MULTIMODAL PERSONALITY TRAIT ANALYSIS

Bonafide record of work done by

HARSHA VARDHAN V M	(20Z217)
KABILAN K K	(20Z223)
PRANAV P	(20Z237)
PREDNYA RAMESH	(20Z239)
SUVAN SATHYENDIRA B	(20Z256)

19Z620 – INNOVATION PRACTICES LAB

GUIDE: Dr.G.R.Karpagam

Dissertation submitted in the partial fulfilment of
the requirements for the degree of

BACHELOR OF ENGINEERING

BRANCH: COMPUTER SCIENCE AND ENGINEERING

of Anna University

April 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

PSG COLLEGE OF TECHNOLOGY
(Autonomous Institution)

COIMBATORE – 641 004

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE – 641 004

Bonafide record of work done by

HARSHA VARDHAN V M	(20Z217)
KABILAN K K	(20Z223)
PRANAV P	(20Z237)
PREDNYA RAMESH	(20Z239)
SUVAN SATHYENDIRA B	(20Z256)

Dissertation submitted in partial fulfilment of the requirements for the degree of

BACHELOR OF

Branch: Computer Science and Engineering

...……………………… ………….………………

Dr.G.R.Karpagam **Dr. Sudha Sadasivam G.**

Faculty guide Head of the Department.

Certified that the candidate was examined in the viva-voce examination held on …………………

………..…………….. …………………………..

(Internal Examiner) (External Examiner)

Certificate

This is to certify that Harsha Vardhan V M (20Z217), Mr. Kabilan K K (20Z223), Mr. Pranav P (20Z237) , Ms. Prednya Ramesh (20Z239) and Mr. Suvan Sathyendira B (20Z256) of BE (CSE) semester 6 has worked on a project entitled Multimodal Personality Trait Analysis under Dr.G.R.Karpagam during April 2023 as a part of the course on 19Z620 - Innovation Practices.

Investigator

Dept. of CSE
PSG College of Technology
Coimbatore

ACKNOWLEDGEMENT

We would like to extend sincere thanks to our Principal, Dr. K. Prakasan, for providing us with this opportunity to develop and complete our project in our field of study.

We would like to express our sincere thanks to our Head of Department, Computer Science and Engineering, Dr. G. Sudha Sadasivam, for encouraging and allowing us to present the project at our department premises for the partial fulfilment of the requirements leading to the award of Bachelor of Engineering degree.

We take immense pleasure in conveying our thanks and a deep sense of gratitude to the Programme Coordinator, Dr N. Arul Anand, Department of Computer Science and Engineering, for providing the necessary resources, facilities, and opportunities to pursue our undergraduate project.

We would like to thank our Tutor, Mrs T. Anusha, Department of Computer Science and Engineering, for her valuable input, guidance, and encouragement throughout our project.

We extend our heartfelt thanks and a deep sense of gratitude to the Faculty Guide, Dr. G. R. Karpagam, Department of Computer Science and Engineering for her valuable insights, suggestions, and inputs that have helped us enhance the quality of our project. Her expertise and knowledge have been instrumental in making this project a success. We would like to thank Dr. Santhi V, Committee Member for her valuable suggestions and motivation throughout the project.

SYNOPSIS

The Multimodal Personality Trait Analysis project has the goal of investigating the connection between various modalities, including speech, facial expressions, body language, and personality traits. Following this, a range of data collection methods will be used, including recording participants in diverse social situations, conducting interviews, and administering personality tests. The data will be meticulously labelled to ensure that the modalities can be analyzed meaningfully.

Personality trait analysis will be carried out by utilizing machine learning algorithms, such as deep neural networks and clustering techniques, to identify patterns and correlations between the modalities and personality traits. The outcomes will be verified through statistical analysis and compared to existing research to guarantee their accuracy and dependability.

Ultimately, the project will wrap up with the interpretation of the outcomes and the creation of suggestions for future research and practical uses of multimodal personality trait analysis. The project has the potential to make contributions to the advancement of novel tools and techniques for personality assessment, and may have important applications in areas such as psychology, human resources, and marketing.

TABLE OF CONTENTS **PAGE**

CHAPTER 1
INTRODUCTION

Chapter 1 deals with the introduction of the project. It describes the motivation, Problem Statement, Project Objectives and block diagram depicting the overall work of the title of project work.

1.1 PROBLEM STATEMENT:

Identify the commonly known big-five personality traits such as Conscientiousness, Agreeableness, Neuroticism, Openness and Extraversion from the given input.

1.2 MOTTO:

With this extensive amount of media data available online, the conventional text-based motivation analysis has evolved into more complex models of multimodal personality trait analysis, which can be applied in the development of virtual assistants, analysis of personality for job interviews, choosing career movie reviews, analysis of news videos, interview videos are a few among the given.

1.3 ORGANIZATION OF REPORT:

Chapter 1: deals with introduction and problem statement of the project
Chapter 2: deals with survey of the existing works in the scope of project work.
Chapter 3: deals with the designed system’s workflow, modules and dataset.
Chapter 4: utilized for model construction and deep insight into the system.
Chapter 5: deals with the hardware and software specifications of the system.
Chapter 6: explains how the system is implemented with the help of modules.
Chapter 7: concludes the model built and provides an insight on how the model can be extended for future work.

CHAPTER 2
LITERATURE SURVEY

Chapter 2 deals with the survey of the existing works in the scope of the project work. It presents the key takeaways from different research papers and articles.

2.1 INTRODUCTION:

In domains like Computer Vision research, personality recognition is being intensively investigated. The ability to develop intelligent systems that reliably understand personalities became a more in-depth reality with the recent rise and popularization of Machine Learning and Deep Learning approaches.
Marketing, psychology, surveillance, and entertainment are just a few examples of industries where personality recognition has gotten a lot of interest due to its practical implications in today's society. A variety of techniques can be used to recognize personality.

Some of them embody architectures like:
Inception-v1, Inception-v3, ResNet-50, Xception, Inception-v4, Inception-ResNets, VGG-16, ResNeXt-50, Bidirectional LSTM (Long Short Term Memory), Convolutional Neural Network (CNN), BERT, VGGish.

The series of steps involved in this process are (however some of the substeps could also be optional):

Pre-Processing:

- Separation of video, audio and text
- Face Detection
- Image Processing
- Smoothing
- Removing Punctuations, Stopwords, Contraction of Words
- Normalization of words
- Extraction of Audio Spectrum
- Extracting Amplitude for each Frequency Frame
- Extraction of MFCC (Mel-frequency Cepstrum Coefficients)

Feature Extraction:

Text-based Features:

- Bag-of-Words TF-IDF Word Embeddings (e.g., Word2Vec, GloVe, BERT)

Audio-based Features:

- Pitch
- Mel-Frequency Cepstral Coefficients (MFCCs)
- Energy
- Spectral Centroid
- Spectral Flux
- Zero Crossing Rate

Visual-based Features:

- Facial Landmarks
- Facial Action Units
- Gaze
- Visual Appearance

Classification/Regression:

Multimodal Deep Learning Models:

- Multimodal CNNs Multimodal
- MFCC
- BERT

Results and Discussion:

- Interpretation and analysis of the results to gain insights into the relationship between different modalities and personality traits.

2.2 RELATED WORKS:

1. Deep Personality Trait Analysis Recognition: A Survey

Talks on a comprehensive survey on existing personality trait recognition methods from a computational perspective. Based on the datasets used and using the recent advances of typical deep learning techniques, including deep belief networks (DBNs), convolutional neural networks(CNNs), and recurrent neural networks(RNNs).These methods are analyzed and summarized in both single modality and multiple modalities, such as audio, visual, text, and physiological signals.

2. A Multi-modal Personality Prediction System

Talks on a solution framework for solving the problem of predicting the personality traits of a user from videos. Ambient, facial and the audio features are extracted from the video of the user based on which the personality is detected.

3. Multimodal analysis of personality traits on videos of self-presentation and induced behavior

Talks on the multimodal deep architectures to estimate the Big Five personality traits from (temporal) audio-visual cues and transcribed speech. For a detailed analysis of personality traits, a new audio-visual dataset, namely: Self-presentation and Induced Behavior Archive for Personality Analysis (SIAP) was collected. In contrast to the available datasets, SIAP introduces recordings of induced behavior in addition to self-presentation (speech) videos. This also shows that the induced behavior indeed includes signs of personality traits.

4. Multimodal Video Sentiment Analysis Using Deep Learning Approaches, a Survey

An overview of various deep learning-based techniques used for multimodal video sentiment analysis. The authors highlight the importance of analyzing emotions and sentiment in videos, and how it can be used in applications such as video recommendation systems, video summarization, and video search engines.

2.3 OBSERVATIONS:

2.3.1 CONVOLUTIONAL NEURAL NETWORK:

Convolutional neural networks are a specialized type of artificial neural network that uses a mathematical operation called convolution in place of general matrix multiplication in at least one of their layers.

Size - 2.60 MB
Accuracy - 0.7235

Several convolution and pooling layers are added before the prediction is made. Convolutional layers help in extracting features. As deeper the network is, more specific features are extracted as compared to a shallow network where the features extracted are more generic.

2.3.2 VGGish:

VGGish is a Convolutional Neural Network (CNN) architecture that is specifically designed for audio classification tasks, such as audio event detection, audio tagging, and audio-based multimedia retrieval.

Features:
Size - 123 MB
Top 1 accuracy – 0.744p 5 accuracy – 0.927
Parameter - 72,141,184
Input Size – 96 x 64

Overall, VGGish is a highly effective and efficient CNN architecture for audio classification tasks, with a relatively small number of parameters compared to other audio classification models. It has been pre-trained on a large dataset of audio signals and can be fine-tuned on a specific audio classification task with only a small amount of additional training data.

2.3.3 BERT:

BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained Transformer-based neural network architecture for natural language processing (NLP) tasks.

Features:

Size - 110 MB
Accuracy – 0.938
Parameter:

- **Uncased** - 110 million
- **Cased** - 340 million

BERT has a large number of parameters, with the base model containing approximately 110 million parameters. However, it is computationally efficient at inference time due to its parallel processing capabilities.

CHAPTER 3
PROPOSED SYSTEM

Chapter 3 deals with the designed system's workflow along with its description, list of modules, and the dataset used for training purposes.

3.1 WORKFLOW:

1. Capture Videos of a pool of participants.
2. After collecting all the data, the data is cleaned by extracting images, prosody, and semantics from the captured video, audio and text respectively.
3. After the creation of corresponding models, a multimodal is built based on which each participant is mapped to a given set of personality traits.

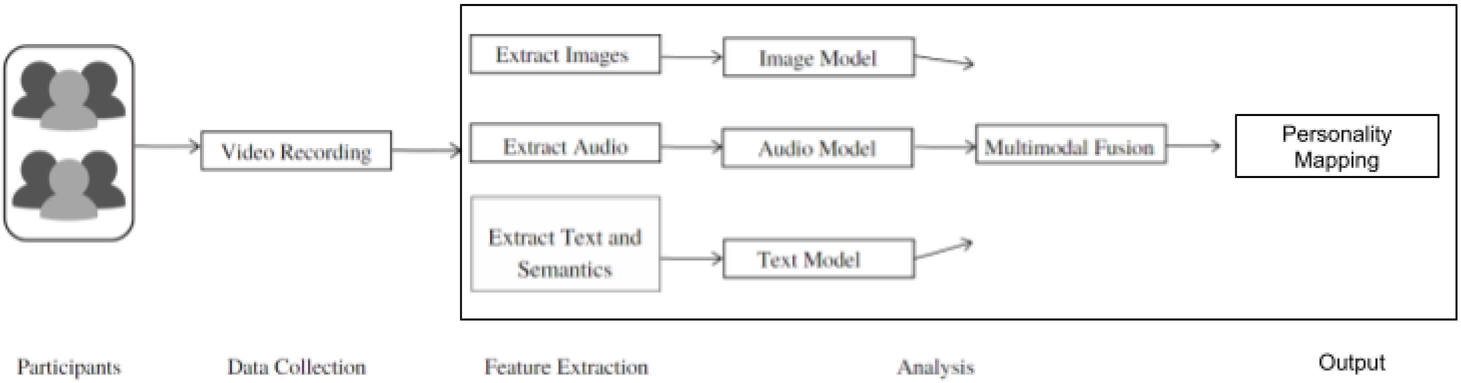


Figure 1: Workflow of the project

3.2 MODULES:

3.2.1 INPUT PREPROCESSING:

1. Audio Extraction Module - It converts the given input video data to wav audio format using Pydub function

2. Text Extraction Module - It takes the extracted wav audio file and converts it to a text using the recognise_google() method of the Recognizer object of the speech_recognition library.

3. Image Extraction Module - It extracts image frames from the input video data using OpenCV.

3.2.2 VIDEO MODEL

1. Preprocessing Module: Extracts images from video input and resizes it to 200x200 pixels for training and testing .Under each of the sub-categories, it is divided into five personality traits.

2. CNN MODEL: The model has two convolutional layers followed by max pooling layers. It has a dropout layer to prevent overfitting and a fully connected layer with ReLU activation function. The output layer uses softmax activation function with 5 units for 5 classes. The model is trained using Adam optimizer with categorical cross-entropy loss function and accuracy as the metric for nb_epochs iterations and batch_size batch size. Validation set is used to evaluate the model after each epoch.

3.2.3 TEXT MODEL

1.Preprocessing Module: Pre-processing for BERT includes removing punctuations, stopwords, and normalization. BERT doesn't require contraction due to its contextual understanding. Dataset will be split into 80% training and 20% testing with 546 rows in the test set.

2. Embedding Module: Pre-trained BERT has contextualized embeddings that map words to vectors and it will be loaded using the Hugging Face library and tokenize pre-processed dataset sentences.
3. Fine-tuning Module: Fine-tune the pre-trained BERT on emotion classification task with training set. Model's weights will be updated, and classification layer added to predict personality labels. Testing set will be used for evaluation.

4. Classification Module: Use a trained model to predict emotions on the test dataset by passing pre-processed and tokenized sentences. Obtain probabilities for each personality class which is stored as an array for each input sentence.

3.2.4 AUDIO MODEL

1. Data Pre-processing: Training data pre-processing involves converting audio files to spectrum, extracting amplitude for frequency frames. MFCC features are extracted by applying DFT, taking log of magnitude, warping frequencies on a Mel scale, and applying inverse DCT. Data is pre-processed and converted into mathematical form for ML model training.

2. Model Building Module: Training an audio model using a CNN involves feeding large amounts of labeled audio data into the network, allowing it to learn relevant features through convolutional filters and optimizing the model through backpropagation.

3.2.5 COMBINER MODULE

The Combiner module acquires and integrates outputs from trained models, assigning weights based on performance. Video, audio and text models have weights of 0.5, 0.1, and 0.4 respectively.

3.3 TRAINING DATASET:

The dataset used in the report comprises 10,000 video clips, with an average duration of 15 seconds, extracted from over 3,000 different high-definition YouTube videos. The videos feature people of different genders, ages, nationalities, and ethnicities, speaking in English to a camera. The videos are labeled with personality trait variables based on the Five Factor Model, which models human personality along five dimensions: Extraversion, Agreeableness, Conscientiousness, Neuroticism, and Openness. The labels were generated using Amazon Mechanical Turk's crowd labeling system.

The dataset is structured as a zip file containing videos in MP4 format. Text transcripts and ground truth values are provided in the form of a pickle dictionary. The ground truth values for each video contain predictions for each class, rather than a probability distribution over the classes that sums up to 1.

Link: https://chalearnlap.cvc.uab.cat/dataset/24/data/41/files/

This dataset is specifically designed for analyzing the Big Five personality traits through the use of video, audio, and text data. The dataset includes video clips in MP4 format, along with text transcripts and ground truth values provided in the form of a pickle dictionary. The data can be used to train models for predicting the Big Five personality traits of individuals based on their speech patterns, facial expressions, and other behavioral cues. This makes it a valuable resource for researchers and practitioners in fields such as psychology, sociology, and computer science, who are interested in understanding and predicting human behavior.

CHAPTER 4
SYSTEM DESIGN

Chapter 4 gives deep insight into the design of the system through use case diagram and sequence diagram and the architecture utilized for model construction using graphics that provide a full explanation of the layers existing deep within the model.

4.1 DESIGN

4.1.1 USE CASE DIAGRAM

The functionality of our system is explained through the following use-case diagram:

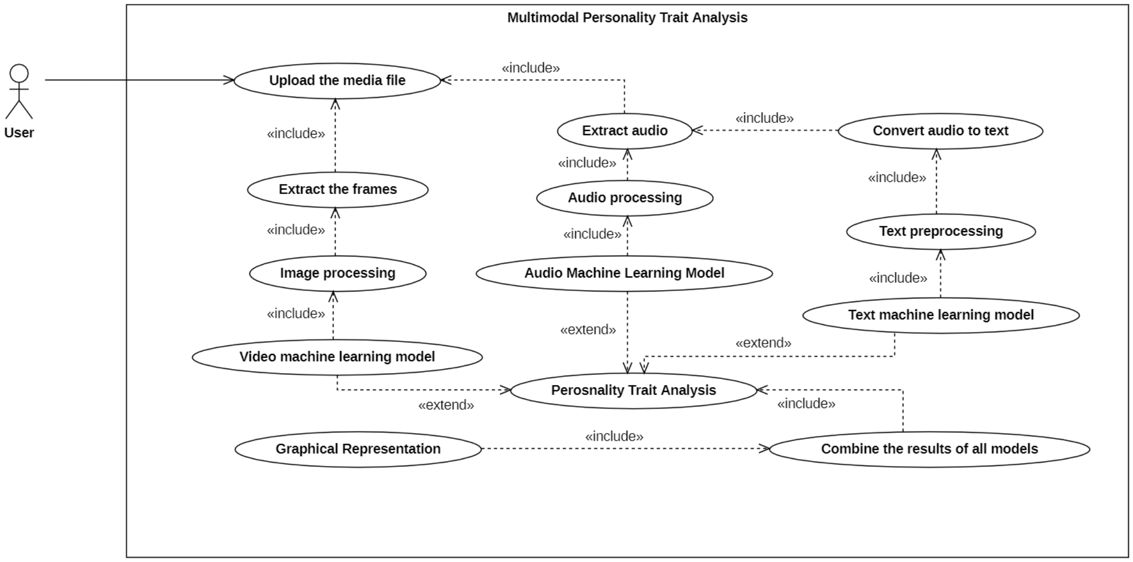


Figure 2: Use Case diagram

4.1.2 SEQUENCE DIAGRAM:

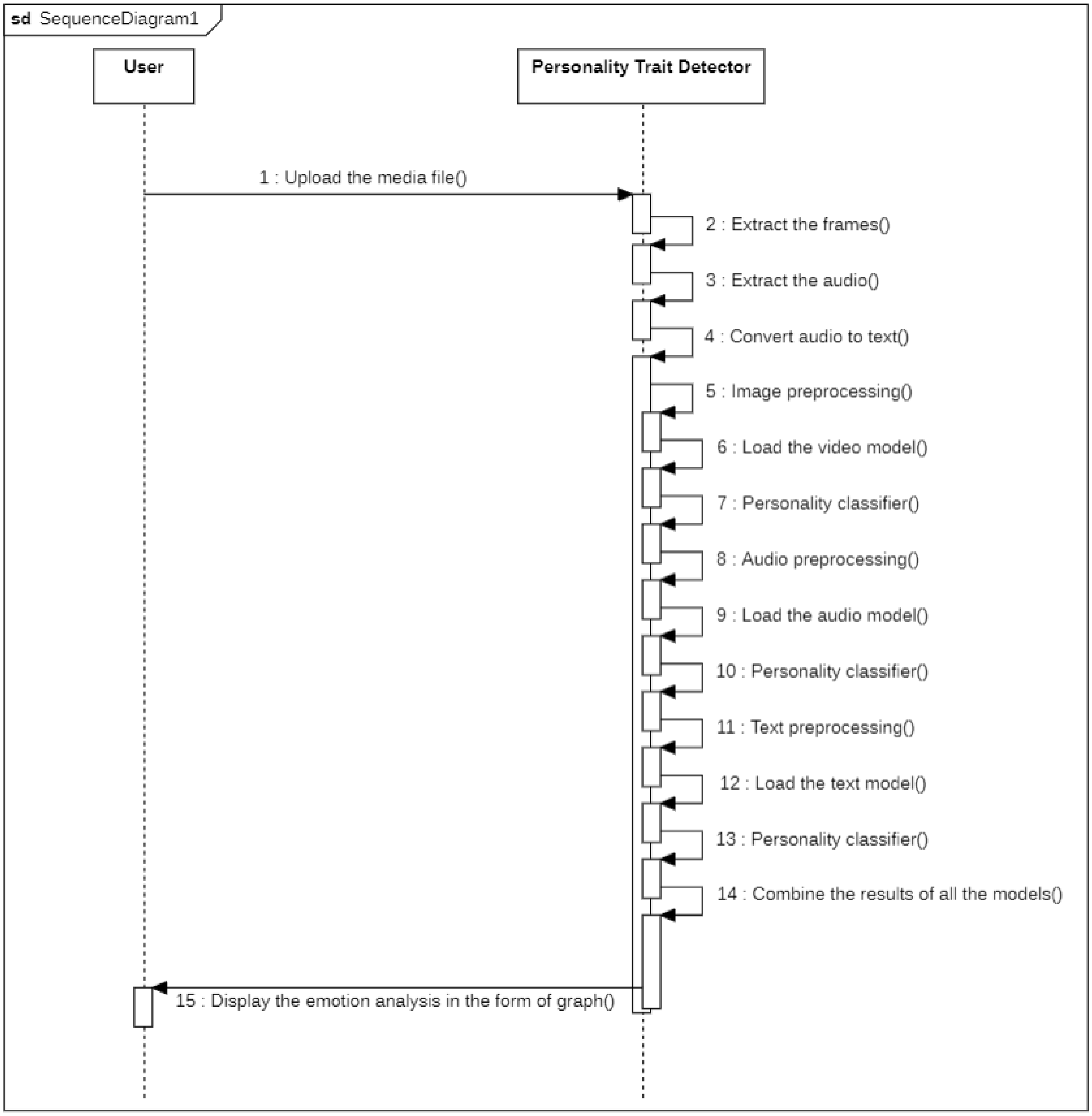

sd SequenceDiagram1
User
Personality Trait Detector
1 : Upload the media file()
2 : Extract the frames()
3 : Extract the audio()
4 : Convert audio to text()
5 : Image preprocessing()
6 : Load the video model()
7 : Personality classifier()
8 : Audio preprocessing()
9 : Load the audio model()
10 : Personality classifier()
11 : Text preprocessing()
12 : Load the text model()
13 : Personality classifier()
14 : Combine the results of all the models()
15 : Display the emotion analysis in the form of graph()

Figure 3: Sequence diagram

4.2 ARCHITECTURE

4.2.1 CONVOLUTIONAL NEURAL NETWORK:

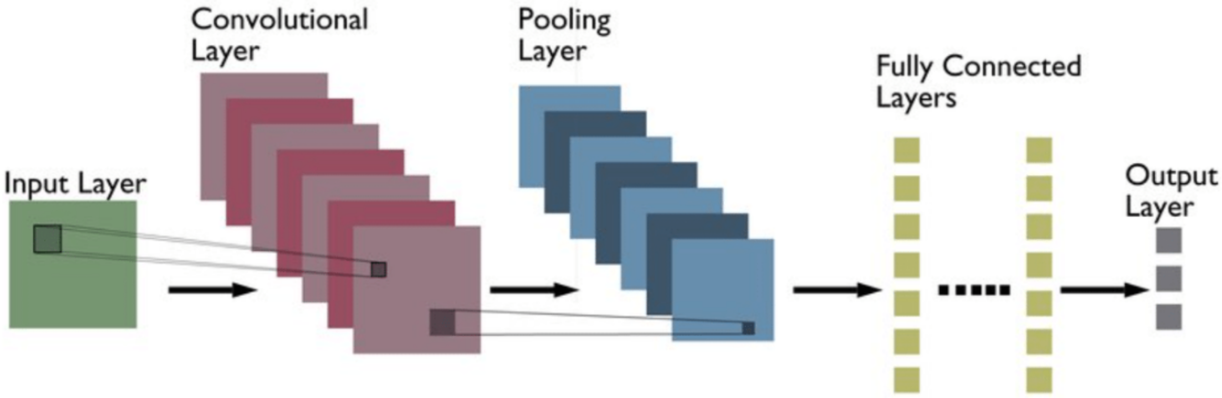
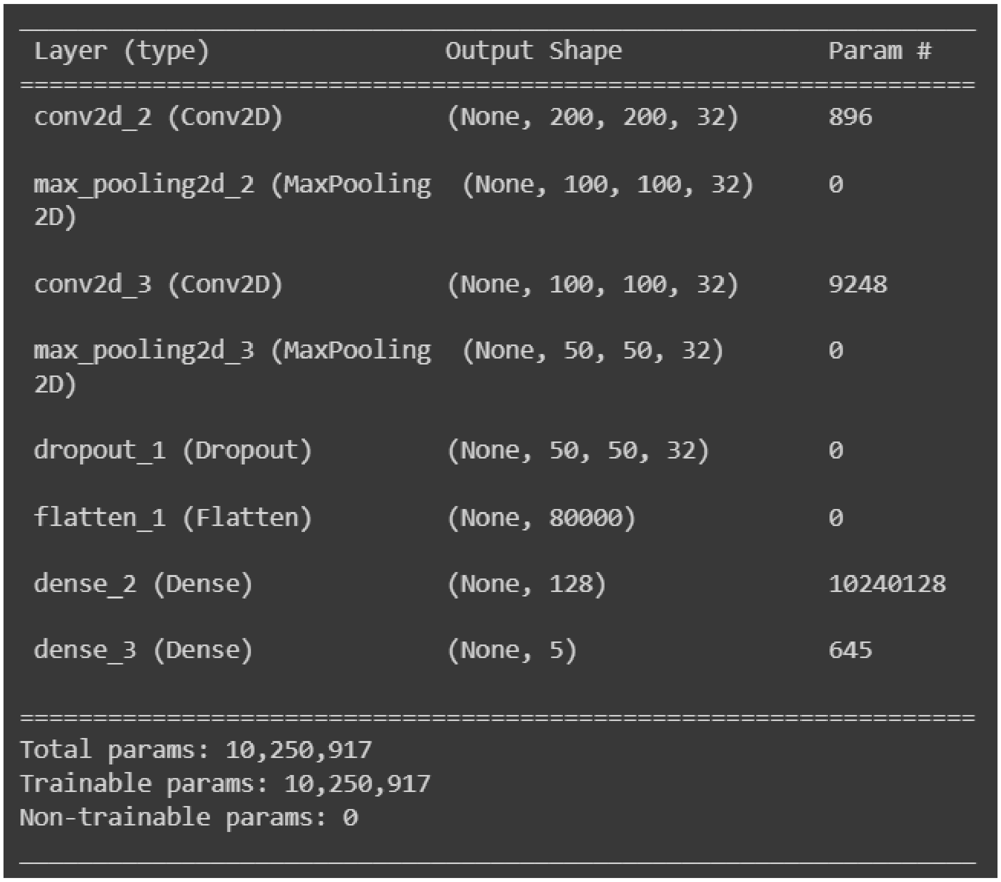


Figure 4: Convolutional Neural Network Architecture for Video Model



```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_2 (Conv2D)           (None, 200, 200, 32)      896

 max_pooling2d_2 (MaxPooling  (None, 100, 100, 32)     0
 2D)

 conv2d_3 (Conv2D)           (None, 100, 100, 32)      9248

 max_pooling2d_3 (MaxPooling  (None, 50, 50, 32)       0
 2D)

 dropout_1 (Dropout)         (None, 50, 50, 32)        0

 flatten_1 (Flatten)         (None, 80000)             0

 dense_2 (Dense)             (None, 128)               10240128

 dense_3 (Dense)             (None, 5)                 645

=================================================================
Total params: 10,250,917
Trainable params: 10,250,917
Non-trainable params: 0
_________________________________________________________________
```

Figure 5: Layers for Video model

4.2.2 VGGish:

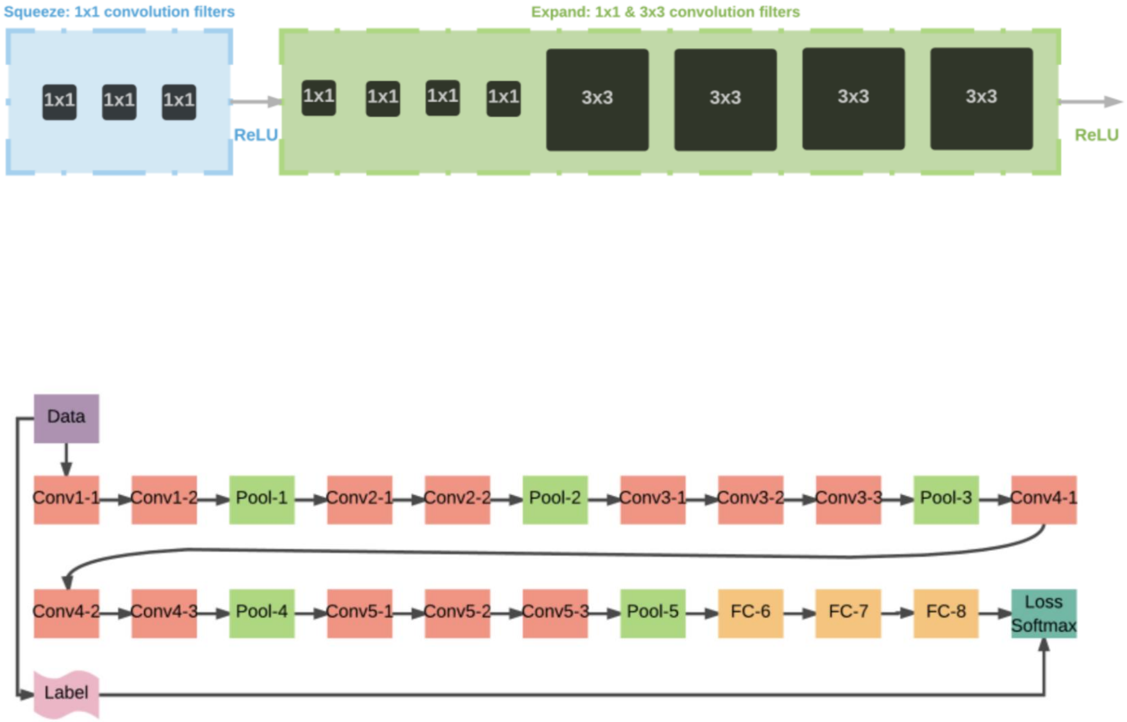
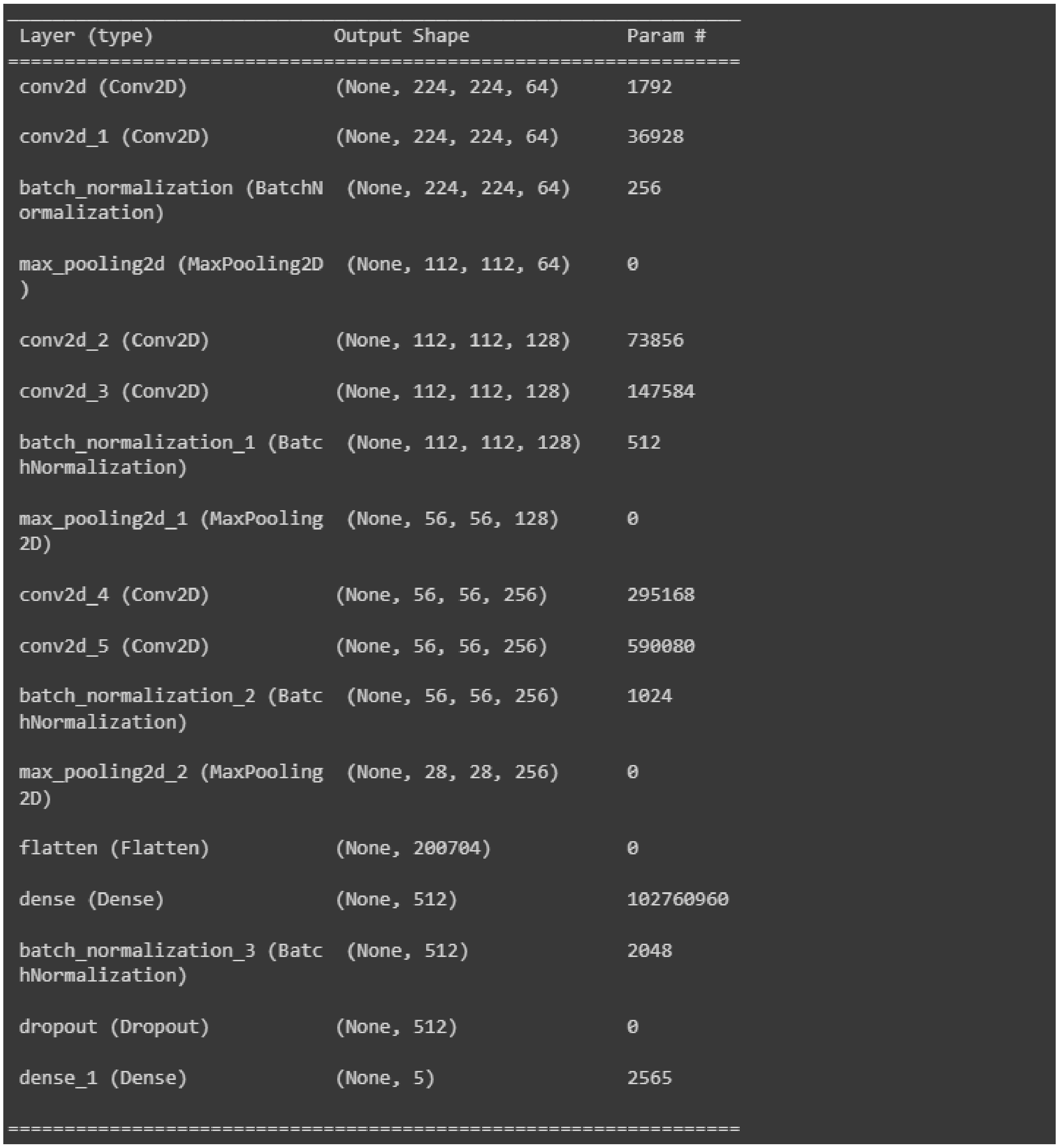


Figure 6: VGGish Architecture for Audio Model



```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 224, 224, 64)      1792

 conv2d_1 (Conv2D)           (None, 224, 224, 64)      36928

 batch_normalization (BatchN  (None, 224, 224, 64)     256
 ormalization)

 max_pooling2d (MaxPooling2D  (None, 112, 112, 64)     0
 )

 conv2d_2 (Conv2D)           (None, 112, 112, 128)     73856

 conv2d_3 (Conv2D)           (None, 112, 112, 128)     147584

 batch_normalization_1 (Batc  (None, 112, 112, 128)    512
 hNormalization)

 max_pooling2d_1 (MaxPooling  (None, 56, 56, 128)      0
 2D)

 conv2d_4 (Conv2D)           (None, 56, 56, 256)       295168

 conv2d_5 (Conv2D)           (None, 56, 56, 256)       590080

 batch_normalization_2 (Batc  (None, 56, 56, 256)      1024
 hNormalization)

 max_pooling2d_2 (MaxPooling  (None, 28, 28, 256)      0
 2D)

 flatten (Flatten)           (None, 200704)            0

 dense (Dense)               (None, 512)               102760960

 batch_normalization_3 (Batc  (None, 512)              2048
 hNormalization)

 dropout (Dropout)           (None, 512)               0

 dense_1 (Dense)             (None, 5)                 2565

=================================================================
```

Figure 7: Layers used in Audio model

4.2.3 BERT:

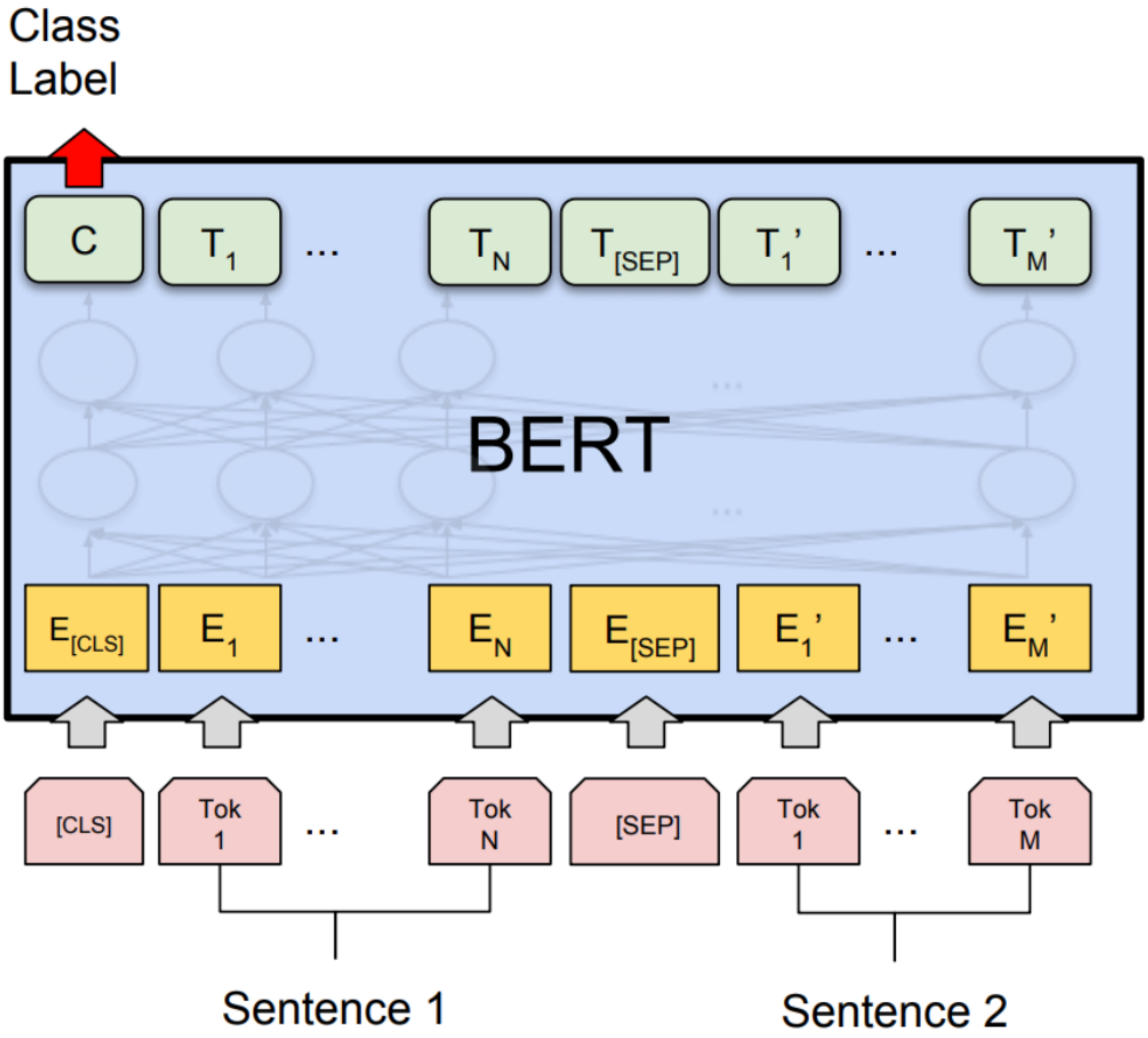


Figure 8: BERT Architecture for Text Model

CHAPTER 5
SYSTEM SPECIFICATIONS

Chapter 5 will focus on discussing the specifications of both hardware and software of the system.

5.1 HARDWARE:

- A high-performance computer with enough RAM and storage space to handle the data and algorithms involved, GPU is preferred.

5.2 SOFTWARE:

- Using the programming language as Python.
- Tools for data pre-processing and cleaning, such as NumPy and Pandas.
- Machine learning libraries such as scikit-learn and TensorFlow.
- Libraries for processing multimedia data, such as OpenCV.

CHAPTER 6
IMPLEMENTATION

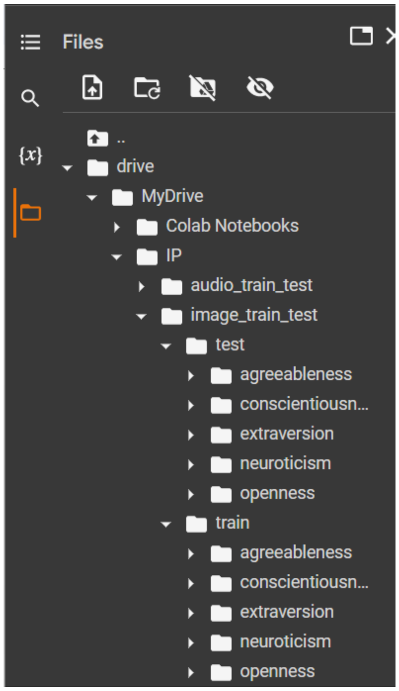
Chapter 6 explains how the system is implemented with the help of modules.

Windows platform with command prompt or Linux with terminal or any suitable IDE for python development or Google Colab is used.

6.1 VIDEO MODEL

6.1.1 Pre-processing:

The dataset is structured in a hierarchical manner with a main directory located at "drive/MyDrive/IP/image_train_test". Within this directory, there are two subdirectories called "train" and "test". These subdirectories contain five subdirectories each, one for each of the five personality traits being studied: "agreeableness", "conscientiousness", "extraversion", "neuroticism", and "openness". Finally, within each of these five trait subdirectories, there are a large number of images.



```
..
drive
  MyDrive
    Colab Notebooks
    IP
      audio_train_test
      image_train_test
        test
          agreeableness
          conscientiousn...
          extraversion
          neuroticism
          openness
        train
          agreeableness
          conscientiousn...
          extraversion
          neuroticism
          openness
```

Figure 9: Hierarchy of split (test and train)

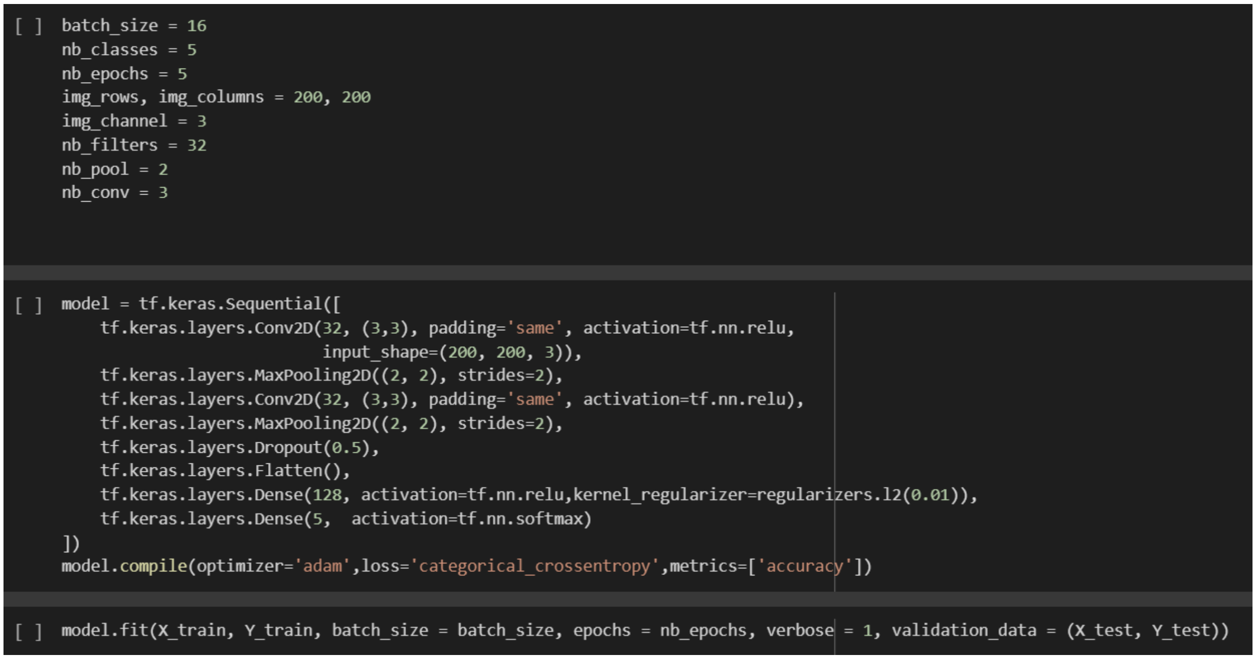
In each category under the train folder, 1000 randomly selected image files from the corresponding subdirectory are resized to 200x200 pixels and adds it, along with its corresponding class number (determined by the index of the category in CATEGORIES), to the training list. The same is done for the test folder but 60 images are selected in random order. The training list and testing list are shuffled. The features(images) and labels(categories) are separated and assigned to x and y lists of the training and testing lists respectively. Images are normalized. Labels are converted to categorical data in y lists.

Figure 10: Image Extraction

6.1.2 CNN MODEL :

The model consists of two convolutional layers with ReLU activation function, each followed by a max pooling layer. The first convolutional layer has an input shape of (200, 200, 3) and outputs 32 feature maps, while the second convolutional layer has an input shape of (100, 100, 32) and also outputs 32 feature maps. A dropout layer is added after the second max pooling layer with a rate of 0.5 to prevent overfitting. The output of the second max pooling layer is flattened and connected to a fully connected layer with 128 units and ReLU activation function, which is then connected to the output layer with a softmax activation function, having 5 units, corresponding to the number of classes.

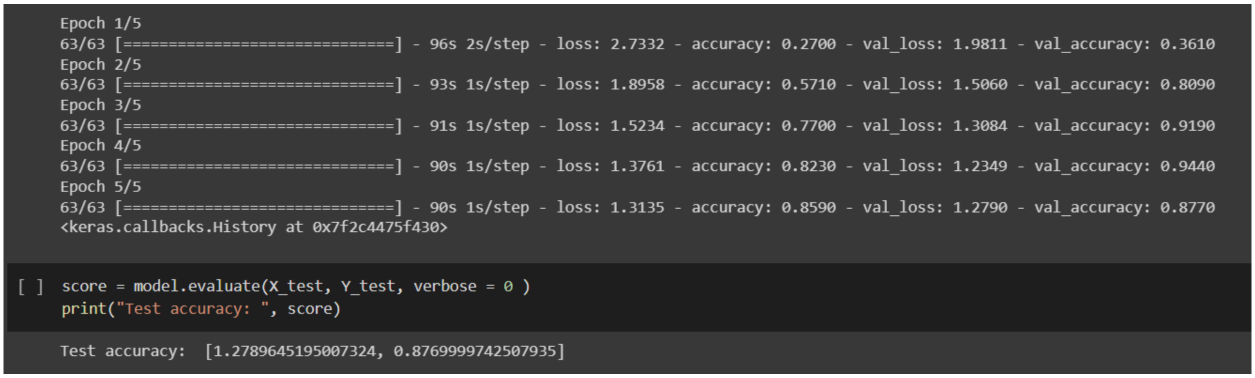
The model is then compiled using the Adam optimizer with categorical cross-entropy as the loss function and accuracy as the metric. Finally, the model is trained using the fit method with the training set X_train and its corresponding labels Y_train. The training is performed for nb_epochs iterations with a batch size of batch_size. The validation set X_test and its corresponding labels Y_test are also provided to evaluate the model after each epoch, and the training progress is displayed with a verbosity of 1.



```
[ ]  batch_size = 16
     nb_classes = 5
     nb_epochs = 5
     img_rows, img_columns = 200, 200
     img_channel = 3
     nb_filters = 32
     nb_pool = 2
     nb_conv = 3
```

```
[ ]  model = tf.keras.Sequential([
         tf.keras.layers.Conv2D(32, (3,3), padding='same', activation=tf.nn.relu,
                                input_shape=(200, 200, 3)),
         tf.keras.layers.MaxPooling2D((2, 2), strides=2),
         tf.keras.layers.Conv2D(32, (3,3), padding='same', activation=tf.nn.relu),
         tf.keras.layers.MaxPooling2D((2, 2), strides=2),
         tf.keras.layers.Dropout(0.5),
         tf.keras.layers.Flatten(),
         tf.keras.layers.Dense(128, activation=tf.nn.relu,kernel_regularizer=regularizers.l2(0.01)),
         tf.keras.layers.Dense(5,  activation=tf.nn.softmax)
     ])
     model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])
```

```
[ ]  model.fit(X_train, Y_train, batch_size = batch_size, epochs = nb_epochs, verbose = 1, validation_data = (X_test, Y_test))
```



```
Epoch 1/5
63/63 [==============================] - 96s 2s/step - loss: 2.7332 - accuracy: 0.2700 - val_loss: 1.9811 - val_accuracy: 0.3610
Epoch 2/5
63/63 [==============================] - 93s 1s/step - loss: 1.8958 - accuracy: 0.5710 - val_loss: 1.5060 - val_accuracy: 0.8090
Epoch 3/5
63/63 [==============================] - 91s 1s/step - loss: 1.5234 - accuracy: 0.7700 - val_loss: 1.3084 - val_accuracy: 0.9190
Epoch 4/5
63/63 [==============================] - 90s 1s/step - loss: 1.3761 - accuracy: 0.8230 - val_loss: 1.2349 - val_accuracy: 0.9440
Epoch 5/5
63/63 [==============================] - 90s 1s/step - loss: 1.3135 - accuracy: 0.8590 - val_loss: 1.2790 - val_accuracy: 0.8770
<keras.callbacks.History at 0x7f2c4475f430>
```

```
[ ] score = model.evaluate(X_test, Y_test, verbose = 0 )
    print("Test accuracy: ", score)
```

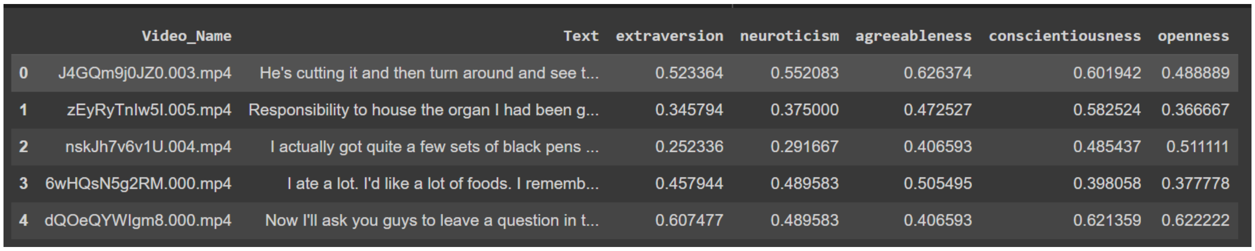
```
Test accuracy:  [1.2789645195007324, 0.8769999742507935]
```

Figure 11: Model training

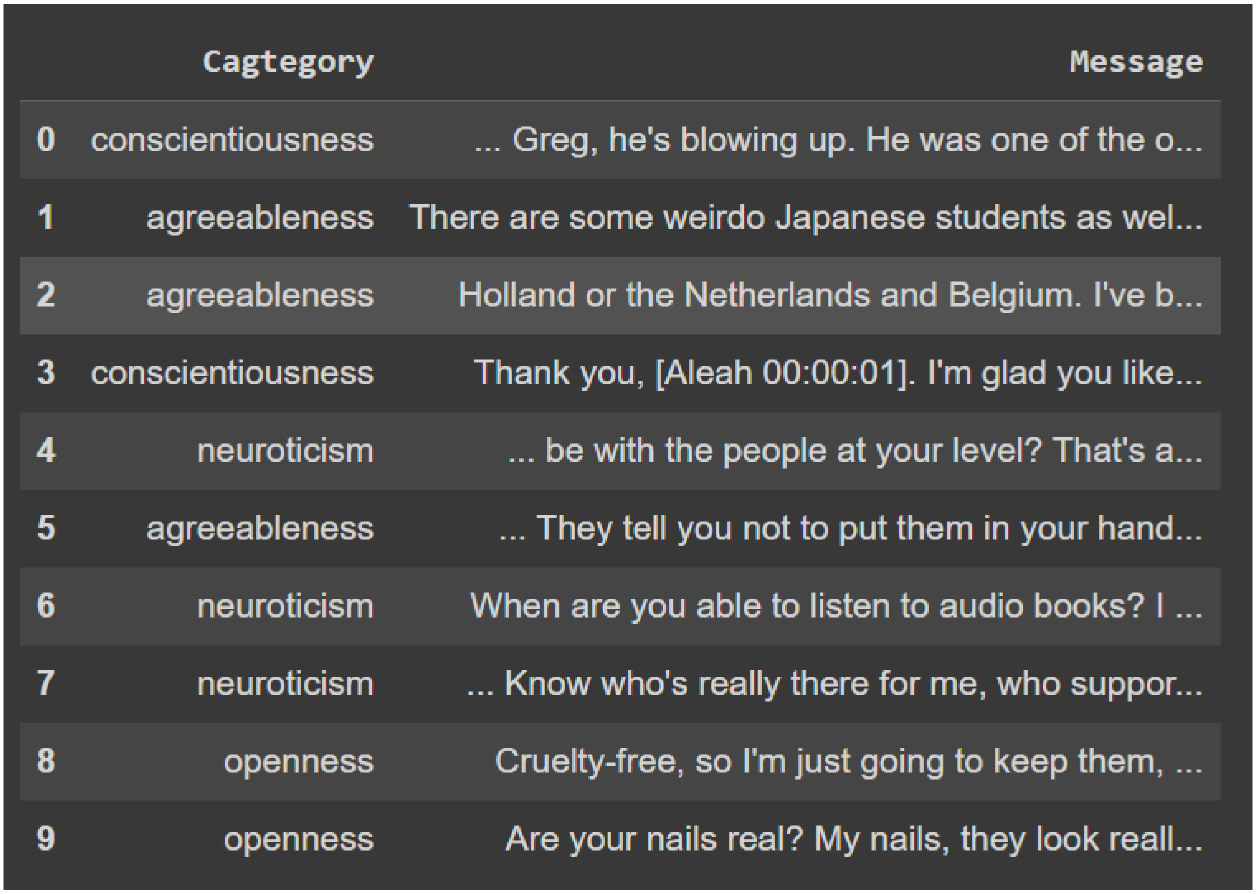
6.2 TEXT MODEL

6.2.1 Pre-processing:

The pre-processing steps for training the BERT transformer model will include removing punctuations, stopwords, and normalization of words. However, for BERT, contraction of words is not needed as it has a better understanding of the context of words. Once the pre-processing is complete, the dataset will be split into training and testing sets. In this case, 20% of the dataset, which is 546 rows, will be used for testing.



	Video_Name	Text	extraversion	neuroticism	agreeableness	conscientiousness	openness
0	J4GQm9j0JZ0.003.mp4	He's cutting it and then turn around and see t...	0.523364	0.552083	0.626374	0.601942	0.488889
1	zEyRyTnlw5I.005.mp4	Responsibility to house the organ I had been g...	0.345794	0.375000	0.472527	0.582524	0.366667
2	nskJh7v6v1U.004.mp4	I actually got quite a few sets of black pens ...	0.252336	0.291667	0.406593	0.485437	0.511111
3	6wHQsN5g2RM.000.mp4	I ate a lot. I'd like a lot of foods. I rememb...	0.457944	0.489583	0.505495	0.398058	0.377778
4	dQOeQYWIgm8.000.mp4	Now I'll ask you guys to leave a question in t...	0.607477	0.489583	0.406593	0.621359	0.622222

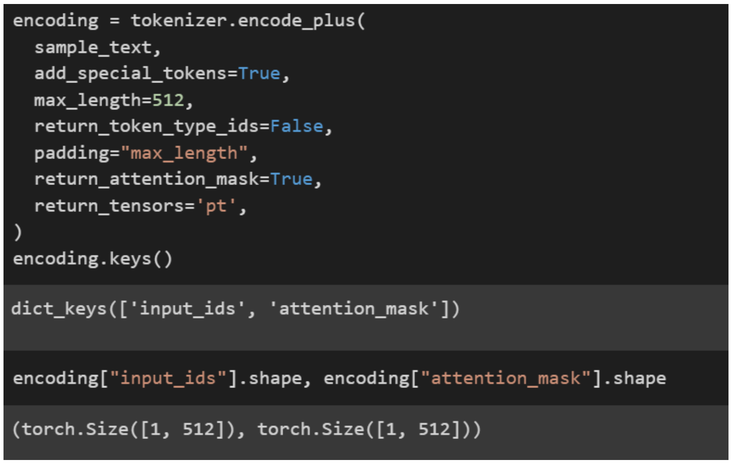


	Cagtegory	Message
0	conscientiousness	... Greg, he's blowing up. He was one of the o...
1	agreeableness	There are some weirdo Japanese students as wel...
2	agreeableness	Holland or the Netherlands and Belgium. I've b...
3	conscientiousness	Thank you, [Aleah 00:00:01]. I'm glad you like...
4	neuroticism	... be with the people at your level? That's a...
5	agreeableness	... They tell you not to put them in your hand...
6	neuroticism	When are you able to listen to audio books? I ...
7	neuroticism	... Know who's really there for me, who suppor...
8	openness	Cruelty-free, so I'm just going to keep them, ...
9	openness	Are your nails real? My nails, they look reall...

Figure 12: Data Pre-Processing

6.2.2 Embedding module:

For the BERT transformer model, a pre-trained BERT model is used, which has its own embeddings. These embeddings map words to vectors that are contextualized based on the sentence in which they appear. Hugging Face library is used to load a pre-trained BERT model and tokenize the pre-processed sentences from our dataset.



```
encoding = tokenizer.encode_plus(
  sample_text,
  add_special_tokens=True,
  max_length=512,
  return_token_type_ids=False,
  padding="max_length",
  return_attention_mask=True,
  return_tensors='pt',
)
encoding.keys()
```

```
dict_keys(['input_ids', 'attention_mask'])
```

```
encoding["input_ids"].shape, encoding["attention_mask"].shape
```

```
(torch.Size([1, 512]), torch.Size([1, 512]))
```

Figure 13: Embedding module

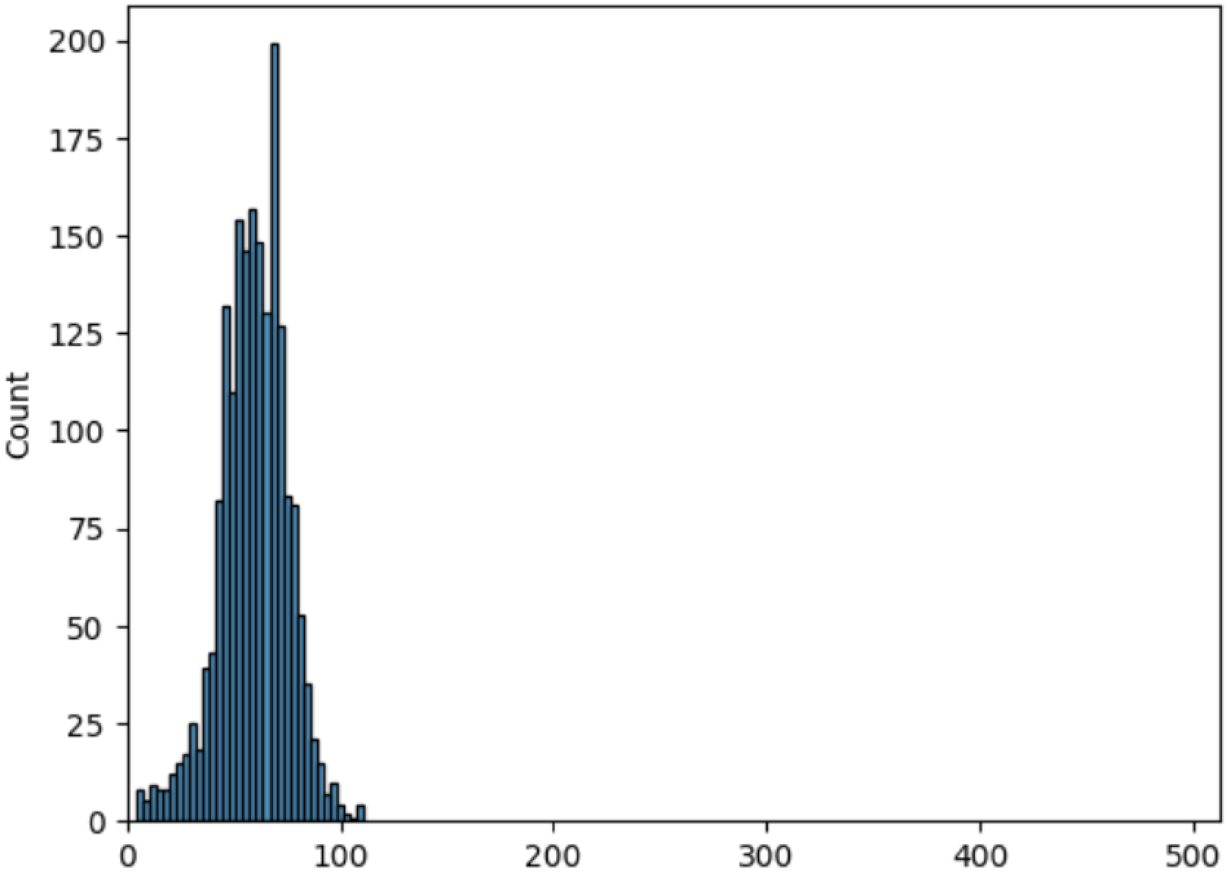


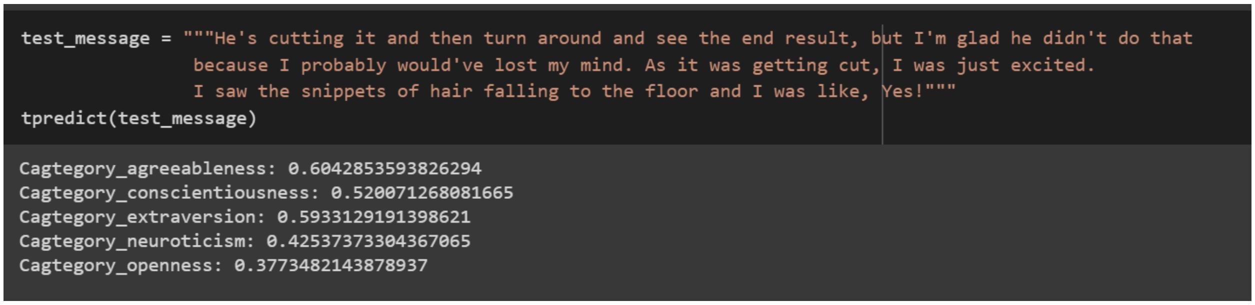
Figure 14: Length of Token vs Count of Length

6.2.3 Fine-tuning Module:

After obtaining the BERT embeddings for our dataset, fine-tuning the pre-trained BERT model on our specific task of emotion classification is done. Andl training the model using the training set and evaluating its performance on the testing set. During training, weights of the BERT model are updated and classification layer on top to predict the personality labels for each input sentence.

6.2.4 Classification Module:

Once the model is trained, to predict the emotions for the test dataset, the pre-processed and tokenized test sentences through the fine-tuned BERT model and obtain the probabilities for each of the personality classes. These probabilities will be stored as an array for each input sentence.



```
test_message = """He's cutting it and then turn around and see the end result, but I'm glad he didn't do that
                because I probably would've lost my mind. As it was getting cut, I was just excited.
                I saw the snippets of hair falling to the floor and I was like, Yes!"""
tpredict(test_message)
```

```
Cagtegory_agreeableness: 0.6042853593826294
Cagtegory_conscientiousness: 0.520071268081665
Cagtegory_extraversion: 0.5933129191398621
Cagtegory_neuroticism: 0.42537373304367065
Cagtegory_openness: 0.3773482143878937
```

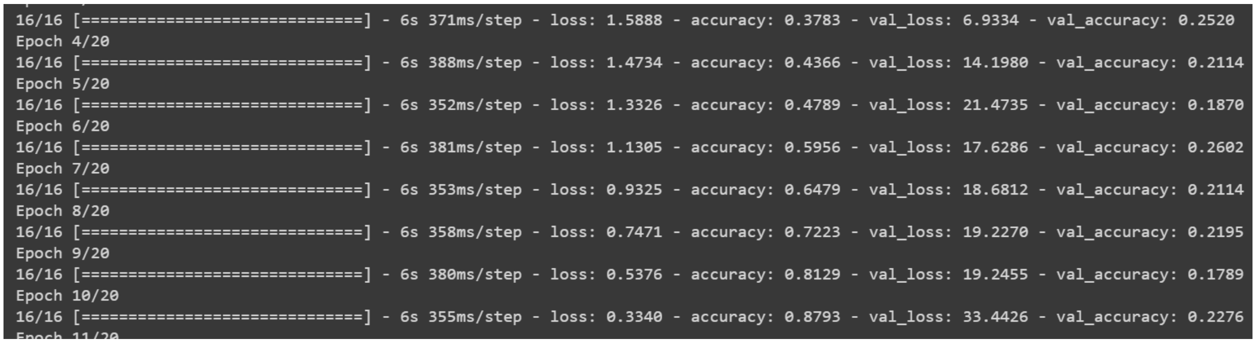
Figure 15: Sample Prediction

6.3 AUDIO MODEL

6.3.1 Data Pre-processing:

The training data is pre-processed by converting all audio files into its respective spectrum. From that, the amplitude for the frequency frame is extracted. This is considered to be pre-processed data. Now, feature extraction steps implemented are extracting MFCC features. This involves windowing the signal, applying the DFT (Direct Fourier Transform), taking the log of the magnitude, and then warping the frequencies on a Mel scale, followed by applying the inverse DCT (Direct Cosine Transform). This indicates that the data is pre-processed and converted into mathematical form so that it can feed into ML Model for training.

6.3.2 Model Building

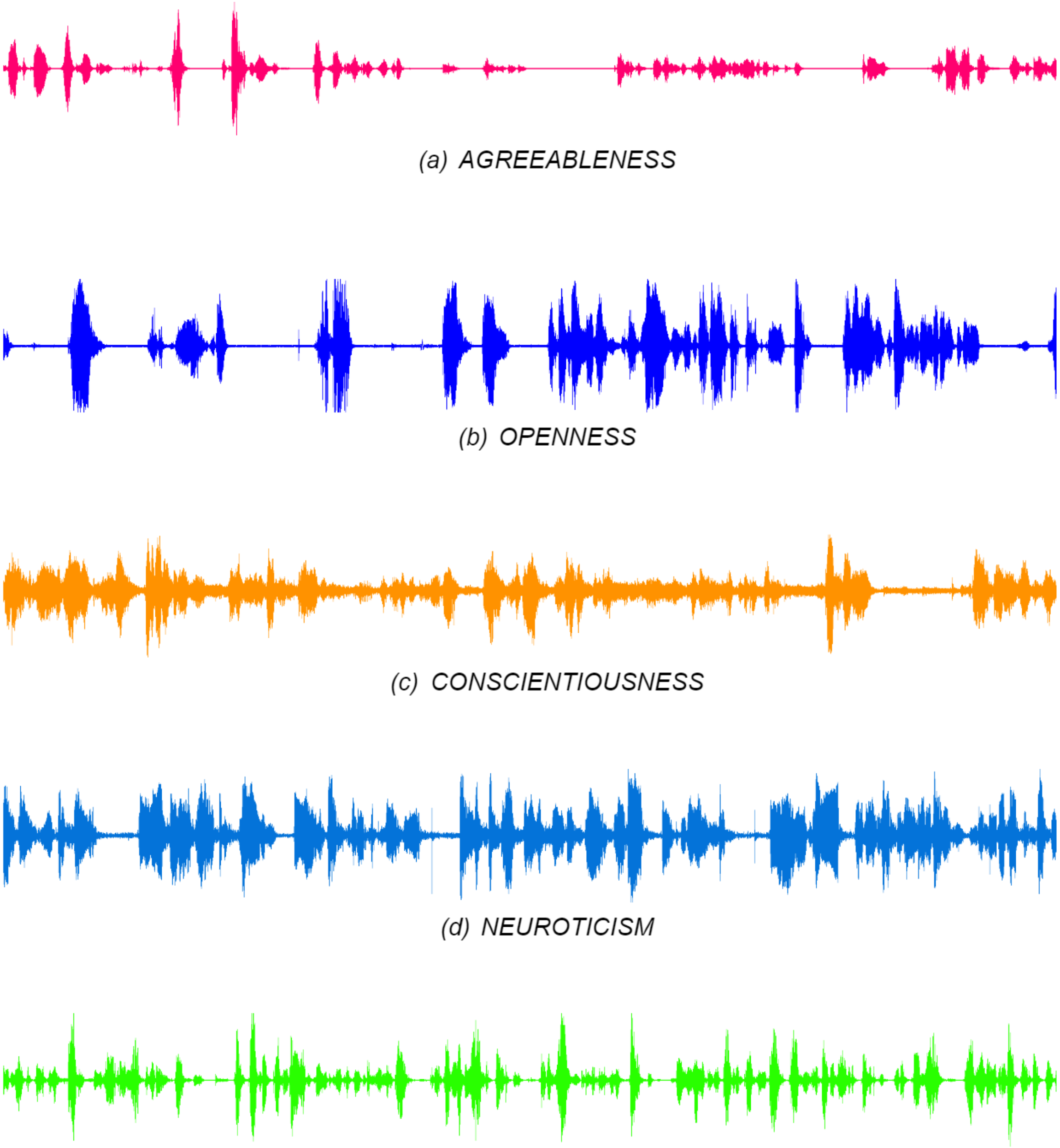


```
16/16 [==============================] - 6s 371ms/step - loss: 1.5888 - accuracy: 0.3783 - val_loss: 6.9334 - val_accuracy: 0.2520
Epoch 4/20
16/16 [==============================] - 6s 388ms/step - loss: 1.4734 - accuracy: 0.4366 - val_loss: 14.1980 - val_accuracy: 0.2114
Epoch 5/20
16/16 [==============================] - 6s 352ms/step - loss: 1.3326 - accuracy: 0.4789 - val_loss: 21.4735 - val_accuracy: 0.1870
Epoch 6/20
16/16 [==============================] - 6s 381ms/step - loss: 1.1305 - accuracy: 0.5956 - val_loss: 17.6286 - val_accuracy: 0.2602
Epoch 7/20
16/16 [==============================] - 6s 353ms/step - loss: 0.9325 - accuracy: 0.6479 - val_loss: 18.6812 - val_accuracy: 0.2114
Epoch 8/20
16/16 [==============================] - 6s 358ms/step - loss: 0.7471 - accuracy: 0.7223 - val_loss: 19.2270 - val_accuracy: 0.2195
Epoch 9/20
16/16 [==============================] - 6s 380ms/step - loss: 0.5376 - accuracy: 0.8129 - val_loss: 19.2455 - val_accuracy: 0.1789
Epoch 10/20
16/16 [==============================] - 6s 355ms/step - loss: 0.3340 - accuracy: 0.8793 - val_loss: 33.4426 - val_accuracy: 0.2276
Epoch 11/20
```

Figure 16: Model training

6.3.3 Spectrographs And Waveforms:

Waveforms:



(a) AGREEABLENESS

(b) OPENNESS

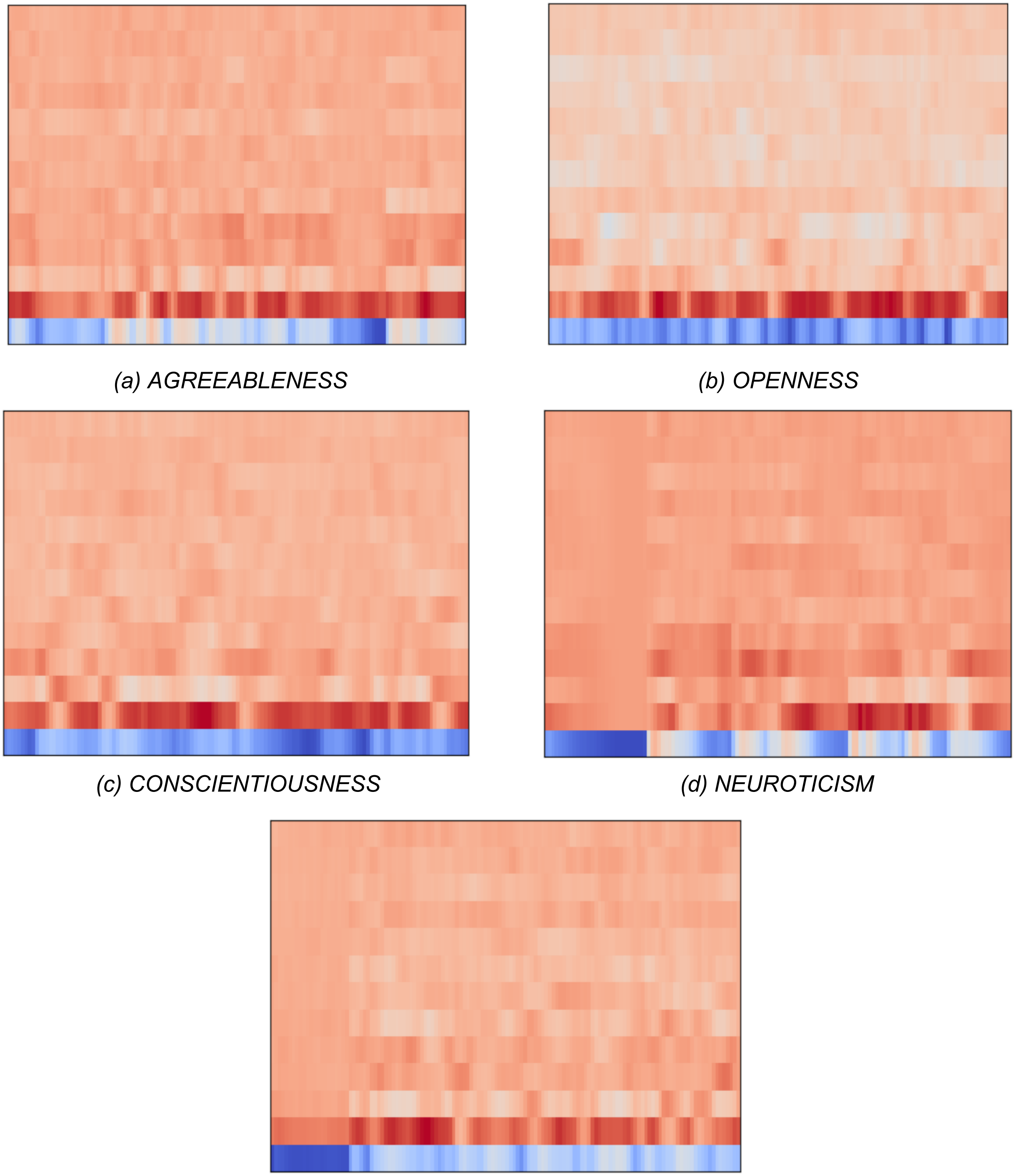
(c) CONSCIENTIOUSNESS

(d) NEUROTICISM

(e) EXTROVERSION

Figure 17: Waveforms of the Big 5

Spectrographs:



(a) AGREEABLENESS

(b) OPENNESS

(c) CONSCIENTIOUSNESS

(d) NEUROTICISM

(e) EXTROVERSION

Figure 18: Spectrographs of the Big 5

6.4 ACCURACY SCORES:

ACCURACY WITH TRAINING DATASET:

- Image - 0.80
- Audio - 0.72
- Text - 0.84

ACCURACY WITH TESTING DATASET:

- Image - 0.81
- Audio - 0.71
- Text - 0.78

6.5 COMBINER MODULE

The main objective of the combiner module is to acquire and integrate the outputs (emotion details) from each model trained. This is implemented by assigning a particular weight to each model's output based on their performance. The weights assigned to video, audio and text models are 0.5, 0.1 and 0.4 respectively.

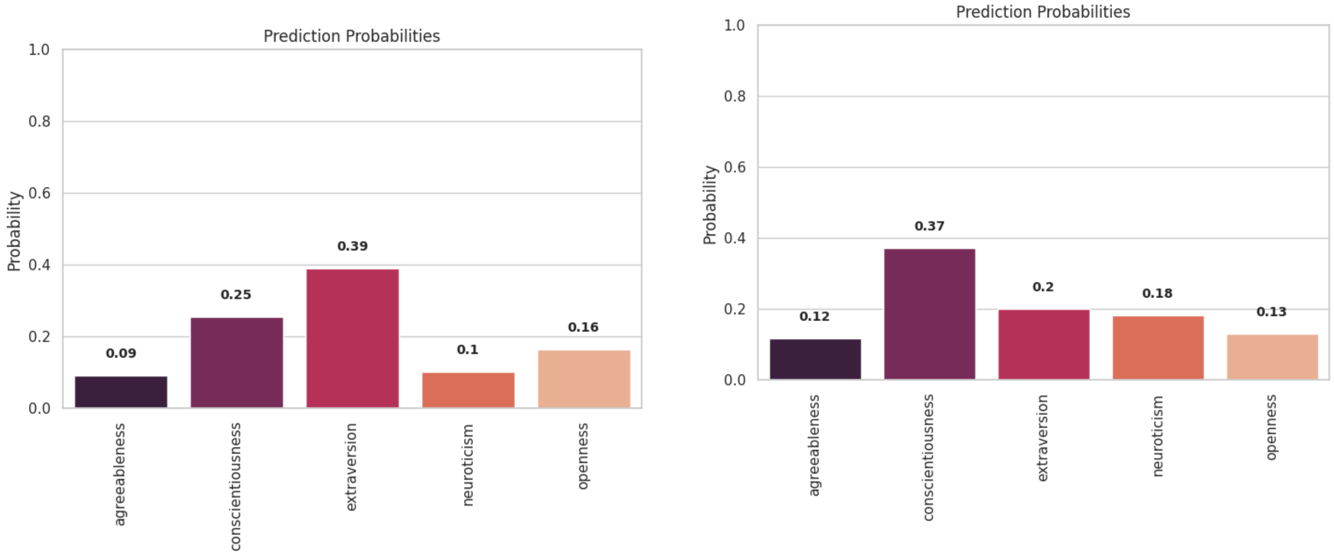


Figure 19: Final Result Personality Trait Analysis of 2 Videos

CHAPTER 7
CONCLUSION

Chapter 7 concludes the model built and provides an insight on how the model can be extended for future work.

EXTENDING THE PROJECT AND FUTURE WORK:

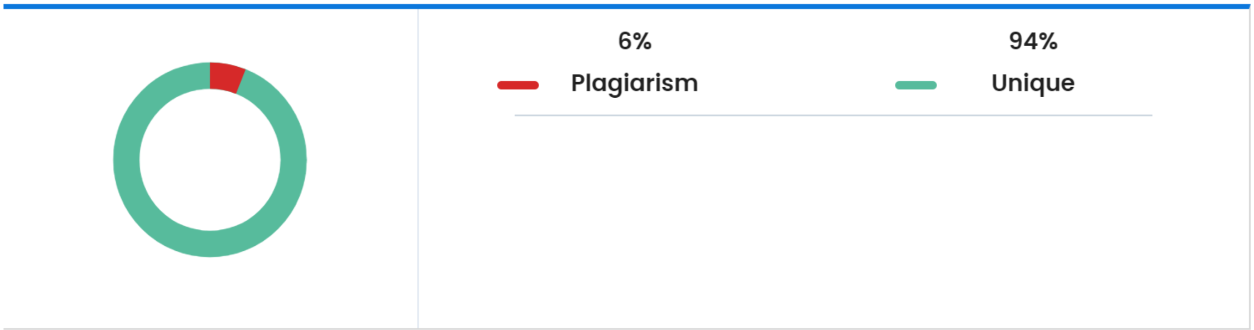
The project can be improved by expanding and diversifying the data to accurately predict personality traits.Future research in this field can focus on further improving the accuracy of the model, by integrating additional data sources, refining the algorithm, and exploring new approaches to multimodal analysis. Additionally, efforts can be made to evaluate the model's performance on a more diverse range of individuals, to ensure that it can be applied in a wide range of contexts.

BIBLIOGRAPHY

1. Zhao, Xiaoming, et al. “Deep Personality Trait Recognition: A Survey.” *Frontiers in Psychology*, vol. 13, Frontiers Media SA, May 2022. *Crossref*, https://doi.org/10.3389/fpsyg.2022.839619.
2. Suman, Chanchal, et al. “A Multi-modal Personality Prediction System.” Knowledge-Based Systems, vol. 236, Elsevier BV, Jan. 2022, p. 107715. Crossref, https://doi.org/10.1016/j.knosys.2021.107715.
3. Giritlioğlu, Dersu, et al. “Multimodal Analysis of Personality Traits on Videos of Self-presentation and Induced Behavior.” Journal on Multimodal User Interfaces, vol. 15, no. 4, Springer Science and Business Media LLC, Nov. 2020, pp. 337–58. Crossref, https://doi.org/10.1007/s12193-020-00347-7.
4. Abdu, Sarah A., et al. “Multimodal Video Sentiment Analysis Using Deep Learning Approaches, a Survey.” Information Fusion, vol. 76, Elsevier BV, Dec. 2021, pp. 204–26. Crossref, https://doi.org/10.1016/j.inffus.2021.06.003.
5. Alameda-Pineda, X., Staiano, J., Subramanian, R., Batrinca, L., Ricci, E., Lepri, B., et al. (2015). Salsa: a novel dataset for multimodal group behavior analysis. IEEE Trans. Pattern Anal. Mach. Intell. 38, 1707–1720. doi: 10.1109/TPAMI.2015.2496269
6. Junior, J. C. S. J., Güçlütürk, Y., Pérez, M., Güçlü, U., Andujar, C., Baró, X., et al. (2019). First impressions: a survey on vision-based apparent personality trait analysis. IEEE Trans. Affect. Comput. 13, 75–95. doi: 10.1109/TAFFC.2019.2930058
7. Zhao, Xiaoming, et al. “Integrating Audio and Visual Modalities for Multimodal Personality Trait Recognition via Hybrid Deep Learning.” Frontiers in Neuroscience, vol. 16, Frontiers Media SA, Jan. 2023. Crossref, doi.org/10.3389/fnins.2022.1107284.

PLAGIARISM REPORT

Duplichecker: www.duplichecker.com



6%	94%
🟥 Plagiarism	🟩 Unique