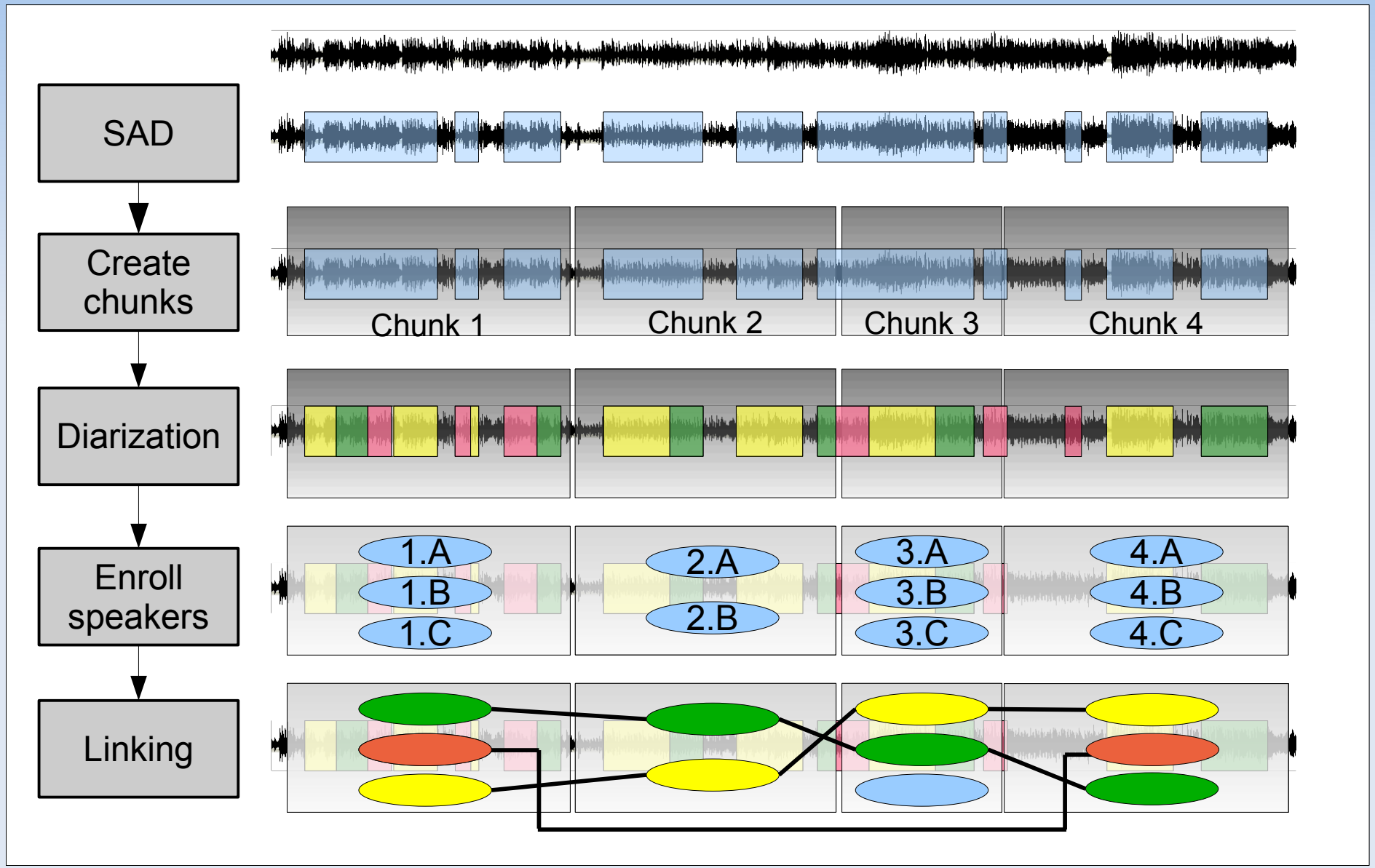
- 3 public-service channels
- From 17:30 hour until midnight
- 1755 hours of broadcast recordings in total

Towards large archives

Three challenges

- Scaling: Increase in data.
- Speaker distribution: not a coherent collection.
- Evaluation: all that data is not annotated!

Towards large archives



marijn.huijbregts@let.ru.nl

Towards large archives

Increase in data:

- First four steps performed in 0.62 times real-time
- Fifth step (scoring enrolled models) takes 0.35 times real-time, but the complexity is quadratic.

Towards large archives

Increase in data:

- First four steps performed in 0.62 times real-time
- Fifth step (scoring enrolled models) takes 0.35 times real-time, but the complexity is quadratic.

Speaker distribution:

- Makes the actual linking step very risky.
- Fortunately not needed for most applications.

Towards large archives

Increase in data:

- First four steps performed in 0.62 times real-time
- Fifth step (scoring enrolled models) takes 0.35 times real-time, but the complexity is quadratic.

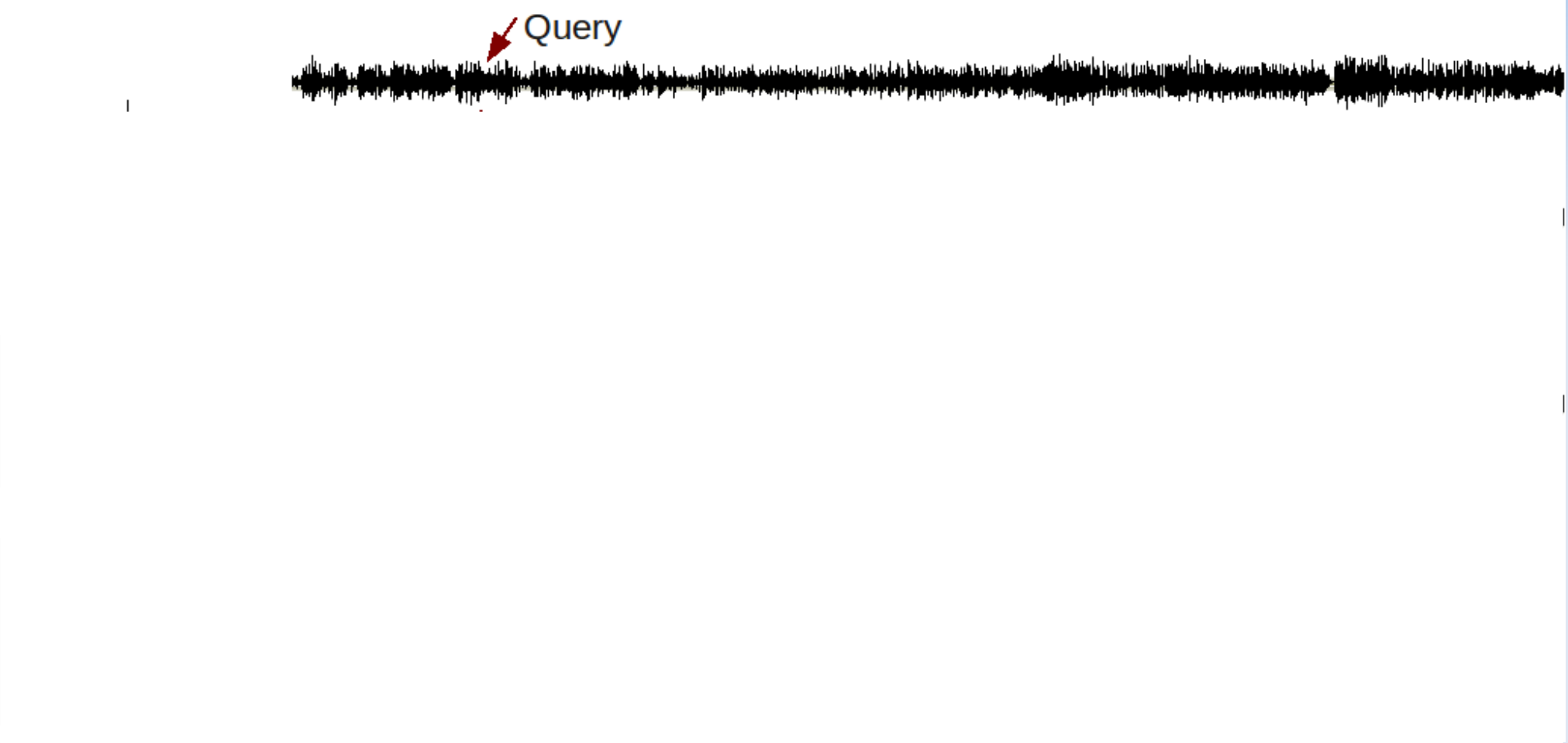
Speaker distribution:

- Makes the actual linking step very risky.
- Fortunately not needed for most applications.

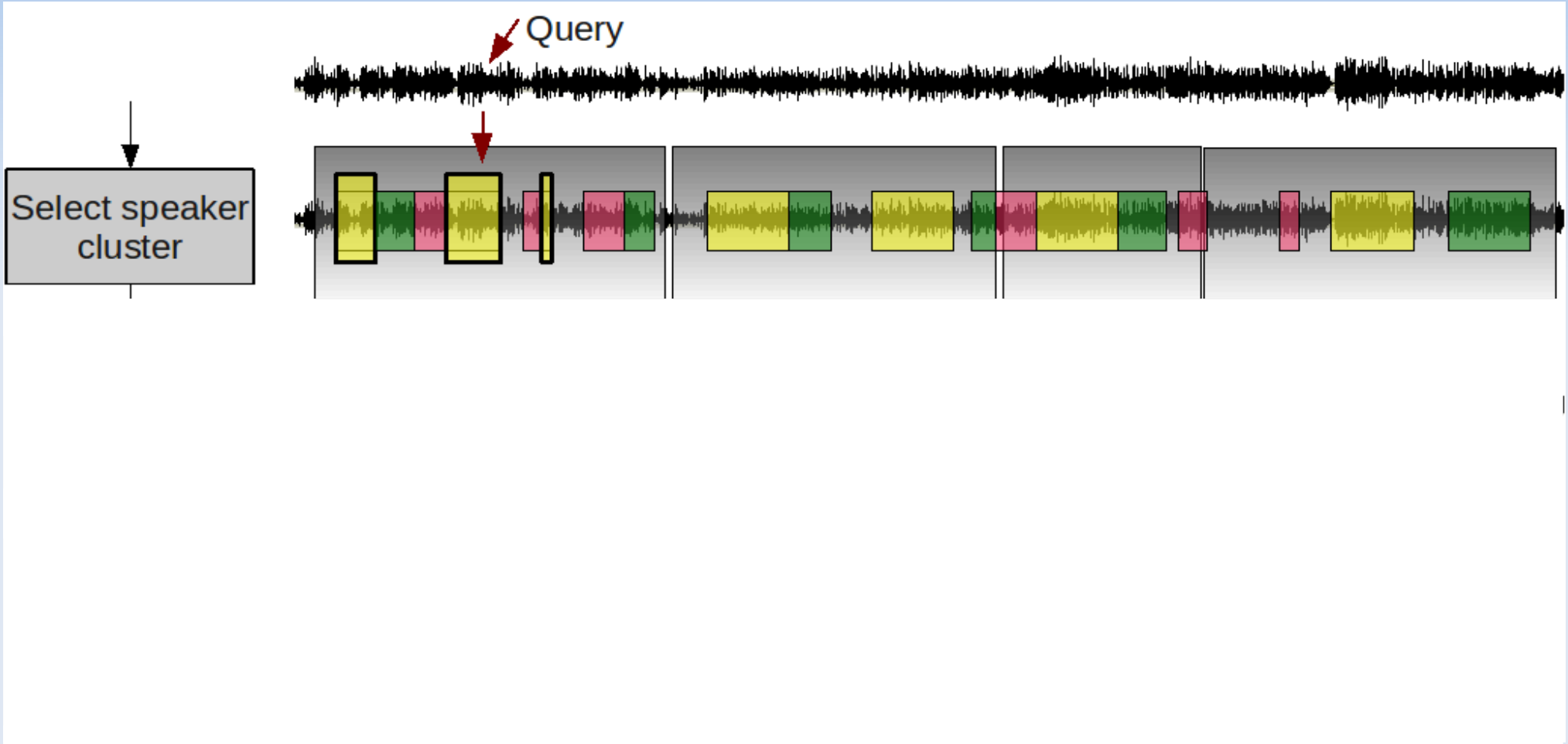
Evaluation:

- A lot of effort needed (benchmarks)
- Evaluated on P@10 only

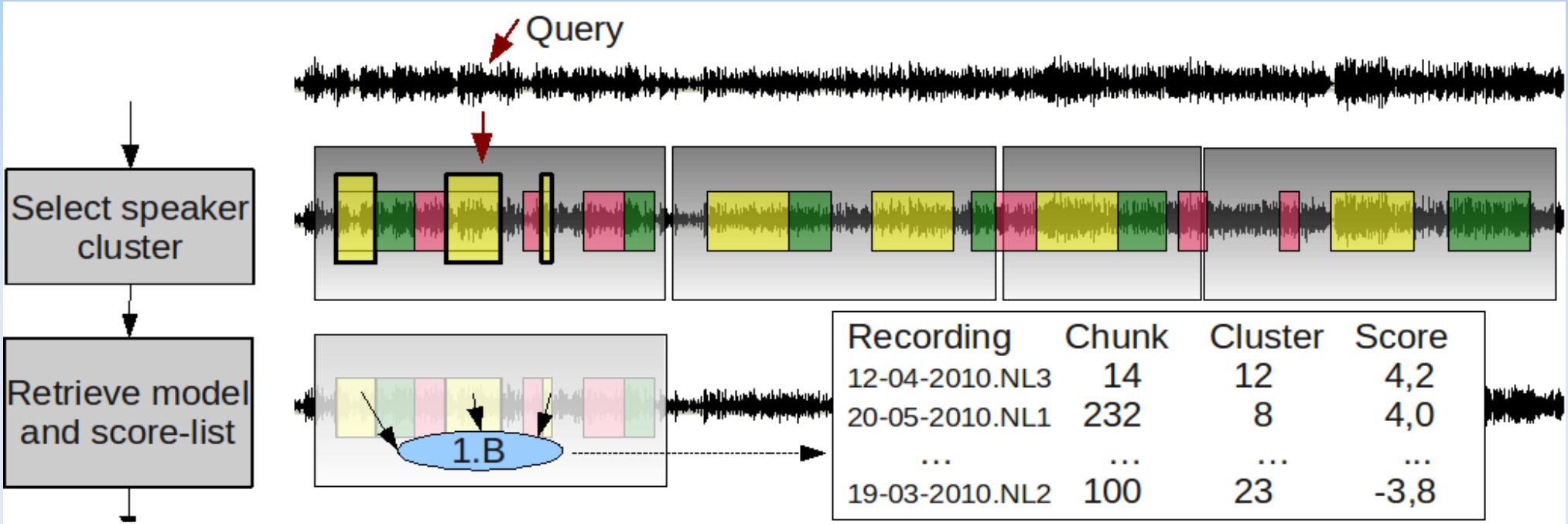
Our retrieval system



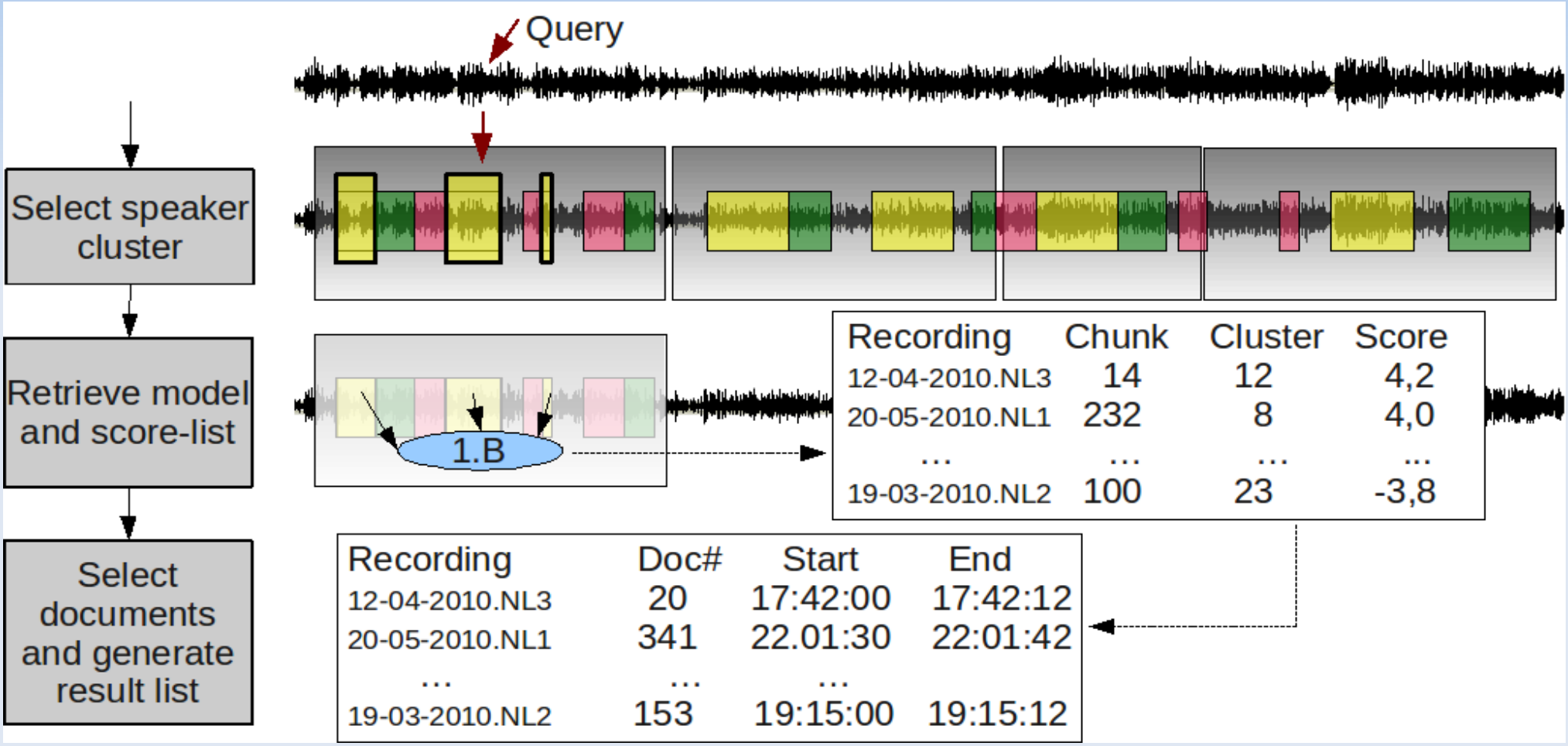
Our retrieval system



Our retrieval system



Our retrieval system



Evaluation

- Hand-picked 30 queries
  - 17 often occuring people
  - 13 less occuring people
- 12 second documents are defined
- P@10 Manually judged

Evaluation

Results:

Description	#queries	P@10
Total evaluation set	30	0.520
Frequently occurring people	17	0.800
Less frequently occurring people	13	0.154
Query represents clean cluster	18	0.756
Query represents polluted cluster	12	0.167

Conclusions

- We were able to scale our query-by-example speaker retrieval system.
- Further scaling is possible, but would take more than one CPU for the last step.
- Purity of the query cluster is very important.

Future work

- Automatically finding the identity of each speaker cluster using meta-data or ASR.
- Data mining: discovering (part of) ones identity by looking at the occurence with other speakers.
  - Always on TV at the same time-slot? → program host
  - Often on TV with other politicians? → politician
  - Lot's of politicians on TV? → election time
  - Know the background of a speaker? → topic detection

Thank you.

marijn.huijbregts@let.ru.nl