Sentiment Analysis on User-generated Video, Audio and Text

Dr. Ashwini Rao,
Assistant Professor, Information Technology Department,
Mukesh Patel School of Technology Management and Engineering, NMIMS, Mumbai
Ashwini.Rao@nmims.edu

Akriti Ahuja
MPSTME, NMIMS, Mumbai, India
Akritiahuja25@gmail.com

Shyam Kansara
MPSTME, NMIMS, Mumbai, India
shyam.kansara97@gmail.com

Vrunda Patel
MPSTME, NMIMS, Mumbai, India
vrupatel97@gmail.com

2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) | 978-1-7281-8529-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICCCIS51004.2021.9397147

***Abstract*—Social Media prevails in today's world for voicing an opinion. That makes it crucial for businesses, artists, content creators, and pretty much anyone else on the internet to analyse what people are saying about them and what they offer. It provides vital information on what they can change and improve upon. Hence, Sentiment Analysis is going to be a significant aspect to help organizations and individuals grow. Contrary to previous work, a broader viewpoint is taken into consideration and any kind of discrepancies is eliminated since the sentiment classified by any one modality is confirmed by the analysis of the other two modalities as well. In all the previous work done on multimodal sentiment analysis, the weightage given to all the modules was the same for fusion but to get perfect results, a trial-and-error logging method is used with many different weightages for fusion. This helped us in knowing which module is best suited for sentimental analysis and how each of them can alter the results obtained. With this type of the development in the sentiment analysis domain, a system which can extract the information in terms of the emotions of the people regarding a specific product from any of the three mediums (text, audio or video) can lead to the better technological aspects in the market. The objective was to build a system which can identify the sentiment categorized into six types: anger, joy, disgust, sadness, fear, and surprise of a video when the data is fed into it. The system developed depicts how much of each of these sentiments are present in a particular input.**

Keywords— Sentiment Analysis, Multimodal Sentiment Analysis, Multimodal Fusion, Human Computer Interaction

I. Introduction

Sentiments, in general, refer to the opinions, emotions, and attitudes. Sentiment analysis or opinion mining is one of many areas of computational studies that deal with the sentiments or opinions related to the subject with opinion-oriented natural language processing. The earliest known sentiment analysis methods date back to the 1950s. It was primarily used on written paper documents.

As of now, it is more profoundly used to mine subjective information from a particular content, including texts, blogs, tweets, social media, reviews, news articles, and comments. Various techniques are involved in doing such analysis like NLP, machine learning methods and statistics. Before, organizations used the information gained or mined to identify new opportunities and target their audience better. A major part of the information era is based on the opinions of the people and organizations conduct survey, opinion polls to know their customers and their target audience better as their competitive strategy to stay in the market and boost their sales.

Multimodal Sentiment Analysis is an emerging field that analyses of data based on text, audio, as well as video. The classification used previously was positive, negative, and neutral only. A proposal is made to use six emotions namely anger, joy, disgust, sadness, fear, and surprise. The advantage this has over the previous approach is that it becomes easy to derive the exact emotion of the author which, for example, enables businesses to take accurate and informed decisions.

In our research work, the text is derived from the video and the sentences are taken from different time frames in the video to get overall sentiment from texts. Tweets analysis was mainly done through R and taking text from a video was not considered. These types of analysis could change what the person wanted to convey because words can have many contextual meanings which could only be acknowledged when taking expressions as well as voice into consideration. For the Audio Analysis purpose, a decision tree classification is done. It builds tree based on regression model and further on, breakdown dataset into smaller subset for classification. Video is first extracted into different image frames which further process into csv will file depicting 708 facial landmarks and facial features resulting in a particular emotion.

Multimodal Sentimental Analysis has four complex modules namely Data Collection, Training, Testing (which form the backend) and Front-end. Data Collection and Front-end can work independently whereas Training and Testing can only be done after data collection has been completed. Data collection consists of Data Pre-processing and Data Processing. Data Pre-processing is the cleaning and division of collected and created datasets into training and testing datasets. This splitting of datasets for training and testing will be in a 70:30 ratio, respectively. Data Processing, audio and text is derived from video the user inputs. Normalization entails error handling in conversion and generalization of database. Training and Testing can be carried out

ISBN: 978-1-7281-8529-3/21/$31.00 ©2021 IEEE

simultaneously, both entails running of algorithms respectively on text, audio, and video for analysis.

We demonstrate the effectiveness of this approach with the training and testing of the publicly available The Ryerson Audi-Visual Database of Emotional Speech and Song (RAVDESS).

II. Literature Survey

Contrary to previous works in multimodal sentiment analysis which focus on holistic information in speech segments such as bag of words representations and average facial expression intensity, this paper gave an idea about the development of a novel deep architecture for multimodal sentiment analysis that performs modality fusion at the word level [1]. Gated Multimodal Embedding LSTM (Long Short-Term Memory) with Temporal Attention (GME-LSTM (A)) model alleviates the difficulties of fusion when there are noisy modalities. LSTM with Temporal Attention performs word level fusion at a finer fusion resolution between input modalities. The main idea is to capture the structure of speech. Sentiment can be expressed by the spoken words, the emotional tone of the delivery and the accompanying facial expressions. As a result, it is helpful to combine visual, language, and acoustic modalities for sentiment prediction. P2FA is a software that is being used computes an alignment between a speech audio file and a verbatim text transcript. Fusing audio, visual and textual clues for sentiment analysis from multimodal content networks, radial basis function networks, and kernel learning [2]. An AU consists of three basic parts: AU number, FACS name, and muscular basis. (Because the six emotions were not enough to relate to all types of emotion, a 7th emotion was introduced for that purpose: contempt). The Active Appearance Model and Optical Flow-based techniques are common approaches that use FACS to understand expressed facial expressions. Exploiting AUs as features like k-nearest neighbors, Bayesian networks, hidden Markov models (HMM), and artificial neural networks (ANN) has helped many researchers to infer emotions from facial expressions. For acoustic features: Gaussian Mixture Model (GMM) and Mel frequency cepstral coefficients (MFCC). EmoSenticNet, an extension of Sentic Net containing about 13,741 common-sense knowledge concepts, including those concepts that exist in the WNA list, along with their affective labels in the set anger, joy, disgust, sadness, surprise, fear. In order to build a suitable knowledge base for emotive reasoning, the so-called blending technique has been applied to Concept Net and EmoSenticNet. Blending is a technique that performs inference over multiple sources of data simultaneously, taking advantage of the overlap between them. [11] propose the Gated Multimodal Embedding LSTM with Temporal Attention (GME-LSTM(A)) model that is composed of 2 modules. The technique was able to better model the multimodal structure of speech through time and perform better sentiment comprehension. The effectiveness of the approach was demonstrated on the publicly available Multimodal Corpus of Sentiment Intensity and Subjectivity Analysis (CMU-MOSI). [12] pose the problem of multimodal sentiment analysis as modeling intra-modality and inter-modality dynamics. The novel Tensor Fusion Network proposed was capable of learning both dynamics end-to-end. They also demonstrated that the three Modality Embedding Subnetworks (language, visual and acoustic) outperformed unimodal state-of-the-art unimodal sentiment analysis approaches.

III. Data sets

The dataset for textual sentiment analysis is a collection of 8,600 tweet which have been manually labelled as one of the six emotions: anger, disgust, fear, joy, sadness and surprise. These tweets contain a lot of special characters, punctuation, and upper-case letters. To normalize the dataset, for better accuracies when using them for sentiment analysis, the data was cleaned and the components that do not contribute to deriving a sentiment was removed.

The audio dataset used in the research work is the RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song) audio dataset. It contains 7356 files (total size: 24.8 GB). The database contains 24 professional actors (12 female, 12 male), vocalizing two lexically matched statements in a neutral North American accent. Speech includes calm, happy, sad, angry, fearful, surprise, and disgust expressions.

Filename Identifiers:

- Modality (01 = full-AV, 02 = video-only, 03 = audio-only).
- Vocal channel (01 = speech, 02 = song).
- Emotion (01 = neutral, 02 = calm, 03 = happy, 04 = sad, 05 = angry, 06 = fearful, 07 = disgust, 08 = surprised).
- Emotional intensity (01 = normal, 02 = strong). NOTE: There is no strong intensity for the 'neutral' emotion.
- Statement (01 = "Kids are talking by the door", 02 = "Dogs are sitting by the door").
- Repetition (01 = 1st repetition, 02 = 2nd repetition).
- Actor (01 to 24. Odd numbered actors are male, even numbered actors are female).

The dataset used for the video sentiment analysis is the Extended Cohn-Kanade Dataset (CK+). The CK+ distribution dataset includes 593 sequences from 123 subjects. The image sequence varies in duration (i.e., 10 to 60 frames) and incorporate the onset (which is also the neutral frame) to peak formation of the facial expressions.

IV. Working

This section will deal with the proposed approaches and the working. A comparison line with respect to algorithms was drawn on the basis of accuracy. It has been done through thorough research and implementation of different of algorithms. The final tools and technologies to be used for the implementation are:

- For Text Analysis: Python, Support Vector Machine (SVM) Algorithm
- For Audio Analysis: Python, Decision Tree Algorithm
- For Video Analysis: OpenCV, Python, Deep Learning Algorithm

The task workflow is as shown in Figure 1. First, the path of the video will be entered on the system into the program so it can fetch the video to start the processing. The video will then be split into smaller clips of 15 to 20 seconds each. The reason behind this is that the voice to text API used, Houndify, cannot handle voice clips with a length that is more than this. It was found out that most of the algorithms work best within this range. The command here is the only place where the user can change the video file path to change the input video to predict its sentiment. The function is then executed using subprocess.

The number of clips created is purely based on the length of the entire video. The longer the video, the more will be the number of clips. Knowing the number of clips created is extremely necessary since it will be required to know the number of times a loop is to be run for executing functions later on. Now, extraction of the audio from each small 20-second video clip is done and it is stored as a .wav file. File handling is a huge part of this research work since there are a huge number of files to be manipulated. Here, a list of names of the video clips and audio clips is created based upon the number of clips generated form this particular video.

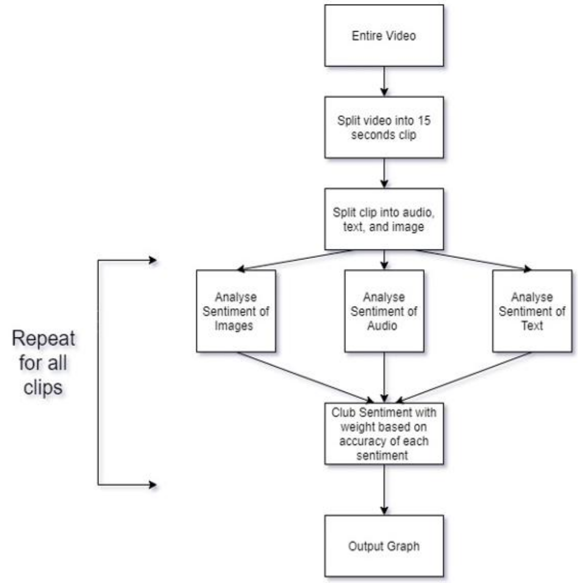

Entire Video
Split video into 15 seconds clip
Split clip into audio, text, and image
Repeat for all clips
Analyse Sentiment of Images
Analyse Sentiment of Audio
Analyse Sentiment of Text
Club Sentiment with weight based on accuracy of each sentiment
Output Graph

Fig. 1. Task Workflow

Next, each file is opened using file handling so the text can be appended into it. Then, the audio file is fetched. Houndify is an API for speech recognition used in this research work. It is not an open-source API but allows a couple of free “minutes” of text extraction. Plans must be purchased in case more minutes are required. So, the Houndify Client ID and Client Key are supposed to be entered over here once the user has registered with Houndify. Lastly, the text is extracted using the API and directly appended into the text file and the text file is closed. This loop repeats for all the audio clips.

The clips that are then further split into image frames (with rate of 1 frame per second), audio clips, and text files. For the images, each small clip has its own folder which contains the images taken at each second. The clip is opened using OpenCV and the frame rate is extracted. Next, for each frame number, the frame is read and stored in the folder with the file name specified, which is derived by using the list of image frame folder names. The image is then written using cv2 and the capture is released. The voice clips are numbered and stored as .wav files. The text is extracted and stored into .txt files.

The entire Textual part of the research work comes under the function Textual Sentiment Analysis. First, we read in the text file that was saved earlier. Next, we read in the entire textual dataset. The names of the sentiments, or the categories, are put into a list for easy access later on. Then, using train_test_split from the sklearn library, the textual dataset is split into training and testing datasets with a train: test ratio of 70:30. This is a standard splitting ratio followed in machine learning.

The initial dataset has 8600 tweets which are classified into the 6 emotions. These tweets contain a lot of junk such as stop words (is, I, the, am, etc.), punctuation and special characters which do not contribute at all to the sentiment of a text. Hence, these should be removed in order to obtain a clean dataset. This gives a higher accuracy when using it. Therefore, this function removes all those punctuation marks and special characters. It also splits conjunctions (for example, what’s is changed to what is). Next, to remove the stop words, a variable is assigned which is a list of all the stop words.

Now, an SVC (Support Vector Machine) pipeline is created wherein the stop words are put through the TFIDF Vectorizer (a function which converts the words into vectors in numeric form) and then through the Linear SVC. The custom text, which has the current text is cleaned using he clean text function defined above. The training dataset and the categories are then fit into the SVC pipeline. The sentiment is then predicted. The output of that is a numeric value. Hence, the prediction is the appended into the respective list.

The audio analysis part is split into two phases. Training with the Audio dataset and analyzing the Sentiment. Now, the entire audio dataset must be read and loaded into the system for training the machine. First, a random audio file from the dataset is loaded to fetch its sampling rate and other data. A path to all the audio files of the dataset is then set. A timer clock is started to be able to measure how long it takes to load the dataset. Then, using the dataset, all MFCCs (Mel-Frequency Cepstral Coefficients) are fetched which forms the actual final dataset for the audio analysis. The Mel-Frequency Cepstral Coefficients of a signal are a small set of features (usually about 10-20) which concisely describe the overall shape of a spectral envelope. This allows us to build a numeric dataset for analysis. Finally, the time taken for loading the entire dataset is recorded and displayed. The dataset is split into testing and training with a train: test split of 70:30 and the training dataset is fit into the Decision Tree Classifier to train the machine.

Now, training part of audio contains, the short audio clip file which is stored in the program as an Audio Segment. Then, the amplitude of the audio is decreased by 25 decibels to bring it to the level of audio in the dataset to increase accuracy. Next, the sample rate is deduced using librosa and the MFCCs are calculated and a list is created of the MFCCs for further analysis. The same process is applied to derive the same final set of values as the dataset. The joblib module is used here. Once that is obtained, the custom prediction variable sores the current prediction which is done using the Decision Tree Classifier which was trained earlier. The output for the classification has different numbers, so the output is then appended into the appropriate list which is declared in the main section of the program.

For video analysis, a list of the emotions is created for the order of categories. A list of all folders with the respective participant numbers (people who helped make the videos for the dataset) is stored. Then a loop is run over all the participants. First, the current participant number is stored. Then, a list of sessions for the current participant is stored using two for loops. Within these, the current session is derived from the name of the file. The file is opened, and emotions are encoded as float and converted into integers. The image path and the path for the neutral image is extracted from the name. A neutral image is used here to find the deviation of the current facial features from the participant's neutral expression to find the value of each facial feature in the current expression. Next, four face detection classifiers are loaded from the OpenCV library.

Next, to be able to store the facial characteristics, it is necessary to first detect the face in the image since in videos, people tend to move quite a lot. So, the data would be much cleaner if only the face of the person was picked up. So first, the image is opened using OpenCV. It is converted into a grayscale image to remove any kind of color complexities which may affect the values. Now, four different classifiers are run to detect the images. The classifiers defined before and are called here. Then, once the faces are detected, they are cut in square images with size of 350 × 350 pixels. This image is then written and stored. The entire function of detecting faces is called at the end for each emotion in the list of emotions declared before. Now, the training and prediction lists are declared. Each item in the training and prediction dataset is put through the same set of functions: reading the image using OpenCV, converting it to grayscale, appending it into training or prediction data and then appending the training/prediction labels. This is repeated for all the emotions in the list of emotions previously declared. The recognizer is the run by training the dataset using fishface recognizer from the Fisherfaces module integrated in OpenCV. The current image is analyzed, and the sentiment is predicted.

Now, everything comes together. All the lists in which the final sentiment of each sentiment class of each modality were to be appended are declared here. Then, all the Preparation and training functions are called. After which, the three functions for analyzing the sentiment are called in a loop which runs over for all the clips that were cut from the main video file in the beginning.

Next, the final output for each sentiment is calculated. The ratio of text: audio: video for the clubbing of the modalities is done in the ratio of 40.22: 27.22: 32.06. This ratio is based on the accuracies of each of these three modalities. Once each of these is calculated, the total percentage of each is printed.

The formula used for each modality within each sentiment is:

$$Sentiment = \left(\frac{\sum modality\_sentiment}{No.\ of\ Clips} \times 100\right) \times Weight \quad (1)$$

V. Testing Methods

Software testing refers to the process of validation and verification of software programs aimed at discovering and rectifying errors as well as ensuring that requirements specified at both the business and the technical level are properly implemented. The various types of testing are shown below:

- Unit Testing: It refers to the testing of the individual component. Here, the individual components are Text, Audio and Video. Unit testing refers to the individual testing of the smallest testable components. It involves providing the requisite input parameters and checking if the outputs match the expected requirement.
- System Testing: System testing refers to a complete testing of the entire software application once all the modules have been integrated into it and ensuring compliance with the specified requirements.

VI. Results

First off, the video to be analyzed is put in the folder. The video used for this run is a video of a man reviewing a brand-new smartphone. It is 7 minutes and 24 seconds long. Here, it is named as "entire video". In the next stage, the video is split into 20 second clips. In this example, the video was split into 22 smaller clips. The file highlighted in Figure 2 is one of the 20-second clips.

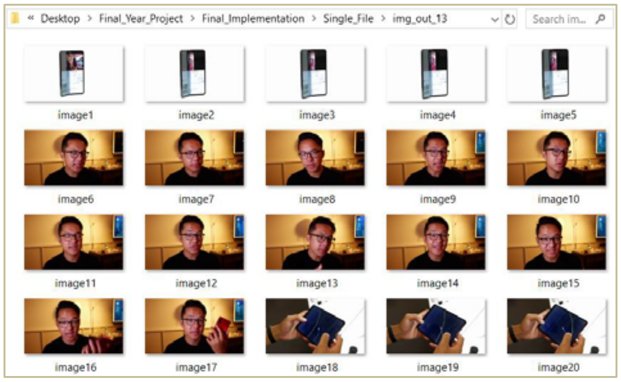
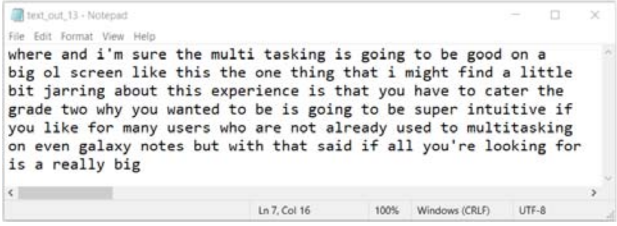


Fig. 2. Screen grags of Video

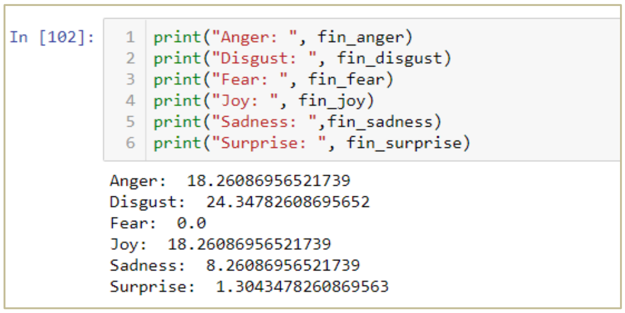
Next, the shorter video clips are turned into screenshots which are taken at the frequency of one frame per second. Further, these clips are split into audio clips and the audio clips are extracted to form text files as shown in Figure 3.

Once this preparation of data for review video is done all the analysis takes place in the backend. The next output is the percentage of each sentiment present in the video. Figure 4 shows the output of those percentages on Jupyter Notebooks.



```
where and i'm sure the multi tasking is going to be good on a
big ol screen like this the one thing that i might find a little
bit jarring about this experience is that you have to cater the
grade two why you wanted to be is going to be super intuitive if
you like for many users who are not already used to multitasking
on even galaxy notes but with that said if all you're looking for
is a really big
```

Fig. 3. Screenshot – Complete run (Text File)



```
In [102]: print("Anger: ", fin_anger)
          print("Disgust: ", fin_disgust)
          print("Fear: ", fin_fear)
          print("Joy: ", fin_joy)
          print("Sadness: ",fin_sadness)
          print("Surprise: ", fin_surprise)
```

```
Anger:  18.26086956521739
Disgust:  24.34782608695652
Fear:  0.0
Joy:  18.26086956521739
Sadness:  8.26086956521739
Surprise:  1.3043478260869563
```

Fig. 4. Screenshot – Complete run -Sentiment Percentages

The final output of the entire program is a bar chart showing percentages in Figure 5 below.

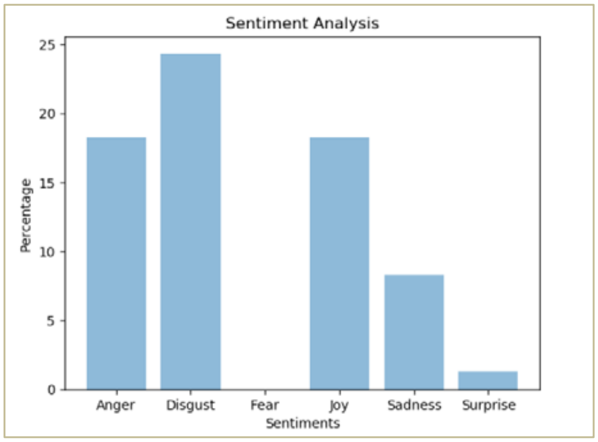


Fig. 5. Screenshot – Complete run -Sentiment Percentages

VII. Conclusion

There were a few shortcomings in this research work which were not handled effectively. First, the entire video clip was cut into multiple shorter clips based on time. Therefore, there was numerous instances wherein the last and the first word in a clip were cut. Then, the analysis may not work effectively for certain cases for people based on their ethnicity, voice modulation, accent etc. A larger database which takes into consideration all these factors would be required to handle this issue. Furthermore, the time required to run the program was too much to be used in the industry today. The time to fetch the output should be much lesser.

The aim of this research work was to successfully build a system which would analyze the sentiment of a video and give an output to the user. Multiple research papers were studied, and a mixture of those methods were implemented in a single program to perform Multimodal Sentiment Analysis successfully. Since the three modalities were analyzed separately and later clubbed together using weights, there is no firm accuracy that was obtained. But based on the observations during testing, the approximate accuracy of this method would be around 70%.

VIII. Future Work

Lot of research work is being done in this field. Some of the aspects to work on in the near future are:

- Speed of the system.
- Increasing the accuracy of the system.
- Creating better and more extensive databases.
- Making a user-friendly version of the Multimodal Sentiment Analysis system.
- Handling the cutting of words at the end of clips by making the program recognize sentences instead of splitting at time intervals.

References

[1] Chen, M.; Wang, S.; Liang, P. P.; Baltrusaitis, T.; Zadeh, A.; Morency, L.-P. 2017. Multimodal sentiment analysis with word-level fusion and reinforcement learning. In Proceedings of the 19th ACM International Conference on Multimodal Interaction, 163–171. ACM.

[2] A. Zadeh, R. Zellers, E. Pincus and L. Morency. Multimodal Sentiment Intensity Analysis in Videos: Facial Gestures and Verbal Messages. IEEE Intelligent Systems, vol. 31, no. 6, pp. 82-88, Nov.-Dec. 2016.

[3] S. Poria, E. Cambria, N. Howard, G.-B. Huang, A. Hussain, Fusing audio, visual and textual clues for sentiment analysis from multimodal content, Neurocomputing 174 (2016) 50-59.

[4] https://www.netguru.com/blog/how-to-estimate-a-machine-learning-project

[5] https://towardsdatascience.com/sentiment-analysis-concept-analysis-andapplications-6c94d6f58c17-learning-project-lifecycle

[6] https://towardsdatascience.com/sentiment-analysis-concept-analysis-andapplications-6c94d6f58c17-learning-project-lifecycle

[7] https://www.lexalytics.com/technology/sentiment-analysis

[8] https://docs.djangoproject.com/en/2.2/

[9] RAVDESS Dataset: https://zenodo.org/record/1188976#.XnudMIgzZPZ

[10] http://www.paulvangent.com/2016/04/01/emotion-recognition-with-python-opencv-and-a-face-dataset/

[11] Majumder, Navonil, Devamanyu Hazarika, Alexander Gelbukh, Erik Cambria, and Soujanya Poria. Multimodal sentiment analysis using hierarchical fusion with context modeling. Knowledge-based systems 161 (2018): 124-133.

[12] Zadeh A, Chen M, Poria S, Cambria E, Morency LP. Tensor fusion network for multimodal sentiment analysis. arXiv preprint arXiv:1707.07250. 2017 Jul 23.