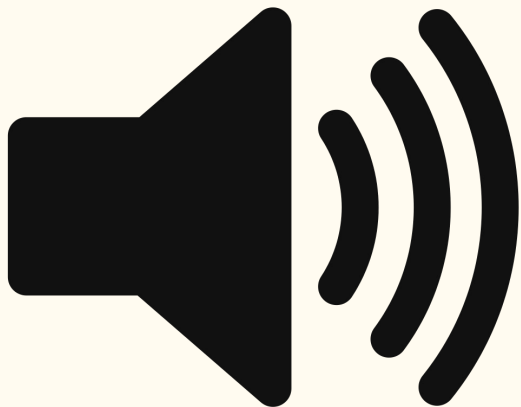


Persona - Speaker Identification System

Nicolas Stencel, Naga Sumanth, Andy Fausak, Daniel Mata

Overview

- Speaker Diarization - Segmenting audio into homogenous segments that are based on speaker identity.
- We realized that voice transcription compliments this idea perfectly and decided to pursue that as well.



[1] How are you guys doing?

[2] Can't complain

[3] Same. What about yourself?

[1] I'm fine thank you.

Overview

- We will be testing our system with a .wav file of a lecture that has already been transcribed. This transcription is our “true” value which we will be using to compare to the transcribed output that our system gives us.
- We are using 3 different APIs: Google, Amazon, and Azure

Google Speech to Text API

- ICSI Dataset
- Needed to enable Speaker Diarization
- Parameters we worked on:
 - Sample_rate_hertz
 - Enable_speaker_diarization
 - Diarization_speaker_count
- Default output from API

```
word: 'things', speaker_tag: 3
word: 'out', speaker_tag: 3
word: 'so', speaker_tag: 3
word: 'Morgan', speaker_tag: 3
word: 'wants', speaker_tag: 3
word: 'to', speaker_tag: 3
word: 'make', speaker_tag: 3
word: 'it', speaker_tag: 3
word: 'hard', speaker_tag: 3
word: 'it', speaker_tag: 3
word: 'doesn't', speaker_tag: 3
word: 'did', speaker_tag: 3
word: 'it', speaker_tag: 3
word: 'did', speaker_tag: 3
word: 'it', speaker_tag: 3
word: 'I', speaker_tag: 3
word: 'didn't', speaker_tag: 3
word: 'even', speaker_tag: 3
word: 'check', speaker_tag: 3
word: 'yesterday', speaker_tag: 1
word: 'either', speaker_tag: 1
word: 'when', speaker_tag: 1
word: 'I', speaker_tag: 1
word: 'started', speaker_tag: 1
word: 'it', speaker_tag: 1
word: 'I', speaker_tag: 1
```

Rand Score

- Normalized output will be used as argument for the `adjusted_rand_score` from `sklearn`
- Need to deal with poor transcribing of words (“ I’m ” v.s. “ I ”, “ ‘m ”)

```
>>> from sklearn.metrics.cluster import adjusted_rand_score
>>> adjusted_rand_score([0, 0, 1, 1], [0, 0, 1, 1])
1.0
>>> adjusted_rand_score([0, 0, 1, 1], [1, 1, 0, 0])
1.0
```

Labelings that assign all classes members to the same clusters are complete but not always pure, hence penalized:

```
>>> adjusted_rand_score([0, 0, 1, 2], [0, 0, 1, 1])
0.57...
```

ARI is symmetric, so labelings that have pure clusters with members coming from the same classes but unnecessary splits are penalized:

```
>>> adjusted_rand_score([0, 0, 1, 1], [0, 0, 1, 2])
0.57...
```

If classes members are completely split across different clusters, the assignment is totally incomplete, hence the ARI is very low:

```
>>> adjusted_rand_score([0, 0, 0, 0], [0, 1, 2, 3])
0.0
```

Google Speech to Text API, Normalization

- Needed to normalize our labels for Rand Score comparison
- Label based on order of initial speaker appearance

```
[3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,  
3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 1, 1, 1,  
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 3, 3, 3, 3, 3, 1,  
1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 3, 3, 3]
```

```
Normalized Labels: {3: 0, 1: 1}
```

```
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1,  
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,  
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0]
```

Outputs: True Label transcript, AWS Console

```
32 </Preamble>
33
34
35 <Transcript StartTime="0.0" EndTime="3021.968">
36
37   <Segment StartTime="0.000" EndTime="3.956" Participant="me018">
38     <NonVocalSound Description="mike noise"/>
39   </Segment>
40   <Segment StartTime="0.056" EndTime="1.861" Participant="me011">
41     Yeah, we had a long discussion about
42   </Segment>
43   <Segment StartTime="2.674" EndTime="6.315" Participant="me011">
44     how much w- how easy we want to make it for people to bleep things out. So -
45   </Segment>
46   <Segment StartTime="5.100" EndTime="7.881" Participant="me018">
47     <NonVocalSound Description="mike noise"/>
48   </Segment>
49   <Segment StartTime="6.260" EndTime="6.864" Participant="fe016" CloseMic="false">
50     Right.
51   </Segment>
52   <Segment StartTime="6.315" EndTime="7.061" Participant="me011">
53     <VocalSound Description="breath"/> <NonVocalSound Description="mike noise"/>
54   </Segment>
55   <Segment StartTime="7.516" EndTime="8.757" Participant="fe016" CloseMic="false">
56     O_K. So this is -
57   </Segment>
58   <Segment StartTime="8.169" EndTime="11.100" Participant="me011">
59     Morgan wants to make it hard. <VocalSound Description="laugh"/>
60   </Segment>
61   <Segment StartTime="8.320" EndTime="13.210" Participant="me018">
62     <NonVocalSound Description="mike noise"/>
63   </Segment>
64   <Segment StartTime="9.751" EndTime="13.549" Participant="fe016" CloseMic="false">
65     The, uh, counter is not <Pause/> moving again. It -
66 </Transcript>
```

Transcription preview

Select download to save a local copy of the transcription.

Text Audio Identification

Speaker 0: Yeah, we had a long discussion about

Speaker 1: how much how easy we wanna make it for people to bleep

Speaker 0: things out.

Speaker 0: Morgan wants to make it hard.

Speaker 0: There is.

Speaker 1: It doesn't

Speaker 1: I didn't I didn't even check. Yes, it didn't move yesterday either when I started it. So I don't know if it doesn't like both. Three. Channel three.

Speaker 1: Yeah. You know, I discovered something yesterday on these wireless ones. You can tell if it's picking up

Speaker 1: breath noise and stuff it

Todo

- Parse True labels information from ICSI dataset transcript XML file.
- We are able to get the Amazon Speaker diarization from AWS Console; we want to be able to get similar output by calling Amazon API.
- We also want to get speaker diarization output from Azure API.
- Compute Rand Score calculation between:
 1. True Label, Google API
 2. True Label, Amazon API
 3. True Label, Azure API
- Pick best 2 out of 3 cloud APIs and use it in the website.
- Website would accept *.wav* file from user, perform speaker diarization - display converted text and display Rand Score.