**Project Title: Speech Based Summarization and Emotion Analysis**

**Members:**

Pradeep Thalasta 4579117326 thalasta@usc.edu

Paridhi Verma 2139220711 vermap@usc.edu

Raksha Nagesh Bysani 3449867668 bysani@usc.edu

Suraj Hebbar Shankar 4376950843 surajhes@usc.edu

**1. Introduction**

Public speaking skills help you communicate important messages inside and outside of the organization. Public speaking is so important that it could be the deciding factor in many things such as your career development, your business growth and even in the relationships you have with your friends and family. We plan to build a tool based on analysis of the recorded speech that can help an individual improve by classifying the feedback of his speech. This is done by highlighting the key points mentioned in the speech and analyzing the over-all emotion. This problem is interesting because it helps the person to improvise the content and the emotion associated with it.

Existing approaches to solve this problem are done in two phases. First by analyzing the sentiment of the speech. Second by summarizing the speech data. These two phases are performed for a corpus based on English Language. In our approach, we try to combine both the phases by analyzing the emotion of the speech as well as summarize the content by using machine learning algorithms, followed by emotion analysis and summarization for German Corpus.

**2. Method**

**Materials:**

We used publicly available emoDB which is a Berlin database for emotional speeches. A database of 700 emotional utterances spoken by actors was recorded as part of a DFG funded research project SE462/3-1 in 1997-1999. Every utterance is named according to the same scheme:

* Positions 1-2: number of speaker
* Positions 3-5: code for text sample
* Position 6: emotion (letter stands for German emotion word)
* Position 7: if there are more than two versions these are numbered a, b, c ....

Example: 03a01Fa.wav is the audio file from Speaker 03 speaking text a01 with the emotion "Freude" (Happiness).

The database has samples for the following emotions: Anger (W), Boredom (L), Disgust (E), Anxiety/Fear (A), Happiness (F), Sadness (T) and Neutral version(N).

**Procedure:**

1. Corpus Building:

Each audio sample was tokenized sentence-wise by taking the audio file and the time intervals spoken for each sentence using aeneas which aligns the spoken data with its respective sentence from the transcript file. The resulting time frames were then used to build the model for classification.

Tools: aeneas.

2. Extracting Model Parameters:

Each audio file was processed along with their respective transcript to align the sentences in audio file with the sentences in the transcripts using aeneas which outputs a time frame for the sentence. This output was then used to create a series of sentence fragments from the audio file using pydub.

Fig 1: Average Loudness and Pitch(F0) plot for different emotions.

Fig 2: Average Energy Entropy values for different emotions.

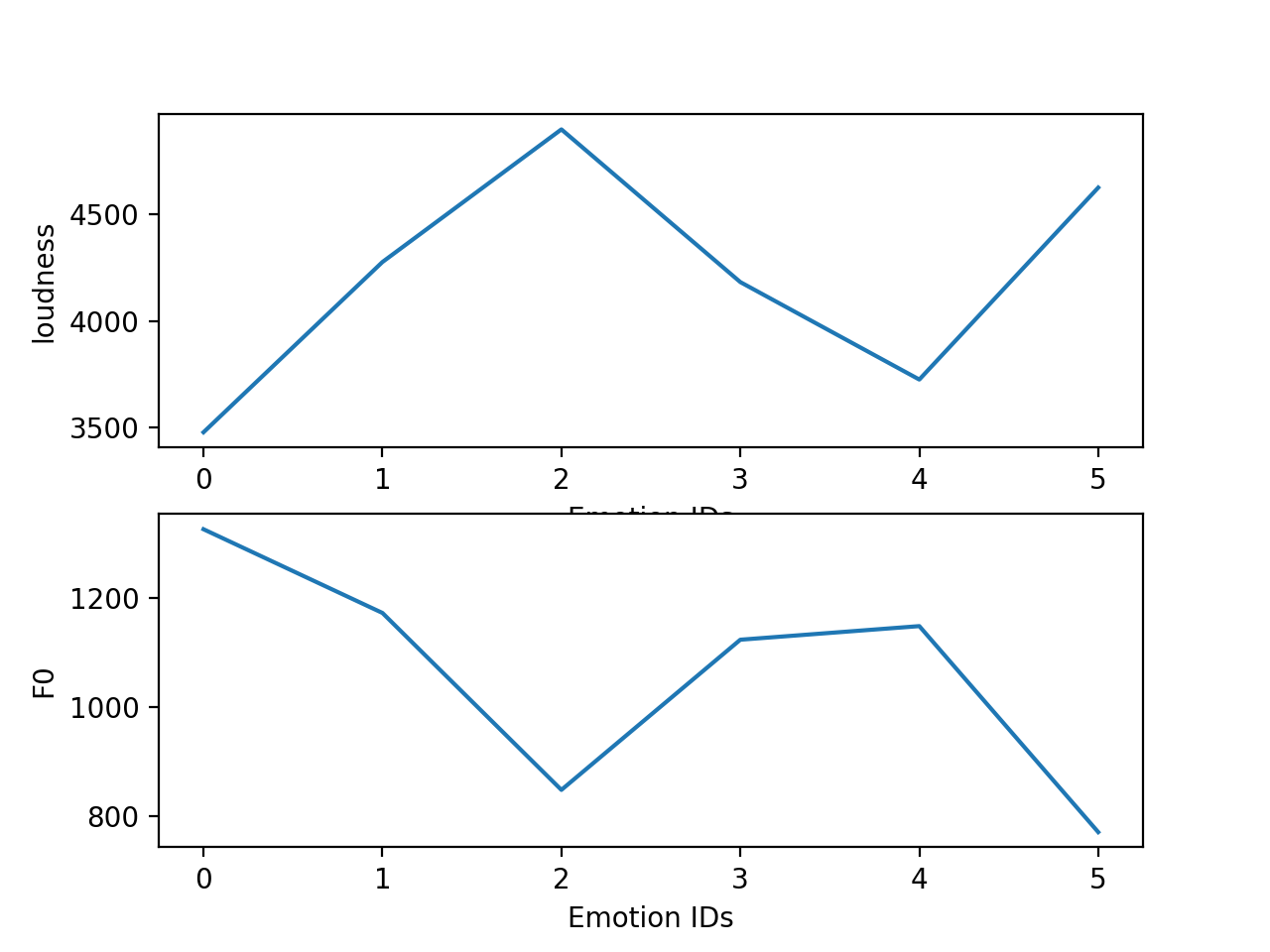
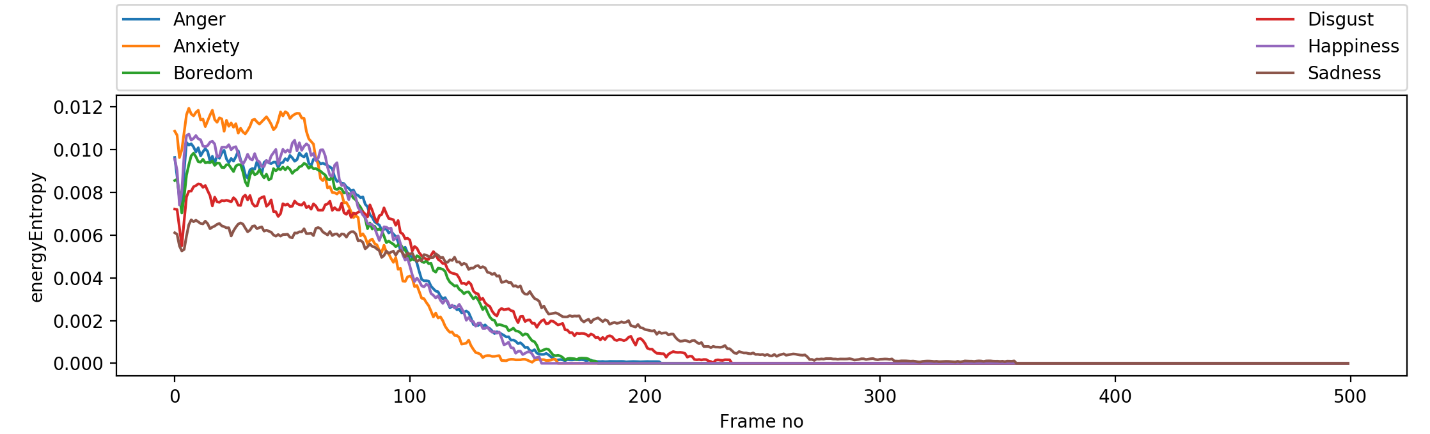
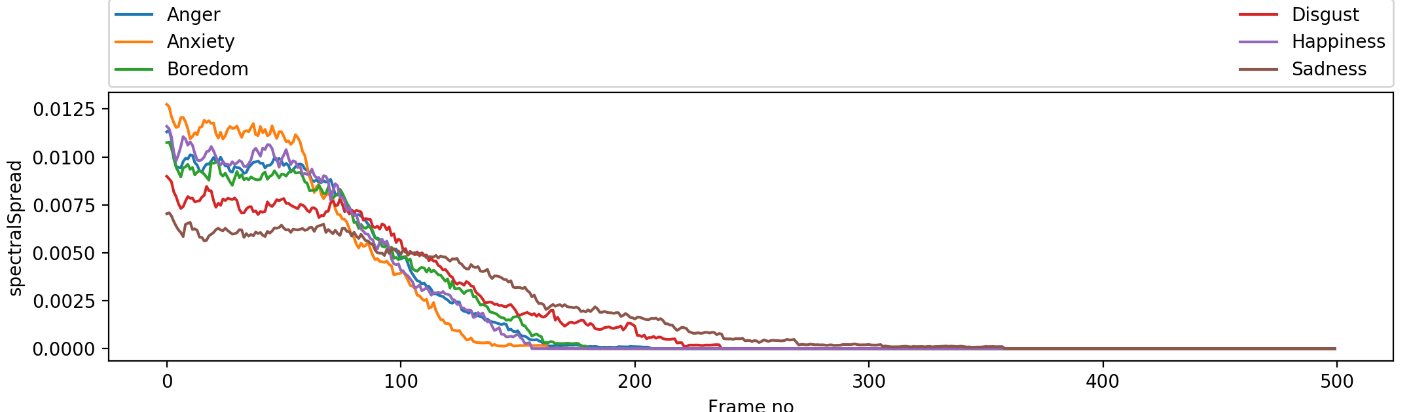
The resulting fragments were used as an input to py AudioAnalysis to extract different features. The result of the feature extraction was

Fig 3: Average Spectral Spread Analysis for different emotions.

saved to a pickle file for each emotion as a pandas DataFrame object.

Tools: pyAudioAnalysis, Pandas, pydub.

3. Emotion Analysis:

We predicted the emotion behind a test speech. The following procedure was followed for this:

* ***Naive Bayes****:*

*Training –*

We used Gaussian Naïve Baye’s to train the model. This is because the features that were extracted were continuous real-valued Gaussian distributions. The model was trained for each of the following features – F0, spectral centroid, MFCC, energy levels, chroma, spectral flux , spectral spread, spectral entropy, ZCR, loudness, energy entropy, chroma deviation and spectral roll-off.

*Classification –*

The emotion for the test data by predicting it individually for each feature. The emotion class with the highest frequency amongst all the features was assigned to the sample data.

Tools:

Pandas, SKLEARN

4. Speech Summarisation:

The objective of this step was to provide an extractive summary of the speech. In this step, we used Text Rank algorithm on the transcript of the speech. Each sentence in the transcript was first split into sentences to construct a bag of word model representing the term frequency using sklearn. The bag of word model was then normalized using TfidfTransformer. The normalization process will re-weight each word based upon its tf-idf, which will diminish the effect of words common to each sentence.

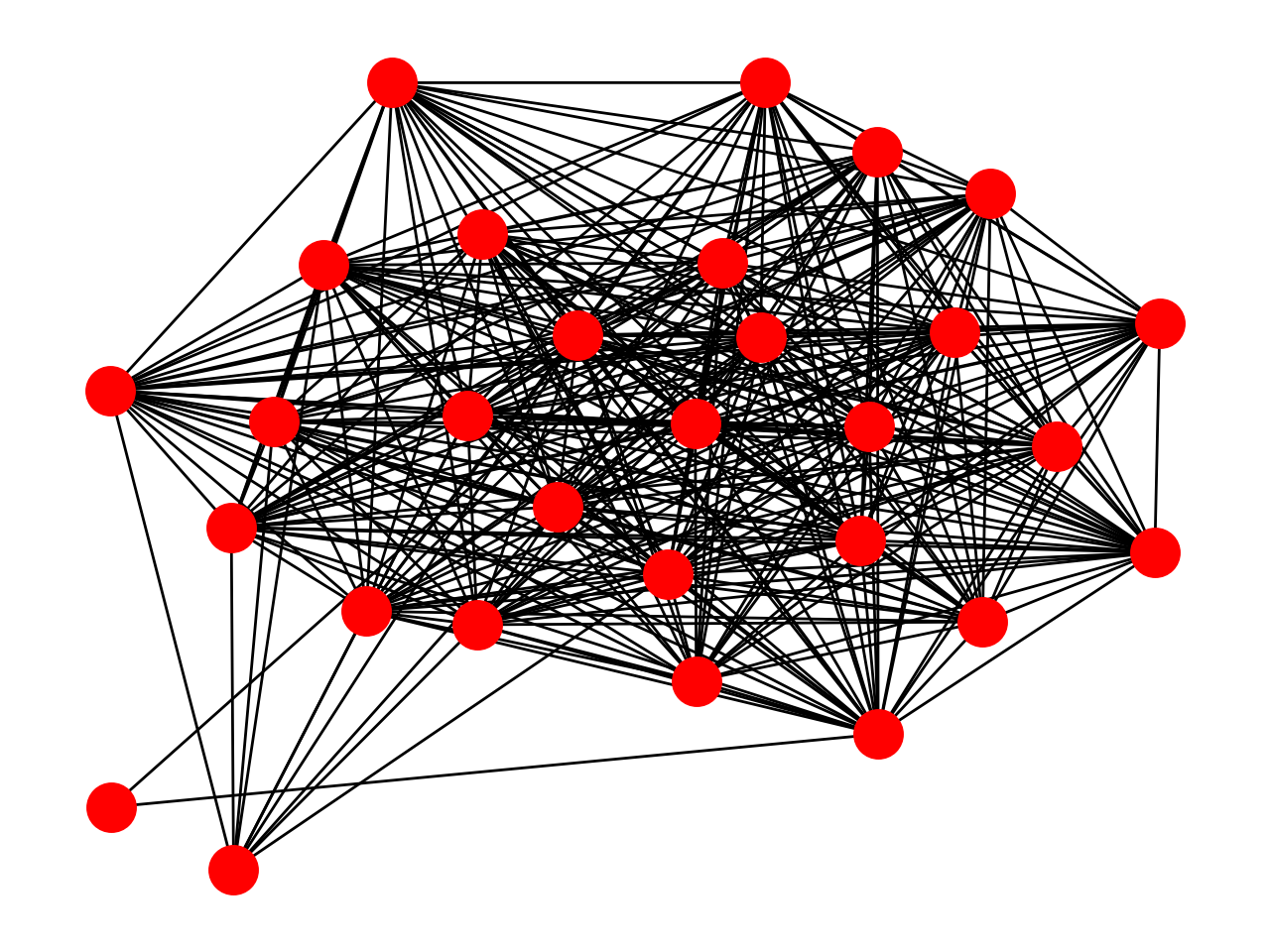
The TextRank algorithm was implemented by considering each sentence as a document by generating the similarity graph between the sentences in the transcript file and was allotted a similarity score based on PageRank algorithm from the [NetworkX](http://networkx.lanl.gov/) graph library. After this step, we had the representative sentences for each sample speech along with their similarity score ranked in descending order.

Fig 1. Representation of sentences as nodes to analyse the similarity between sentences for summarisation using TextRank algorithm.

Tools/Packages: sklearn, [NetworkX](http://networkx.lanl.gov/)

5. Extracting Test Data:

Crawler:

Developed a crawler in Java for fetching the data from YouTube to generate test data for German audio files along with its transcript. One of the Ted Talk video’s url was given as the seed url for the crawler and crawler was made to parse the data from that page, get the related links from that page and create a queue. After processing a page, the crawler would then fetch a set of URLs from the queue (depending upon the number of threads active) and perform the similar operation to get new related links by traversing and fetching data in breadth first manner. The crawler was made to run at a depth of 16, and a csv file was created for maintaining the useful links for the data.

Once the crawler operation is finished, the csv file is given as an input to a python script. The python script is implemented using youtube-dl package to get the subtitles from the YouTube videos and create the relevant transcripts from the video and generate the audio for the corresponding links.

**Evaluation:**

1. Corpus:

The test data was collected by running a crawler, which would collect different audio files along with their transcripts. The corpus was collected from Ted talks in German language, Hitler’s speech during World War II, sentences spoken by a native German Speaker having various emotions. The test data corpus was manually annotated with its emotion and its summary for every sample audio file.

2. Emotion detection:  
The sentences in the given test audio file were aligned with the sentences in its transcript to obtain a fragment of the sound for feature extraction and emotion analysis. Each sentence was then classified as one of the trained emotion based on different features extracted.

\*\*Mention accuracy\*\*

3. Speech Summarisation:

The summary generated by our algorithm was then compared with other summarization tools available online using document similarity approach. Apart from this, the generated summary was manually evaluated to check for its validity and correctness.

\*\*Mention avg similarity for n documents\*\*

**3. Results**

The given audio files were tokenized based on sentences and were analyzed for different emotions in the audio sample. There houses a scope for further optimize the accuracy based on algorithms like Neural Networks, and HMMs. Obtaining the audio files with minimal background noise plays an important role in classifying the test samples with higher accuracy. Summaries were also generated for the given transcript to give a gist of the contents in the speech.

**4. Discussion**  
 In this project, we have presented an overview of how we can classify the emotion behind the speech and summarized the speech using summarization techniques. It can be inferred from our results that our speech classifier is efficient and performs well when we have sufficient features which helps in identifying the emotion behind it.

The main features of the project include Naïve Bayes method for classifying the speech to an emotion.

For summarization of the speech, we use the transcript of the speech to generate its summary. We measure the accuracy of the summarized data using the cosine similarity between the summarized data and the summary obtained by tools available online.

Backing good accuracy of our system, we can use our method at practical levels to identify the confidence of a speaker and the emotion behind his speech. Also, it can be extended further to improve the speech of the user by providing feedbacks and finding the quality of the speech.

\*\*\*Mention the precision here \*\*\*\*\*

[WRITE ABOUT THE PRECISSION RATE OF TOKENIsING AUDIO TO FETCH THE FEATURE VALUES FOR SAMPLE DATA]

**Future research**:

Although our model is an initial step, it has vast potential which can be explored further. One such method is to use Neural Network models such as Tensor Flow to construct the classifier with reinforced learning.

Also, it can act as a training kit for students to improve their public speaking abilities helping them professionally and personally. It can be extended to use HMM to perform speech synthesis, which be capable of generating high quality speech as well as low quality one to use as a reference for public speech training.

**5. References**

1. [European Language Resources Association (ELRA) 2016] Mathieu Chollet, Torsten Wortwein, Louis-Philippe Morency, Stefan Scherer, “**A Multimodal Corpus for the Assessment of Public Speaking Ability and Anxiety**”.
2. [IEEE Transactions on Speech and Audio Processing ( Volume: 12, Issue: 4, July 2004 )] S. Furui, T. Kikuchi, Y. Shinnaka. “**Speech-to-text and speech-to-speech summarization of spontaneous speech**”.
3. Felix Burkhardt, Astrid Paeschke, Miriam Rolfes, Walter Sendlmeier und Benjamin Weiss.  
   “**A Database of German Emotional Speech**” [dataset].

**6. Division of Labor between teammates**

**Emotion Analysis:** Raksha, Pradeep

**Speech Summarisation:** Pradeep

**Corpus Building:** All

**Evaluation**: Suraj, Raksha

**Test Data:** Paridhi, Suraj

**Research Papers**: All

**7. Word Count for the document:** 1596 words