International Institute of Information Technology, Bangalore

Reading elective : Final report

(under the guidance by : Prof. Dinesh Babu)

Classification of Native and Non-Native English Speakers

Arunaav
IMT2016108

Sudharsanaraj R
IMT2016130

Introduction :

- After an extensive research on speech related features for speaker classification, we believe that prosodic features play a major role in capturing the rhythmic and intonational properties in the speech. Prosody is linked to linguistic units such as syllables, and it is manifested in terms of changes in measurable parameters such as fundamental frequency (F_0), duration and energy. In this project, a syllable-like unit is chosen as the basic unit for representing the prosodic characteristics. Approximate segmentation of continuous speech into syllable-like units is obtained by locating the vowel onset points (VOP) automatically. The knowledge of the VOPs serve as reference for extracting prosodic features from the speech signal.The process of extracting features is explained in detail in the later part of the paper.
- We intend to create a deep learning system that takes spoken sentences sliced at a phoneme-level as input, and classifies each phoneme into a binary-class output, where each class is an indicator native and non-native english speakers. We have recordings of non-native and native english speakers, with labeling at phoneme-level to train and test our model.

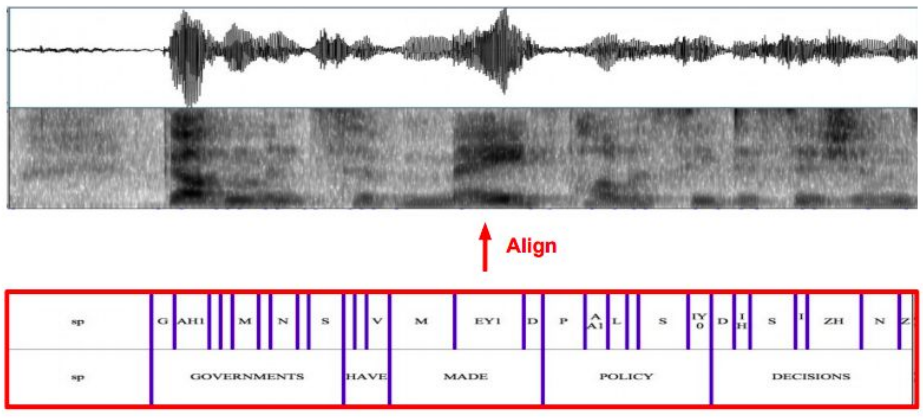
Data :

- We used LibriSpeech and L2-ARCTIC datasets for training and testing the model.
- The LibriSpeech corpus is derived from audiobooks that are part of the LibriVox project, and contains 1000 hours of speech sampled at 16 kHz and the corpus is freely available for download.All the speakers in this dataset are native English speakers.

- The L2-ARCTIC corpus includes recordings from twenty-four (24) non-native speakers of English whose first languages (L1s) are Hindi, Korean, Mandarin, Spanish, Arabic and Vietnamese, each L1 containing recordings from two male and two female speakers.
- Both the data sets are huge in size (LibriSpeech is of 20GB size and L2-ARCTIC is 7.5GB),due to computational constraints we took ~3000 samples of wav files from both the datasets.

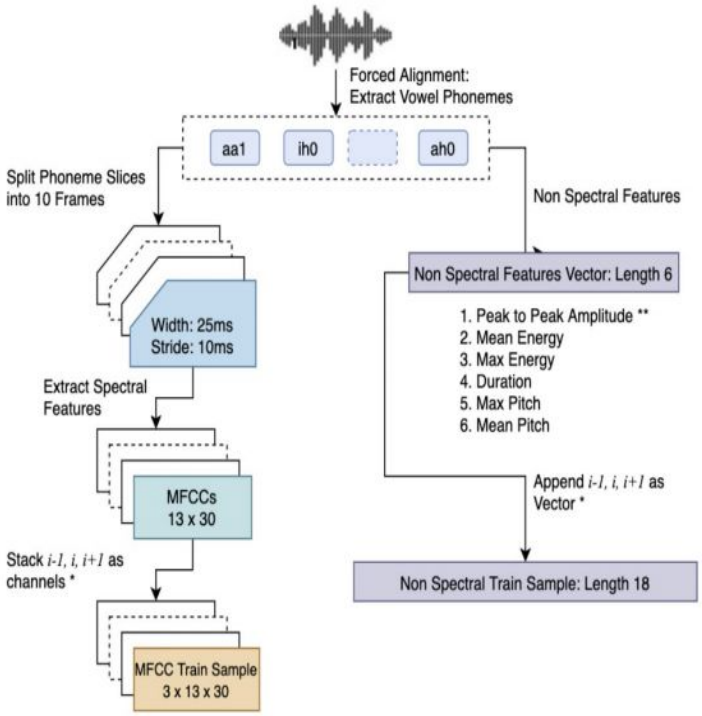
Preprocessing :

- The first step is to determine the time boundaries of each pronounced phoneme in order to extract acoustic features from each syllable.
- Performing forced phoneme alignment using gentle (built on kaldi). In forced alignment, the acoustic model is given an exact transcription of what is being spoken in the speech data. The system then aligns the transcribed data with the speech data, identifying which time segments in the speech data correspond to particular phonemes in the transcription data.



- After the alignment is done,we need to extract the syllable nucleus.
- In phonetics and phonology, the nucleus (sometimes called peak) is the central part of the syllable, most commonly a vowel. In addition to a nucleus, a syllable may begin with an onset and end with a coda, but in most languages the only part of a syllable that is mandatory is the nucleus.
- For native and non-native classification,syllable duration plays a major role but extraction of exact syllable duration is a complicated task .So,instead we tried to extract the syllable nucleus duration.
- Using aligned data,for each word we can extract vowel phones (consider vowel phonemes as syllable nucleus)
- Split the wav file and create new wav files using start and end time of vowel phoneme.

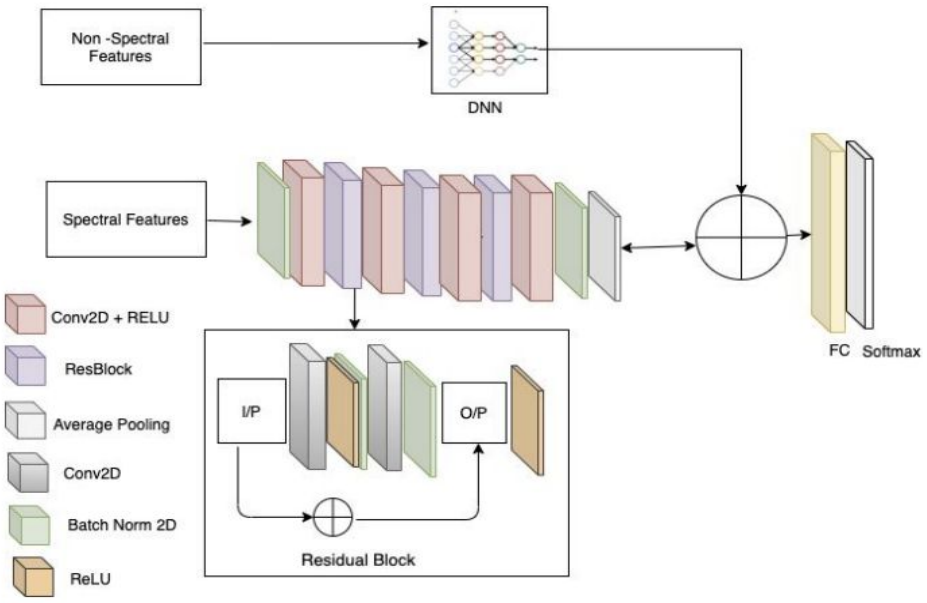
Feature extraction:



- We sliced out the phoneme of interest (vowel phoneme) as well as the preceding and succeeding phoneme.
- For each phoneme, extracted out six temporal features (peak to peak amplitude, mean energy, max energy, duration, max pitch, mean pitch) over syllable nucleus.
    - **peak to peak amplitude :** difference between the max and min peak values of the syllable nucleus.
    - **mean energy :** the mean value of energy calculated from each frame.
    - **max energy :** the maximum energy value of a set of frames.
    - **duration :** Duration features are obtained based on the word and phone alignments of human transcriptions.
    - **max pitch :** we can find pitch value in a frame using zero crossing method and take the max value out of all the frames.
    - **mean pitch :** average the pitch values of input frames.

- As the speech signal is distributed over multiple frequencies, used features on the Mel Scale. Each phoneme is split into n non-overlapping frames of 10ms and the first 27 Mel Frequency Cepstral Coefficients (MFCC) are extracted.
- To adjust for variable length of phoneme the maximum number of frames is set to a constant N. If $n > N$, the middle N frames are used and if $n < N$, zero padding is used.
- The MFCCs are arranged as a three dimensional matrix of shape N x 27 x 3 where the three channels are the features from the three phonemes. Also, the temporal features which are arranged sequentially into a vector of length $6 \times 3 = 18$.

Model Training :



- The architecture consists of two different neural networks whose outputs are combined before passing onto the softmax layer.
- The first part is a CNN which takes the MFCCs as input and outputs a vector.
- The second part is a deep neural network which takes in the vector of non-spectral acoustic features and outputs a vector.
- The two vectors are concatenated and passed onto a fully connected (FC) layer followed by a softmax layer. Two different networks are required because non-spectral features cannot be treated similarly.

Model experiments :

- Initially we tried extracting phoneme sequences for audio files,we trained a GRU model over Timit dataset with MFCC and log energy values as input features to predict phoneme sequences.Our plan was, to extract phoneme sequences and align it to original audio, so that we can calculate syllable based features.

- But we got 78% accuracy and extracting syllable features using this model will give us terrible end results ,so we switched to kaldi pre-trained forced aligner.
- Kaldi is an open source toolkit made for dealing with speech data.It is written mainly in C/C++, but the toolkit is wrapped with Bash and Python scripts.We used Gentle module, which is built over kaldi to get alignment files.

Results :

```
Test set: Average loss: 0.2411, Accuracy: 11737/15370, (76.3624)%

After epoch: 17, train_loss: 0.2061352279701081 68, test loss is: 0.22113061705078449,
train_accuracy: 79.76631419057078, test_accuracy: 76.36239427456083
```

- We got 79.76 percent training accuracy and 76.3 percent test accuracy .

```
[0.9997513890266418, 0.0002486590819898993],
[0.9391326308250427, 0.06086738407611847],
[4.598945356482087e-12, 1.0],
[0.9999722242355347, 2.7724508981918916e-05]]
```

```
In [32]:
t=0
n=0
s=len(temp)
for i in temp:
    t+=i[0]
    n+=i[1]
```

```
In [29]:
output=[t/s , n/s]
```

```
In [33]:
print('The original label of wav file is:{} and the predicted output is [{},{}]'.format(wave_file_label,output[0
```

```
The original label of wav file is:0 and the predicted output is [0.6203913199837674,0.3796086919389276]
```

- In the above picture we tried classifying a speaker ,we took a wav file of native speaker at random and the result of probability prediction of the speaker being native is 0.62 .
- Similarly we took another wav file from L2-ARCTIC (non-native speech corpus) ,the results are given in the picture below.

```
[2.6420357457368482e-08, 1.0],
[2.5465018548692653e-12, 1.0],
[0.9999586343765259, 4.137381256441586e-05],
[8.230202297454525e-08, 0.9999998807907104],
[2.81497705145739e-07, 0.9999997615814209],
[0.9999873638153076, 1.2679217434197199e-05],
[2.0302454162290928e-11, 1.0],
```

```
In [28]:
t=0
n=0
s=len(temp)
for i in temp:
    t+=i[0]
    n+=i[1]
```

```
In [29]:
output=[t/s , n/s]
```

```
In [30]:
print('The original label of wav file is:{} and the predicted output is [{},{}]'.format( wave_file_label,output[
```

```
The original label of wav file is:1 and the predicted output is [0.3182120560104184,0.6817879402937157]
```

References :

- Huang, Z., Chen, L., & Harper, M.P. (2006). An Open Source Prosodic Feature Extraction Tool. *LREC*.
- Mary, Leena & Yegnanarayana, B.. (2008). Extraction and representation of prosodic features for language and speaker recognition. Speech Communication. 50. 782-796. 10.1016/j.specom.2008.04.010.
- Li, Kun & Mao, Shaoguang & Li, Xu & Wu, Zhiyong & Meng, Helen. (2017). Automatic lexical stress and pitch accent detection for L2 English speech using multi-distribution deep neural networks. Speech Communication. 96. 10.1016/j.specom.2017.11.003.
- Shahin, Mostafa & Epps, Julien & Ahmed, Beena. (2016). Automatic Classification of Lexical Stress in English and Arabic Languages Using Deep Learning. 10.21437/Interspeech.2016-644.
- https://github.com/lowerquality/gentle
- https://github.com/kaldi-asr/kaldi/tree/7ffc9ddeb3c8436e16aece88364462c89672a183
- http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/
- https://towardsdatascience.com/how-to-start-with-kaldi-and-speech-recognition-a9b7670ffff6