Constraint Based Speaker Diarization Module

Abstract

The framework proposed will implement speaker diarization module for a massive heterogeneous television news corpus using LIUM and pre-processing steps which will allow a user to use it's own segmentation information or can use auto-segmentation option. Another contribution will be to train a speaker identification module within a network using technique called “Constrained Global Clustering” across video files. This framework will be then compiled with open-source toolKit called “voice-id” to use Multi-Modal approach which also incorporates information from visual features.

1. Data

Red Hen Labs has approximately 300,000 hours is very diverse collection and each network might have to be addressed differently to maximize accuracy (which will be assumed at the beginning)

<table>
<thead>
<tr>
<th>Data Type</th>
<th># Files</th>
<th>Features</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. “.tpt files” (mainly for CNN data)</td>
<td>30,000 files (Approximately 10 hrs for evaluation, the ones with least number of speakers)</td>
<td>*Gold Segmentation *Gold Speaker Identity Information</td>
<td>*Evaluation *Training</td>
</tr>
<tr>
<td>2. Forced Aligned Data</td>
<td></td>
<td>Gold Segmentation</td>
<td>Training *semi-supervised</td>
</tr>
<tr>
<td>3. Raw Broadcast News</td>
<td></td>
<td>Auto Segmentation</td>
<td>Training *unsupervised</td>
</tr>
</tbody>
</table>

Also data will be pre-filtered according to number of speakers (say. 20 speakers) to get the training data for a particular network for both Data_Type 2 and 3.

2. Main Tool Kits Used

- LIUM for speaker diarization [1]
- FFMPEG (for extracting “wav” audio from Mp4 file)
- SOX for audio format conversion
- BeamformIt v2 toolkit for single channeling
- Voice ID for using features from Visuals
3. Pre-Processing Module

All audio files are first treated with Wiener filter noise reduction. Then, for the multiple distant microphone (MDM) condition only, a single virtual channel for each show is created using the BeamformIt v2 toolkit [3, 4] with a 500ms analysis window and a 250ms frame rate. This latter stage is not necessary for the single distant microphone condition (SDM) condition. Most the data available from different networks is Multi Channel for Red Hen Labs. This is the only difference between the diarization systems to be made for the MDM and SDM experiments which will be considered for now.

Data Filter: Depending on data statistics, data will be divided into its availability from various Networks. This will be done to check if we have enough data for each network to have a different network to do cross-clustering to identify speakers from the same network.

NOTE: Proposed framework is very generic but filters are applied to maximize performance for each network separately.

Evaluation Data (data with “.tpt” files): only CNN data has already marked “speaker identification” labels. So evaluation will be done for CNN data only.

4. Diarization Module

4.1 Why LIUM:

- As Lium can be used to accommodate text features by providing Text files where each line corresponds to a frame (Using files having correct alignment) - That information can be simply given in terms of word vectors obtained using a model created by Word2Vec. Even if alignments are partially wrong, then also using vector space models, right semantic meaning can be composed as we can handle some noise by composing vectors from the context too. This will mainly be done for Cross-Clustering within a network.

- There are numerous works done for using Multi-Modal approach for Diarization using LIUM specially Voice_ID [5] while will be combined towards the end of the project, Bost et al [6] Each step is independent of next-step so it gives leverage to try different algorithms for segmentation, clustering.
4.2 Diarization Steps

Features Used: MFCC Features, Text Features (from Word2Vec trained on transcriptions)

4.2.1 Level 1: Single Show Diarization
This will mainly be done for data having no segmentation information. This will also allow a user to remove non-speech events which are there in the initial segmentation founded/provided (depending upon use-case). Textual Features will be used for this step after an initial segmentation is done. For Textual features we will use a context of 5 seconds before and after and composition of word-vectors will be simple aggregation to begin with. It is expected that there will some delay in the captions (1-10 seconds), we can begin with the padding (like finding median for the entire dataset). Level 1, will be improved after whole pipeline development, when using “.tpt” files, so check errors occurred due to bad segmentation. Various methods will be tried mainly:
- combination of Generalized Likelihood Ratio (GLR) and the Bayesian Information Criterion (BIC) for segmentation from IRIT ToolKit [7]
- using LIA_SpkSeg [8]
4.2.2 Level 2
This is done by LIUM to cluster speakers which actually refer to same speaker in different segments, where LIUM combines the output of two different clustering algorithms namely CLR Clustering and i-vector based clustering. 
	- ".tpt" files be again used after Level 2
	 1. Missed speech (MissSp)
	 2. false alarm speech (FA)
	 3. speaker error (SpkErr)
	 4. overall diarization error rate (DER)

4.2.3 Level 3 : Constraint Based Global Clustering
Now the clusters diarization obtained will be further clustered to obtain clusterization over data for each Network (initially for CNN data). This will be done following the work by Bost et al [6] Evaluation will be done based on parameters described in Level 2 using ".tpt" data from CNN Network. The results on this Level depends upon results from previous Level. Also, upper limit will be found using ".tpt" files as we have Gold annotated diarization cluster information in that dataset
In the input distance matrix 'D', cannot-link constraints between two instances are inserted by setting their distance to +Infinity and the linear algebra library used will be Armadillo.

4.2.4 Combining Voice ID for visual features
  
Pro's : Produces Better Results
Cons : Slow, as the implementation is in Python and LIUM’s implementation is in JAVA

5. Evaluation

Evaluation data will be kept the same for all models i.e a part of data having .tpt files

There will be four major parameters for Evaluation :
  1. Missed speech (MissSp)
  2. false alarm speech (FA)
  3. speaker error (SpkErr)
  4. overall diarization error rate (DER)

Spreadsheet for Evaluation Results (will be populated as we go ahead)

Link : https://docs.google.com/spreadsheets/d/1yoMn2xiLJf6s2M37L5Gf2TAvGHhKjgsgrFVi82RWBI/edit#gid=0
6. Timeline

6.1 Milestones

- **Mid-Term Milestone**: Audio Based Speaker diariaztion Module (allows both auto-segmentation and also allows model to use already provided segmentation file along with Mp4 file from Red Hen Labs). The deliverable will have following major features:
  - Audio Based Speaker Diarization Tool
  - Different Features based on input methods (Textual Features or not)
  - Allows change in pre-processing parameters
  - Allows Different Segmentation Techniques
  - Diarization Results over the CNN data for both models (auto-seg and gold-seg) based on parameters described in Section 5

- **End-Term Milestone**: Generic speaker diarization package for heterogeneous TV news data, which will have primarily following features
  - Different Input Methods (one can provide a segmentation or not)
  - Different Features based on input methods (Textual Features or not)
  - Recognizes speakers which are present in the same network but in different news sessions
  - Combined Multi-Modal Techniques for Diarization (Visual Features take into account)
  - Documentation/ Evaluation results obtained over news corpus from Red Hen Labs.

<table>
<thead>
<tr>
<th>Weeks</th>
<th>Short-Term AIM</th>
<th>Progress (✓/✗)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Share and Divide the data according to different Experiments (with/without segmentation, clean/unclean segmentation removal)</td>
<td></td>
</tr>
<tr>
<td>2,3</td>
<td>Setup Pre-Processing module (exploration about parameters may be needed) and testing it with LIUM ⇒ Share segmented data and segmentation evaluation results</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Working System till the work described in Section 2.2 ⇒</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Generating Textual Features (Word2Vec Model on Transcriptions or generic news corpora) for the Training data which has alignment information and combining it with earlier framework. Compare results with modal without the textual information trained on same data in earlier week</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Evaluate the overall diarization results for CNN network before cross-clustering for each network. Also cleaning model pipeline to make it a deliverable</td>
<td></td>
</tr>
</tbody>
</table>

**Mid-Term Evaluation : Milestone Checkpoint Evaluation**

| 8 | Implementing Level 3 (Section 2.3) part of the framework - Share codes of this module and trained model for different Network. Share results of Cross-Clustering within a network for CNN data using “.tpt” files. Also provide trained models for other networks. |
| 9 | Combining Multi-Modal Technique used in Voice ID or work done by Bost et al [6] with the existing framework and deciding which one works better for our data problem. |
| 10 | Setup Multi-Modal Framework Pipeline and Share speaker diarization results for using Multi-Modal Framework for CNN data |
| 11 | Test the code for bugs and try various use cases suggested by mentor. Or some other directions discussed during the project. |
| 12 | Cleaning up the codes, and packaging it to deliver it as tool. Also documenting the work done till now (may be as a paper), share evaluation results for entire project |

**Final Evaluation : End-Term Milestone Evaluation**

**References**


