National University of Computer and Emerging Sciences

Intelligent Mock Interviewer

Omar Amir (14L-4015)
Shaoor Munir (14L-4223)
Zainab Iftikhar (14L-4304)

Supervisors:
Dr. Mehreen Saeed
Dr. Mirza Mubasher Baig

B.S. Computer Science
Final Year Project: May 2018

Department of Computer Science
FAST NU, Lahore, Pakistan

Anti-Plagiarism Declaration

This is to declare that the above publication produced under the:

title: _____

is the sole contribution of the author(s) and no part hereof has been reproduced on **as it is** basis (cut and paste) which can be considered as **Plagiarism.** All referenced parts have been used to argue the idea and have been cited properly. I/We will be responsible and liable for any consequence if violation of this declaration is determined.

Date: _____

Student 1

Name: Omar Aamir

Signature: _____

Student 2
Name: Shaoor Munir

Signature: _____

Student 3
Name: Zainab Iftikhar

Signature: _____

Table of Contents

List of Tables

List of Figures

Abstract

The personality of a candidate has a lot to say about the job position that will be best suited for him. The aim of this project is to develop an application which can evaluate the personality traits of a candidate during an interview and, based on that personality assessment, determine if they are the right choice for a job position. The project is research based in nature and will be developed using Python and MATLAB. To assess the personality of an interviewee, it will make use of visual, audio and semantic features of the transcript of a video interview. Analysis of these features will help generate an evaluation report which can be used by interested parties to decide whether the interviewee is a suitable candidate. After extensive research on the topic, the analysis of an interview video is divided into three parts: video analysis, audio analysis and semantic analysis of text. The video analysis is done by extracting frames from the video after a specific interval and getting an estimate through pre-trained machine learning regression models. Tests performed on different popular machine learning models show that a multi-layer perceptron is the best choice for analyzing the big five personality traits from a video. An aggregation of different multi-layer perceptron models trained on facial features like eyes and smile, audio features related to emotion is used to build a prototype which assesses the personality of a person through video recorded via the webcam. Textual part is used to reinforce and increase confidence in the personality values returned by combinations of audio and video.

Chapter 1: Introduction

The purpose of Intelligent Mock Interviewer is to develop a smart interviewing system which shall determine the personality of a person. This assessment will be based on the Big Five personality assessment scale [1], given by Lewis Goldberg in 1900. The dimensions through which the personality will be judged are named extraversion, agreeableness, openness to experience, conscientiousness, and neuroticism.

With the advancement of technology, the world is moving towards the automation of manual procedures. The term "interviewing process" refers to the job interviews, therapy sessions, surveillance interviews and so much more. Even though it is a large-scale process and demands automation, the interviewing process is still carried out manually. Our aim is to automate this procedure and facilitate employees in deciding whether the interviewee is suitable for the position they are applying for.

The saturation of equally skilled candidates in the industry leads to the demand for a prominent feature that can lead to the appointment of the right person – the candidate's personality. Our interviewer system shall evaluate the personality traits of an interviewee and determine if the person is right for the job they are applying.

1.1 Goals and Objectives

The goal of Intelligent Mock Interview is to judge the personality of an interviewee and determine if they are a suitable candidate. This goal can be further divided into the following objectives:

- **Speech to Text Conversion**
  The conversation with the interviewee will be converted into text for processing. The semantic meaning of the conversation will help in determining the personality traits of the interviewee.

- **Audio Analysis**
  Audio is the primary source of communication and its analysis is vital to our project. Acoustic features of the audio will be used for determining the emotional state of the interviewee.

- **Video Analysis**
  The facial expressions of an interviewee have a lot to say about the personality of that individual. The interview will be recorded and will be analyzed at runtime.

- **Job Recommendation**
  Different job positions demand different measures of a trait in the candidate. For example, a managerial position would require the candidate to be open, agreeable and in most cases, an extrovert. Our system shall check if those personality traits are present in the candidate before recommending them for the job.

1.2 Project Scope

Due to the vast paradigm of interviewing processes, the initial scope of the project is limited to keep things simple. The initial scope of the project is as follows:

- **Personality Analysis**
  The software will determine what traits are dominant in the personality of the interviewee by analyzing the acoustic and semantic parameters of the audio of the interviewee as well as their facial expressions.

- **Job Recommendation**
  Based on the interviewee's personality traits, an evaluation report will be generated which will help the interviewer in analyzing the feasibility of the candidate.

Throughout FYP-I, we have looked at many different software and projects working on automating the interview process. The project was divided into three main parts: video processing, audio processing, and textual analysis. We looked at different learning techniques to evaluate videos for FYP-I. We chose the best model which fit our data and then worked on optimizing it to work well on unseen interview footage. In FYP-II, we have worked on integrating audio and text with video to generate a more accurate score. Audio is utilized by using an open source tool to extract emotions from the interview file while text is used to confirm the score generated by audio and video through looking for words which are linked to high or lows of a trait. The prototype is working on that optimized model which helps in the estimation of Big Five traits.

Chapter 2: Literature survey

By a quick assessment of the previous research and software currently being used in the market, it is abundantly clear that not a lot of work has been done regarding automation of interview process using the Big Five model. Keeping this in view, we divided our research into two main parts; first part encompasses the software currently being used to automate parts of the interview process, while the second part focuses on research done on analysis of a person's personality and emotions using the visual and auditory features.

Before proceeding with the literature review, here are a few definitions of terms being used frequently in the document:

2.1 Definitions

Following are the definitions of the terms which will be used later in this literature review:

- **Big Five personality traits** [1]
  It is related to a psychological theory which uses some common language descriptors to define a human being's personality and characteristics. These common language descriptors form the five personality traits: openness to experience, extraversion, conscientiousness, agreeableness, and neuroticism.

- **Interview**
  Although in some research and references, the interview is taken as a general job interview, the term interview, for this project, can be taken to mean job interviews, therapy sessions, surveillance interviews or any form of communication in which judging a person's personality might be helpful.

- **Acoustic features** [2]
  Acoustic features are those features of an audio signal which can be experimentally recorded and observed. These phonetic features do not require a semantic understanding of the signal and depend on the characteristics of the sound being produced. Acoustic features may include the volume of the sound, pitch, frequency, and timbre.

- **Prosodic features** [3]
  Prosodic features deal with the aspects of speech with are not related to the phonemes. These features make use of the auditory quality of the sound being produced and are much more difficult to observe and analyze as compared to acoustic features. Pauses took during the speech, stress level on individual words and change of volume of specific sections of the speech.

Considering the scope of the project, we have divided the literature review into following three parts:

i. Automated interview systems currently in use
ii. Personality analysis using visual features
iii. Emotion detection using audio features

2.2 Automated Interview Systems Currently in Use

Automated interview system is a broad term which can mean one of the two things. They can be either management system aiding hiring managers to go through interviews or intelligence systems determining the feasibility of candidates.

i. **Automated Interview Management Systems**
Automated interview system can refer to a system which allows sorting the applications, automatically conducting an interview and recording the whole processes. This enables hiring managers to go through and look at those interviews later. In this case, there is little to no intelligence incorporated into the system. Formally stating it is a simple management system which keeps track of candidates and remembers the comments made by the managers while looking at those interviews.

ii. **Intelligent Interview Systems**
These systems, unlike the former, incorporate some sort of intelligence. They analyze the content of the interview and give a preliminary report on the feasibility of the candidate. Such systems make use of advanced analytic tools including machine learning algorithms which must be trained over a large dataset.

We looked at both kind of systems currently in use and inferred what areas they were or were not covering.

2.2.1 Automated Interview Management Systems

Most of the automated interview management systems that were studied were limited to storing the video interviews and making them available to access and watch through different portals. Some included a few intelligent features which could assist in making decisions for hiring managers, but they were not sufficing for replacing the manual interviewing process. Following is a brief description of the interview management systems investigated:

2.2.1.1A Portable and Intelligent Interview System [4]

This software was developed as a final year project by Tso Hei Lok, Cheng Man Fung Kevin, Fung Chin Pan and Lau Hiu Tsun in the year 2014-15 under the supervision of Dr. Cheng Reynold and Dr. Chui Chun Kit from the University of Hong Kong. The key features of this system are as follows:

- It integrates the video conferencing and recording function.
- It has offline support to overcome network failures.
- It incorporates intelligence for decision making with the aid of data-mining techniques and statistical data.
- It uses free-text-analysis on the comments by interviewers to give recommendations on whether the candidate should be selected or not.

2.2.1.2 Arya [5]

Arya is a commercial software by LeoForce, a recruiting AI company, which aims to automate the recruitment process by sorting through the applicants based on their CVs and selecting those candidates whose background and qualities closely match the job requirement. It does not offer any help in the actual interview process, instead, it helps to filter out the candidates who should be lined up for the interview.

2.2.1.3 InterviewStream [6]

InterviewStream is another commercial program that is dedicated to helping job seekers practice their interview skills. The development of the software started in 2016 and is still in process. It differentiates from other interview management systems by allowing hiring managers to record questions using their own voice and video input, which is then played to the candidates when they are applying for the job. The responses by the candidates are then cataloged and made available to hiring managers through the online portal or mobile applications.

2.2.1.4 Mya [7]

Mya is an interview management system which works in similar ways to Arya. The software was launched in 2015. In addition to going through the candidate resumes and sorting them based on their relevance to the job description, it also provides a communication channel to the applicants. Candidates, in any phase of the interview process, can query Mya about their progress and it will, using natural language processing, will communicate with them the details of their performance throughout the whole process.

2.2.1.5 Automated Telephonic Interview System (ATIS) [8]

ATIS, in development since 2015, is a replacement for the regular telephonic interviews. It supports recording predefined questions and automated calling of potential candidates. The candidates can then cycle through the questions through a key press on their phones or ATIS automatically determines when a candidate has finished answering a question based on the duration of the pause taken by the user while answering the questions.

The summary of the above-mentioned systems is provided below. The table outlines the development years, developing teams and the areas that the systems cover.

Name of System	**Years of Active Development**	**Developed by**	**Areas Covered**
A Portable and Intelligent Interview System	2014 – 2015	Hong Kong University Final Year Project	Video conferencing and recording function Offline support Data-mining techniques to make intelligent decisions Free-text-analysis on the comments by viewers to give recommendations
Arya	2014 – Present	Commercial Software	Automates the recruitment process by searching for candidates with specific background and qualifications.
InterviewStream	2016 – Present	Commercial Software	Allows customers to use a pre-recorded interview for all the candidates. Candidates can then take interview from any app or web browser; the application will record the interview and catalog it according to the job they have applied for. HR Managers can then watch the interviews and shortlist candidates.
Mya	2015 – Present	Commercial Software	Works as a complete recruiting assistant.

			Automates the recruiting process like Arya but it also engages directly with the candidates. Candidates can contact Mya through an app or SMS and it will respond to their questions regarding the interview process, their progress through the whole process.
Automated Telephonic Interview System (ATIS)	2015 – Present	Commercial Software	Allows telephonic interviews to be conducted without any human input from company's side. Pre-recorded questions Software will record answers to questions independently, judging the ending of one answer through a long pause or keypad press on the candidate's phone

Table 1: Automated Interview Management Systems Summary

2.2.2 Intelligent Interview Systems

In addition to the above-mentioned interview management systems, there are a few systems which focus on becoming a replacement for the whole interview process. They do so by analyzing the content of the interview through visual and/or audio features and predicting the suitability of the candidate for the job in question.

2.2.2.1 Matlda [9]

Matlda was developed as a research project in the Research Center for Computers and Communication and Social Invention (RECCSI) for nearly eight years starting from 2011. It is a humanoid robot which was designed to take interview of candidates and monitor their responses to the questions being asked.

Matlda was initially specialized to interview for sales positions and it included a corpus of 76 questions which focused on assessing the interviewee's skills and professional expertise. Its most impressive feature was to judge the facial expressions of the interviewee and analyze if they were nervous. In such cases, Matlda would try to relax the candidates by playing soothing music.

The preliminary report by Matlda includes the analysis of the candidate based on their attachment to the position they have applied for. It gave recommendations to help decide if the candidate was suitable for the position.

Currently, the direction of the project has been shifted and the robot is now working as a care robot for patients suffering from psychological issues, one such is Dementia [10]. It monitors their stress level and tries to soothe them when they are distress.

2.2.2.2 HireVue [11]

HireVue, a commercial interview software launched by HireVue, Inc, was initially started as a simple interview management system, but since last few years, it has added some intelligent features to help hiring managers in deciding the suitability of a candidate for the job.

HireVue analyzes the interview videos and monitors about 200 different facial features to generate a score to help decide if the interviewee is suitable for the given job. The actual method used in calculating the score is proprietary and is not disclosed for further research.

The intelligent interview systems are summarized in the following table with their years of active development, their developing teams and the areas that they cover.

Name of System	Years of Active Development	Developed by	Areas Covered
Matlda	2011 – Present	Research Center for Computers, Communication and Social Innovation (RECCSI)	Takes interview with candidates Specializes in interviewing candidates for sales positions Analyses the facial expressions of candidates for signs of nervousness Judges if the candidate is emotionally attached to the job Helps in recruitment
HireVue	2004 – Present	Commercial Software	Detects emotion Helpful in pattern recognition Determine the feasibility of interviewee

Table 2: Intelligent Interview Systems Summary

2.3 Personality Analysis using Visual Features

A big part of our automated interview process shall rely on analyzing the visual cues and judging the personality of the interviewee through those cues. Most interviewers take only a few seconds to make up their mind about a candidate. Based on this concept, a great amount of research has been done in the Speed Interviews Project by ChaLearn Looking at People.

2.3.1 ChaLearn Looking at People Speed Interview Project [12]

The ChaLearn Looking at People Speed Interview Project is aimed towards improving automated systems which can analyze and predict the personality of an interview candidate within a few seconds. The focus of this project is to analyze the candidates on the basis of the Big Five personality traits and use them to evaluate the suitability of a candidate for a particular job.

For personality analysis using visual features, we focused on a coopetition of this project which focused on analyzing a dataset of 10,000 fifteen second interview videos and predicting the Big Five traits of individuals in those videos.

2.3.2 2017 Looking at People CVPR/IJCNN Coopetition [13]

2017 Looking at People CVPR/IJCNN Coopetition is the latest coopetition to focus on personality assessment through visual features. The main goals of the coopetition were set to be:

- Analysis of Big Five personality traits using the First Impressions V2 [14] dataset
- Determination of an interview callback score using the Big Five personality trait values

2.3.3 Dataset used

The First Impressions V2 dataset used to train and test the visual analysis algorithms in the coopetition included 10,000 fifteen second videos. The key features of the dataset are:

- Each fifteen-second video includes only one individual
- The individual talks about any random topic in the video
- Each frame of the video contains more than half of the individual's facial features
- The training data was labeled by Amazon Mechanical Turks (AMTs)
- A principled process was adapted to make sure that the labels were reliable
- The videos include Big Five personality trait labels, transcriptions and an interview callback score which determines how likely is the candidate to get a callback for the position

2.3.4 Winning teams

The learning strategy, feature extraction techniques and the methods used by the top three teams in 2017 Looking at People CVPR/IJCNN Coopetition are briefly described below:

2.3.4.1 BU-NKU

The BU-NKU team made use of a kernel-based extreme learning method to train their system on both visual and acoustic features.

2.3.4.1.1 Learning Strategy

Learning strategy used by BU-NKU was:

- Kernel ELM (after min-max normalization) is used for visual features.
- Kernel ELM (feature z-normalization and instance level L2-Norm) is used for acoustic features.

2.3.4.1.2 Feature extraction technique

BU-NKU used four different types of features to analyze the videos. They are noted below:

- Local Gabor Binary Patterns were extracted from using the Local Gabor Binary Patterns from Three Orthogonal Planes (LGBP-TOP) technique [15]
- Facial features were extracted using VGG DCCN model [16] (Visual Geometry Group, Oxford University's Deep Convolutional Neural Network), trained on the Facial Expression Recognition 2013 (FER-2013) [17] Dataset
- Acoustic features from the audio were extracted using the open source openSMILE tool
- Pre-trained VGG-VD19 [18] (Very Deep Convolutional Networks for Large-Scale Image Recognition by Karen Simonyan and Andrew Zisserman) model from MatConvNet is used to detect scene features from the first frame of the video

2.3.4.1.3 Method

Following method was used to train the system:

- Instead of directly classifying the data set, they first use Viola & Jones Face detector [19] to isolate face from surroundings.
- The output from Viola & Jones Face detector was fed into IntraFace to detect facial landmarks.
- All four features were then extracted using the models discussed above.
- The features were fused using Kernel Extreme Learning Method
- Audio and scenic feature are fused together
- Facial features from LGBP-TOP and VGG face pre-training are fused together
- A random forest was used to predict the values of the Big Five traits and the interview callback method.
- The architecture can be written as RF (FF(LGBPTOP_face, DCNN_face), FF (IS13, DCNN_Scene)).

2.3.4.2 PML

PML used a combination of Support Vector Regression and Gaussian Progression Regression to estimate the Big Five traits and the interview callback value.

2.3.4.2.1 Learning Strategy

The learning strategy used by PML was:

- Support vector regression for Big Five personality traits
- Gaussian Progression Regression for Interview callback value

2.3.4.2.2 Feature Extraction Technique

PML detected the faces using an off-the-shelf face detector and then two different texture descriptors were used on resulting frames:

- Local Phase Quantization (LPQ) [20]
- Binarized Statistical Image Features (BSIF) [21]

The result of these descriptors was kept in a Multi-Level Pyramid (PML). Features were extracted from the Pyramids of both LPQ and BSIF and they were joined by concatenation in a single vector [22]. Each frame had its separate vector and the whole video's features were calculated by finding the means of the individual frame vectors.

2.3.4.2.3 Method

The PML features calculated from each frame of the video are fed into 5 different non-linear Support Vector Regressors (SVRs), one for each big-five trait, which then estimates the score for each video. These five scores are then used as an input for calculating the interview score. All five features are fed into a Gaussian Progression Regression which calculates the interview callback score for each video.

2.3.4.3 ROCHCI

ROCHCI used Gradient Boosting Regression to predict the Big Five Traits and the interview callback score.

2.3.4.3.1 Learning Strategy

Gradient Boosting Regression is used as a learning technique.

2.3.4.3.2 Feature Extraction Technique

Feature extraction technique used by ROCHCI was:

- Visual features were extracted using Facial Action [23] facial tracker and SHORE face analysis tool [24]
- Audio features were extracted using PRAAT [25]
- For textual analysis, they used some handpicked features which are not specified in the report

2.3.4.3.3 Method

Gradient Boosting Regression is used on all the extracted features and then a prediction is made for the 6 variables including 5 personality traits and the interview callback probability.

The following table recaps the learning strategies, feature extraction techniques, and methods used by the top three teams of Looking at People CVPR/IJCNN Coopetition:

Team	**Learning Strategies**	**Feature Extraction Techniques**	**Method**
BU-NKU	Kernel ELM (after min-max normalization) used for Visual features Kernel ELM (feature z-normalization and instance level L2-Norm) used for Acoustic features	Local Gabor Binary Patterns from Three Orthogonal Planes (LGBP-TOP) Facial features extracted using VGG DCCN model and trained on the Facial Expression Recognition 2013 (FER-2013) Dataset Acoustic features Pre-trained VGG-VD19 used to detect scene features from the first frame of video	Firs Viola & Jones face detector is used to isolate face from surroundings, then IntraFace is used to detect facial landmarks. The four different types of features are fused at feature level using Kernel ELM. • Audio and scenic feature are fused together • Facial features from LGBP-TOP and VGG face pre-training are fused together Random forest is then used to predict the values of the six target variables
PML	Support vector regression for 5 traits Gaussian Progression Regression for Interview callback value	Off-the-shelf face detector used for feature extraction Texture descriptors used on resulting frames: • Local Phase Quantization (LPQ)	PML features are fed into 5 different non-linear Support Vector Regressors (SVRs) for estimation These five scores are then used as an input for calculating the

		• Binarized Statistical Image Features (BSIF) The result of these descriptors was kept in a Multi-Level Pyramid Features from each frame was joined by concatenation in a single vector Whole video's features were calculated by finding the means of the individual frame vectors	interview score by using Gaussian Progression Regression
ROCHCI	Gradient Boosting Regression used as a learning strategy	Visual features were extracted using Facial Action facial tracker and SHORE face analysis tool Audio features were extracted using PRAAT For textual analysis, they used some handpicked features which are not specified	Gradient Boosting Regression was used on all the extracted features and then a prediction was made for the 6 variables including 5 personality traits and the interview callback probability

Table 3: Personality Analysis using Visual Features Review Summary

2.4 Emotion detection using audio features

Emotion detection from speech signals is becoming an intensely researched field. Though gestures and facial expressions play a primary role in determining the emotional state of a person, a major contributing factor for this assessment has emerged in recent years, namely acoustic features. Emotions can help our system in determining the personality of our candidates, especially along the neuroticism dimension. We went through some different papers which focused on detecting emotions through audio features. Some interesting findings from the papers are given below.

2.4.1 Features used for emotion detection using speech signals

Speech signals, with its acoustic and prosodic features, is playing a vital role in determining the emotional state of the speaker. The following two approaches were used for this purpose:

i. Emotion Detection using acoustic features

Acoustic features of speech, along with differentiating between different individuals, are now being used to determine the emotional state of a person. These features determine the wave properties of a signal. These features were used in [26] [27] [28].

The basic idea is to analyze acoustic differences that occur when we say the same thing differently. Each experiment used a different set of classifiers. The dataset in all the experiments comprised of professional actors speaking an utterance (dates or numbers) under

different emotional states. One experiment [28] concluded that subject based dataset is more efficient and accurate. Another study [27] first asked experts to label utterances. Humans achieved an accuracy of 69%. The study then used a recurrent neural network which could classify emotions with an overall accuracy of 59%. This determined that emotion is a highly subjective notion, and not only does it depend on the emotional state of a speaker, it also depends on the emotional state of the receiver. This experiment also found out that emotion exists in only some parts of an utterance.

One study [26] divided the emotions in six opposing pairs (happy and sad, interest and boredom, shame and pride, hot anger and elation, cold anger and sadness, despair and elation) There were three data sets, one from all speakers, one from only male speakers, and one from only female speakers. The input to the classifiers were different vectors of acoustic features. The results of the experiment showed that highest recognition accuracy obtained from all speakers' dataset was within 68-83%, from male-only speakers was 90-96%, and from female only speakers was within 69-96%. The study concluded that separating out male and female speakers play a big role in accurate classification. Also, using SVM and male-only speakers' dataset has resulted in the highest accuracy of detecting emotions so far.

ii. Emotion Detection using prosodic features and acoustic features

Though prosodic features improve the effectiveness of emotion detection, they are not widely used as they are subjective and depend primarily on the speaker. A person may be loud without being angry and another person may speak softly even when angered. The role of the prosodic features is debatable, and they have been used only in one experiment that we studied. [3] In this experiment, the prosodic features were combined with acoustic features. First, acoustic features were analyzed. They were implemented through classical hidden Markov models. The overall accuracy obtained was 65.4%. Next prosodic features were combined with the acoustic features. Prosodic features were implemented using suprasegmental states. These states were used for determining the intensity and pitch over a syllable or a word which are subject to vary. They cannot be implemented using hidden Markov model that requires a fixed time frame for observing the speech signal. The accuracy obtained this time was 72.25%, an absolute 7% improvement. This study concluded that prosodic features do improve the effectiveness of emotion detection using speech analysis.

The following table overviews the team's name, paper, dataset, features examined, and classifiers used:

Team	**Paper**	**Date Set**	**Features**	**Classifiers**
Raunaq Shah Michelle Hewlett Machine Learning Project, 2007, Stanford University	Emotion Detection from Speech	Linguistic Data Consortium [29]	Pitch Mel Frequency Cepstral Coefficients Formants	K-Means Support Vector Machines
Vladimir Chernykh Grigoriy Sterling	Emotion Recognition from Speech with	The Interactive Emotional Dyadic Motion Capture [30]	Acoustic features, 34 features including 12 MFCC	Recurrent Neural Networks

Pavel Prihodko	Recurrent Neural Networks			
Thomas S Polizin Alex H Waibel	Detecting Emotions in Speech	Corpus contained questions, statements, and orders spoken by 5 drama students according to the label provided (happy, sad, angry, afraid, and neutral)	Acoustic and Prosodic (pitch and intensity) features	Hidden Markov Models
Assel Davletcharova Sherin Sugathan Bibia Abraham Alex Pappachan James	Detection and Analysis of Emotion from Speech Signal	30 subjects aged 20-45 with equal proportion of men and women	Mel frequency cepstral coefficients Heart Rate Emotional State of the Person	Naïve Bayes Lazy IB1 RBF Network Logistic Ada Boost M1 Bagging Random Tree

Table 4: Emotion detection using audio features

The facial expressions and the way that a person looks during the interview are key features in assessing their personality. In our literature review, we have looked at some interview systems which are already being utilized in the industry. Although most of those systems restrict themselves to perform as an interview cataloging and managerial system, a few include some form of intelligence to give recommendations about a candidate. We have also looked at work that has been done so far on analyzing the personality of a person through visual features. A very good reference point for that is the CVPR/IJCNN Coopetition which focuses on determining the Big Five personality traits of a person through a fifteen-second video clip of them speaking about themselves. To analyze the effect of how a person speaks during an interview to their personality, we looked at multiple papers which focused on the analysis of emotion through acoustic and prosodic features. Those papers help conclude that training systems using both male and female speakers resulted in less accuracy as compared to systems which were trained separately for each gender. It is also apparent from the results that using prosodic features improves the overall accuracy of the emotion detection system.

Chapter 3: Requirements and design

The purpose of this section is to identify all the high-level requirements for the intended system i.e. Intelligent Mock Interviewer. Adding to the functional requirements, this section also highlights nonfunctional requirements and constraints of the system.

Following are the high-level requirements for Intelligent Mock Interviewer:

3.1 Functional Requirements

Functional requirements of the system include:

- **Registration**
  FR1. The system shall allow the interviewee to register.

- **Login**
  FR2. The system shall allow interviewees and hiring managers to log in.

- **Logout**
  FR3. The system shall allow interviewees and hiring managers to log out.

- **Maintain Profile**
  FR4. The system should allow the interviewee to change his password.
  FR5. The system should allow the interviewee to change his profile picture.
  FR6. The system should allow the interviewee to change his email address.

- **Take Interview**
  FR7. The system shall allow the interviewee to take an interview.
  FR8. The system shall record the video of the interview.
  FR9. The system shall record the audio of the interview.

- **Determine Personality**
  FR10. The system shall determine the five personality scores of an individual after the interview.
  FR11. The system shall evaluate the job score of an individual.

- **Show Results**
  FR12. The system shall display the personality assessments and job feasibility of the interviewee.

- **View Candidates Feasibility**
  FR13. The system shall display to the hiring manager the job feasibility scores of all the interviewees who underwent through the interview process of a particular job.
  FR14. The system shall allow hiring managers to view a candidate’s personality assessed values.
  FR15. The system shall allow hiring managers to view a candidate’s interview.

- **Manage Interview Statistics**
  FR16. The system shall save the interviewee’s interview statistics to the database.

FR17. The system shall let the interviewee see his old interview statistics.

3.2 Non-Functional Requirements

The following section describes all the non-functional requirements of the system.

- **Usability**
  U1. The system shall require less than a half hour training time.
  U2. The system shall be platform independent.

- **Performance**
  As the system has not been developed, the accuracy provided below would be our best effort to achieve but is not guaranteed.
  P1. The system shall assess the personality traits up to 70% accuracy.

- **Security**
  S1. The system shall be protected against SQL injections.
  S2. The system shall be protected against Cross Site Scripting.
  S3. The system shall log out all users after a period of inactivity.

- **Compatibility**
  C1. The system shall be compatible with devices that support web access.
  C2. The user interface for the system shall be compatible with Google Chrome, Mozilla Firefox, Microsoft Edge and Internet Explorer.

- **Reusability**
  R1. The system shall be reused in other personality identification system.

3.3 Assumptions

The following assumptions are considered for the system's specification:

- A user's history could not exceed more than 20 because of memory constraints
- Intended users are computer literate
- Intended users know how to communicate in English (limitation imposed because the semantic analysis of interview transcript will be done keeping English language rules in view)

It is also assumed that users have proper access to the basic software and hardware requirements described in section 3.8.

3.4 Use case Overview

Following are the identified use cases of the system:

Use Case Identifier	Initiating Actor	Use Case Title	Description
UC1	Interviewee	Register	The initiating actor creates an account.
UC2		Email Verification	The system sends an email to a newly registered user to verify his account.

UC3		Fill Form	A user of the system fills forms associated with signing up.
UC4	Interview Hiring Managers	Login	A user of the system logs in to the system.
UC5		Authenticate	A user is authenticated based on his roles.
UC6	Interviewee	Maintain Profile	A user maintains or edits his profile for his account.
UC7		Change Password	A user changes his password.
UC8		Change Profile Picture	A user changes his profile picture.
UC9	Interviewee	Take Interview	The initiating actor undergoes the interview process.
UC10		View Interview Results	The initiating actor sees his interview assessment scores i.e. the five personality traits.
UC11	Interviewee	View Previous Interview Statistics	The initiating actor views the previous results of his interviews.
UC12	Interviewee Hiring Managers	Logout	A user signs out of the system.
UC13	Hiring Managers	View Interviewees Feasibility	The initiating actor views the job score of all the candidates' who has given the interview for a specific job.
UC14	Hiring Managers	View Interviewees Personality Scores	The initiating actor views the personality scores of all the candidates'.
UC15	Hiring Managers	View Interviewees Interview	The initiating actor views the interview video of a candidate.
UC16	Hiring Managers	View Interviewees Information	The initiating actor views the profile of a given candidate.

Table 5: Overview of the Use Cases of Intelligent Mock Interview

3.5 Use Case Diagram for Interviewee

The interviewee is the primary actor for the given use cases. The figure shown below extend and include relationships of the use cases. The use cases Logout, Maintain Profile, Take Interview and View Previous Interview Scores uses the use case Login, but the association has been left out for the sake of simplicity.

The interviewee registers for an account if he does not have one. The account is verified when he uses the verification link. Once verified, the interviewee can log in and take interview. He can maintain his account, and see his previous interview statistics, too.

The following is the use case diagram for the actor interviewee:

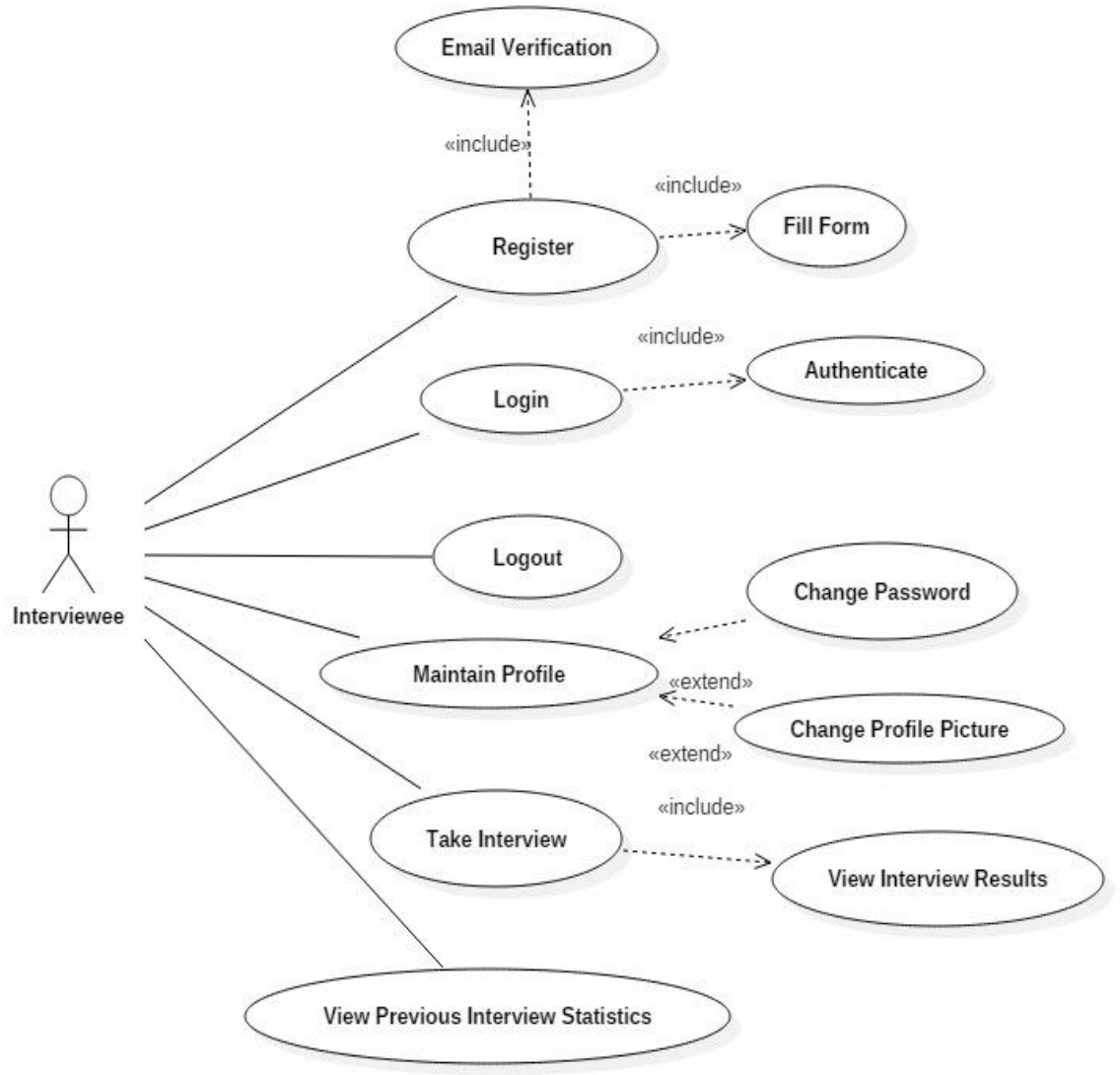

Figure 1: Use Case Diagram for Interviewee

3.6 Use Case Diagram for Hiring Manager

The hiring manager is the primary actor in this scenario. The figure shown below extends and include relationships of the use cases. All the other use cases use the Login use case for successful operation.

The hiring manager logs into the system with a company's credentials. He can see the feasibility scores of all the interviewees that have taken the interview for a job. The ability to open a job position and set the job options have been left out of the system for now. The primary focus is on the evaluation of personality scores of a candidate. Manager can then look the personality scores and the video of an individual candidate.

The following is the use case diagram for the actor hiring manager:

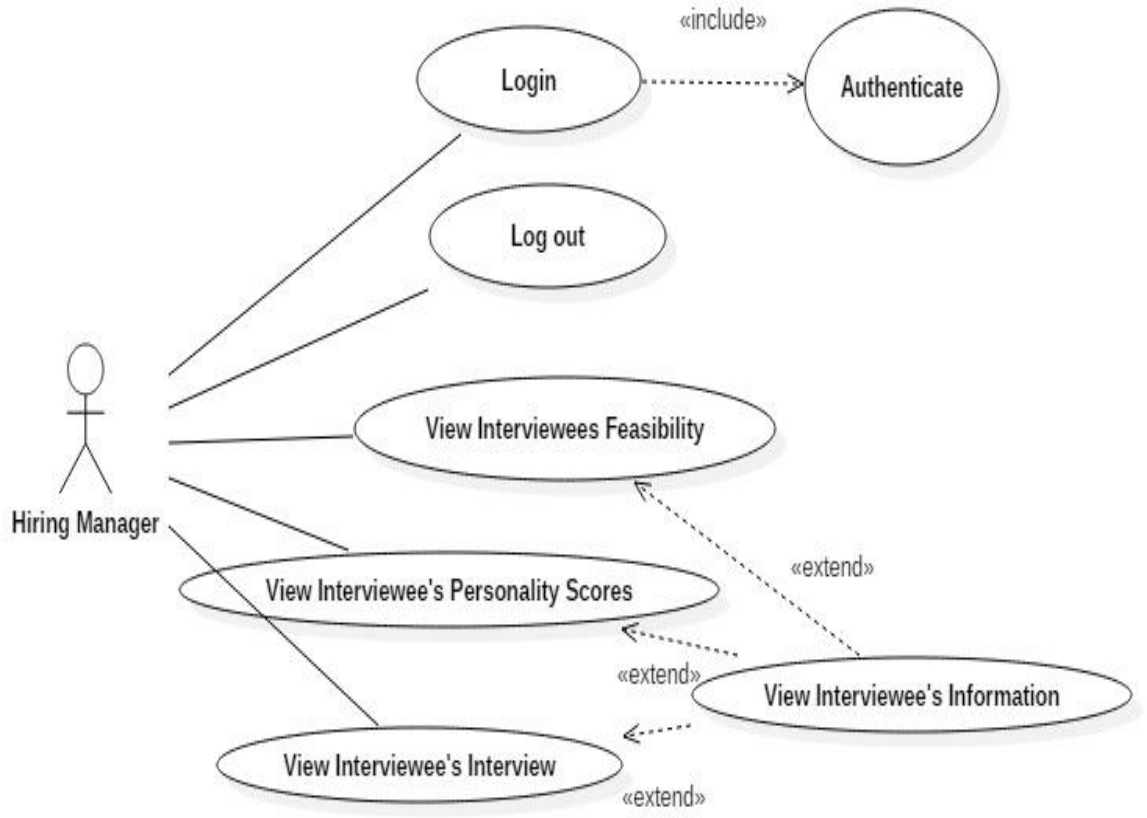

Figure 2: Use Case Diagram for Hiring Manager

3.7 System Architecture

The system has been divided into the following parts:

- Corpus Preprocessing
- Model Trainer
- Video Processor
- Frames Processor
- Audio Processor
- Text Processor
- Personality Detector

The corpus processing will generate persistent trained models which can be used by video, audio and text processor while generating the personality score from an interview video.

Following diagram shows an overview of the system architecture:

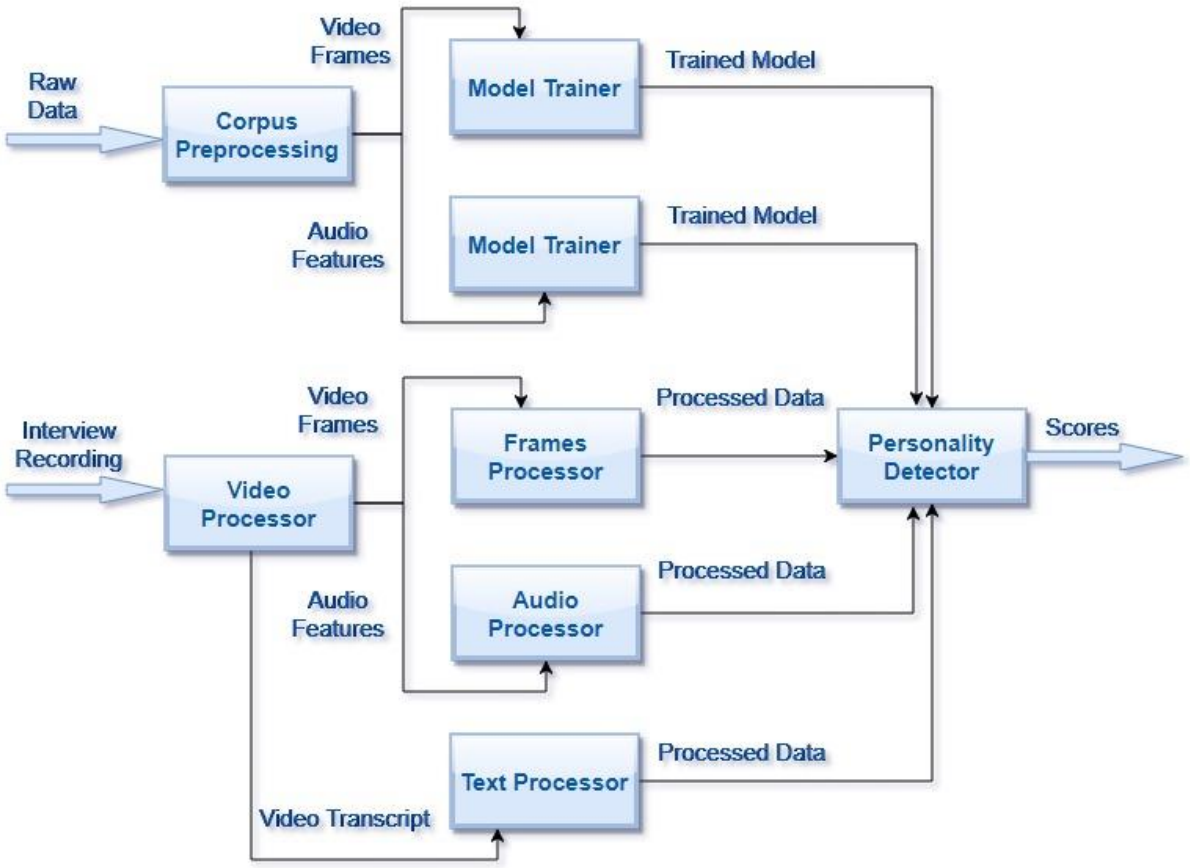

Figure 3: System Architecture for Intelligent Mock Interviewer

3.8 Hardware/Software Requirements

The system shall be needing the following hardware and software requirements for successful operation:

3.8.1 Hardware Requirements

We need the following hardware:

- **Laptop**
  The application is web-based; hence a laptop is necessary to use the application. Generation of minimum requirements is not possible at this point due to lack of integration between modules, but the system is currently being run on 8 GB RAM and Intel Core i7 processor.

- **Camera**
  A camera is needed to record the video of the interview.

- **Microphone**
  A microphone will be required for recording the audio.

3.8.2 Software Requirements

We need the following software for a successful application:

- **PyCharm**
  We will be developing the software using Python. The IDE we shall be using is PyCharm.

- **Scikit-learn**
  Scikit-learn is used to implement commonly used machine learning algorithms.

- **Viola Jones**
  Viola Jones is required to extract features from the video.

- **MySQL**
  To store the interview and the data of an interviewee, a database is required. We will be using MySQL for managing the database.

There are two main actors in our system: interviewee and hiring manager. The interviewee is restricted to viewing only his own interview recordings and can access his previously analyzed scores. The hiring manager can view a list of all the interviews recorded through the system and can easily check which candidates were most feasible for the job posted. The backend of the program will mainly be developed using Python while the front end would require a browser to work. Due to the focus being on analyzing the video and audio of the interviewee, a webcam-enabled laptop and a microphone are also added to the requirements.

Chapter 4: Description of prototype

The project is mainly divided into three main modules; video analyzer, audio analyzer and text analyzer. A brief description of how this prototype works for each module is given below:

4.1 Dataset collection

To train models for the video analyzer, we have used the First Impressions V2 dataset provided by ChaLearn [14]. The dataset includes 10,000 videos with labels for all Big Five traits. The labels were generated using the AMTs (Amazon Mechanical Turks) and the process was supervised by experts to ascertain the validity of the process. Following is a brief description of the dataset:

- Dataset contains 10,000 videos, where each video is exactly fifteen-second long
- The videos are then divided into 6,000 training videos, 2,000 validation videos and 2,000 test videos
- All videos have only one person in the frame, with at least half of their facial features visible in every frame
- The size of a video is 1280 by 720 pixels
- The labels and transcription for each video are provided in separate pickle files in the form of python dictionaries

The training videos are used to generate a machine learning model which will be used to predict the Big Five traits of an interviewee using the interview video.

4.2 Video Analysis

4.2.1 Model training

The dataset described above is used to train different machine learning models which are used to predict each of the Big Five personality traits. The training process is divided into following steps:

i. Frame extraction
ii. Feature extraction
iii. Model training

4.2.1.1 Frame extraction

The first step in training the machine learning models is to extract individual frames from the videos provided in the dataset. A single, fifteen-second video recorded at 30 frames per second has 450 unique frames. So, to train 6,000 fifteen second videos, a total of 2,700,000 frames would have to be processed. But according to a research conducted by Philippe G. Schyns, Lucy S. Petro, and Marie L. Smith in 2009 [31], it takes the human brain 200 milliseconds to register a change in facial expression. Keeping that in view, we extracted a frame from the video after every 200 milliseconds, thus allowing for a complete change in facial expressions.

By using an interval of 200 milliseconds, the number of extracted frames from a single video is reduced to 75, with the number of frames extracted from all 6,000 videos coming down to 450,000. This results in a big improvement in the total time required to train a single model.

4.2.1.2 Feature extraction

The next step was to decide what features we wanted to extract from those frames. The selection of these features depends on their direct correlation with how much influence they can have on judging the personality of a person. For this prototype, we have focused on following features:

- Complete facial profile, which should include how much of the face is visible in the frame along with the current position of all facial features with reference to each other
- Both eyes (extracted separately to account for frames where only right or left eye might be visible)
- Smile (if applicable in the frame)

So, each extracted frame is then divided into four sub-images: face, left eye, right eye, and smile. In an ideal case where all four features are present in the extracted frames, we get a total of 1,800,000 different training images.

However, the image of a face, left eye, right eye or a smile cannot be directly used as a feature. The visual noise, along with the large number of different RBG (Red, Blue, and Green) pixels present in the image hampers the amount of useful data which can be obtained through an image. To process the image further and extract useful information from it, we first convert it into an LBP (Local Binary Pattern) [32] [33].

4.2.1.2.1 Conversion to LBP and generating feature histogram for an image

A Local Binary Pattern from an image is generated by first converting the image to grayscale so that each pixel has a single value instead of the three RBG values. Each pixel's value is then compared to its neighbor pixels' values (most commonly to eight nearest neighbors in a radius of 1 pixel). If the neighbor has a pixel value higher than the current pixel, it is given a weight of 1, otherwise, it is given a weight equal to 0. By comparing a pixel with its eight neighbors, an 8-bit number is generated which is stored as the new value of that pixel.

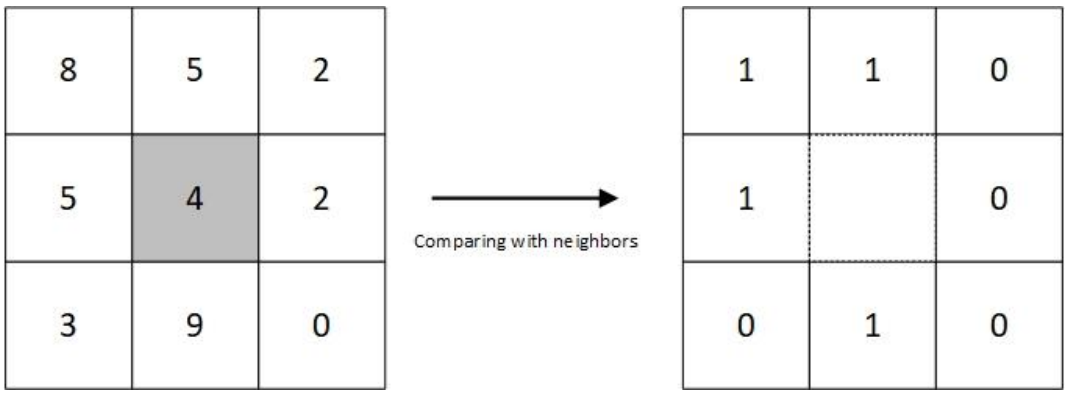

Figure 4: Generating LBP for a pixel

After repeating this process for all the pixels, we get an LBP representation of the image in which each pixel has a value between 0 and 255. The number of pixels in the whole image is still the same and thus it might seem impractical to use it as a feature for machine learning. To remedy that, we calculate the number of pixels that have a value from 0 to 255; this count is then used to generate a histogram in which the x-axis represents the values from 0 to 255 and the height y represents the number of pixels that have that value. So, for example, the entry (25, 6) will represent that 6 pixels in the image have a value of 25. A flow chart showing this whole process is given below:

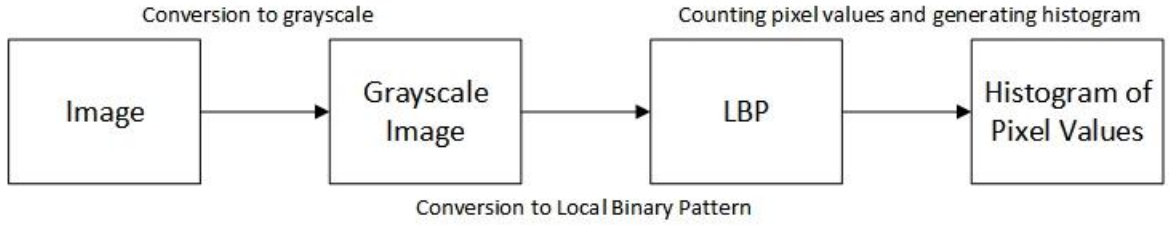

Figure 5: Generating a feature histogram from an image

4.2.1.3 Model training

Using the method described in previous sections, a feature histogram is generated for each image of a face, left eye, right eye and smile found in a frame. That histogram, along with the Big Five trait value for the video is passed along to the respective model for the feature and the Big Five trait. Each trait has four different models (one for each feature) which need to be trained for every frame. So, a total of 20 different models are trained for this prototype.

A flowchart describing the whole training process is shown below:

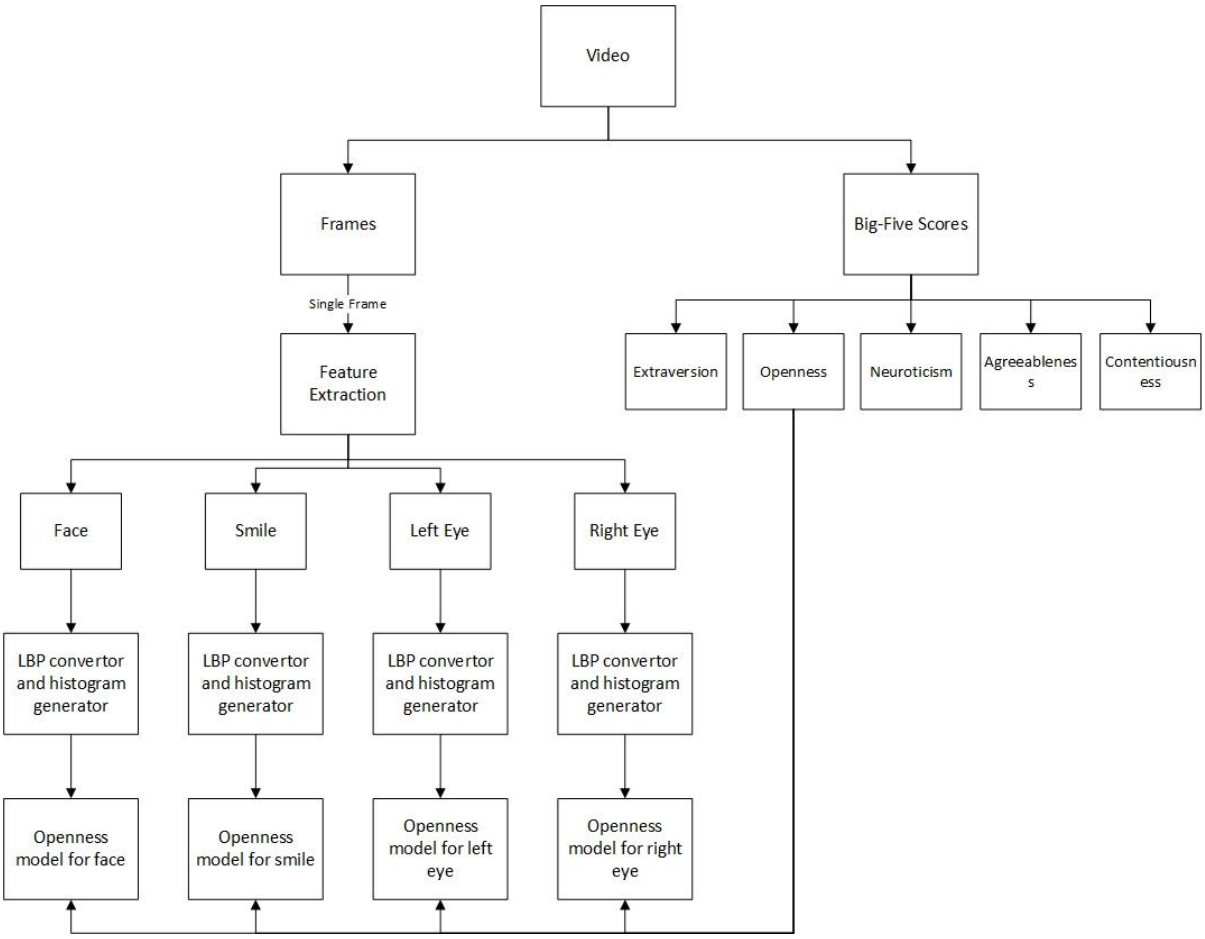

Figure 6: Openness model training process

4.2.2 Predicting trait values for a video

The prototype currently uses the laptop webcam to record a video and then analyze it based on individual frames. Like the training process, a frame is extracted after 200 milliseconds and all four features (face, left eye, right eye, and smile) are extracted from it. A feature histogram is then generated from the four feature images and a prediction is obtained using the pre-trained models. The final score for each trait is obtained by aggregating the scores obtained through individual features.

A flowchart describing the prediction process for openness trait is shown below:

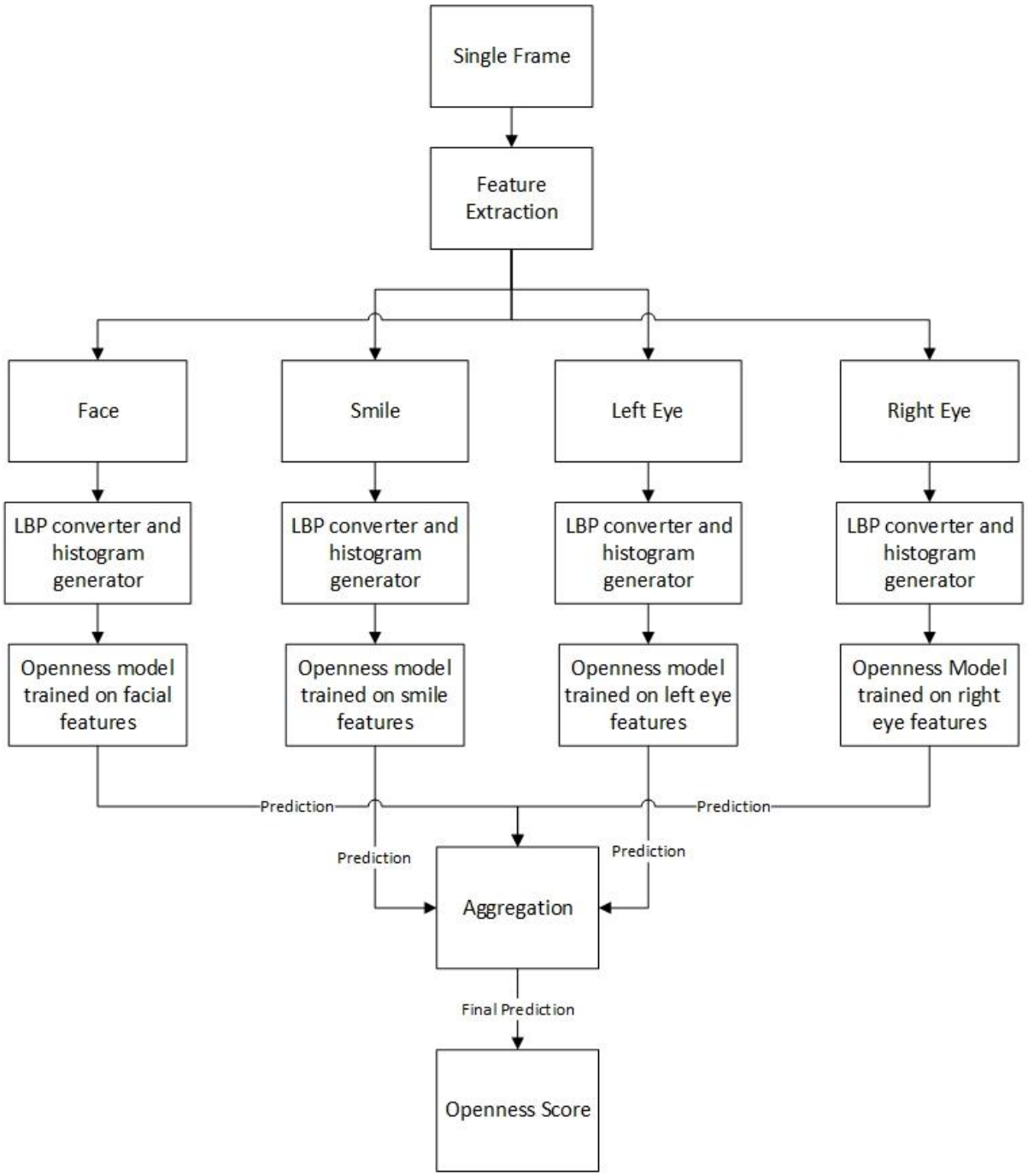

Figure 7: Prediction process for openness trait

Using the above training and prediction technique, our prototype can record a live video using a webcam and predict the values of all the Big Five traits by extracting individual frames after every 200 milliseconds (performance and test scores are discussed in the next chapter). The total score for the video is then calculated as an aggregation of the individual scores of each video. So, in this way, we can give an early estimate of a person's personality through analysis of their video interview.

4.3 Audio Analysis

How confident an interviewee is in his interview, how well he answers the questions, how good are his speaking skills, all these traits depict the personality and credibility of the interviewee. The first impression of an interviewee on the interviewer is how confidently and boldly he gives an introduction about himself. The first two minutes of an interview can make or break the remaining interview. If the person has a solid start and speaks without hesitation then the remaining interview is likely to go smooth as well, however if a person starts to stammer or his voice becomes shaky it would depict that the person is shy, introvert, unconfident etc. Thus, the sound features also have a decisive role in determining the personality traits of the interviewee.

4.3.1 Audio features

The audio features are mainly divided into two types:

- Acoustic features
- Prosodic features

4.3.1.1 Acoustic Features

The acoustic features determine the wave properties of a signal. These features are quantitative and can be measured. Some examples of acoustic features are pitch, frequency, amplitude. These features are focused upon during the model training.

4.3.1.2 Prosodic Features

Prosodic features are those features of an audio signal which cannot be measured directly rather they are inferred. Features such as loudness are prosodic features. The role of these features in determining the personality of a person is questionable since a person can be inherently loud, that doesn't mean that he is angry or a person can speak softly when he is angry. Hence these features were combined with acoustic features to increase the performance of the model.

4.3.2 Model Training

The audio from the fifteen second videos from the Looking at people challenge was used in developing our model. Thus, the audio analysis consisted of two major parts:

- Audio Extraction
- Audio Analysis

4.3.2.1 Audio Extraction:

The audio was extracted from the videos using an online tool FFmpeg. This tool took a video in .mp4 format as input and returned the .avi audio file which is used for further processing.

The audio was processed using an open source library OpenVokaturi. OpenVokaturi is an emotion recognition software that can recognize the emotions in a voice just as well an any human would. The creators of this library claim to have an accuracy of up to 65-70 percent.

4.3.2.1.1 Emotional Features from Audio

The OpenVokaturi processes the audio signal and returns five different emotional values:

- Neutrality
- Happiness
- Sadness
- Fear

- Anger

These emotional values are closely linked to the Big Five Personality Traits (Openness, Extraversion, Conscientiousness, Neuroticism, Agreeableness). These five emotional values are used as features for training of each of the big five personality traits.

Using these values, we can train a classifier to predict the personality of an interviewee through his sound.

4.3.2.1.2 Training Process

The following figure describes the model training procedure. As described above the audio file is extracted using the FFmpeg tool and features are extracted from this audio file using OpenVokaturi.

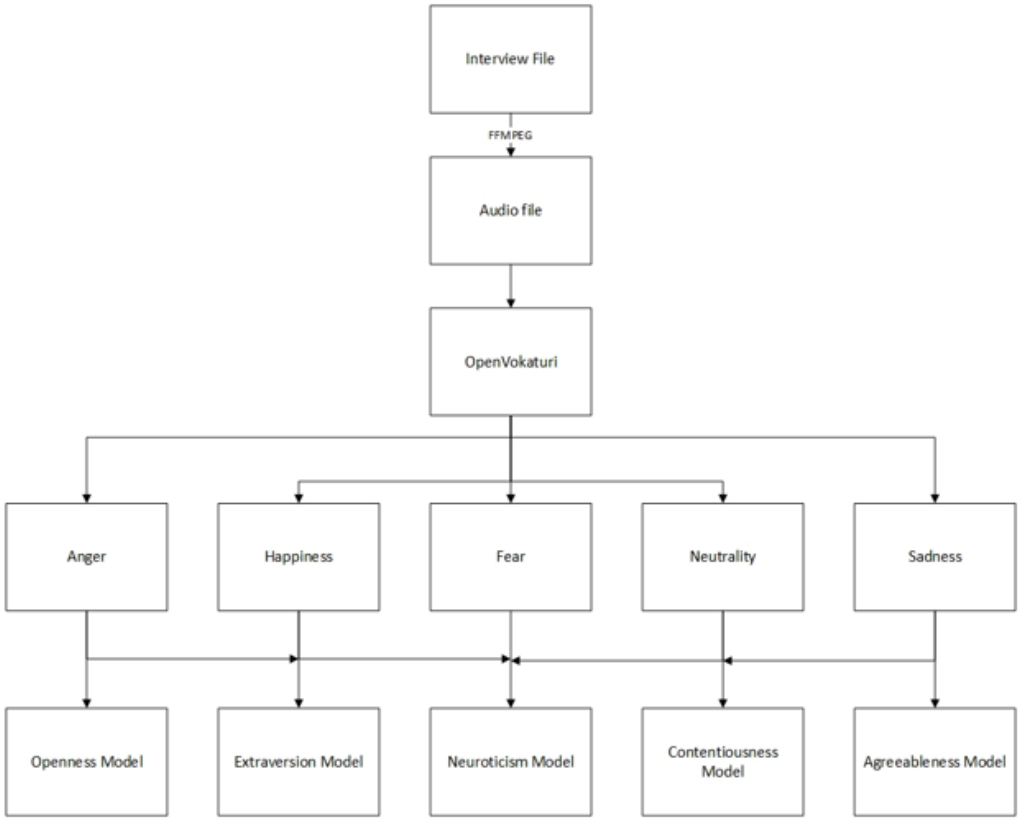

Figure 8 Training process using emotional values from OpenVokaturi

The five audio features received from OpenVokaturi are trained using a Neural Network. A model is being generated for each trait. For the training of each model all the five emotional values returned by OpenVokaturi are used. The neural network used for training contains 4 layers and there are 190 neurons in each layer. These values have been optimized using the validation set. As a result, a model is generated for each trait.

4.3.3 Model Prediction:

Once trained the models are used for predicting the personality traits using an input audio file. The figure below describes the steps involved in making a prediction for an audio file.

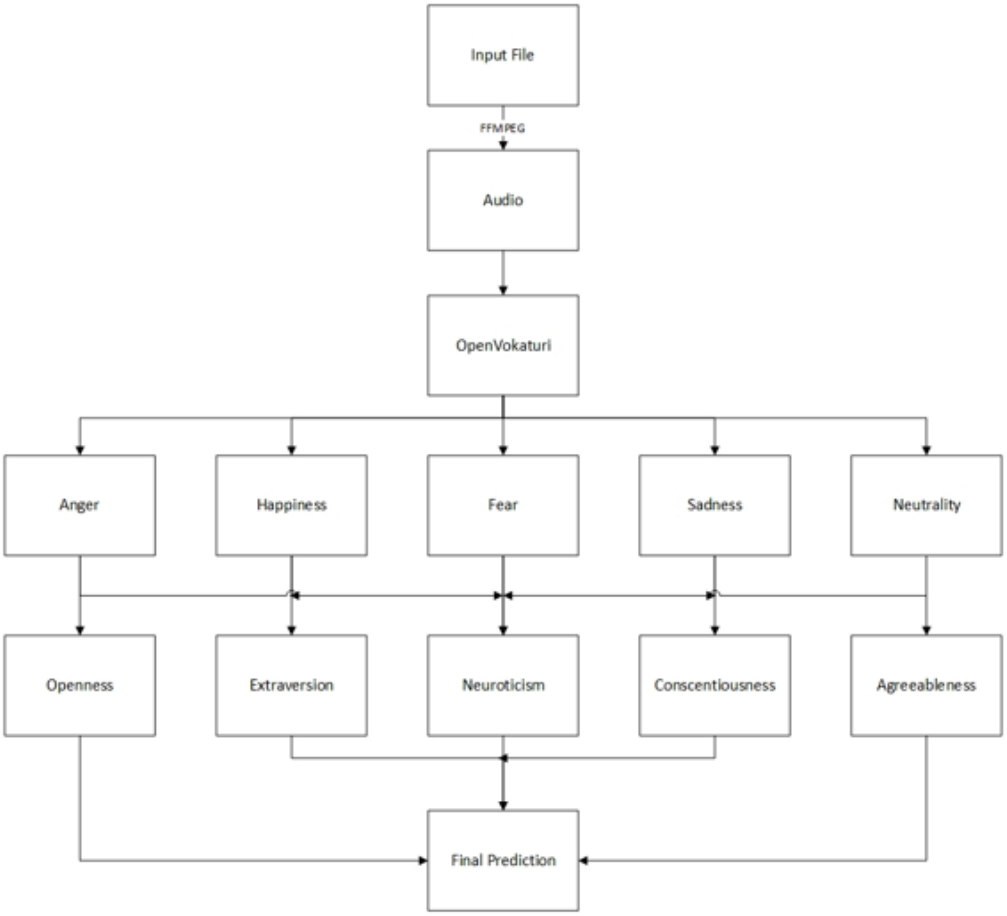

Figure 9 Prediction process using emotional values from OpenVokaturi

Just like in the training process the audio is extracted from the video using FFmpeg tool and the five features are extracted from the audio using OpenVokaturi. The five features are fed into each of the five personality trait models. The models using these features predict the individual traits of the user. These traits are combined to give a final prediction of the big five personality traits of the user.

4.4 Text Analysis

Text analysis is a vital part of determining the personality of a person. How well a person answers the questions and how accurate and relevant those answers are, has a lot to say about the personality of a person. In our project we have used the text analysis to further support the results given by facial expression analysis and sound analysis. Using the bag of words technique [34], we identify the polarity of the text and tweak the results of the facial and sound analysis for a better accuracy.

4.4.1 Libraries Used

The following Python libraries have been used for the text analysis process:

- nltk (Natural Language Toolkit)
- nltk.tokenize
- nltk.corpus

4.4.2 Technique Used

The bag of words technique has been used to analyze the text. Following steps have been followed to analyze the text and train the classifier.

4.4.2.1 Training Data:

The dataset from the PyCharn Looking at people has been used. That data set along with the fifteen second videos contains the transcription of the videos as well. The transcription data set has been divided into following three parts:

- Training Data
- Validation Data
- Testing Data

Each of these data sets are saved in a pickle file, the corresponding big five personality trait values are places in a separate pickle file. Both files are loaded into a python dictionary. Python dictionary stores data as a key value pair. The video name is placed as the key and the corresponding transcription and big five personality trait values are placed as the values. Thus, the dictionary is ready for preprocessing and text analysis.

4.4.2.2 Text Preprocessing

In any text analysis application, the first and most important step is to preprocess your data before training it. The text in raw form contains many punctuations, stop words etc. that can hinder the performance of the model. Text preprocessing ensures that the unusable portions of the transcription is removed before feeding it to the training model. The text pre-processing consists of following two steps:

- Text Segmentation
- Removal of stop words.

4.4.2.2.1 Text Segmentation

Text segmentation is the process of dividing text into meaningful units of information. In order to determine the personality of a person we would need to analyze the meaning of the words and their context. For this reason, the text is first divided into words using the nltk.tokenize library. The word_tokenize () function takes a string and divide it into words. We have used the words from this function in our training procedure.

4.4.2.2.2 Removal of Stop Words

Stop words are the commonly occurring words in a language. The words like ‘a’, ‘and’, ‘but’, ‘how’ etc. are stop words. These words hinder the performance of language processing application therefore they are removed in-order to improve the accuracy. Since these stop words have no contribution in determining the personality of a person we removed these words using the nltk.corpus library.

4.4.2.3 Bag of Words

After pre-processing the text, the bag of words is created. The words in each transcription are saved in a python dictionary with their big five personality trait values. Once all the words are saved in a dictionary the minimum, maximum and average of these words is calculated. Based on these measurements the words are divided into bags [35]. Two bags are made for each of the big five personality traits, one representing the high end and the other representing the low end, thus a total of 10 bags are created. Those words which have a high average and a high maximum value for a certain personality are places in the high end back of that personality on the other hand if they have a low average and low minimum value then they are placed in the low-end bag. The minimum, maximum and average value selection is as following:

- Minimum < 0.3 and average < 0.45 (Low end bag)
- Maximum > 0.7 and average > 0.55 (High end bag)

4.4.3 Testing Process

The polarity of the testing sentence determined from the bag of words is used to tweak the results of the software. The speech from the interview is converted into text. This text is preprocessed and each

word is compared with [43–52] the words in the bags and is assigned a polarity. Counts of words for each personality high-end and low-end is calculated. These counts are compared with the results from the facial recognition and sound recognition modules. If the results match then the values are tweaked a little i.e. if the facial and sound recognition show that a person is an extrovert and the majority of the words from the text belong to the high extroversion bag then we increase the confidence value of the extroversion for that person. However, if there is a mis-match i.e. the results are of high extroversion but the majority text words belong to the low extroversion class then the value of extroversion is decreased. The amount of tweaking to be done is calculated using a Neural Network and validation set.

Chapter 5: Experimental results and analysis

To find out the best machine learning model that fit our data, we divided the results and experimentation portion into two phases:

i. Finding out the best machine learning model
ii. Finding out optimal hyperparameters for the model selected in the first phase

5.1 Selection of best model

For a selection of best model, we took a smaller sample size of our training videos and trained three different models with default parameters provided in the scikit-learn library. The three models tested were:

i. Support vector machine with a linear kernel
ii. Support vector machine with an RBF kernel
iii. Multi-layer perceptron regressor with 1 hidden layer consisting of 100 neurons

5.1.1 Dataset used for the first phase

Following dataset configuration was used for the first phase:

- 1,000 training videos (15 seconds each)
- Each video has resolution of 1280 x 720 pixels, which is then down-sized to 320 x 180 pixels
- Frames are extracted after every 200 milliseconds (time taken for brain to register a facial expression)
- A total of 75 frames per video are extracted, resulting in a total of 7,500 frames in total for the training data

5.1.2 Results

20 different models were trained (4 for each trait) for 1,000 videos. They were then evaluated using unseen testing data. Two different evaluation techniques were used:

- The formula used for absolute percentage error calculation was:

$$\textbf{Absolute Error Percentage} = \frac{\text{abs(Expected value-Experimental value)}}{\text{Expected value}} \times 100 \quad (1)$$

- Root mean square error [36] was also used to evaluate the deviation of output from the expected output.

A table describing the absolute percentage errors for all three models is given below:

	Linear support vector machine	**Support vector machine with RBF kernel**	**Multi-layer perceptron**
Openness	29.5419	18.6807	27.7751
Extraversion	16.2693	15.7303	15.5483
Neuroticism	37.5374	36.9677	36.3218
Agreeableness	20.1457	19.8011	19.6090
Contentiousness	32.3631	32.6581	32.8110
Average error	27.1715	24.7676	26.4130

Table 6: Absolute percentage errors

The table describing the RMSE (root mean square errors) for all three models is given below:

	Linear support vector machine	**Support vector machine with RBF kernel**	**Multi-layer perceptron**
Openness	0.19168	0.19376	0.18624
Extraversion	0.08705	0.08263	0.07503
Neuroticism	0.16114	0.15874	0.15589
Agreeableness	0.13726	0.13491	0.13360
Contentiousness	0.15071	0.15136	0.15428
Average error	0.14557	0.14428	0.14101

Table 7: Root mean square error

5.1.3 Key findings

Some key findings from the results of these experiments are:

i. Performance of multi-layer perceptron during the training phase was the best among all three models. Support vector machine with RBF kernel performed the slowest while training the model for new videos.
ii. Neural networks had the least overall root mean square error while support vector machine with RBF kernel had the least absolute percentage error.

Due to a large performance difference, while training the model, and overall better RMSE, we chose multi-layer perceptron for the next phase of experimentation.

5.2 Optimizing hyperparameters

The next phase was to optimize parameters for the model selected. Based on performance during training and the overall error scores, we chose to proceed with a multi-layer perceptron. Following key hyperparameters were tested for the selected model:

i. Number of neurons in a hidden layer
ii. Number of hidden layers
iii. Activation function
iv. Solver
v. Technique for updating learning rate
vi. Alpha value

All the different hyperparameters were tested individually to view their effect on the learning process.

5.2.1 Number of neurons in a hidden layer

The first hyperparameter tested was the number of neurons in a hidden layer. The starting point was chosen to be 100 and was iterated over 10 steps with increments of 10 until the number of neurons reached 200. RMSE was calculated for each step and plotted to view the change in RMSE over different values.

Following results were obtained after training and testing the model:

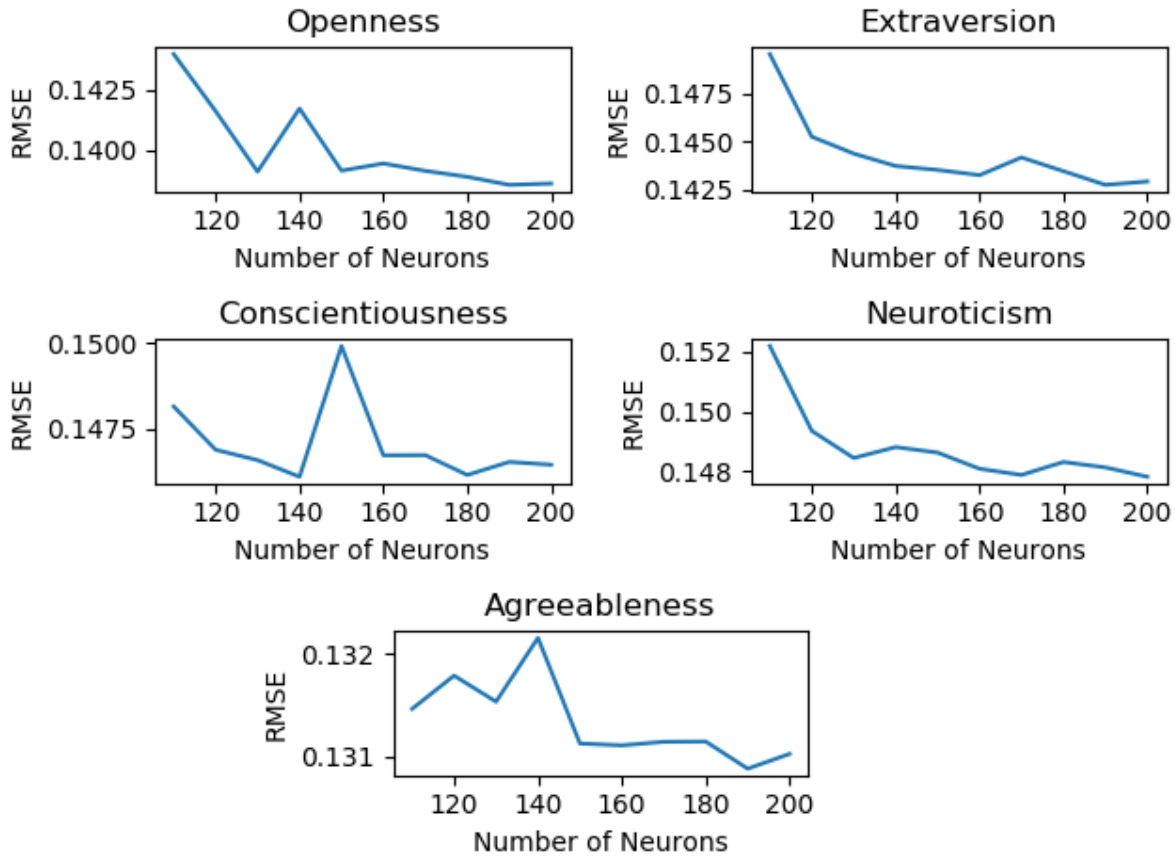

Figure 10: RMSE values for different number of neurons

Out of five traits, three had the minimum RMSE at 190 neurons while 2 had their minimum value at 200 neurons. Due to simple numerical lead, we chose to conduct rest of the experiments with 190 neurons in the hidden layer.

5.2.2 Number of hidden layers

After deciding on the number of different hidden layers, the next step was to select how many hidden layers to use. Usually using many hidden layers can be counterproductive and reduce the prediction accuracy of unseen data [37].

To reduce the effect of overfitting the data over many hidden layers, we limited the maximum number of hidden layers to 4. The training process was repeated each time for 1 to 4 hidden layers, with all the hidden layers having 190 neurons each (selected after running the previous test). The test results were generated in each iteration and cross validated to calculate the RMSE values.

The results achieved after the experiment were:

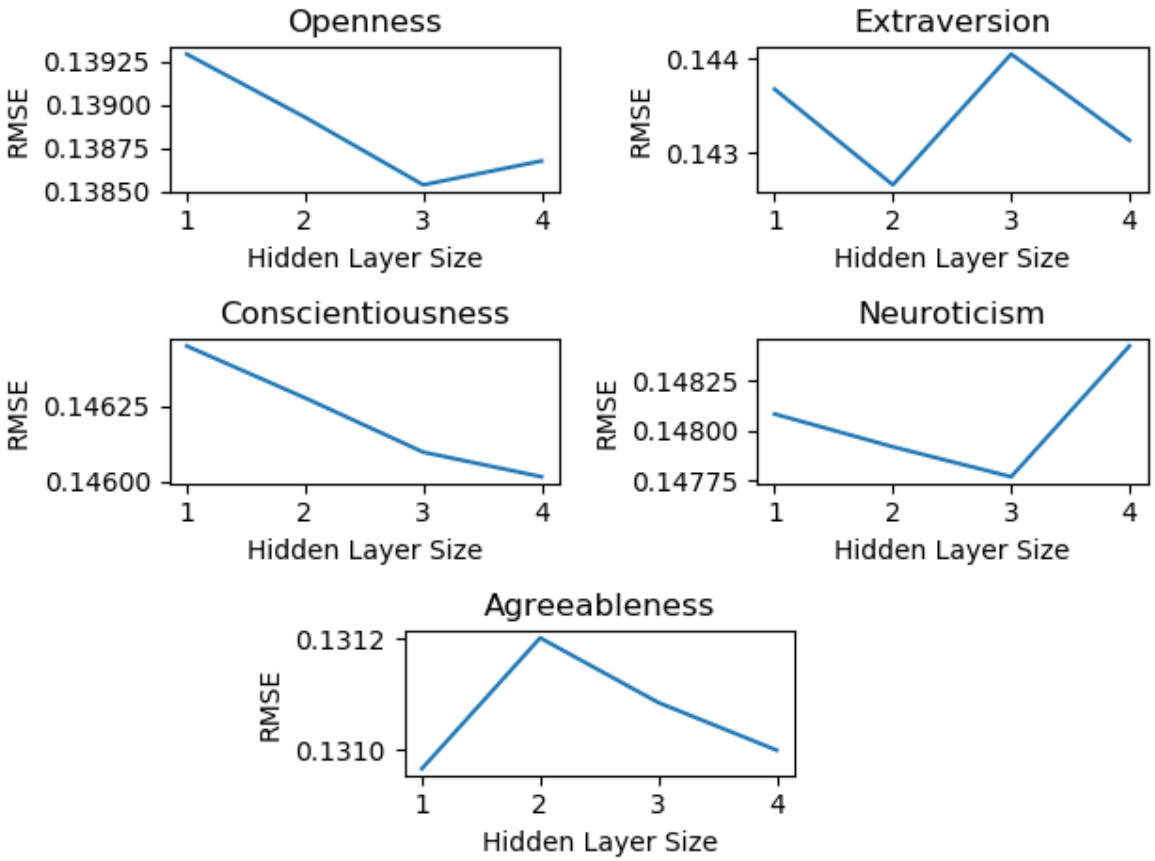

Figure 11: RMSE values for different number of hidden layers

The least RMSE value over all five traits were observed while using four different hidden layers with 190 neurons each in every hidden layer.

5.2.3 Activation function

An activation function defines the output of any node for a given input to that node. The activation functions allowed for the multi-layer perceptron in scikit-library are:

- **identity**
  This returns the same value as the input. Its equation is given as:

$$f(x)=x \tag{2}$$

- **logistic**
  This function returns a sigmoid value. The equation for this function is given as:

$$f(x)=\frac{1}{1+e^{-x}} \tag{3}$$

- **tanh**
  The value returned by this function is the hyperbolic tangent value. It can be represented in equation form as:

$$f(x)=\tanh(x) \tag{4}$$

- **relu**
  This function returns a simple rectified linear value. This value can be easily represented using a max function:

$$f(x)=max(0, x) \quad (5)$$

The following results were obtained after running the test:

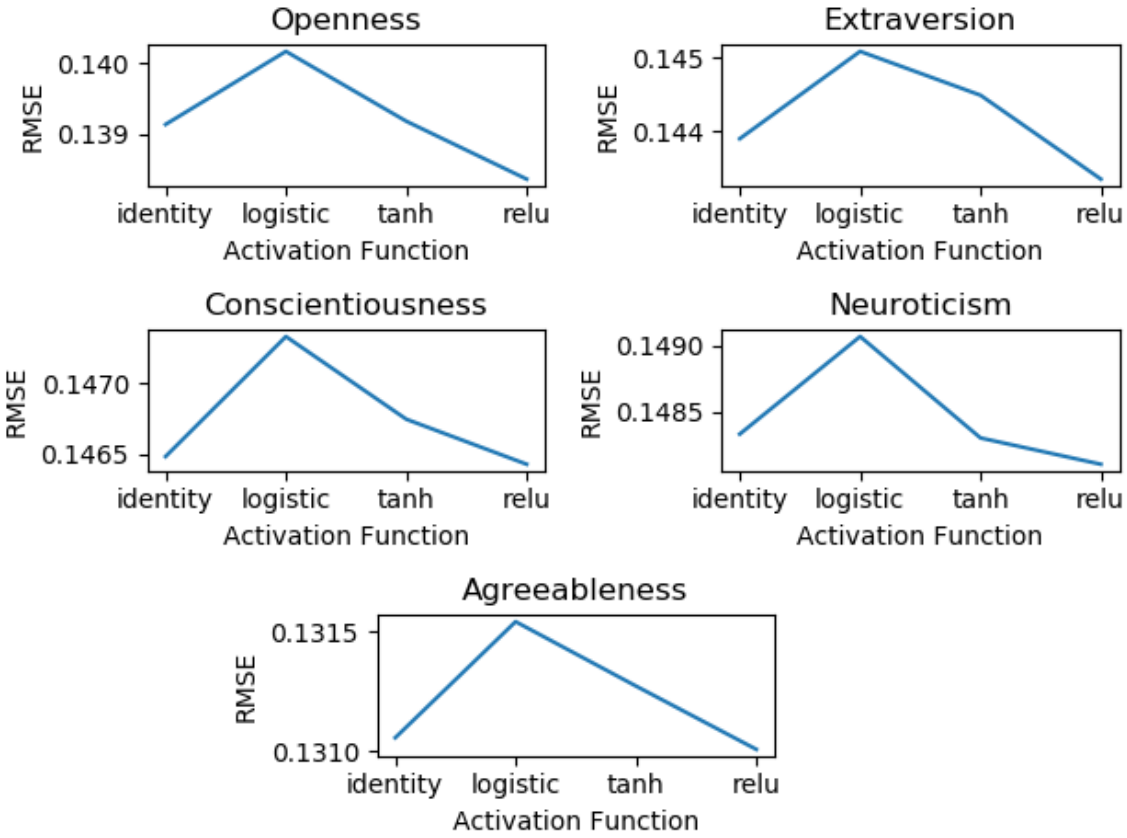

Figure 12: RMSE values for different activation functions

The most effective activation function in term of RMSE in all five different traits seems to be the *relu* function which returns a rectified linear value.

5.2.4 Solver

The solver is used by the algorithm for optimizing weights on each iteration over the network. Following different solvers are used by the multi-layer perceptron in scikit library:

- **lbfgs**
  An optimizer based on quasi-Newton methods [38].
- **sgd**
  A stochastic gradient descent based solver.
- **adam**
  A stochastic optimizer based on the research by Kingma, Diederik, and Jimmy Ba [39].

The results obtained by testing for each solver were:

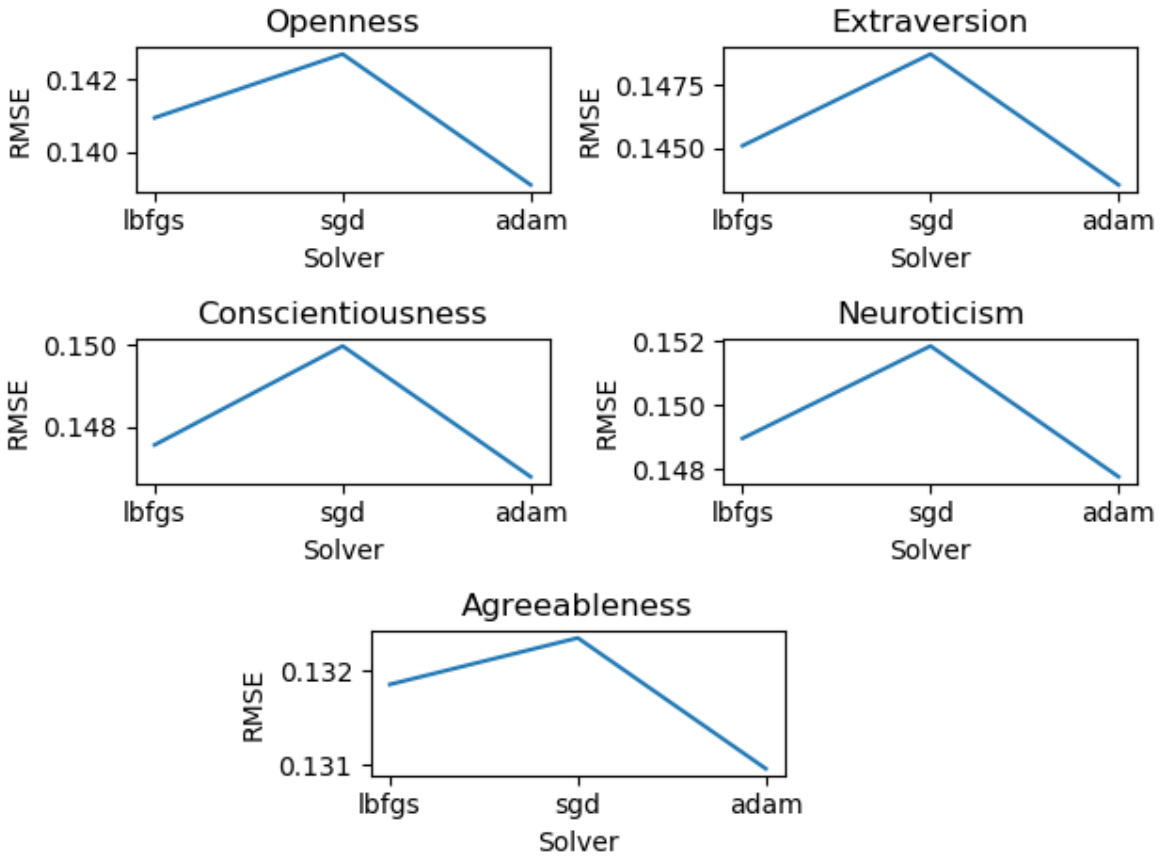

Figure 13: RMSE values for different solvers

Overall the five different learning techniques, *adam* proved to be the most effective solver when it comes to RMSE.

5.2.5 Technique for updating learning rate

The learning rate determines how much the weights are updated on each iteration while training the network. Learning rate can be kept constant or it can change over the course of the training process. Following techniques are employed by the scikit-learn multi-layer perceptron to update the value of the learning rate:

- **Constant**
  This keeps the learning rate same across all iterations.

- **Invscaling**
  This technique gradually decreases the learning rate by a factor of inverse scaling exponent (parameterized when using invscaling as learning rate updating technique). The generally used formula for a decrease over a step k is:

$$\textbf{New learning rate} = \frac{\textbf{Initial learning rate}}{\textbf{(Inverse scaling exponent)}^{k}} \quad (6)$$

- **Adaptive**
  Adaptive initially keeps the learning constant if there is a decrease in training loss. As soon as there are two iterations in which the training loss does not decrease, the learning rate is reduced to one-fifth of its current value.

The results obtained for different techniques for updating learning rate are:

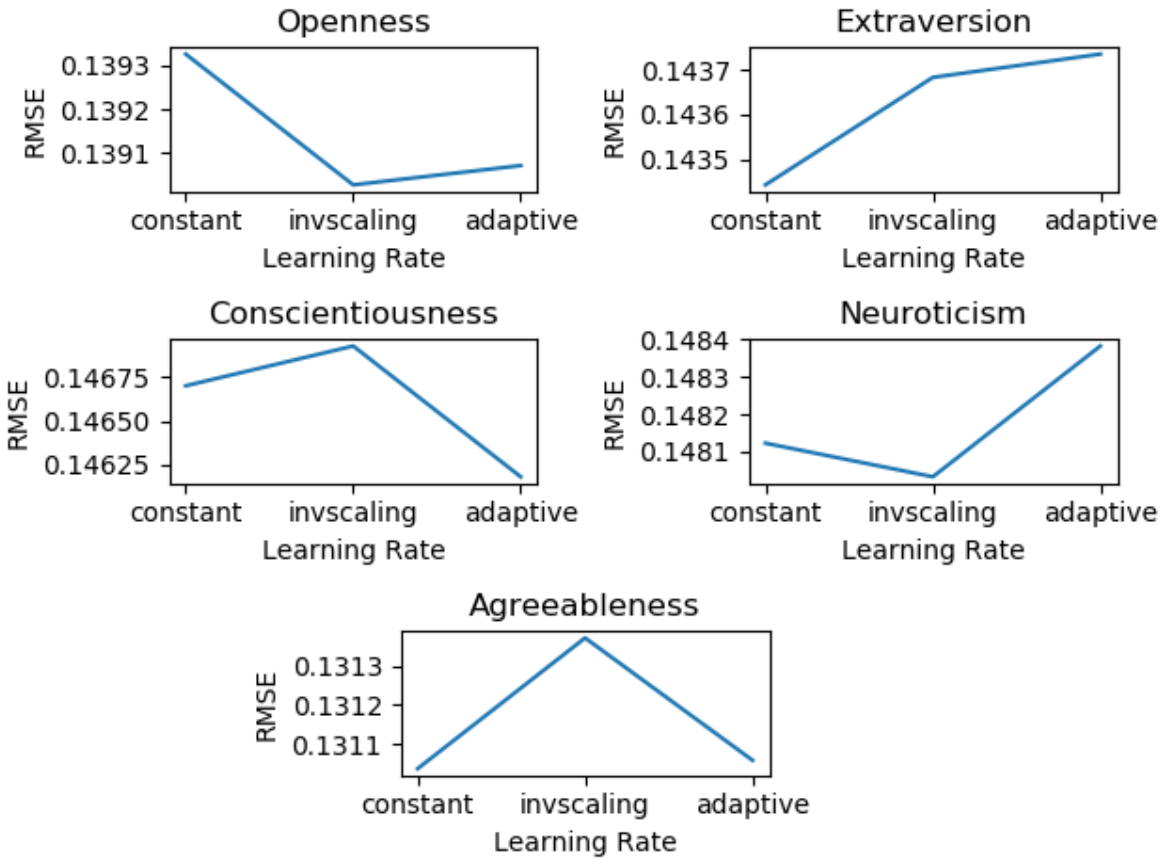

Figure 14: RMSE values for different techniques for updating learning techniques

Both adaptive and constant techniques provided good results during the testing process. However, the adaptive technique was marginally better than constant due to a smaller difference in RMSE values during cases it didn't perform well.

5.2.6 Alpha value

The alpha value is one of the most important hyperparameters used in scikit library's implementation of a multi-layer perceptron. This parameter regularizes the L2 loss function [40]. It decides the balance between minimizing the cost function and overfitting the model. The default value in scikit library is set as 0.0001. We tested the alpha values ranging from the default value of 0.0001 to 1.

These are the results which were obtained:

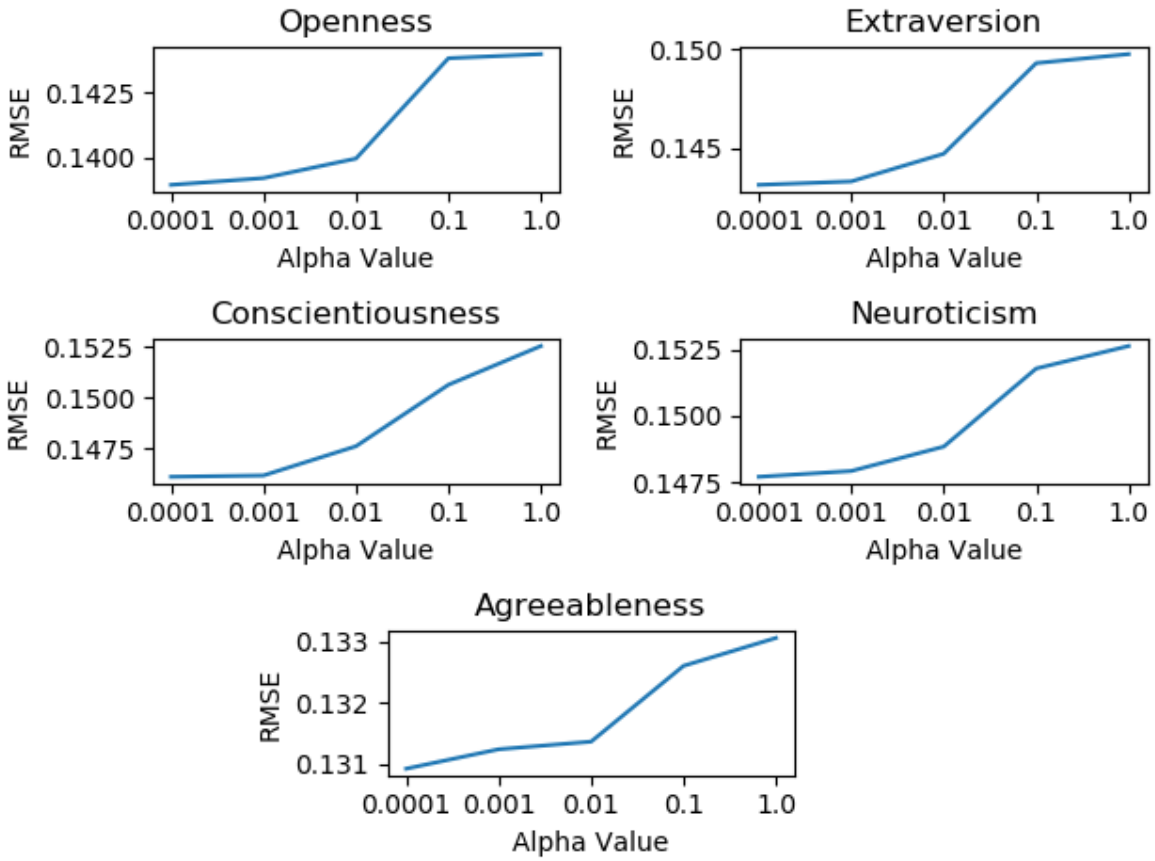

Figure 15: RMSE values for different alpha values

The results obtained for different alpha values were easiest to interpret among other tests. The graphs clearly show that using an alpha value of 0.0001 produces the best result in all five traits.

After optimizing hyperparameters and running the test on 2,000, we ran the tests again on unseen videos using both the optimized parameters and the default values. Following results were obtained for RMSE in each case:

	After optimization	**Before optimization**
Openness	0.13895	0.18690
Extraversion	0.14388	0.17543
Neuroticism	0.14879	0.15689
Agreeableness	0.13155	0.13360
Contentiousness	0.14618	0.15530

Table 8: Comparison of optimized and unoptimized model

The results show that optimization did help significantly in openness and extraversion traits. There are improvements in other traits too, but the margin is smaller as compared to aforementioned traits.

Chapter 6: Conclusions

The world is adopting the trend of automation at a very fast pace in the past few years. From business decisions to predicting a marathon winner, every process is being automated in every observable discipline. This report has put its focus on one such process, so-called an interview. The interview systems explored in our research spectrum could be broadly categorized into two; management systems and intelligent systems designed for the evaluation of a candidate undergoing an interview.

6.1 Initial research

We chose to research further about the intelligent systems; the features and parametric they use for making decisions about hundreds of candidates. The evaluating factor was decided to be a candidate's personality, defined by Big Myers along the dimensions of openness, extraversion, agreeableness, conscientiousness, and neuroticism. After research on intelligent systems, it was decided to use three metrics for the evaluation; a candidate's facial expression during the interview process, the acoustic features of the audio of the interview and the transcript of the interview.

6.2 Work completed

The current work has focused on training our models for facial expression's analysis of the video. Three different type of models were tested in the beginning: Linear SVM, SVM with RBF kernel and a Multi-Layer Perceptron. Object detection in the frames was carried out through Viola Jones' [19] Haar Cascades whereas processing the frames of an image were implemented through Local Binary Patterns [15].

Upon evaluations of the models, the Multi-Layer Perceptron was chosen as the base of our prototype and the model was further optimized by testing many variable hyperparameters. Due to lack of time and computational resources required to perform an extensive grid cross-validation technique on the training videos, the hyperparameters were optimized using a mixture of best choice forward and hit and trial method.

The scope of the project has been explored through much research. The implementation has started surfacing by first working on the video analysis. The challenges faced during the project was in the last phase of video analysis. Currently, an aggregate has been used to give a good assessment of the five personality traits. This, although has been giving more than good results, could be improved by employing a rate of change of measure in the position of the features. The current work has also left out other important features for facial expressions' analysis namely forehead, cheeks and eyebrows.

For FYP-II, we have broadened the analysis to integrate the audio and transcript of the interview with the video analysis. We are using OpenVokaturi to extract emotional values from the audio of the interview and are training models based on that. The text part is used to instill confidence into the analysis by finding out words that match the result given by combination of both audio and video.

References

[1] S. E. Seibert and M. L. Kraimer, "The Five-Factor Model of Personality and Career Success," Journal of Vocational Behavior, Cleveland, 2001.

[2] J. Rong, Y. P. P. Chen and M. Chowdhury, "Acoustic Features Extraction for Emotion Recognition," IEEE/ACIS International Conference, 2007.

[3] T. S. Polzin and A. H. Waibel, "Detecting Emotions in Speech," School of Computer Science, Carnegie Mellon University, 1998.

[4] T. H. Lok, C. M. F. Kevin, F. C. Pan and L. H. Tsun, "A Portable and Intelligent Interview System," The University of Hong Kong, 2015.

[5] "Arya - AI Recruiting Technology," 2014. [Online]. Available: https://goarya.com.

[6] "Video Interviewing Software|InterviewStream," 2016. [Online]. Available: https://interviewstream.com.

[7] "Mya | Your Team's A.I. Recruiter," 2015. [Online]. Available: https://hiremya.com.

[8] "ATIS - Automated Interview Telephony System," 2015. [Online]. Available: http://umstechlabs.com/index.php/products/cloud-telephony/automated-telephonic-interview-system.

[9] "Human Centered Innovations | About Matlda," 2011. [Online]. Available: https://www.hc-inv.com/about-matlda.

[10] D. M.-T. Chu, P. R. Khosla, S. M. S. Khaksar and D. K. Nguyen, "Service Innovation through Social Robot Engagement to Improve Dementia Care Quality," Assitive Technology, 2016.

[11] "HireVue Video Intelligence," 2004. [Online]. Available: https://www.hirevue.com.

[12] "Speed Interviews Project - ChaLearn Looking at People," 2011. [Online]. Available: http://gesture.chalearn.org/speed-interviews.

[13] "2017 Looking at People CVPR/IJCNN Coopetition," 2017. [Online]. Available: http://chalearnlap.cvc.uab.es/challenge/23/description/.

[14] "First Impressions V2 (CVPR'17)," 2017. [Online]. Available: http://chalearnlap.cvc.uab.es/dataset/24/description/.

[15] T. Ahonen, A. Hadid and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006.

[16] O. M. Parkhi, A. Vedaldi and A. Zisserman, "Deep face recognition," British Machine Vision Conference, 2015.

[17] D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner and W. Cukierski, "Challenges in representation learning: A report on three machine learning contests," 2013.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[19] P. Viola and M. Jones, "Robust Real-time Object Detection," Second International Workshop on Statistical and Computational Theories of Vision, 2001.

[20] V. Ojansivu and J. Heikkilä, "Blur Insensitive Texture Classification using Local Phase Quantization," International Conference on Image and Signal Processing, 2008.

[21] J. Kannala and E. Rahtu, "Binarized Statistical Image Features," Pattern Recognition Conference (ICPR), 2012.

[22] S. E. Bekhouche, A. Ouafi, F. Dornaika, ATaleb-Ahmed and A. Hadid, "Pyramid Multi-Level Features for Facial Demographic Estimation," In press, 2013.

[23] "FacialAction," 2013. [Online]. Available: https://github.com/go2chayan/FacialAction.

[24] "Facial Recognition Software SHORE," 2017. [Online]. Available: https://www.iis.fraunhofer.de/en/ff/sse/ils/tech/shore-facedetection.html.

[25] "Praat: doing phonetics by computer," 2014. [Online]. Available: http://www.fon.hum.uva.nl/praat/.

[26] M. A. Shah, "Emotion Detection from Speech," Hewlett-Packard Software Community, 2012.

[27] V. Chernykh, G. Sterling and P. Prihodko, "Emotion Recognition From Speech With Recurrent Neural Networks," arXiv:1701.08071v1, 2017.

[28] A. Davletcharova, S. Sugathanb, B. Abraham and A. P. James, "Detection and Analysis of Emotion from Speech Signal," Procedia Computer Science, 2015.

[29] "Linguistic Data Consortium," 2017. [Online]. Available: https://www.ldc.upenn.edu.

[30] "The Interactive Emotional Dyadic Motion Capture (IEMOCAP) Database," 2004. [Online]. Available: http://sail.usc.edu/iemocap.

[31] G. S. Philippe, L. S. Petro and M. L. Smith, "Transmission of Facial Expressions of Emotion CoEvolved," PLoS ONE, Glasgow, United Kingdom, 2009.

[32] L. Wang and D. He, "Texture classification using texture spectrum," Pattern Recognition, Vol 23, 1990.

[33] L. Wang and D. He, "Texture Unit, Texture Spectrum, and Texture Analysis," IEEE Transactions on Geoscience and Remote Sensing, 1990.

[34] Y. Zhang, R. JIn and Z. Zhou, "Understanding bag-of-words model: a statistical framework," *International Journal of Machine Learning and Cybernetics*, vol. 1, no. 4, p. 43–52, 2010.

[35] J. Yang, Y. Jiang, A. G. Hauptmann and C. Ngo, "Evaluating bag-of-visual-words representations in scene classification," *Proceedings of the international workshop on Workshop on multimedia information retrieval*, pp. 197-206 , 2007.

[36] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?," Geoscientific Model Development Discussions, Volume 7, Issue 1, 2014, 2014.

[37] N. M. Wagarachchi and A. Karunananda, "Mathematical Modeling of Hidden Layer Architecture in Artificial Neural Networks," 3rd International Conference on Information Security and Artificial Intelligence, 2012.

[38] D. F. Shanno, "Conditioning of quasi-Newton methods for function minimization," Mathematics of Computation, 1970.

[39] K. D. P. and J. L. Ba, "ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION," ICLR 2015, 2015.

[40] R. C. Moore and J. DeNero, "L1 and L2 Regularization for Multiclass Hinge Loss Models," Symposium on Machine Learning in Speech and Natural Language Processing, 2011.