Diarization of overlapping speech:
Methods and ensembles

Desh Raj

ISCA SIG-ML Seminar
May 05, 2021
Overview

• A brief **background in diarization**:
  • The task and its applications
  • A traditional solution, and the problem of overlapping speech

• **Methods**:
  • Overlap-aware clustering
  • Separate then diarize

• **Ensemble**: An approach for combining overlap-aware diarization systems
Background

What is speaker diarization?

Task of “who spoke when”

Input: recording containing multiple speakers

Output: homogeneous speaker segments

Background

Applications of Diarization

- Psychotherapy and human interaction
- Child language acquisition
- Collaborative learning
- Meeting transcription
- Cocktail party problem
Background
What makes Diarization difficult?

Input: recording containing multiple speakers

Output: homogeneous speaker segments

1. The recording may be very long with arbitrary silences/noise.
2. Number of speakers may be unknown.
3. Overlapping speech may be present.

Example from CHiME-6 challenge (best system achieved >30% error rate)

The traditional solution
“Clustering-based” systems

- **Key idea**: formulate Diarization as a clustering problem
- Cluster small segments of audio
- Each cluster represents a distinct speaker

Clustering-based diarization
SAD extracts speech segments from recordings

Spectral energy-based
GMM-based
Hybrid HMM-DNN
End-to-end SAD
Clustering-based diarization
Embeddings extracted for small subsegments
Clustering-based diarization
Embeddings extracted for small subsegments


Clustering-based diarization
Pair-wise scoring of subsegments

Clustering-based diarization

Clustering based on the affinity matrix, followed by optional resegmentation

Agglomerative hierarchical clustering
Spectral clustering
Variational Bayes (VBx)

Clustering-based diarization
How well does it perform?

- **Winning system in DIHARD I (2018) and II (2019)**
- DIHARD contains “hard” Diarization evaluation with recordings from several domains
- But **Diarization error rates (DER) still high**: 37% in DIHARD I and 27% in DIHARD II

\[
\text{DER} = \frac{\text{Missed speech} + \text{False alarm} + \text{Speaker error}}{\text{Total speaking time}}
\]


Clustering paradigm assumes single-speaker segments

So overlapping speakers are completely ignored!

“Roughly 8% of the absolute error in our systems was from overlapping speech … it will likely require a complete rethinking of the diarization process … This is an important direction, but could not be addressed …”
- JHU team (2018)

“Given the current performance of the systems, the overlapped speech gains more relevance … more than 50% of the DER in our best systems … has to be addressed in the future …”
- BUT team (2019)
Overlap-aware spectral clustering

Overlap Detector → Speech Activity Detection → Embedding extractor → Pair-wise Scoring → Overlap-aware spectral clustering → Affinity matrix → Diarization labels

Overlap-aware spectral clustering
Overview of differences

Regular spectral clustering
(Ng-Jordan-Weiss algorithm):

• Estimate number of speakers (say, $K$)
• Compute Laplacian $L$ of affinity matrix
• Apply K-means clustering on first $K$ eigenvectors of $L$

Overlap-aware spectral clustering
Overview of differences

Regular spectral clustering
(Ng-Jordan-Weiss algorithm):

- Estimate number of speakers (say, $K$)
- Compute Laplacian $L$ of affinity matrix
- Apply K-means clustering on first $K$ eigenvectors of $L$

Overlap-aware spectral clustering
Overview of differences

Alternative formulation:
multi-class spectral clustering

New formulation for spectral clustering

The basic clustering problem: a graph view

New formulation for spectral clustering
The basic clustering problem: a graph view

Speaker A

Speaker B

Edge weights within a group

Edge weights across groups
New formulation for spectral clustering

The basic clustering problem: a graph view

Speaker A

Speaker B

maximize

Edge weights within a group

Edge weights across groups
New formulation for spectral clustering
The basic clustering problem: a graph view

The basic clustering problem: a graph view

maximize $\epsilon(X) = \frac{1}{K} \sum_{k=1}^{K} \frac{X_k^T A X_k}{X_k^T D X_k}$
subject to $X \in \{0,1\}^{N \times K}$, $X1_K = 1_N$. **K speakers, N segments**
The basic clustering problem: a graph view

**New formulation for spectral clustering**

The basic clustering problem: a graph view

**maximize**

- **Edge weights within a group**

\[ \epsilon(X) = \frac{1}{K} \sum_{k=1}^{K} \frac{X_k^T A X_k}{X_k^T D X_k} \]

**subject to**

- \( X \in \{0,1\}^{N \times K} \)
- \( X1_k = 1_N \)

Affinity matrix

Diagonal matrix containing degree of nodes
New formulation for spectral clustering

The basic clustering problem: a graph view

maximize $\epsilon(X) = \frac{1}{K} \sum_{k=1}^{K} \frac{X_k^TAX_k}{X_k^TDX_k}$

subject to $X \in \{0,1\}^{N \times K}$,

$X1_K = 1_N$. 

Final cluster assignment matrix
New formulation for spectral clustering
This problem is NP-hard!

\[
\begin{align*}
\text{maximize} \quad & e(X) = \frac{1}{K} \sum_{k=1}^{K} \frac{X_k^T A X_k}{X_k^T D X_k} \\
\text{subject to} \quad & X \in \{0, 1\}^{N \times K}, \\
& X 1_K = 1.
\end{align*}
\]

Remove the discrete constraints to make the problem solvable.
New formulation for spectral clustering

Relaxed problem has a set of solutions

\[
\begin{align*}
\text{maximize} & \quad \epsilon(X) = \frac{1}{K} \sum_{k=1}^{K} \frac{X_k^T A X_k}{X_k^T D X_k} \\
\text{subject to} & \quad X \in \{0,1\}^{N \times K}, \\
& \quad X 1_K = 1.
\end{align*}
\]

Taking the Eigen-decomposition of D^{-1}A

Set of solutions to the relaxed problem

and its orthonormal transforms
New formulation for spectral clustering

Now we need to **discretize** this solution!

Find a matrix which is **discrete** and also close to any one of the **orthonormal transformations** of the relaxed solution.

subject to

\[ X \in \{0,1\}^{N\times K}, \]
\[ X1_K = 1_N. \]
New formulation for spectral clustering

Now we need to **discretize** this solution!

and its orthonormal transforms

Iterate until convergence
Let us now make it overlap-aware

Suppose we have \( v_{OL} \)

and its orthonormal transforms

subject to

\[
X \in \{0,1\}^{N \times K},
\]

\[
X1_K = 1_N + v_{OL}.
\]

Discrete constraint is modified to include overlap detector output
Let us now make it overlap-aware
Modify non-maximal suppression to pick top 2 speakers

and its orthonormal transforms

Iterate until convergence
Hybrid HMM-DNN overlap detector
(Can also use other methods, e.g. end-to-end)
Results on AMI Mix-Headset eval
12.0% relative improvement over spectral clustering baseline

<table>
<thead>
<tr>
<th>System</th>
<th>DER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spectral clustering</td>
<td>26.9</td>
</tr>
<tr>
<td>AHC</td>
<td>28.3</td>
</tr>
<tr>
<td>VBx</td>
<td>26.2</td>
</tr>
<tr>
<td>Overlap-aware SC</td>
<td>24.0</td>
</tr>
</tbody>
</table>

AMI data contains 4-speaker meetings

---


Díez et al., “Speaker diarization based on Bayesian HMM with eigenvoice priors,” Odyssey 2018.
Results on AMI Mix-Headset eval
Comparable with other overlap-aware diarization methods

<table>
<thead>
<tr>
<th>System</th>
<th>DER</th>
</tr>
</thead>
<tbody>
<tr>
<td>VB-based overlap assignment</td>
<td>23.8</td>
</tr>
<tr>
<td>Region proposal networks</td>
<td>25.5</td>
</tr>
<tr>
<td>Overlap-aware SC</td>
<td>24.0</td>
</tr>
</tbody>
</table>

Does not require **matching training data** or **initialization** with other diarization systems.


## Results: DER breakdown on AMI eval

<table>
<thead>
<tr>
<th>System</th>
<th>Missed speech</th>
<th>False alarm</th>
<th>Speaker conf.</th>
<th>DER</th>
</tr>
</thead>
<tbody>
<tr>
<td>AHC/PLDA</td>
<td>19.9</td>
<td>0.0</td>
<td>8.4</td>
<td>26.9</td>
</tr>
<tr>
<td>Spectral/cosine</td>
<td>19.9</td>
<td>0.0</td>
<td>7.0</td>
<td>28.3</td>
</tr>
<tr>
<td>VBx</td>
<td>19.9</td>
<td>0.0</td>
<td>6.3</td>
<td>26.2</td>
</tr>
<tr>
<td>VB-based overlap assignment</td>
<td>13.0</td>
<td>3.6</td>
<td>7.2</td>
<td>23.8</td>
</tr>
<tr>
<td>RPN</td>
<td>9.5</td>
<td>7.7</td>
<td>8.3</td>
<td>25.5</td>
</tr>
<tr>
<td>Overlap-aware SC</td>
<td>11.3</td>
<td>2.2</td>
<td>10.5</td>
<td>24.0</td>
</tr>
</tbody>
</table>
Results: DER breakdown on AMI eval

Missed speech decreases significantly

<table>
<thead>
<tr>
<th>System</th>
<th>Missed speech</th>
<th>False alarm</th>
<th>Speaker conf.</th>
<th>DER</th>
</tr>
</thead>
<tbody>
<tr>
<td>AHC/PLDA</td>
<td>19.9</td>
<td>0.0</td>
<td>8.4</td>
<td>26.9</td>
</tr>
<tr>
<td>Spectral/cosine</td>
<td>19.9</td>
<td>0.0</td>
<td>7.0</td>
<td>28.3</td>
</tr>
<tr>
<td>VBx</td>
<td>19.9</td>
<td>0.0</td>
<td>6.3</td>
<td>26.2</td>
</tr>
<tr>
<td>VB-based overlap assignment</td>
<td>13.0</td>
<td>3.6</td>
<td>7.2</td>
<td>23.8</td>
</tr>
<tr>
<td>RPN</td>
<td>9.5</td>
<td>7.7</td>
<td>8.3</td>
<td>25.5</td>
</tr>
<tr>
<td>Overlap-aware SC</td>
<td>11.3</td>
<td>2.2</td>
<td>10.5</td>
<td>24.0</td>
</tr>
</tbody>
</table>
## Results: DER breakdown on AMI eval

### Speaker confusion increases

<table>
<thead>
<tr>
<th>System</th>
<th>Missed speech</th>
<th>False alarm</th>
<th>Speaker conf.</th>
<th>DER</th>
</tr>
</thead>
<tbody>
<tr>
<td>AHC/PLDA</td>
<td>19.9</td>
<td>0.0</td>
<td>8.4</td>
<td>26.9</td>
</tr>
<tr>
<td>Spectral/cosine</td>
<td>19.9</td>
<td>0.0</td>
<td>7.0</td>
<td>28.3</td>
</tr>
<tr>
<td>VBx</td>
<td>19.9</td>
<td>0.0</td>
<td>6.3</td>
<td>26.2</td>
</tr>
<tr>
<td>VB-based overlap assignment</td>
<td>13.0</td>
<td>3.6</td>
<td>7.2</td>
<td>23.8</td>
</tr>
<tr>
<td>RPN</td>
<td>9.5</td>
<td>7.7</td>
<td>8.3</td>
<td>25.5</td>
</tr>
<tr>
<td>Overlap-aware SC</td>
<td>11.3</td>
<td>2.2</td>
<td>10.5</td>
<td>24.0</td>
</tr>
</tbody>
</table>
Need more robust x-vector extractors

T-SNE plot of x-vector embeddings
More results: DER on LibriCSS

LibriCSS data contains 8-speaker meetings
Overlap-aware Diarization
Several new methods proposed recently


A different paradigm
Separate, then diarize

- Work on small windows (2 to 8 seconds) — assume at most 2/3 speakers in the window
- Stitch the window output streams

Traditional single-speaker clustering systems can be used in this method!

Results on LibriCSS data

Using 2 different continuous speech separation methods

Mask-based MVDR

2.4s chunks; 2 streams

Takuya Yoshioka, Hakan Erdogan, Zhuo Chen, and Fil Alleva, “Multi-microphone neural speech separation for far-field multi-talker speech recognition,” ICASSP 2018

Sequential neural BF

10s chunks; 3 streams

Works well in practice
Winner of VoxSRC Track 4 (Diarization)

Fig. 1. System Diagram

<table>
<thead>
<tr>
<th>Team</th>
<th>Method</th>
<th>DER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Huawei</td>
<td>VBx</td>
<td>9.5</td>
</tr>
<tr>
<td>Sugou</td>
<td>VB-based overlap assignment</td>
<td>7.2</td>
</tr>
<tr>
<td>DKU-DukeECE</td>
<td>VB-based overlap assignment</td>
<td>6.5*</td>
</tr>
<tr>
<td>BUT</td>
<td>VBx + overlap handling</td>
<td>4.0</td>
</tr>
<tr>
<td>Microsoft</td>
<td>CSS + spectral clustering</td>
<td>3.7</td>
</tr>
</tbody>
</table>

Machine learning tasks benefit from an ensemble of systems.

For example, ROVER is a popular combination method for ASR systems.

Problem

Why is it hard to combine diarization systems?

• System outputs may have different number of speaker estimates.
• System outputs are usually in different label space.
• There may not be agreement on whether a region contains overlap.
Solution

DOVER-Lap performs “map and vote”

- System outputs may have different number of speaker estimates.
- System outputs are usually in different label space.
- There may not be agreement on whether a region contains overlap.

Label mapping: Maximal matching algorithm based on a global cost tensor

Solution
DOVER-Lap performs “map and vote”

- System outputs may have different number of speaker estimates.
- System outputs are usually in different label space.
- There may not be agreement on whether a region contains overlap.

Label voting: Weighted majority voting considers speaker count in region

DOVER-Lap extends DOVER
Diarization Output Voting Error Reduction

Hypothesis A  e.g. AHC
Hypothesis B  e.g. SC
Hypothesis C  e.g. VBx

Assumption: The input hypotheses do not contain overlapping segments.

Preliminary: how DOVER works

Pair-wise incremental label mapping

Hypothesis A

Hypothesis B

Hypothesis C
Preliminary: how DOVER works
Pair-wise incremental label mapping

Hypothesis A

<table>
<thead>
<tr>
<th></th>
<th>a₁</th>
<th>a₂</th>
<th>a₃</th>
<th>a₄</th>
</tr>
</thead>
<tbody>
<tr>
<td>b₁</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b₂</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b₃</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Hypothesis B

<table>
<thead>
<tr>
<th></th>
<th>b₁</th>
<th>b₂</th>
<th>b₃</th>
</tr>
</thead>
<tbody>
<tr>
<td>a₁</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a₂</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a₃</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a₄</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Hypothesis C

<table>
<thead>
<tr>
<th></th>
<th>c₁</th>
<th>c₂</th>
<th>c₃</th>
<th>c₄</th>
</tr>
</thead>
</table>

Score all speaker pairs

Hungarian algorithm

This is the same algorithm that is used to map hypothesis to reference for DER computation.
Preliminary: how DOVER works
Pair-wise incremental label mapping
Preliminary: how DOVER works
Pair-wise incremental label mapping

Hypothesis A

1 2 3 4

Hypothesis B

2 1 4

Hypothesis C

1 3 4
Preliminary: how DOVER works
Label voting using rank-weighting

Hypothesis A

Hypothesis B

Hypothesis C

Speaker 1

Speaker 2
Preliminary: how DOVER works
Label voting using rank-weighting

Hypothesis A

Hypothesis B

Hypothesis C

DOVER

Speaker 1

Speaker 2
Preliminary: how DOVER works
Label voting using rank-weighting

Hypothesis A
Hypothesis B
Hypothesis C
DOVER

Speaker 1
Speaker 2

Voting using rank-based weights

53
Preliminary: how DOVER works
Label voting using rank-weighting

Hypothesis A
Hypothesis B
Hypothesis C
DOVER

Speaker 1
Speaker 2
2 limitations of DOVER

1. Incremental pair-wise label assignment does not give optimal mapping
2. Voting method does not handle overlapping speaker segments
DOVER-Lap label mapping
Change incremental method to global

Hypotheses can contain overlapping segments.
DOVER-Lap label mapping
Compute “tuple costs” for all tuples

Hypothesis A
Hypothesis B
Hypothesis C

\[x + y + z\]
DOVER-Lap label mapping

Compute “tuple costs” for all tuples
DOVER-Lap label mapping

Compute “tuple costs” for all tuples

Hypothesis A

Hypothesis B

Hypothesis C
DOVER-Lap label mapping

This gives us a “global” cost tensor

Hypothesis A

Hypothesis B

Hypothesis C

Global cost tensor
DOVER-Lap label mapping
Pick tuple with the lowest cost and assign them same label
DOVER-Lap label mapping
Pick tuple with the lowest cost and assign them same label
DOVER-Lap label mapping
Discard all tuples containing these labels
DOVER-Lap label mapping
Pick tuple with lowest cost in remaining tensor
DOVER-Lap label mapping

Pick tuple with lowest cost in remaining tensor
DOVER-Lap label mapping
Repeat until no tuples are remaining
DOVER-Lap label mapping
Repeat until no tuples are remaining
DOVER-Lap label mapping

If no tuples remaining but labels left to be mapped, remove filled dimensions and repeat
DOVER-Lap label mapping

Final mapped labels

Hypothesis A

1
2
3
4

Hypothesis B

1
2
3

Hypothesis C

1
2
3
4
DOVER-Lap label voting
Consider 3 hypotheses from overlap-aware diarization systems

Hypothesis A

Hypothesis B

Hypothesis C

Speaker 1

Speaker 2
DOVER-Lap label voting
Divide into regions (similar to DOVER)
DOVER-Lap label voting

Estimate number of speakers in each region

Hypothesis A

Hypothesis B

Hypothesis C

Estimated # of speakers

1 1 1 1 2 2 1 1 1 1 1 1

# speakers = weighted mean of # speakers in hypotheses

Weights -> obtained by ranking hypotheses by total cost
DOVER-Lap label voting
Assign highest weighted N speakers in each region

Hypothesis A

Hypothesis B

Hypothesis C

DOVER-Lap

Speaker 1

Speaker 2
### DOVER-Lap results: AMI

Effect of global label mapping algorithm

<table>
<thead>
<tr>
<th>System</th>
<th>Spk. conf.</th>
<th>DER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overlap-aware SC</td>
<td>10.1</td>
<td>23.6</td>
</tr>
<tr>
<td>VB-based overlap assignment*</td>
<td>9.6</td>
<td>21.5</td>
</tr>
<tr>
<td>Region proposal network</td>
<td>8.3</td>
<td>25.5</td>
</tr>
<tr>
<td>Average</td>
<td>9.3</td>
<td>23.5</td>
</tr>
<tr>
<td>DOVER</td>
<td>10.6</td>
<td>30.5</td>
</tr>
<tr>
<td>+ global label mapping</td>
<td>5.1</td>
<td>25.0</td>
</tr>
</tbody>
</table>

AMI data contains 4-speaker meetings
DOVER-Lap results: AMI
Effect of rank-weighted majority voting

<table>
<thead>
<tr>
<th>System</th>
<th>Spk. conf.</th>
<th>DER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overlap-aware SC</td>
<td>10.1</td>
<td>23.6</td>
</tr>
<tr>
<td>VB-based overlap assignment*</td>
<td>9.6</td>
<td>21.5</td>
</tr>
<tr>
<td>Region proposal network</td>
<td>8.3</td>
<td>25.5</td>
</tr>
<tr>
<td>Average</td>
<td>9.3</td>
<td>23.5</td>
</tr>
<tr>
<td>DOVER</td>
<td>10.6</td>
<td>30.5</td>
</tr>
<tr>
<td>+ global label mapping</td>
<td>5.1</td>
<td>25.0</td>
</tr>
<tr>
<td>DOVER-Lap</td>
<td>7.6</td>
<td>20.3</td>
</tr>
</tbody>
</table>
Results: Breakdown on LibriCSS
Effectively combines complementary strengths

LibriCSS data contains **8-speaker meetings**
Results from DIHARD-3

Top 2 teams used DOVER-Lap for system fusion in DIHARD-3

#1: USTC team combined clustering, separation-based, and TS-VAD systems

#2: Hitachi-JHU team combined VB-based and EEND-based systems


New analysis

Label mapping is a graph partitioning problem, DOVER-Lap algorithm is exponential!

New analysis
Modified Hungarian algorithm is fast and accurate

<table>
<thead>
<tr>
<th>System</th>
<th>Spk. conf.</th>
<th>DER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agglomerative Hierarchical</td>
<td>10.1</td>
<td>23.6</td>
</tr>
<tr>
<td>VB-based overlap assignment*</td>
<td>9.6</td>
<td>21.5</td>
</tr>
<tr>
<td>Region proposal network</td>
<td>8.3</td>
<td>25.5</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>9.3</strong></td>
<td><strong>23.5</strong></td>
</tr>
<tr>
<td><strong>DOVER-Lap</strong></td>
<td><strong>7.6</strong></td>
<td><strong>20.3</strong></td>
</tr>
<tr>
<td><strong>Hungarian (modified)</strong></td>
<td><strong>8.2</strong></td>
<td><strong>20.9</strong></td>
</tr>
</tbody>
</table>

A word on evaluation
https://github.com/desh2608/spyder

- 4-5x faster than md-eval.pl and dscore
- Use from Python or as CLI tool
- Selectively evaluate on single-speaker or overlapping regions
- Other metrics (JER) coming soon…

```python
import spyder

# reference (ground truth)
ref = [("A", 0.0, 2.0), # (speaker, start, end)
      ("B", 1.5, 3.5),
      ("A", 4.0, 5.1)]

# hypothesis (diarization result from your algorithm)
hyp = [("1", 0.0, 0.8),
       ("2", 0.6, 2.3),
       ("3", 2.1, 3.9),
       ("1", 3.8, 5.2)]

metrics = spyder.DER(ref, hyp)
print(metrics)
# DERMetrics(duration=5.10,miss=9.80%,falsealarm=21.57%,conf=25.49%,der=56.80%)
```
Summary

Diarization is a useful but difficult task.

Clustering-based systems fall short on handling overlapping speech, but small modifications inspired from mathematical insights can change this.

Continuous Speech Separation (CSS) works well with clustering-based systems, but well-trained separation module is required.

Ensembles work. Use DOVER-Lap for your challenge submissions.
Collaborators

**Overlap-aware spectral clustering:** Zili Huang, Sanjeev Khudanpur

**CSS-based diarization:** Zhuo Chen, Hakan Erdogan, Maokui He, Zili Huang, Shinji Watanabe

**DOVER-Lap:** Paola Garcia, Zili Huang, Shinji Watanabe, Dan Povey, Andreas Stolcke, Sanjeev Khudanpur
Acknowledgments

Some of the work reported here was done during JSALT 2020 at JHU, with support from Microsoft, Amazon, and Google. Thanks to all the members of the team for insightful discussions.

Special thanks to:
Takuya Yoshioka (Microsoft), for providing data simulation scripts.
Shota Horiguchi (Hitachi) for suggesting a modification for DOVER-Lap.