# **Speaker Diarization**

Speaker diarization คือ กระบวนการประมวลผลข้อมูลเสียงที่มีผู้พูดหลายคน เพื่อแยกแยะว่าในข้อมูลเสียงนั้นมีผู้พูดกี่คน และแต่ละคนได้พูดในช่วงเวลาไหน เป็นระยะเวลาเท่าใด

ตัวอย่าง ผลลัพธ์ที่ได้จากกระบวนการ speaker diarization ในกรณีที่มีผู้พูดจำนวน 3 คน:  
  
 Speaker\_00 เริ่มพูด ณ เวลา 10.9s ใช้เวลาพูดรวม 5.2s

Speaker\_02 เริ่มพูด ณ เวลา 17.9s ใช้เวลาพูดรวม 1.4s

Speaker\_02 เริ่มพูด ณ เวลา 18.9s ใช้เวลาพูดรวม 2.4s

Speaker\_01 เริ่มพูด ณ เวลา 22.3s ใช้เวลาพูดรวม 10.7s

# **Methods**

## Experiment Setting

นำโมเดลจาก [Pyannote](https://github.com/pyannote/pyannote-audio) มาทำการ Fine-tuning หรือเทรนเพิ่มเติมด้วย Dataset ของข้อมูลการประชุมที่ได้เตรียมไว้เพื่อเพิ่มขอบเขตความรู้ของโมเดลให้รู้จักข้อมูลที่มีลักษณะเป็นการประชุมและข้อมูลภาษาไทย จากนั้นนำโมเดลที่ได้มาปรับปรุงค่า Hyperparameters เพื่อให้เหมาะสมกับลักษณะของอินพุตที่กำหนดไว้

# **Models**

main

* [pyannote](https://pyannote.github.io/) ([Demo link](https://github.com/pyannote/pyannote-audio/blob/develop/tutorials/intro.ipynb))
* [Paper](https://arxiv.org/pdf/1911.01255.pdf)

ค่า hparams ของ pretrained:

* [Segmentation](https://huggingface.co/pyannote/segmentation/blob/main/config.yaml)
* [Diarization](https://huggingface.co/pyannote/speaker-diarization/blob/main/config.yaml)

ปรับแก้ไข hparams, การเทรนโมเดล:

* [Pytorch Lightning](https://www.pytorchlightning.ai/)
* [Pyannote Github](https://github.com/pyannote/pyannote-audio) [ Model=LightningModule, Task=LightningDataModule]

# **Results**

rttm files format

SPEAKER {file\_name} 1 {start\_time} {duration} <NA> <NA> {speaker\_name} <NA> <NA>

example:

SPEAKER ES2011a 1 58.56 5.72 <NA> <NA> FEE041 <NA> <NA>

SPEAKER ES2011a 1 65.0 12.78 <NA> <NA> FEE044 <NA> <NA>

# 

# **Datasets**

## Resources

### AMI Corpus

Downloads: <https://groups.inf.ed.ac.uk/ami/AMICorpusMirror/amicorpus/>

Description: <https://groups.inf.ed.ac.uk/ami/corpus/>

Metadata: <http://shachi.org/resources/5073>

### Common Voice

Version: cv-corpus-9.0-2022-04-27

Downloads: <https://commonvoice.mozilla.org/th/datasets>

Description: <https://www.kaggle.com/datasets/mozillaorg/common-voice>

### Clips from Parliament

Id: Amarat, Chuan, Prayut, Sutin, Visanu

Resources:

[Live :The Politics special การอภิปรายทั่วไปเพื่อลงมติไม่ไว้วางใจรัฐมนตรีเป็นรายบุคคล วันที่ 27 ก.พ.](https://www.youtube.com/watch?v=xQKT66VMZeQ)

["ชวน"สอนมวย"ประธานวิป-ส.ว." ต้องรักษาสัญญาสุภาพบุรุษ : Matichon TV](https://www.youtube.com/watch?v=sdB7dj5zw2c)

[ลำดับเหตุการณ์ป่วน "ชวน" มีดทื่อ ยื้อไม่ไหวสภาล่ม สู่ขิต : Matichon TV](https://www.youtube.com/watch?v=RJ4OgEXRm7U)

[Live :The Politics special การอภิปรายทั่วไปเพื่อลงมติไม่ไว้วางใจรัฐมนตรีเป็นรายบุคคล วันที่ 25 ก.พ.](https://www.youtube.com/watch?v=yMvHWLLvfvc&t=12660s)

["อมรัตน์" อัดน่าเศร้า ชนชั้นนำคับแคบไม่เข้าใจประชาธิปไตย แนะรัฐบาลนิรโทษกรรมให้เยาวชน : Matichon TV](https://www.youtube.com/watch?v=oOq6XdroqwM)

[อมรัตน์ เสนอจัดงบกระบวนการยุติธรรมใหม่ แนะตัดงบศาลรัฐธรรมนูญ 100 % : Matichon TV](https://www.youtube.com/watch?v=v_ksTjT6jHA)

Tools: [ffmpeg](https://ffmpeg.org/), [Audacity](https://www.audacityteam.org/)

## Train Datasets

ami

Set up ตาม <https://github.com/pyannote/AMI-diarization-setup>

Length: 100 hr

amiF07

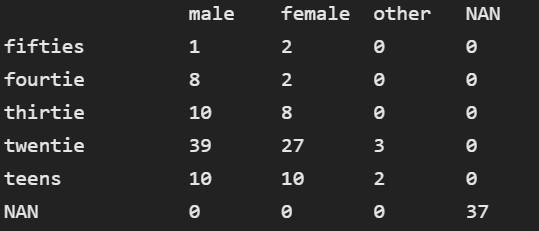
ami ที่ rttm ตัด row ที่ duration น้อยกว่า 0.7 sec ออก (30% ของ row ทั้งหมด)

cv00

ไฟล์เสียงต่อกันตรงๆ(cv\_concat) + ไฟล์เสียง overlap กันด้วย downmix(cv\_diarization)

length: concat 20 ไฟล์ ไฟล์ละ 20 min + overlap 20 ไฟล์ ไฟล์ละ 30 min รวม 16 hr

ปัญหา: Dataset ไม่ distributed age กับ gender + False alarm เยอะ(คาดว่าจากการไม่ตัด silence)



cv01

Run cv00 ใหม่แต่ลด silence threshold ลง

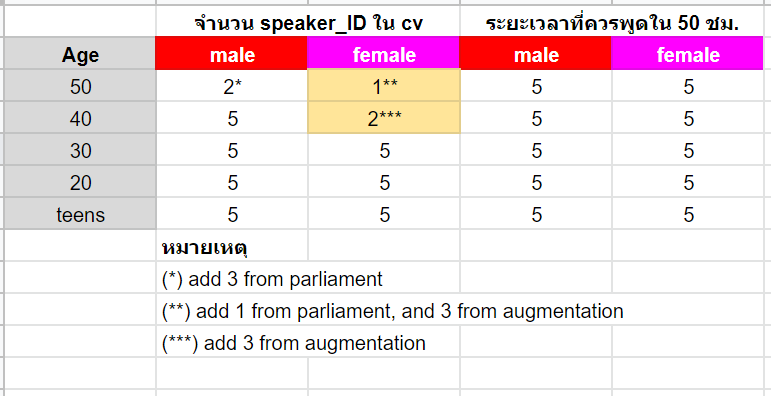
ผลลัพธ์: error น้อยลงเทียบกับ cv00 แล้วลด false alarm ได้มากกว่า

age00

Filter เฉพาะ id ที่อายุมากกว่า 40-50 เพิ่มคลิปเสียงที่ตัดมาจากประชุมสภา

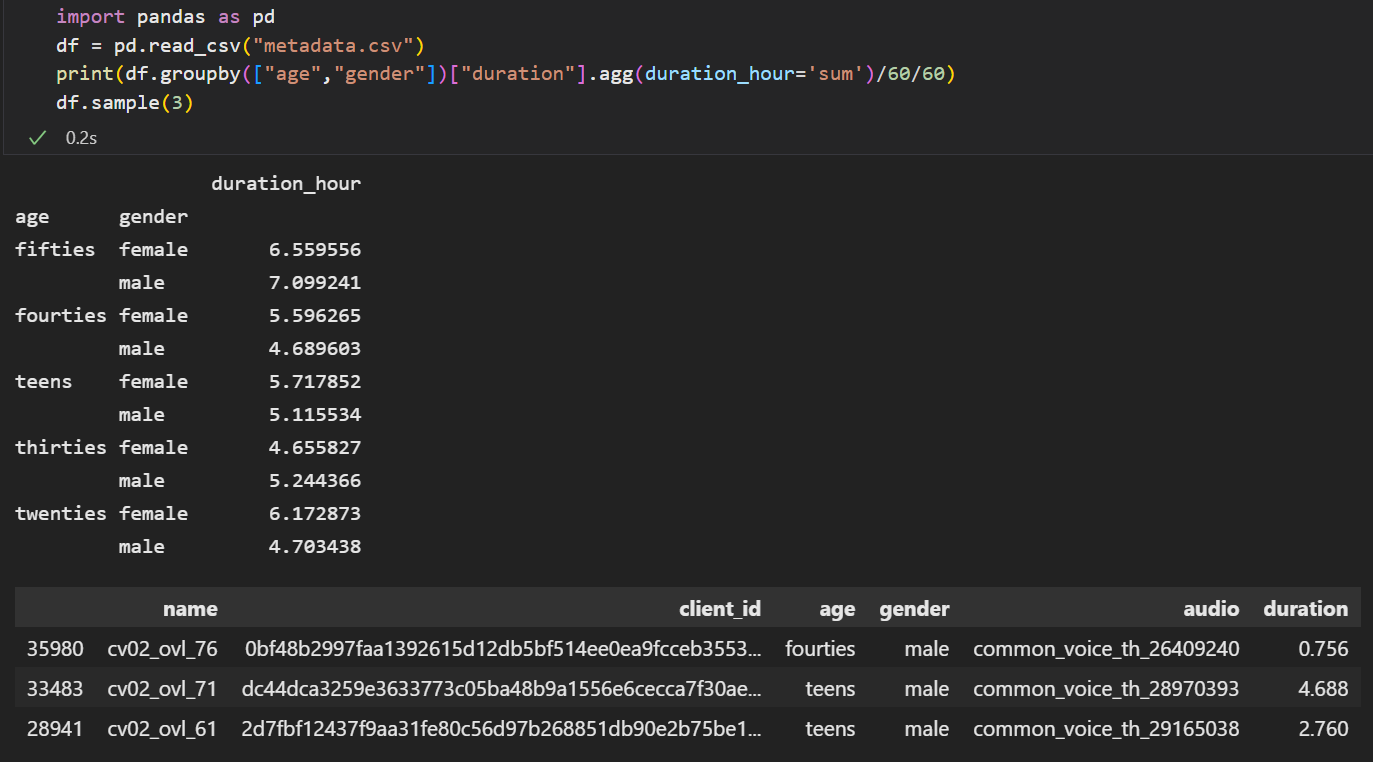
cv02 (\*\*latest)

Distributed age and gender as shown



Length: 50hr

มี metadata ให้ไป query ดู distribution และ เวลาจริง

Ex. 

# **Evaluation**

<https://pyannote.github.io/pyannote-metrics/reference.html>

# **Further Improvement**

1. [Suggestion from devs](https://github.com/pyannote/pyannote-audio#pretrained-pipelines-do-not-produce-good-results-on-my-data-what-can-i-do)
2. Label Parliament เพิ่ม
3. AMI Filter
4. เพิ่ม Dataset ที่ Overlap กันมากกว่า 2 คน
5. Tune Embedding blocks ให้เป็น 5 sec เหมือน Segmentation (Pretrained(ver.1) tune มาที่ 2 sec ทั้งคู่)
6. Train from scratch