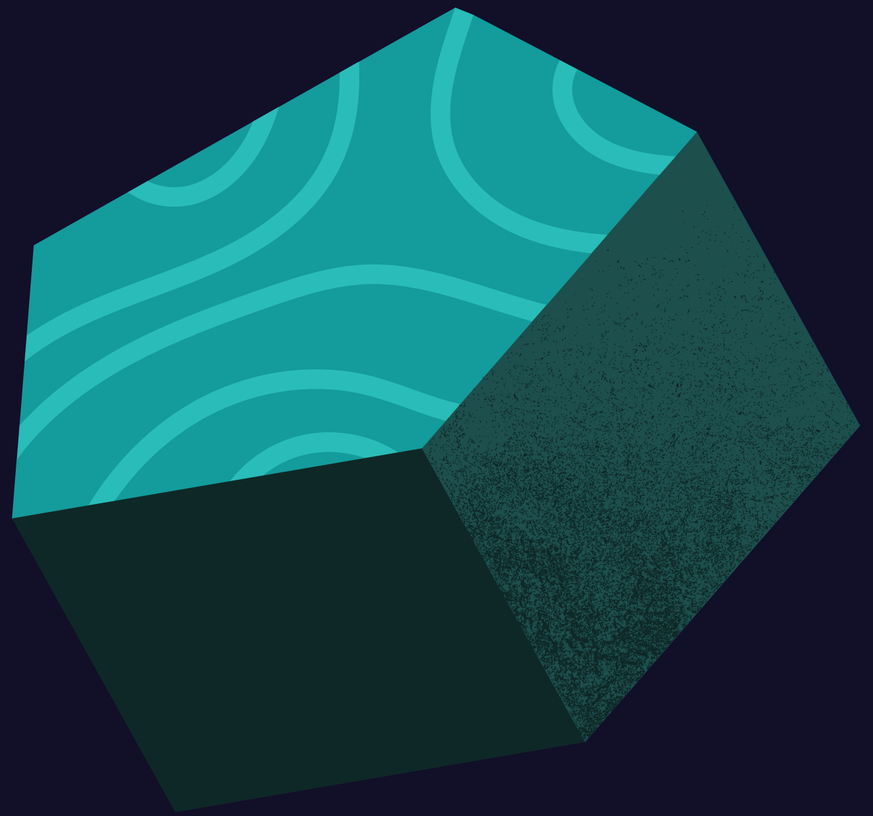
Speaker Diarization

MTC-AIC2

Today's Agenda

1. Introduction
2. ASR Model Improvement
3. Offline Diarization
4. Online Diarization
5. Results
6. Online Diarization Challanges
7. Results



Omar Ismail

Mohamed Motawie

Mohamed Tamer

Introduction

Speaker diarization is the process of partitioning an audio stream into segments based on the identity of the speaker. Essentially, it answers the question "who spoke when" in a given audio or video recording. This technology is commonly used in various applications, including:

- **Transcription Services**: Differentiating speakers in a transcription, making it clear who is speaking at any given time.
- **Meetings and Conferences**: Identifying different speakers in meeting recordings, which is helpful for minutes or notes.
- **Broadcast Media**: Segregating different speakers in radio and TV shows, podcasts, and other multimedia content.

Unigram Model Improvement

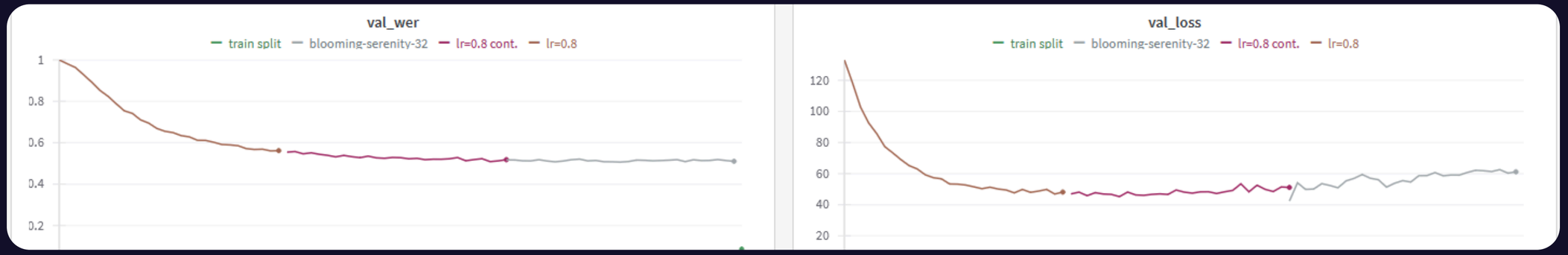
We made the following improvements to the unigram based model:

- Use precision 16 instead of 32.
- Add the missing tokens and thus removing the english characters from the model tokens.
- Limit max token length to 2 characters to prevent whole words from being added to the model vocabulary


```
'<unk>', '<fill>', '<overlap>', '<laugh>', '__', ' ', '▁م', '▁ف', '▁ق', '▁نا', '▁ش', '▁و', '▁ب', '▁ك', '▁د', '▁ع', '▁ح', '▁ر', '▁أ', '▁ا', '▁ب', '▁ه', '▁ي', '▁ت', 'ال',
'في', '▁إ', '▁غ', '▁ص', '▁لي', '▁ط', '▁ان', '▁يه', '▁أ', '▁ده', '▁ع', '▁ني', '▁لا', '▁ما', '▁ك', '▁ج', '▁إن', '▁ها', '▁ين', '▁عا', '▁بع', '▁ض', '▁دي', '▁وا', '▁من', '▁ه', '▁ف', '▁لى', '▁ول', '▁له', '▁كل', '▁يا', 'ا
ت', '▁ع', '▁ز', '▁ار', '▁بي', '▁يا', '▁حا', '▁دا', '▁مش', '▁في', '▁كن', '▁لأ', '▁هو', '▁را', '▁ست', '▁عم', '▁ج', '▁ور', '▁ذ', '▁ى', '▁لو', '▁طب', '▁ظ', '▁يب', '▁ره', '▁إ', '▁قو', '▁حد', '▁ث', '▁مع', '▁حد', '▁رب', 'جـ
ه', '▁اع', '▁نه', '▁ام', '▁اء', '▁شر', '▁صر', '▁رف', '▁زي', '▁اب', '▁عن', '▁كر', '▁يق', '▁كو', '▁رو', '▁جا', '▁قا', '▁قد', '▁يز', '▁وق', '▁ضر', '▁صل', '▁رض', '▁اج', '▁ئ', '▁ؤ', '▁ء', '▁آ', '▁ً', '▁ق', '▁،', '▁ِ', '▁١',
ج', '▁أ']
```

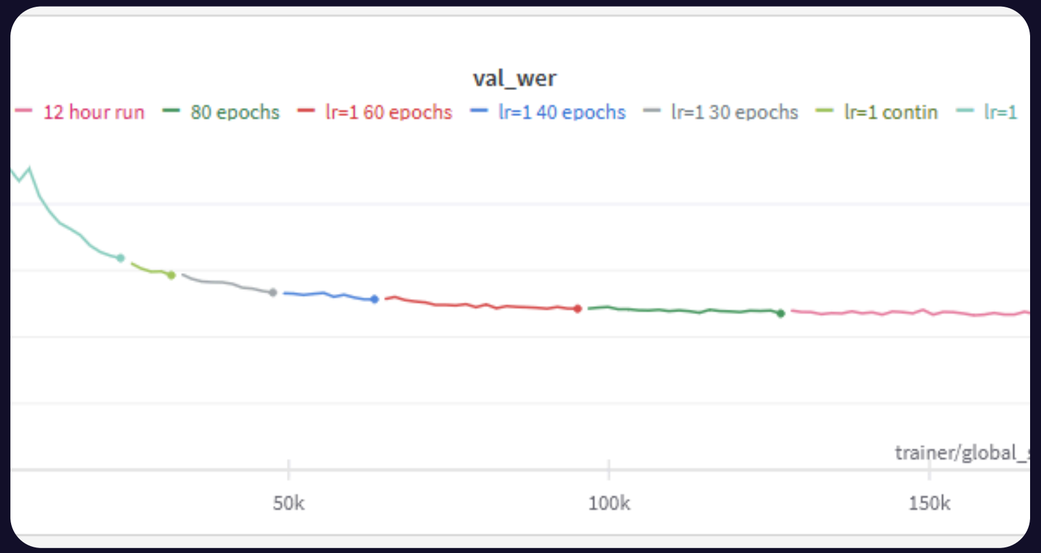
Unigram Model Unstability

- Reducing precision to 16 made the model unstable when using the same learning rate as 32 version.
- To make the training stable the learning rate had to be reduced to 0.8.
- However, lowering it caused the loss and WER to plateau in an early stage of training compared to the char based moel.



Char Model Continutation

- Due to the instability in the unigram model, we continued training the char model.
- The char-based tokenizer is appropriate for Arabic as it has complex letters with various diacritics and forms.
- Achieve 10.840672 MLD, a significant improvement over the 12.995944 MLD achieved in phase one.

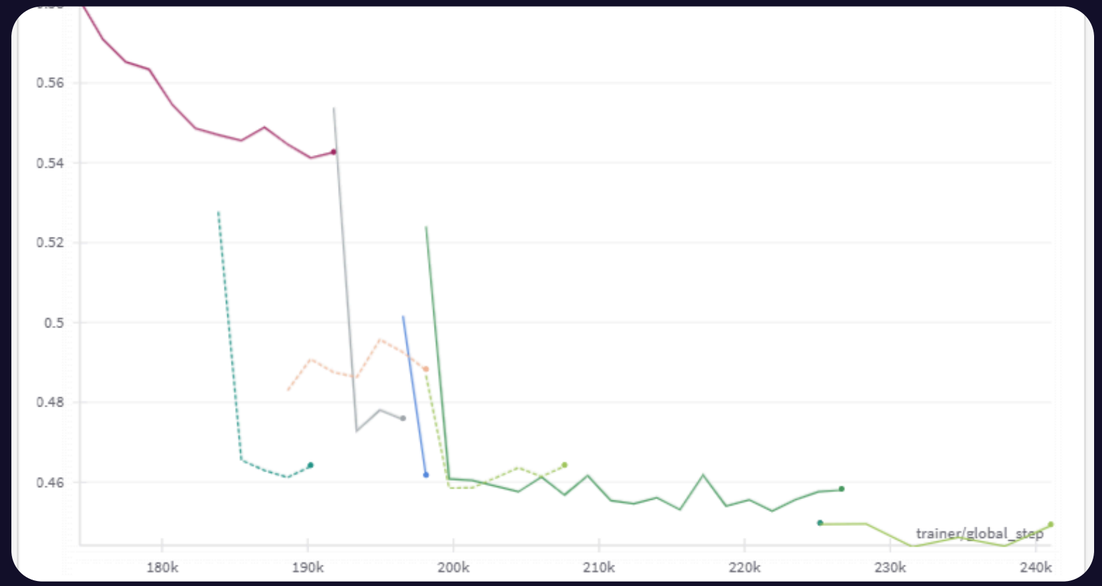



```
' ', '<unk>', '<fill>', '<overlap>', '<laugh>', '▁', ' ', 'ج', 'إ', 'ش', 'ق', 'ح', 'س', 'أ', 'ف', 'ك', 'د', 'ر', 'ع', 'ب', 'ت', 'و', 'ه', 'م', 'ن', 'ي', 'ل', 'ا',
چ', 'آ', 'ا', 'ٱ', 'ق', 'ٔ', 'أ', 'ؤ', 'ئ', 'ظ', 'ء', 'ث', 'ذ', 'ع', 'ة', 'ض', 'ز', 'ى', 'خ', 'ط', 'ص']
```

Dynamic Adjustment of Dropout Rates

We discovered that tweaking the dropout rates during training was a game-changer for us. By starting with high dropout rates, we forced the model to avoid relying too much on specific neurons, which helped it learn more robust features. After a few epochs, we gradually lowered the dropout rates, letting the model fine-tune and capture more complex patterns.

This strategy helped us achieve the lower MLD score of 8.9, a notable improvement over our previous results.



Offline Speaker Diarization

Offline diarization involves processing pre-recorded audio files to identify and segment speakers.

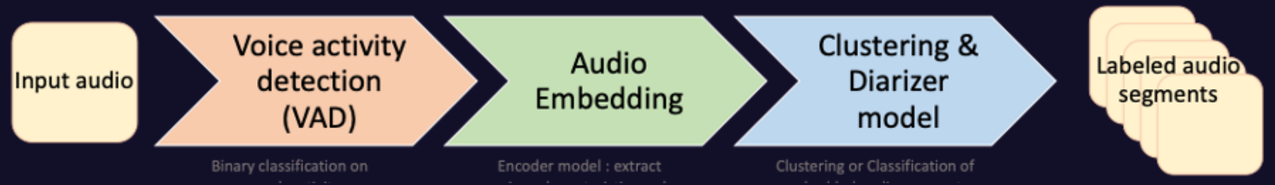
This method allows us to apply complex and accurate algorithms since we have access to the entire audio file at once.

It's particularly useful for post-event analysis, transcription, and detailed review where immediate results aren't necessary.

Offline Speaker Diarization

A typical speaker diarization pipeline consists of:

- **Voice Activity Detector (VAD):** Detects the presence of speech for an audio file generating segments for speech activity.
- **Segmentation:** Further breaks down the detected speech segments into smaller segments ensuring that each segment is small enough for accurate speaker embedding extraction.
- **Speaker Embedding Extractor:** Extracts speaker embedding vectors containing voice characteristics from raw audio signal.
- **Clustering Module:** Groups the extracted speaker embeddings into clusters, where each cluster represents a unique speaker.



Offline Speaker Diarization

We attempted to use the following pipelines:

1. Pyannote:

- VAD model: pynet
- Embedding model: wespeaker-voxceleb-resnet34-LM
- Agglomerative clustering: Hierarchical clustering method used to group objects based on their similarities.

2. NeMo:

- VAD model: MarbelNet
- Embedding model: Titanet-Large
- Clustering Model: NME-SC
- Neural Diarizar: MSDD

Offline Speaker Diarization

We decided to use nemo for the following reasons:

- Its embedding shows better EER results compared to wespeaker-voxceleb-resnet34-LM.
- Wespeaker-voxceleb-resnet293-LM achieved better results than Titanet-large, however this is due to it having more parameters which would increase inference time.
- Better clustering model (NME-SC)

Offline Speaker Diarization
NME-SC

- The new framework estimates the row-wise binarization threshold pp and the number of clusters kk using the NME value derived from the eigengap heuristic.

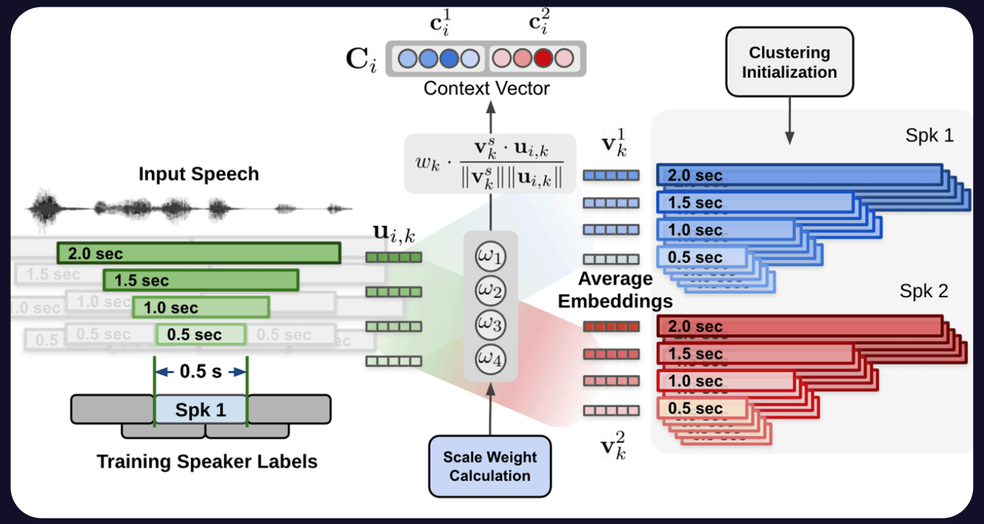
- The process involves creating an affinity matrix with raw cosine similarity values, binarizing it, symmetrizing it, computing the Laplacian, performing SVD, and calculating the eigengap vector.

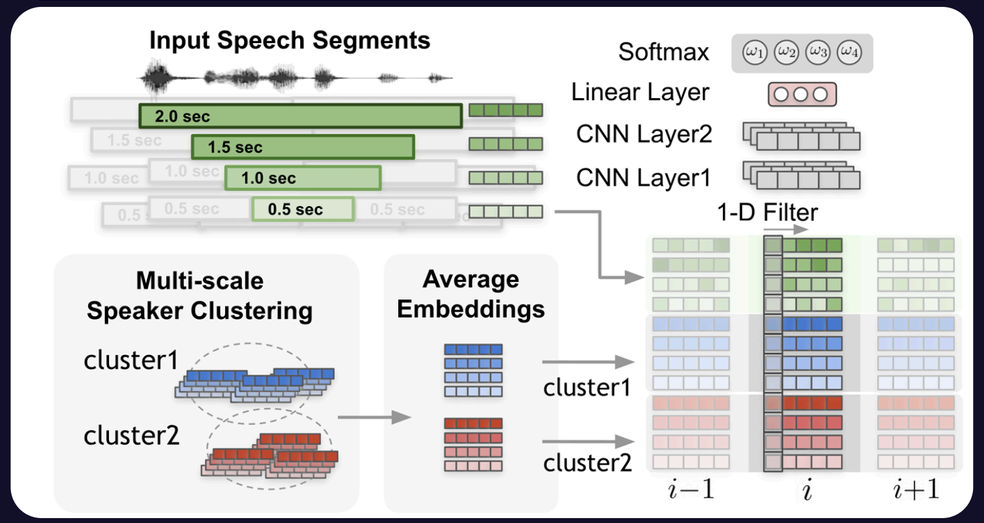
- The NME value **gp** is used to find the optimal **p** and **k** number of clusters, with the ratio **r(p) = r / gp** serving as a proxy for the diarization error rate (DER).

- **p** value should be minimized to get an accurate number of clusters, while the **gp** value should be maximized to get the higher purity of clusters. Thus, the ratio **r(p) = p / gp** is calculated to find the best **p** value by getting a size of the **p** value in proportion to **gp**.
-

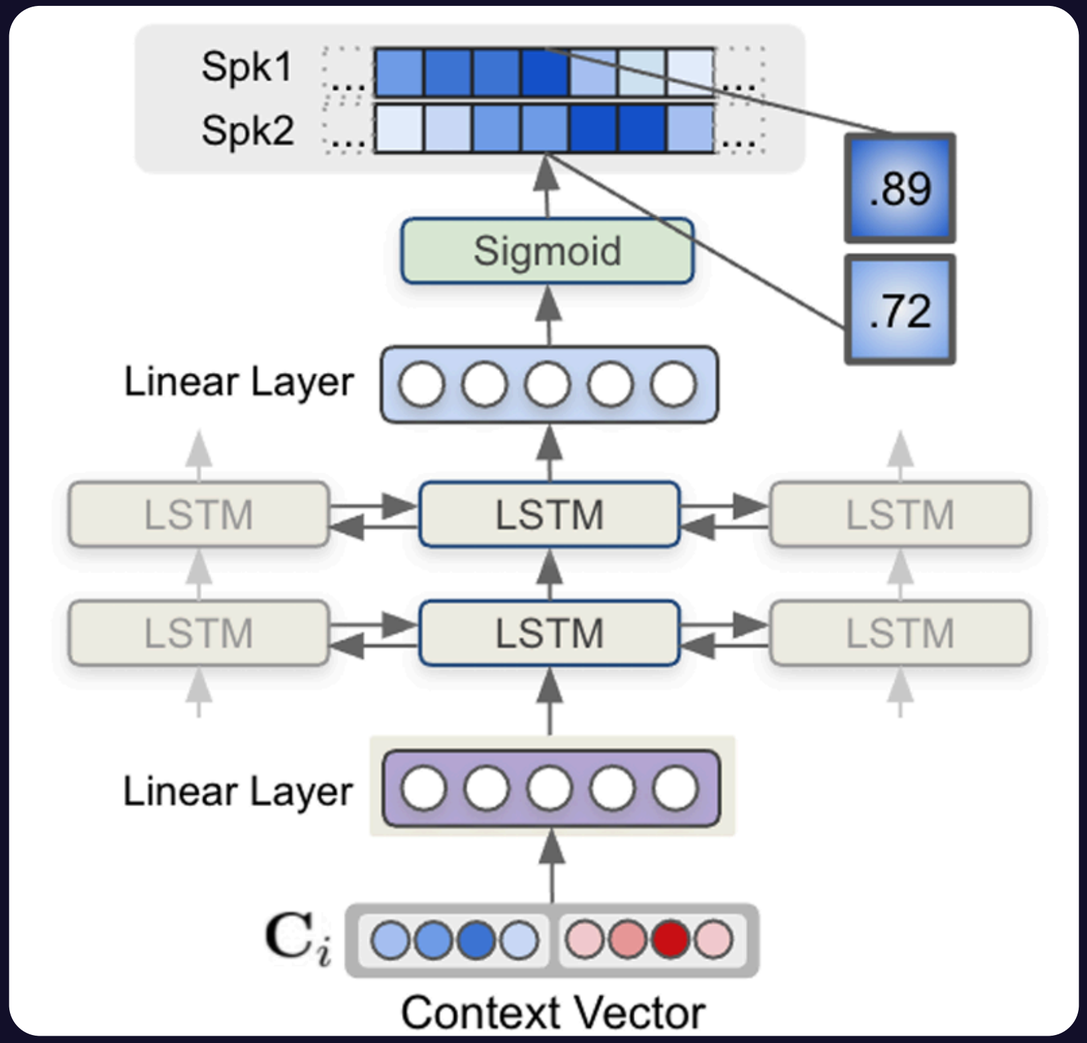
Offline Speaker Diarization
MSDD

- The MSDD approach offers overlap-aware diarization, flexibility in speaker numbers, and improved performance without extensive parameter tuning. It can handle a variable number of speakers without being constrained by a fixed number during training.
- The multi-scale approach is fulfilled by employing multi-scale segmentation and extracting speaker embeddings from each scale.
- When combining the features from each scale, the weight of each scale largely affects the speaker diarization performance.
- The final speaker labels are estimated using the context vector.





Offline Speaker Diarization
MSDD



Offline Diarization Challenges

- Sometimes the configuration requires to be set hard-coded as the configuration is not always set to be the same across the system, example **enhanced_count_thres** variable is always set to 80.

- Audio files that have more than 1 speaker the initialization can't cluster correctly and just outputs 1 cluster presenting a single speaker showing that having a low DER doesn't necessary mean that the model is performing well.

- Thus, problem is probably caused by using the dummy clusters, the default value is 3 dummy clusters and this results to have 4 clusters in total most of the time.

Dummy Clusters	Enhanced Threshhold	Mean Error of Number of Clusters	Diarization ER
0	40	9.25	0.4583
0	80	20.314	0.6226
1	40	8.655	0.4533
1	80	19.493	0.6156
2	40	8.084	0.4489
2	80	18.694	0.6085
3	40	1.605	0.3486
3	80	**1.535**	**0.3222**
4	40	7.044	0.4388
4	80	1.574	0.3260
5	40	1.660	0.3532
5	80	1.588	0.3275

Offline Diarization

Experiments

- Tuned the **rp** and **sigmoid** thresholds to arabic SADA dataset to optimizer the model performance on arabic speech.

- After testing values for **rp** from 0.03 to 0.5, we found **rp = 0.25** gave the best DER results.

- Tested the following range of values **0.5 <= sigmoid threshold <= 0.9**, but showed no significant improvement compared to default value of **0.75**.

- Fine-tuned the MSDD module but it showed no improvement. 1 epoch took 6 hours so we trained for 5 epochs only.

- Tuned the multiscale_weights which are the weights given to each scale on the custom SADA dataset, following values: **[1, 1, 0.4, 1, 1]** made a slight improvement.

Offline Diarization

Results

```
[NeMo I 2024-07-19 19:55:47 der:176] Cumulative Results for collar 0.25 sec and ignore_overlap True:
     FA: 0.0028  MISS 0.1820  ⟶                 Diarization ER: 0.3237 , Confusion ER:0.1390
[NeMo I 2024-07-19 19:55:47 speaker_utils:93] Number of files to diarize: 1500
[NeMo I 2024-07-19 19:55:56 der:176] Cumulative Results for collar 0.25 sec and ignore_overlap False:
     FA: 0.0027  MISS 0.1877  ⟶                 Diarization ER: 0.3278 , Confusion ER:0.1374
[NeMo I 2024-07-19 19:55:56 speaker_utils:93] Number of files to diarize: 1500
[NeMo I 2024-07-19 19:56:05 der:176] Cumulative Results for collar 0.0 sec and ignore_overlap False:
     FA: 0.0206  MISS 0.2040  ⟶                 Diarization ER: 0.3882 , Confusion ER:0.1636
```

```
[    '[00:00.00 - 00:27.88] speaker_0: شكلها يعني بتبقى مش واقعيه أو شكلها مش '
     'حقيقي ومش مش الواقع بتاع الحاله دي و ده في نفس الحاجه في كل الكونتن يعني'
     ' مثلا لو أنا بعمل حاجه عن الأمراض النفسيه فدلوقتي في في وعي بالأمراض'
     ' النفسيه دلوقتي والحاجات دي كلها ما ينفعش أعمل حاجه تشوه كل ده لثقافة'
     ' البلد عشان في الأول في الاخر برضو مسلسلات الأفلام دي بتثقف ف البلد دي'
     'بتبقى أكتر حاجه بنركز فيها طبعا',
     '[00:28.56 - 00:35.04] speaker_2: إيه أصعب حاجه يعني إيه اللي خليك تكون '
     'جري وتاخد الدور ده',
     '[00:35.96 - 00:39.56] speaker_1: والله مش جري خالص مخفتش',
     '[00:39.60 - 00:39.84] speaker_2: يعني',
     '[00:40.28 - 01:06.80] speaker_1: جدا جدا جدا ده دو صعب هو بس أنا اتحمست '
     ' أوي إن أنا يعني أعمل حاجه مختلفه يمكن خدت القرار حتى كنت بقول كده أنا خدت'
     ' القرار بسرعه أوي إن أساتذاره أوي اما عرض عليا الموضوع تحمست أوي فكره إن'
     ' هو دور جديد أوي معانا أنا ممكن أبحب نفسي فيك كومثل إم بعد ما خدت القرار'
     'كتشفت بقى اللي أنا عملته في نفس اللي هو لأ هحاجه']
```

Online Speaker Diarization: Diart

- Our first attempt was to use diart pipeline which uses pyanote models by default.
- It sends audio as 2 seconds chunks, transcribe and dairize it then send next chunk.
- Modify embedding model to use nemo Titanet-Large.
- We integrated our ASR model with it to output both transcriptions and diarization.
- It couldn't keep the profile of the speakers, and somethimes in long runs the pipeline starts to misclassify the speakers and keep predicting the same words
- The used mic to capture the audio was bad so this affected the results

يعني بتبقى
قي متا ها مش حقيقيه
وق بتع الحاله دي
وده في نفس الحاجات ف
يعني مثلا لو أنا بعمل حاجه
الأمراض النفسيه ندلوقتي
كده
إوعي بالأمر ا
دلوقتي وجوتحاجات كلها ميفعش أناا
حاجه مو
في البلد في الأول في ا
سكفي الأفرام
فشي طبعا حاجه
بناجي فيها طبعا
إيه أسعار حاجهيع
لي
كم إيه وتاخد

مش ت خالص
وصح
بس حمست أوي
يعني عمل حاجه
يفع
أ را
وز امكسد قرار ام
أوي ع
حمست أوي
دور جديد أوي
أنا أنا ممكن أحبه
أمثل بعد ماخ
كتش
لك
ح

example: audio file 12

Custom Pipeline

- VAD Model: Used pyannote VAD modle to detect conversation segments within the audio stream.
- Multi-Scale Segmentation: Generates sub-segments at different temporal resolutions.
- Efficient Storage: Segment intervals are stored for efficiency, and to minimize redundancy.
- Developed a custom clustering algorithm that uses NME-SC to estimate cluster numbers before employing k-means for precise cluster predictions.
- Implements a limit on the number of points per cluster to boost efficiency, merging points when exceeding thresholds based on proximity and similarity.

THANK YOU