BACHELOR OF SCIENCE IN
COMPUTER SCIENCE AND ENGINEERING

Inspiring Excellence

Personnel security system of nuclear power plants using machine learning for psychological, behavioral and social media activity analysis.

Authors
FABIHA NEAZ
HASAN TAWQIR AHMED
MARIUM TASNUVA GRANTHI
SHUVRO DEV SAHA

Supervisor
Dr. Md. Ashraful Alam
Assistant Professor
Department of CSE

A thesis submitted to the Department of CSE
in partial fulfillment of the requirements for the degree of
B.Sc. Engineering in CSE

Department of Computer Science and Engineering
BRAC University, Dhaka - 1212, Bangladesh

December 2018

We are proud to dedicate this research respectfully to our beloved parents, honorable supervisor, esteemed faculty members, friends and well wishers who keep trust in us …

Declaration

It is hereby declared that this thesis /project report or any part of it has not been submitted elsewhere for the award of any Degree or Diploma.

Authors:

FABIHA NEAZ
Student ID: 14301031

HASAN TAWQIR AHMED
Student ID: 14101203

MARIUM TASNUVA GRANTHI
Student ID: 14301020

SHUVRO DEV SAHA
Student ID: 14101200

Supervisor:

Dr. Md. Ashraful Alam
Assistant Professor, Department of Computer Science and Engineering
BRAC University

December 2018

The thesis titled .....
Submitted by:
FABIHA NEAZ Student ID: 14301031
HASAN TAWQIR AHMED Student ID: 14101203
MARIUM TASNUVA GRANTHI Student ID: 14301020
SHUVRO DEV SAHA Student ID: 14101200
of Academic Year ..... has been found as satisfactory and accepted as partial fulfillment of the requirement for the Degree of ..... (An example is shown. It must be replaced by the appropriate Board of Examiners)

1. Dr. Md. Ashraful Alam
Assistant Professor
University Building 5, Level 3, BRAC University

2. Md. Abdul Mottalib, Ph.D
Professor and Chairperson
University Building 5, Level 4, BRAC University

Acknowledgements

We would like to offer our earnest appreciation to our thesis supervisor Dr. Ashraful Alam, Assistant Professor of department of Computer Science and Engineering, BRAC University for guiding us on how we should approach the idea and for his huge help in completion of our thesis. His input, supervision, and guide have been of tremendous incentive throughout our research work. We are thankful to BRAC university CVIS lab for helping us with all the equipment needed for our research. Besides, we are thankful to the BRAC University Faculty Staffs of the Computer Engineering Department who have been a light of guidance for us through the entire study period in BRAC University, particularly in building our base in education and upgrading our insight. Finally, our heartiest gratitude to Prof. Dr. Md. Golam Rabbani, Chairperson , Neuro Developmental Disability Protection Trust, Ministry of Social Welfare and President, Bangladesh Association of Psychiatrists for helping us to generate the question set.

Abstract

In this research, a novel Personnel Security Model is designed and demonstrated for detecting suspicious people in an organization especially for nuclear power plants. The proposed system composed of three subsystems and a final decision making system such as i. Application software for performing a dynamic questionnaire session of individual employee of the power plant, ii. Eye blink and response time counter for lie detection during the questionnaire session and iii. Another sub system is also introduced for sentiment analysis from social media activities. Then, based on the outputs of three sub-systems, final decision is generated. In first sub-system, According to the science of psychology, suspicious people can be detected by asking some questions, by their response time and their eye blinking, lie can also detected. On the other hand their social media posts can also reflect a person's actual psychological condition. In this study a person's answers of the psychological questions, their eye blinking and response time corresponding to the question, and their social media activity are taken in consideration to extract as parameters or features for the final prediction model to find out whether a person is suspicious or not. Experimental results and analysis have been presented to justify the validity of the proposed method.

Table of contents

List of figures

List of tables

Chapter 1

1. Introduction

1.1 Motivation

The personnel security system was first introduced amid the World War II to support the classification system and of actualizing the Truman and Eisenhower Administrations' programs to explore the devotion of Federal Government authorities. Over the past 50 years, loads of directives and additional regulations have been issued to tailor the framework to explicit needs and retort to specific concerns, creating a layering of principles and, thusly, certain redundancies and other inadequacies. Recognizing the threat of cyber-attacks on Nuclear Power Plant (NPP), the Nuclear Regulatory Commission (NRC) has been active in updating the regulations and guidance to include a cyber-security component. Getting motivated on this note, we have come up with this approach to secure industrial environment. IAEA (International Atomic Energy Agency) has detected 3 main securities in nuclear power plant platform which are basically- i. Personnel ii. Physical and iii. Information Security. By securing personnel selection, we believe we can secure the nuclear power plant industries better than before[4]. An updated personnel security system furthermore ought to designate more consideration and resources for screen, evaluate, and encourage current specialists, precisely those in places of most prominent affect ability and the people who have advanced toward getting to be in threat because of changes or distresses in their lives. More prominent security care and understanding should provoke a more secure workplace, as faculty end up being more capable about the key security concerns and imperative perils, and what instruments exist to respond to these challenges. Between individual human correspondences combines not just spoken language yet likewise non-verbal signs, for example, hand motions, facial expressions and tone of the voice, which are used to express sensation and give reaction. The focus purpose of our research is to building up a computer system that can identify its employees' psychological condition and conform to them appropriately with the end goal in

order to increase social presence. This system will have the capacity to distinguish whether an employee is suspicious or non-suspicious. A liar is somebody who intends to captivate or deceive the other part even by a genuine data and with not bad intention. Lying is a sort of trickiness, which was always classed as an impulse part of human instinct. Hence, thinking about techniques to get it is not a recent domain. In this work we are intending to propose a system for emotion and lie identification for nuclear power plant. Our fundamental target is to find the proper features that can portray emotions of people and ensure personnel security of nuclear power plant by identifying employees' psychological behavior. Lying is considered as the most common human act that merits spending time thinking about it. Lying is a form of trickery, which was always categorized as an predisposition part of human nature. Vrij et al. (2000) in their paper have researched on the theory of lie detection, which came up with the principal ways to discover liars: physiological responses measure, speech analysis and behaviors recognition based on multiple traits like faces, facial expression [14]. The objective is to interrogate the workers of the nuclear power point on regular basis and store them in software .A real-time algorithm to distinguish eye blinks in a video sequence from a standard camera is proposed. Current landmark indicators, prepared on in-the-wild data sets show excellent vigor against face resolution, varying illumination and facial expressions. We demonstrate that the landmarks are recognized absolutely enough to dependably estimate the dimension of the eye transparency. The proposed algorithm accordingly assesses the facial landmark positions, extracts a single scalar quantity Eye Aspect Ratio (EAR) describing the eye openness in each frame. Finally blinks are identified either by a SVM (Support Vector Machine) classier recognizing eye blinks as an example of EAR esteems in a short worldly window or by concealed Markov model that estimates the eye states pursued by a straightforward state machine perceiving the blinks as per the eye conclusion lengths. The proposed algorithm has equivalent outcomes with the state-of-the-art methods on three standard data sets. And the third and final aim of us is to detect employee crime through their social media activities. Now-a-days, internet-based life is a typical stage for individuals of various age and occupation and it helps to detect a person's characteristics and personality, as people share their day-to-day activities here. Furthermore, they interact with their friends and family through social media. The posts, remarks and interacting with various people can give us an idea of what a person is up to, or what is one's present mental state. Constantly following their activities, recording them and using them to distinguishing their emotional state is one of the procedures to detect crime and averting it.

1.2 Literature review

For a start, we went through different research on nuclear power plant security issues and also on detecting human expression through Eye Blinking while giving answer to the dynamic questions and it came to our knowledge that, eye blinks decrease when cognitive demand increases. Lying is more cognitively demanding than truth telling. Liars are tending to be more inclined than honest persons to monitor and regulate their manner so that they will appear honest to the lie detector, which should be cognitively demanding. Leal and Vrji (2008) discussed tested the hypothesis [6] that when liars faces less cognitive demand, the level of their eye blink increases, on the other hand, eye blinks demand has reduced after the lie is told. A research was done on total of 13 liars and 13 truth tellers in a target period. Liars and truth tellers both told the truth in two baseline periods. Their eye blinks during the time and directly after the time were recorded. And a light difference was noticed that the eye blinks in liars was strikingly different from the pattern obtained in truth tellers Periods [6]. Remeseiro et al. proposed a methodology, which includes the detection of the face and eyes of a person, and create a quantitative vector by analyzing the low-level features of the eye section. The vector is classified into two categories considered either open or closed eyes by using machine learning algorithms. The effectiveness of the proposed methodology was errors under 5% [10]. It takes more on average time to convey a intentionally false response than a honest one because it requires the truth to be acknowledged and then reformed. Three experiments [11] were performed where subjects indicated whether presented numbers were greater or lower than a given customary number, and to give false response on half the trials. Results found, lying to add some extra time to response independently of other factors like method of response. Additionally, true Yes Response Time were shorter than true No ones, producing an interface with Response Time could reliably distinguish truth from lies for Yes responses but not so easily for No responses. Al-gawwam Benaissa (2018) invented a new technique which estimates the facial landmark positions and extracts the vertical distance between eyelids for each video frame . A Savitzky–Golay (SG) [7] filter is engaged to level the obtained signal while keeping the peak information to detect eye blinks. Pimplaskar et al. proposed a method to calculate eye-position and their direction grounded on initial centroid analysis technique. It was implemented by tracking eye position within high and low obstruction condition. They used the connected component technique and centroid method to track blinking of eyes on OpenCV platform [9]. In basis of our study for twitter sentiment analysis, we reviewed several papers of several writers. Kumar and Sebastian (2012) investigated a paradigm to mine the sentiment from Twitter, where users post real time reactions and state opinions. The paper expound a fusion tactic using both corpus based and dictionary-based methods to determine the semantic orientation of the opinion

words in tweets [5]. Agarwal et al. examined sentiment analysis on Twitter data, introduced POS-specific prior polarity features, and used a tree kernel. The features and the tree kernel perform approximately at the same level [1]. Sharlan et al. reported a design of a sentiment analysis, extracting tweets. Results classify tweets into positive and negative, which was epitomized in a pie chart and html page. The program was developed by using a Linux server or LAMP [3]. Twitter offers an unprecedented [2] opportunity to create theories technologies that search and mine for sentiments to uncover the sentiment, opinion words are extracted in the tweets. The corpus-based method is usually used to find the semantic orientation of adjectives and dictionary-based method can be used to find the semantic orientation of verbs and adverbs. The overall tweet sentiment can be calculated using linear equation which can incorporate emotion intensifiers too. This work is exploratory in nature and the prototype evaluated is a preliminary prototype. The initial results show that it is a motivating technique. INSAG-12 [12] states in paragraph 242 that the design and operation of nuclear power plants should provide adequate measures to protect the plant from damage and to prevent the unauthorized release of radioactive material arising from unauthorized acts by individuals or groups, including trespass, unauthorized diversion or removal of nuclear material, or sabotage of the plant.

1.3 Thesis contribution

The personnel security system that we have built up basically has three subsystems – Dynamic questionnaires, Image processing from video footage and social media activity analysis of the nuclear power plants' employees; combined into another subsystem which got us the final result to say whether the employee is suspicious or non-suspicious. The main aim of our research was to introduce the fourth subsystem, the combined one. Using the results of the first three subsystems as the inputs of the fourth subsystem was the new idea of our research, so that we can detect a person is suspicious or non-suspicious on the basis of three different attributes. Also we have introduced the system in the nuclear power plants platform, which is completely a new concept. The questions of the questionnaires subsystem were set on the basis of an employee's psychological thoughts and behavior, his day to day work life routine, pressure and responsibilities. In this system, there will be set of questions where the different employee face different question, the question set will be dynamic. So that the answers can give us an approximate correct impression to define the person's mental state or motive. The data acquisition through image processing is also done at the same time when the person is answering the questions. The eye blinks and time duration of answering each question are count to detect the person's irregularities and deception motive. Also now a days

people share their daily life, thoughts, activities and talk openly with their friends and family through social media. A regular analysis of social media activities of a person can help us to get knowledge whether a person is in a happy, sad, angry or depressed state. All attributes and the whole structure of our research is a new approach to ensure IAEA's (International Atomic Energy Agency) nuclear security plan in Bangladesh.

1.4 Thesis Orientation

In this paper, the first part is the introduction, where we have discussed the motivation that worked behind our research, the literature review and the contribution of our system, the part that we newly invented and the system that is our unique creation. In the second chapter, we have discussed the proposed model. As our system has total four subsystems, so in this chapter we have given description of each of our subsystems along with proper explanations, calculations and diagrams. In chapter 3, we have showed the result of our prepared system, discussed about the algorithms that we have used, which worked well for us and which don't. Lastly at chapter 4, we have concluded our paper with the future works that we want to do and extend this research. The references are mentioned at the end of our paper.

Chapter 2

Proposed Model

2.1 System Architecture of the proposed model

The data set creation will be the first step for us to accomplish our goal. For our research we will need to create four types of data to finally establish some result. To create data set the system will ask questions and collect answers and at the same time it will count the blink of the employee and measure the answering time of the person for each question. Lastly, we will also track some posts from twitter. In figure 2.1 we make a proposed model for personnel security of a nuclear power plant.

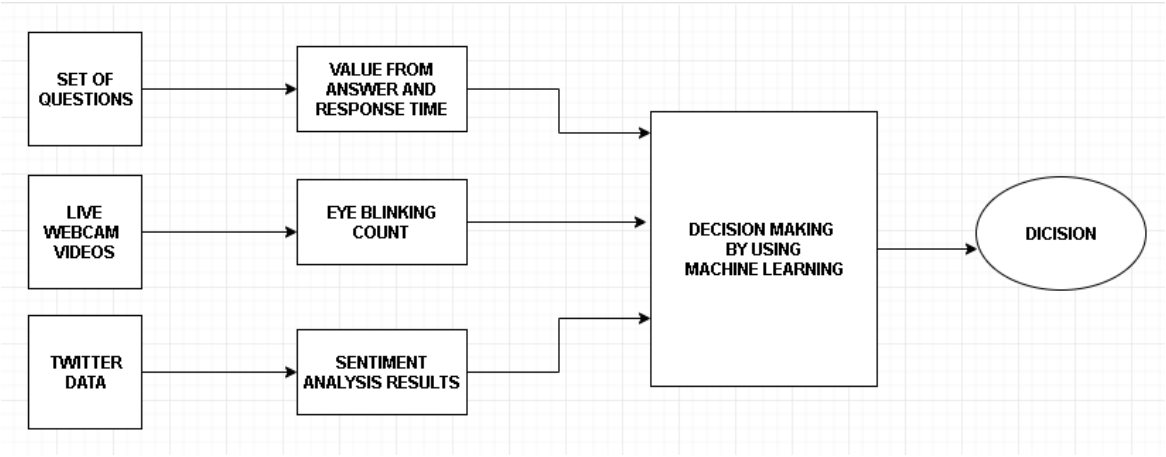

Fig. 2.1 System architecture of the proposed model

2.2 Questionnaires

We have dynamic multiple choice questions for each person, which model shown In figure 2.2. Nasri et al.(2000) in their research paper have stated that, there are two ways of test strategies, Control Question test (CQT) and Guilty knowledge Test (GKT). Where CQT is formed with set of controlled and relevant questions. GKT test on the other hand is constituted of multiple-choice questions around information that only a guilty could know [8]. The writers have also claimed that, strategic use of unexpected questions for proclamation of guilties when the number of suspects are more than one, can be a great help to detect a liar. For our calculation purpose we have used the luminosity method which is more sophisticated version of the average method. In this method, we are more sensitive to green than other colors, so green is weighted most heavily. Likewise, we have assigned non-suspicious answer with the highest value of 0.65, natural answer had the value of 0.20 and suspicious answer had the value of 0.15. We have set the question answers value parameter as according: For non-suspicious person =0.41 to 0.60, Neutral person = 0.20 to 0.40, Suspicious person =0 .15 to 0.20 the main task is to take the values of the 4 questions and make an average of them. If the value is greater than 0.50; then the person will be detected as non-suspicious. If the value is greater than 0.30 and less than 0.50; then the person will be detected as neutral. And if the value is less than 0.30; then the person will be detected as suspicious.

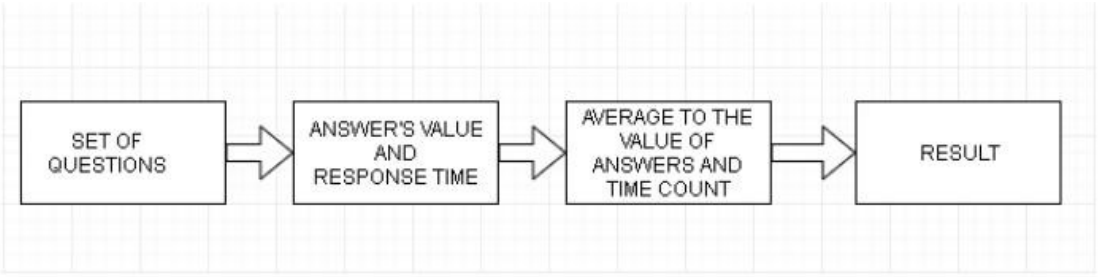

Fig. 2.2 Model of Questionnaires

2.2.1 Data collection through questionnaires

Here, we have asked some personal a set of questions. The sample questions are in figure 2.3.

Here, each multiple choice questions have different value which we have set as parameters. For example (A =0.65 ; B=0.20 ; C=0.15) According to the answers of the multiple

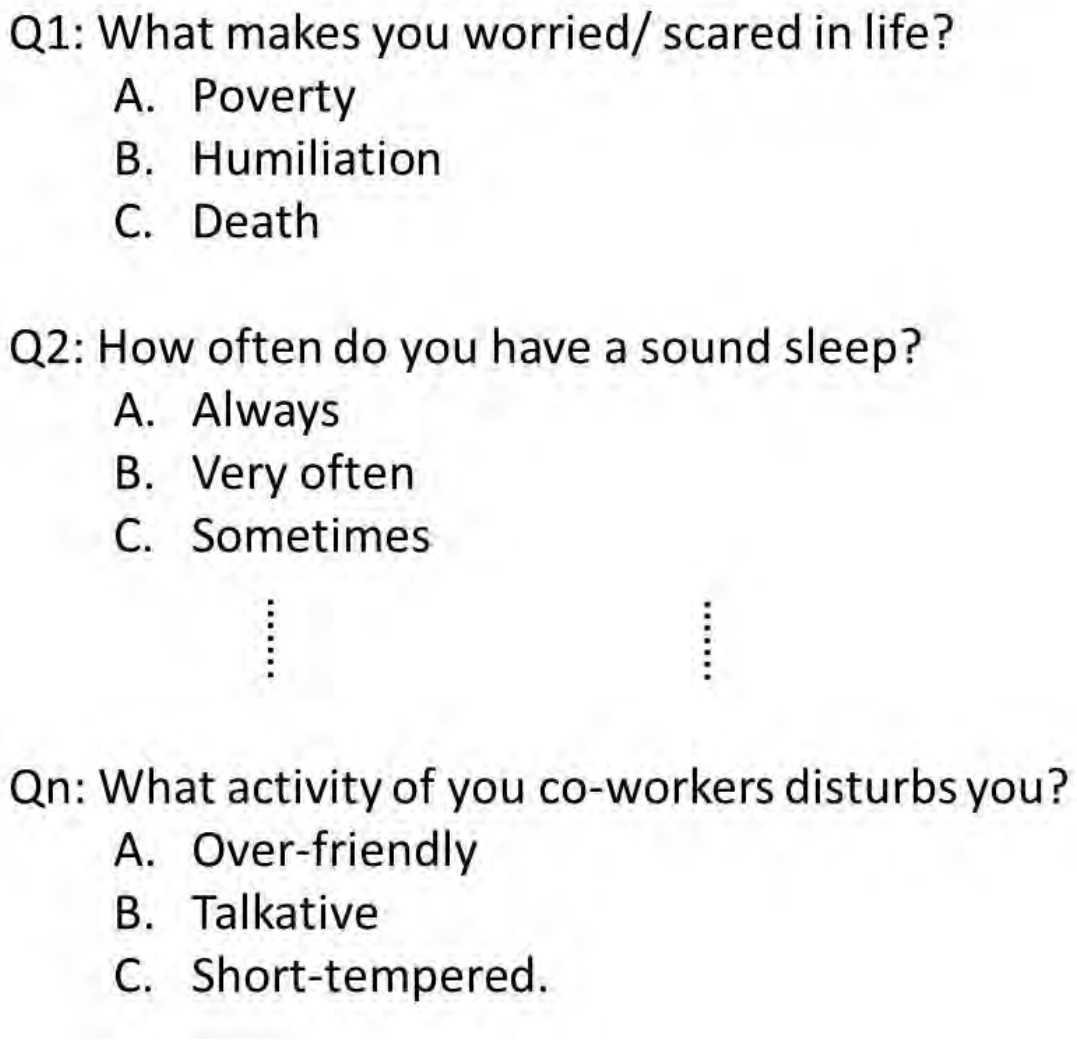

Fig. 2.3 Sample of questionnaire

choice questions, we get the value from the following equation:

$$P_Q = \frac{V_{Q1} + V_{Q2} + V_{Q3} + ... + V_{Qn}}{N} \tag{2.1}$$

Here, in 2.1 we added all the value that we get from the questionnaires then divide it with the total number of the questions. Again, we have also calculated the total response time by using a stopwatch.

Let's assume, the Response time = T
So, Response time of Question 1 = T_{Q1}
Response time of Question 2= T_{Q2}
.
.
.
Response time of Question n = T_{Qn}

In equation 2.2 we calculated the average response time of a person where N is the total number of the questions.

$$P_T = \frac{T_{Q1} + T_{Q2} + T_{Q3} + ... + T_{Qn}}{N} \tag{2.2}$$

2.3 Data acquisition through image processing

We will expand upon this information and build up a Computer vision application that is equipped for detecting and counting blinks in video streams utilizing facial milestones and OpenCV. To manufacture our blink detector, we will be figuring a metric called the eye aspect ratio (EAR), introduced by Soukupová and Cech in their 2016 paper, Real-Time Eye Blink Detection Using Facial Landmarks Contrasting traditional image processing methods for computing blinks, which commonly include some combination of: Eye localization, Thresholding to discover the whites of the eyes, Determining if the "white" area of the eyes dissolves for a time frame (indicating a blink).
The eye aspect ratio is rather a substantially much more exquisite solution that includes an

exceptionally basic estimation dependent on the ratio of distances between facial landmarks of the eyes. This strategy for eye blink detection is quick, productive, and easy to actualize.

2.3.1 Understanding the "eye aspect ratio" (EAR)

We have applied facial landmark detection to localize significant sections of the face, including eyes, eyebrows, nose, ears, and mouth: Likewise, we can extract specific facial structures by knowing the records of the particular face parts:

In terms of blink detection, we are only interested in two arrangements of facial structures — the eyes. Each eye is represented by 6 (x, y)-coordinates, initial at the left-corner of the eye (as if someone is looking at a person), and then working clockwise around the remainder of the region. There is a relation between the width and the height of these coordinates. Based on the work by Soukupová and Cech in their 2016 paper, Real-Time Eye Blink Detection using Facial Landmarks, we have derived an equation that reflects this relation called the eye aspect ratio (EAR):

$$EAR = \frac{||P_2 -_P 6|| + ||P_3 - P_5||}{2||P_1 - P_4||} \tag{2.3}$$

where P1 to P6 are 2D facial landmark locations. The numerator of this equation computes the distance between the vertical eye landmarks while the denominator figures the distance between horizontal eye landmarks, weighting the denominator appropriately since there is only one set of horizontal points but two sets of vertical points. By using this equation, we have found out that the eye aspect ratio is approximately constant while the eye is open, but will rapidly fall to zero when a blink is taking place. We have simply utilized this equation and relied on the ratio of eye landmark distances to determine if a person is blinking. To make this clearer; consider the following figure from Soukupová and Cech. On the upper left (figure 2.3) we have an eye that is completely open — the eye aspect ratio here would be larger and moderately constant over time. However, once the individual blinks (upper-right) the eye aspect ratio decreases significantly, moving towards zero. The bottom figure plots a graph of the eye aspect ratio over time for a video clip. As the eye aspect ratio is constant, and after then it quickly drops close to zero, then increases again, indicating a single blink has taken place.

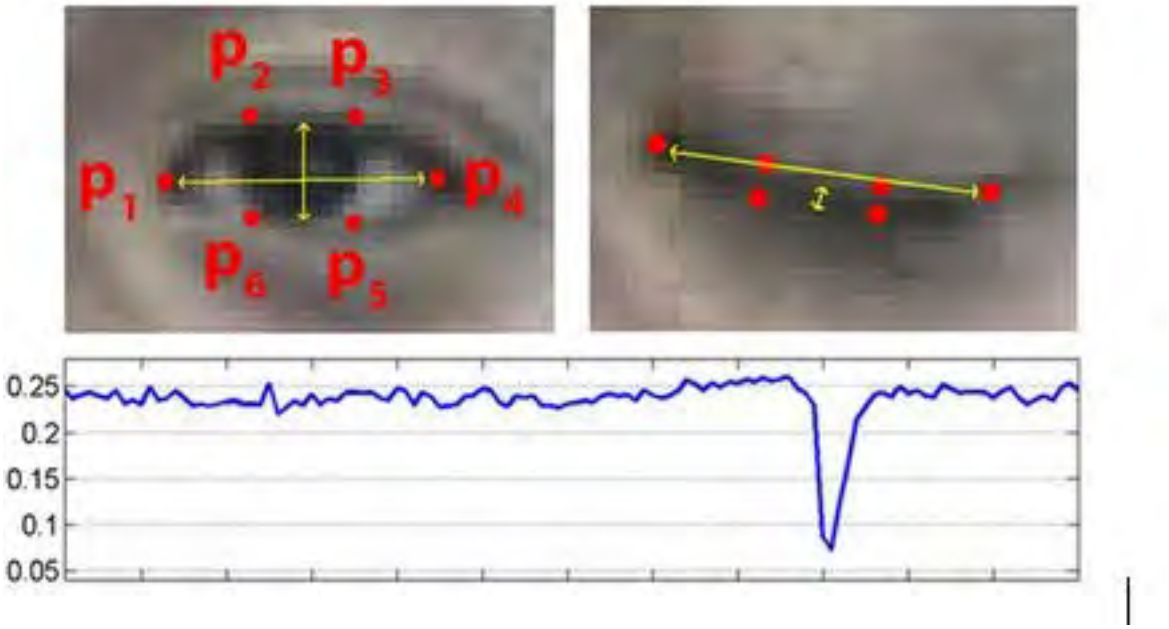

Fig. 2.4 Top-left: A visualization of eye landmarks when then the eye is open. Top-right: Eye landmarks when the eye is closed. Bottom: Plotting the eye aspect ratio over time. The dip in the eye aspect ratio indicates a blink (Figure 1 of Soukupová and Cech).

2.3.2 Calculations of the data

Here, In Figure 2.5 we try to show our Model of data acquisition through image processing. We collected the live video when a person attend in the questionnaires session then step ahead through one by one section in the model.

Let's assume, eye blink per question is BQ and We have calculated the average blinking for each person. Now, Lets assume that average eye blink is PB. The equation of average eye blinking is shown in 2.4.

$$P_B = \frac{B_{Q1} + B_{Q2} + B_{Q3} + ... + B_{Qn}}{N} \tag{2.4}$$

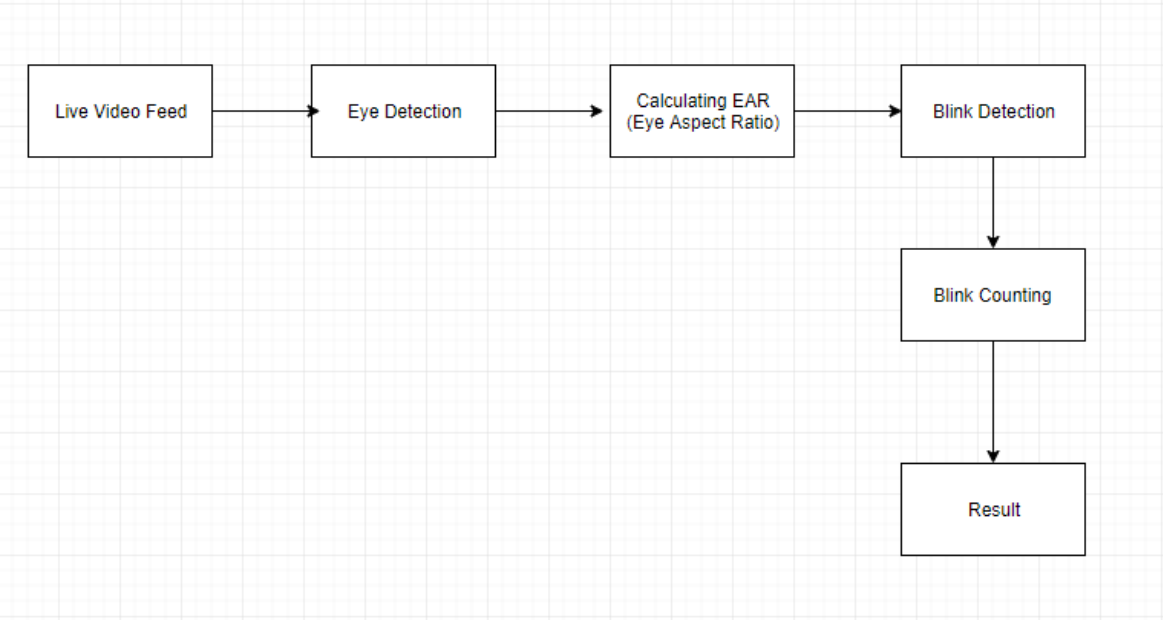

Fig. 2.5 Model of data acquisition through image processing

2.3.3 Eye blink detection with OpenCV, Python, and dlib

We have done eye-blinking detection in four parts: In the first part we have detected the eye aspect ratio and how it very well may be utilized to decide whether a person is blinking or not in a given video frame.

For this, we have written a Python, OpenCV, and dlib code to perform facial landmark location and detect blinks in video streams. Based on this application we have applied our method to identify blinks in instance web cam streams along with video files. We have done eye-blinking detection in four parts: In the first part we have detected the eye aspect ratio and how it very well may be utilized to decide whether a person is blinking or not in a given video frame. For this, we have written a Python, OpenCV, and dlib code to perform facial landmark location and detect blinks in video streams. Based on this application we have applied our method to identify blinks in instance web cam streams along with video files. OpenCv is a library of programming function for computer vision. It lets us to setup a pipeline for our computer vision project. It can load data from image files, videos, capturing devices. It also helps in feature extraction. OpenCV contains a long list of existing algorithms so we don't have to implement them our self. Applying machine learning algorithms it make recognition

and detection. Dlib is a frame work that helps to identify the facial landmarks which will help us out to detect face and eye. In order to define face landmarks, we have set a range of (x,y) value, for that we have calculated the full points, face points, jawline points, right eye brow points, left eye brow points, nose points, right eye points, left eye points, mouth outline points, mouth inner points. We have also defined some thresholds for the EAR value. One threshold is used to detect the eye to be closed and another threshold to define the number of consecutive frame the eye need to be "closed" in order for us to detect it as a wink. Before calculating the value of EAR, we drew the outline of the eyes.

2.3.3.1 The process

In order to build our blink detector we will be computing a matrix called EAR introduced by Soukupova and Chech (2016) in their paper. In our system, we will use the EAR method using the following steps [13]:
With the end goal to figure the Eye Aspect Ratio, we have to identify the layout purposes of the eyes first. Furthermore, with the end goal to identify the eye focuses, we first need to recognize the face, and after that distinguish the face landmarks. Then we will be using the Eye aspect ratio method to detect when an eye gets closed. For that we have initiate vertical eyed landmarks (x,y) – co ordinates A and B. Then we have calculated the distance between the horizontal eye landmark OC.At last we have calculated the value using the equation, 2.3. The figure 2.6 represents the process of our eye blink detection system using OpenCV and Dlib. To develop the system we have used OpenCV and Dlib. Firstly, the web cam starting method was called in order to start the webcam. After writing the function to start the webcam we have tried to detect the face, Then we have written the method to detect the eyes. We use the convex Hull and draw contour functions of OpenCv for this, Then we calculated the EAR for each eye and draw the value on the video frame. With the EAR value of each eye calculated, we check whether the value for each eye has gone below the threshold, which we have set earlier.

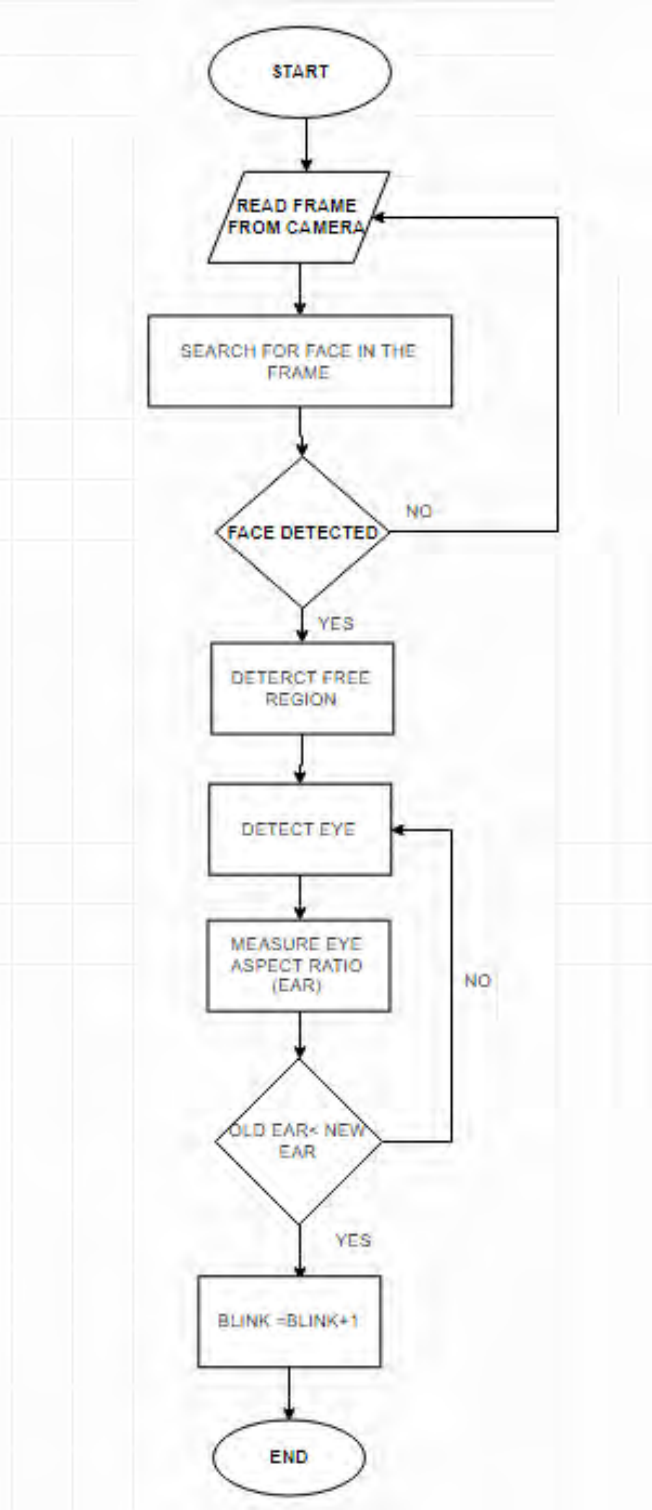

Fig. 2.6 Image Processing work flow

2.3.3.2 The output

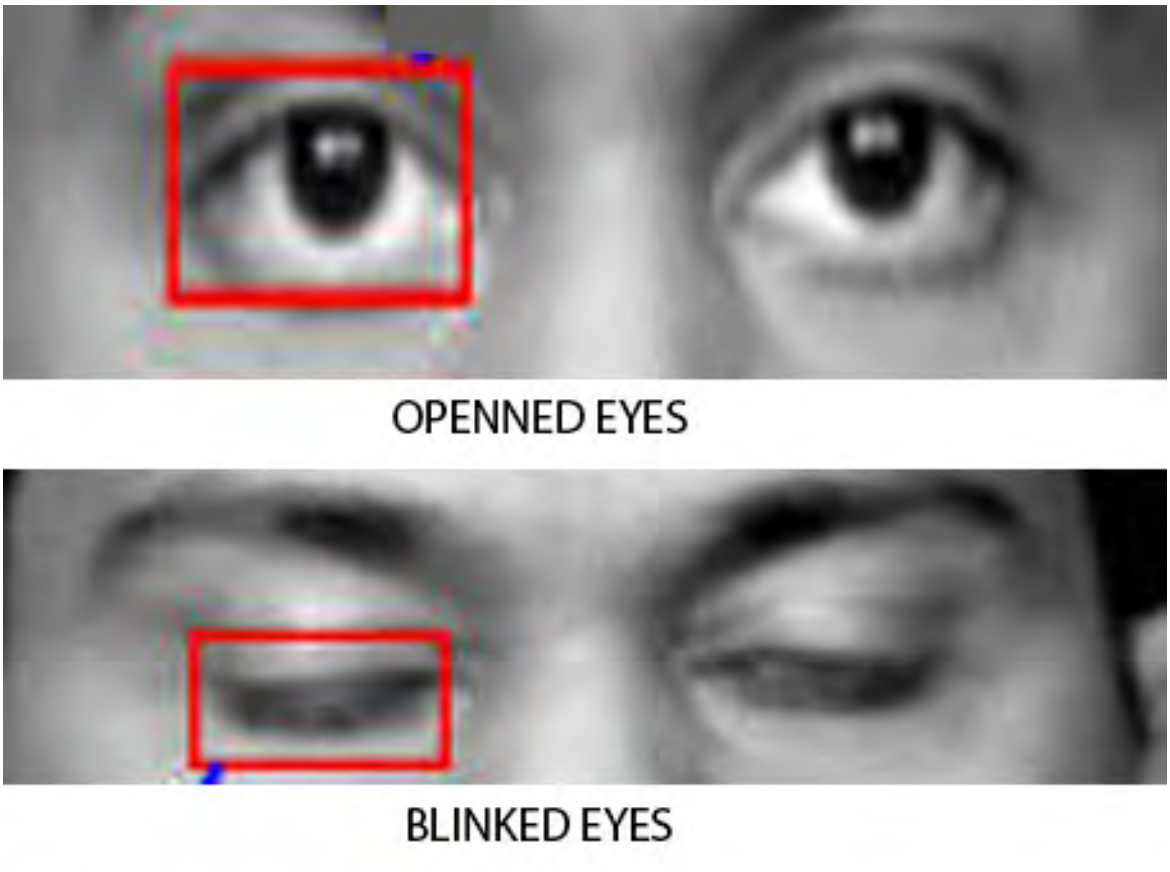

Fig. 2.7 Output of image processing code

In figure 2.7 the output shows that we can detect the blinks of live video feed using the algorithm, that we shown in figure 2.6.

Table 2.1 Sample table of Eye Blink Count

	EAR 1	EAR 2	EAR 3	EAR 4	EAR 5	EAR 6	EAR 7	EAR 8	EAR 9	Total Blink
Q1	0.25	0.10	0.05	0.23	0.22	0.08	0.19	0.05	0.18	4
Q2	0.23	0.05	0.24	0.03	0.28	0.25	0.07	0.25	0.60	4
Q3	0.26	0.03	0.02	0.09	0.26	0.03	0.20	0.25	0.50	5
Q4	0.30	0.28	0.25	0.08	0.27	0.23	0.05	0.18	0.90	4
Q5	0.28	0.30	0.26	0.04	0.09	0.09	0.25	0.08	0.10	4

In table 2.1 we calculated the total eye blink per person. While the person was answering

question EAR was calculated. For the accurate measurement, we have noted each points where EAR has been changed. If the EAR less than or equal 0.1, a blink was counted.

2.4 Social Media Activity analysis

2.4.1 Data collection using Web Scraping

Web Scraping is the procedure of data extraction from resources that are situated on the World Wide Web and the order of scratched and unstructured information (normally found in HTML pages) in an organized shape like Spreadsheets or database tables. Information shown by most websites must be seen utilizing an internet browser. They do not offer the usefulness to spare a duplicate of this information for individual utilize. The main alternative at that point is to physically reorder the information - an extremely monotonous activity that can take numerous hours or now and again days to finish. Web Scraping is the method of computerizing this procedure, so that rather than physically replicating the information from sites, the Web Scraping programming will play out a similar undertaking inside a small amount of the time.
Web scraping has turned into a fundamental skill to secure in the present computerized world, not just for tech organizations and not just for technical positions. On one side, compiling vast data sets are principal to Big Data analytic, Machine Learning, and Artificial Intelligence; on the opposite side, with the blast of computerized data, Big Data is ending up a lot less demanding to access than any time in recent memory.
For our research we have decided to do our sentiment analysis by using twitter posts of the employees. Python 3 is used to do web-scraping as python 3 is easier and quicker. We have used two main packages- they are request and beautiful soup. To request for the posts on the twitter , “request” package is used. The source codes have been stored in a variable which is request type. Next, we have used beautiful soup package which is a Python library for pulling data out of HTML and XML files. It works with parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers work. In our study, beautiful soup takes in two parameters-one is the data type from the request package another is the csv format that we want to store our posts into.

Figure 2.8 shows that, how we collect data from twitter and successfully done all the step one by one in the hole process and generate the result for our main system.

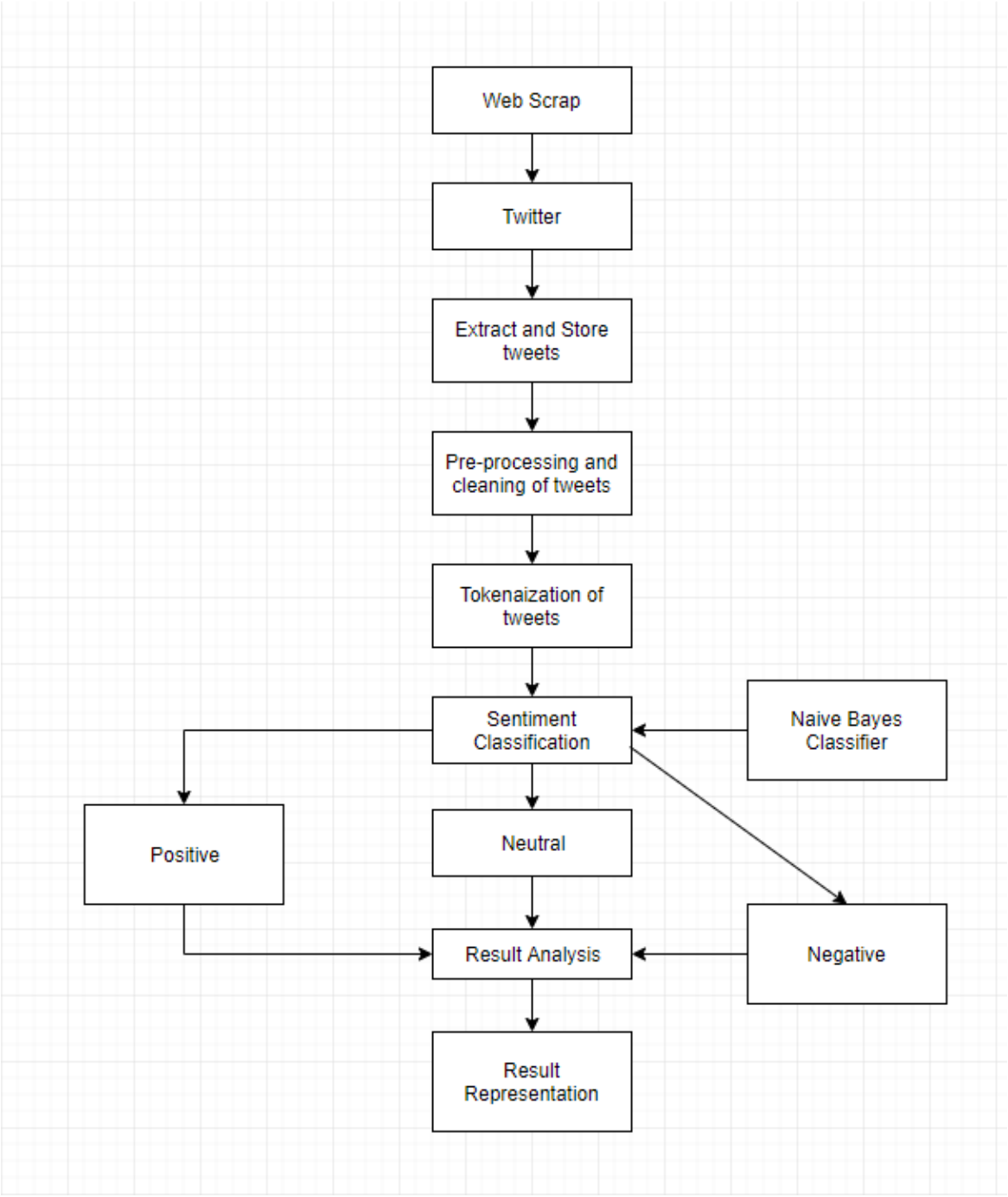

Fig. 2.8 Model of data collection from social media activity

2.4.2 Text-Preprocessing

The authors used text preprocessing to set up the data set. Natural language processing toolkit is utilized to pre-process the data. For the most parts tweets consist of message alongside usernames, special character, stop words, emoticons, hash tags, time stamps, URL's and so forth. Therefore, to make this data fit for mining authors pre-process this data by using various function of NLTK. In pre-processing extracting main message from the tweet, then removing the unwanted words or characters and then replacing all emoticons and abbreviations with their corresponding meanings is the first task. Once these are done the preprocessed tweet is clean. Sample tweet and processed tweet. Cleaning of twitter data is essential since tweets contain several syntactic highlights that may not be useful for analysis. The pre-processing is done so that data represented to just regarding words that can easily categorize the class. Next, the texts are tokenized by splitting the content into spaces and punctuations marks and then from bundle of words, in this case the authors used library to tokenize the tweet.

2.4.3 Data Classification

For sentiment analysis and accuracy testing utilizing machine learning the most important thing is the data set. 1) Naïve Bayes Classifiers are probabilistic classifiers which come under machine learning technique. These classifiers are depend on applying the Bayes' hypothesis with solid (naïve) assumption of independence between each match of pairs. Let us assume, there is a dependent vector from to and a class variable 'y'. So, according to Bayes' theorem,

$$\frac{P(X_1,...,X_n) = P(Y)P(X_1,...,X_n|Y)}{P(X_1,...,X_n)} \tag{2.5}$$

The proposed classifier is used to calculate the probability for a given words having a place with a specific class. It is used because it is easy to train and discover the classification. Pre-processed data is given as an input to train the classifier and that is applied on test set to generate positive, negative or neutral sentiment.
For twitter sentiment analysis, we have set some parameters for negative, positive and neutral posts. For negative posts, the parameter will be =0.3. For neutral posts, the parameter will be .01 and for positive posts, the parameter will be = .6.

2.4.4 Generating Results

The next step after using this algorithm is we get the result as following:

Positive = x%

Negative = y%

Neutral = z%

From this result we have calculated our result for each person.
Let assume twitter result denoted as PTW.
There, we have assumed that neutral result as positive.
So,

Case 1: If x>y and x>z; then PTW is positive.

Case 2: If y>x and y>z; then PTW is negative.

Case 3: If z>x and z>y; then PTW is positive.

2.4.5 Implementation

Firstly, for sentiment analysis it is very important to pre process the sentences. In the previous part we have preprocessed the text which contains of reducing the stop words, normalizing the sentences and tokenize them into words. For that process we have used the NLTK, which is a natural language processing tool. There are built in methods for removing the stop words from the sentence. Stop words are words which are used in the sentence for appending the sentence or dividing the sentence into parts such as, “the”, “but”, “and” etc. Next, we have Normalized the sentence which is also done by using two different methods called stemming and lemmatizing. Although, these two methods look similar by book they are actually not that much similar. Stemming converts a single word into its base form or stems that means the word is reduced form unnecessary suffixes whereas on the other hand lemmatization means converting the word into root, which means the word is converted into more dictionary form. Furthermore, we had to exclude the emoticons from the sentence as it doesn’t mean anything to the machine.
Secondly, after the text-preprocessing part we had to split the full sentence into words which is also done by a method of NLTK called word_tokenize. We tokenized the sentences into words and found out the positive and negative words by a method called nltt.pos_tag and nltk.neg_tag which actually finds out the positive and negative words from a collection of

words.
Next, we made a feature set consisting of positive words and negative words. After creating the feature set we trained the Naïve Bayes classifier which will determine the positive and negative words. So for positive word the classifier will give a result of 1-.06 and for negative word the classifier will show result below 0.3. Here we set the threshold value as 0.1 that means we avoided the words providing result below or equal to the threshold value. We trained the naïve bayes classifier with around 2000 feature sets. As the feature set increases the accuracy will also increase. However, larger number of feature set requires high processing power to classify which in our case is not applicable as the machine resource is limited. Furthermore, we saved the classification result as pickle format which is a built in format for saving results in python.
Next, we tested the data with the classifier. So for that we had to apply the naïve bayes classification on a feature set consisting of positive and negative words and see how the classifiers works. The classifiers will separate the words based on their polarity which are in this case positive, negative and neutral. For accuracy finding we had to compare the saved classification result and our tested classification result. This is also done using a method offered by Scikit learn library which we used for machine learning.
Moreover, the accuracy of the Naive bayes classifier varies depending on the number of feature set. As mention earlier, in this research we gradually increased the number of feature sets. We started from 500 feature set and stopped at 2000 which resulted an accuracy shift from 69.87% to 74.49%. The more the number of the feature set the more accurate the classifier.
We only used Naïve Bayes classifier in our research as it is considered as one of the most accurate in terms of text classification. There are also many classifiers for text classification but in terms of usage Naïve Bayes is the most familiar in the world of Sentiment analysis although Random forest classifier produces the best result for text classification.

2.5 Decision Making Process

To create our training dataset we have collected all the process described in section 2.2 ,2.3 , 2.4. We have collected data from 100 persons by using the previously used method.

2.5.1 Data Collection from the previously used methods

From the previously mentioned subsystems, we have collected the value of

PQ which is the result of a persons questionnaires test.

PB the value of a persons blinking at each question.

PT the time duration of each question.

PTW the twitter result and social media activity analysis process . for example,

Table 2.2 Sample table of data collection

Person 1	P_{Q1}	P_{B1}	P_{T1}	. . .	P_{TW1}
Person 2	P_{Q2}	P_{B2}	P_{T2}	. . .	P_{TW2}
Person 3	P_{Q3}	P_{B3}	P_{T3}	. . .	P_{TW3}
Person 4	P_{Q4}	P_{B4}	P_{T4}	. . .	P_{TW4}
.	.	.	.		.
.	.	.	.		.
.	,	.	.		.
Person n	P_{Qn}	P_{Bn}	P_{Tn}	. . .	P_{TWn}

2.5.2 Creating the final training data set

For our training set we have added a result column to training our own system. For that we have worked with some parameters.
According to Soukupova [13] the eye blink increases when a suspicious person answers the questions. Using this we have taken 5 blinks per questions as our parameter. On the other hand, the response time increases if a suspicious person answers the questions [13]. Basing on this, we have taken our response time of 5 seconds as parameters. For the question answering part, the value of suspicious answer was 0.15, value of neutral answer was 0.20 and the value of non- suspicious answer was 0.65. Therefore, all the parameters are shown below:

Case1:

Parameter of P_Q :

If 0.20 < PQ < 0.65= non-suspicious

If 0.15 < PQ <0.20 = suspicious

Parameter of P_B :

If PB > 0.5 =suspicious

If PB < 0.5 = non-suspicious

If PB >= 1 ; then we have added the value of blink to 0.9

Parameter for P_T :

If, PT > 0.5 = suspicious

If, PT < 0.5 = non-suspicious.

Here, if PT >= 1 then the value of was converted to 0.9.

On the other hand, twitter data was converted to a float number.

If, PTW =positive then PTW = 6

If, PTW=negative then PTW = 3

To make the data analysis easier, we have multiplied all the values from 2.1, 2.2, 2.3, 2.4 with 1/10 Using this parameter we have added our result column. The following diagram shows the process of calculating result.

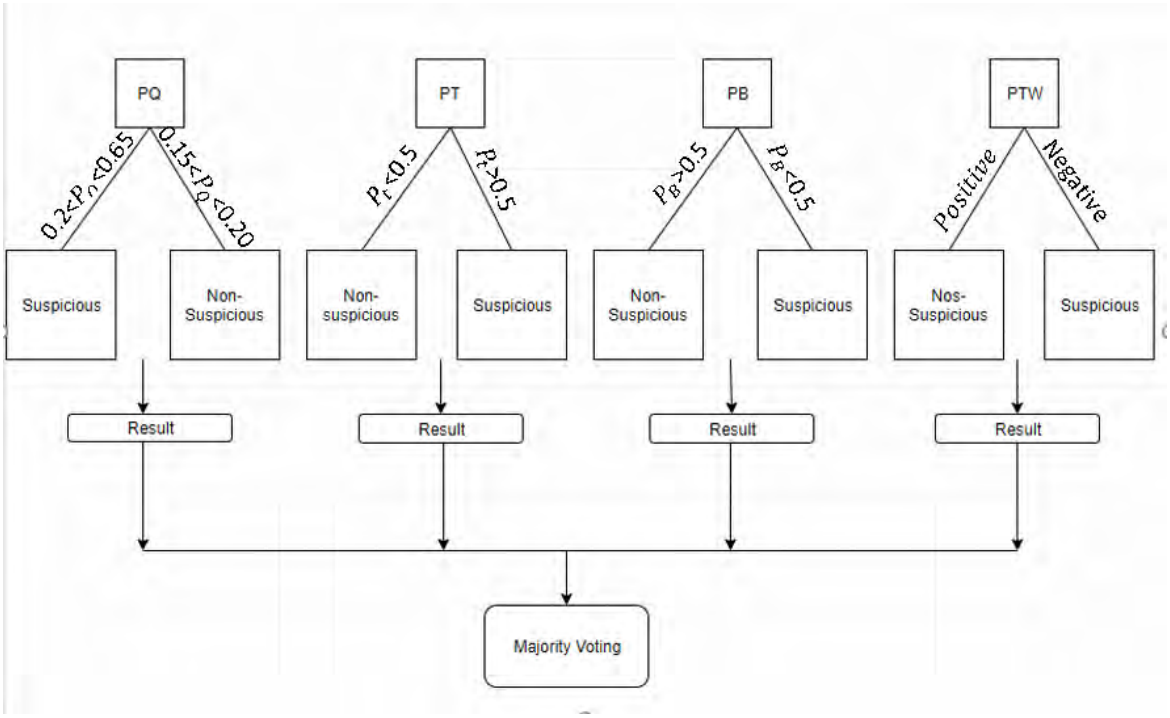

Fig. 2.9 Model of decision making process

In this part, we have used majority voting to determine the result. If PQ, PB, PT, PTW gives suspicious value then the final result will be suspicious. If three among the four attributes (PQ, PB, PT, PTW) is non-suspicious, then the final result will be non-suspicious. We have added the suspicious value =1 and the non-suspicious value =0 in the training data set. Our training data set is given below:

Table 2.3 Data set for Decision Making Process

	Ques(PQ)	Eye(PB)	Time(PT)	Twitter(PTW)	Result
P1	0.17	0.57	0.33	0.3	0
P2	0.19	0.84	0.37	0.6	0
P3	0.58	0.35	0.58	0.6	0
P4	0.35	0.6	0.64	0.3	0
P5	0.49	0.31	0.31	0.3	1
P6	0.22	0.42	0.84	0.1	1
P7	0.18	0.57	0.61	0.6	0
P8	0.24	0.87	0.84	0.1	0
P9	0.59	0.76	0.68	0.1	1
P10	0.22	0.55	0.61	0.3	1
P11	0.31	0.41	0.34	0.6	0
P12	0.55	0.36	0.46	0.6	1
P13	0.15	0.61	0.74	0.3	0
P14	0.19	0.42	0.49	0.3	0
P15	0.46	0.55	0.57	0.1	1
P16	0.35	0.73	0.61	0.3	0
P17	0.21	0.83	0.34	0.6	0
P18	0.51	0.61	0.67	0.6	0
P19	0.53	0.81	0.76	0.3	1
P20	0.46	0.59	0.9	0.6	1
P21	0.27	0.41	0.65	0.3	0
P22	0.37	0.9	0.31	0.1	0
P23	.22	0.33	0.45	0.6	0
P24	0.55	0.61	0.36	0.3	0
P25	0.31	0.44	0.38	0.3	1
P26	0.37	0.82	0.65	0.6	1
P27	0.15	0.57	0.56	0.1	0
P28	0.19	0.46	0.68	0.6	0
P29	0.38	0.39	0.85	0.6	0
P30	0.19	0.31	0.78	0.6	0
P31	0.37	0.46	0.75	0.3	0
P32	0.27	0.39	0.78	0.3	0
P33	0.31	0.49	0.57	0.1	1
P34	0.37	0.43	0.34	0.6	0
P35	0.41	0.34	0.79	0.3	0
P36	0.33	0.84	0.71	0.6	0
P37	0.21	0.56	0.73	0.1	1
P38	0.41	0.48	0.38	0.3	0
P39	0.57	0.97	0.46	0.6	1
P40	0.28	0.34	0.57	0.1	0
P41	0.36	0.67	0.49	0.1	1
P42	0.48	0.68	0.67	0.1	1
P43	0.23	0.9	0.61	0.3	1

Table 2.4 Data set for Decision Making Process

	Ques(PQ)	Eye(PB)	Time(PT)	Twitter(PTW)	Result
P44	0.36	0.34	0.78	0.3	0
P45	0.38	0.35	0.62	0.3	0
P46	0.35	0.64	0.66	0.6	0
P47	0.64	0.84	0.33	0.6	1
P48	0.35	0.79	0.46	0.1	1
P49	0.47	0.68	0.45	0.3	1
P50	0.53	0.37	0.68	0.1	1
P51	0.23	0.89	0.52	0.3	0
P52	0.15	0.39	0.75	0.6	0
P53	0.47	0.56	0.54	0.3	1
P54	0.4	0.35	0.61	0.1	1
P55	0.29	0.46	0.88	0.3	1
P56	0.55	0.63	0.9	0.6	1
P57	0.47	0.36	0.33	0.1	1
P58	0.29	0.31	0.34	0.3	0
P59	0.23	0.32	0.45	0.1	0
P60	0.57	0.34	0.67	0.6	0
P61	0.43	0.62	0.6	0.1	1
P62	0.19	0.61	0.54	0.3	0
P63	0.35	0.81	0.61	0.1	1
P64	0.44	0.94	0.4	0.6	0
P65	0.47	0.92	0.54	0.6	0
P66	0.27	0.81	0.32	0.3	0
P67	0.19	0.78	0.69	0.3	0
P68	0.29	0.9	0.49	0.1	1
P69	0.4	0.71	0.47	0.6	0
P70	0.53	0.41	0.54	0.1	1
P71	0.59	0.84	0.61	0.3	1
P72	0.23	0.76	0.54	0.1	1
P73	0.25	0.84	0.72	0.3	0
P74	0.58	0.9	0.49	0.6	1
P75	0.39	0.64	0.9	0.1	1
P76	0.29	0.34	0.36	0.3	0
P77	0.19	0.94	0.54	0.6	0
P78	0.38	0.87	0.57	0.3	0
P79	0.55	0.62	0.64	0.1	1
P80	0.51	0.31	0.6	0.3	0
P81	0.2	0.54	0.74	0.3	0
P82	0.41	0.39	0.84	0.6	0
P83	0.31	0.49	0.59	0.3	0
P84	0.41	058	0.71	0.1	1
P85	0.29	0.67	0.7	0.3	0

Table 2.5 Data set for Decision Making Process

	Ques(PQ)	Eye(PB)	Time(PT)	Twitter(PTW)	Result
P86	0.19	0.31	0.41	0.6	0
P87	0.49	0.32	0.51	0.3	0
P88	0.47	0.33	062	0.1	0
P89	0.48	0.55	0.75	0.6	1
P90	0.16	0.64	0.81	0.3	0
P91	0.33	0.35	0.82	0.1	1
P92	0.44	0.61	0.76	0.3	1
P93	0.51	0.71	0.47	0.3	0
P94	0.16	0.39	0.38	0.6	0
P95	0.6	0.45	0.39	0.3	0
P96	0.52	0.81	0.49	0.1	1
P97	0.42	0.82	0.54	0.6	1
P98	0.23	0.9	0.55	0.3	0
P99	0.29	0.87	0.78	0.3	0
P100	0.47	0.31	0.79	0.1	1

Chapter 3

Result Analysis

3.1 Result Analysis

After making the training data-set we used machine learning to evaluate the data. For that we have used Random Forest Algorithm, SVM model, logistic regression model and decision tree algorithms.
Random forest is an ensemble learning method which is very suitable for supervised learning such as classification and regression. In random forest we divided train set to smaller part and make each small part as independent tree which its result has no effect on other trees besides them. Random forest gets training set and divided it by "Bagging = Bootstrap Aggregating" which is algorithm to increase accuracy by prevent over fitting and decrease variance. It starts to divide data set to 60% as unique decision tree and 30% as overlapping data. It also solves the problem of data over fitting and under fitting. On the other hand, it is very well known for its accuracy.
SVM's are great when we have no clue on the information. Functions admirably with even unstructured and semi organized information like content, Images and trees. The bit trap is genuine quality of SVM. With a suitable portion work, we can take care of any unpredictable issue. Dissimilar to in neural systems, SVM isn't understood for neighborhood optima. It scales moderately well to high dimensional information. SVM models have speculation by and by, the danger of over fitting is less in SVM.
Logistic regression model is more robust: the independent variables don't have to be normally distributed, or have equal variance in each group. It may handle nonlinear effects. Normally in Logistic regression model distributed error terms are not assumed .It does not require that the independents be interval and It does not require that the independents be unbounded.

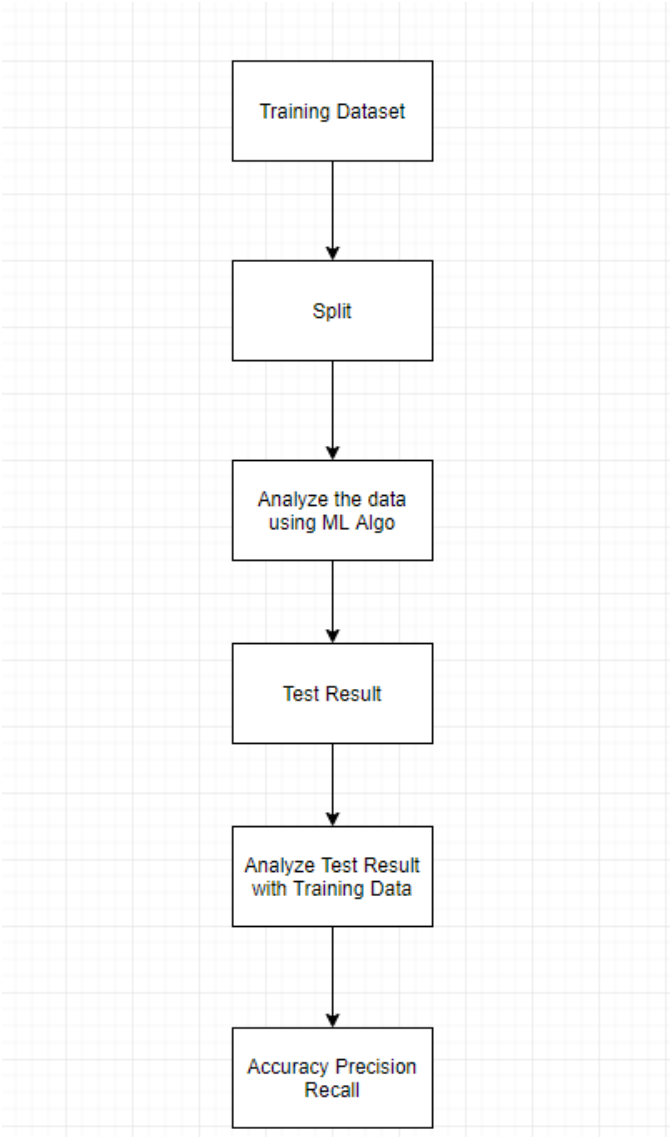

Fig. 3.1 Model of data analysis

3.1.1 Data Analysis

After training our system with the training dataset, we have tested it. To do that, we have split our data set into two parts. The values of PQ, PB, PT and PTW will be the testing part and the system will give a new result using the algorithm and training the data. Then it will compare with the previous result and give the value of accuracy, precision and recall. The whole process is showed in the figure 3.1.

3.1.2 Implementation

To implement our system and train our system, we have followed the steps below figure 3.2:

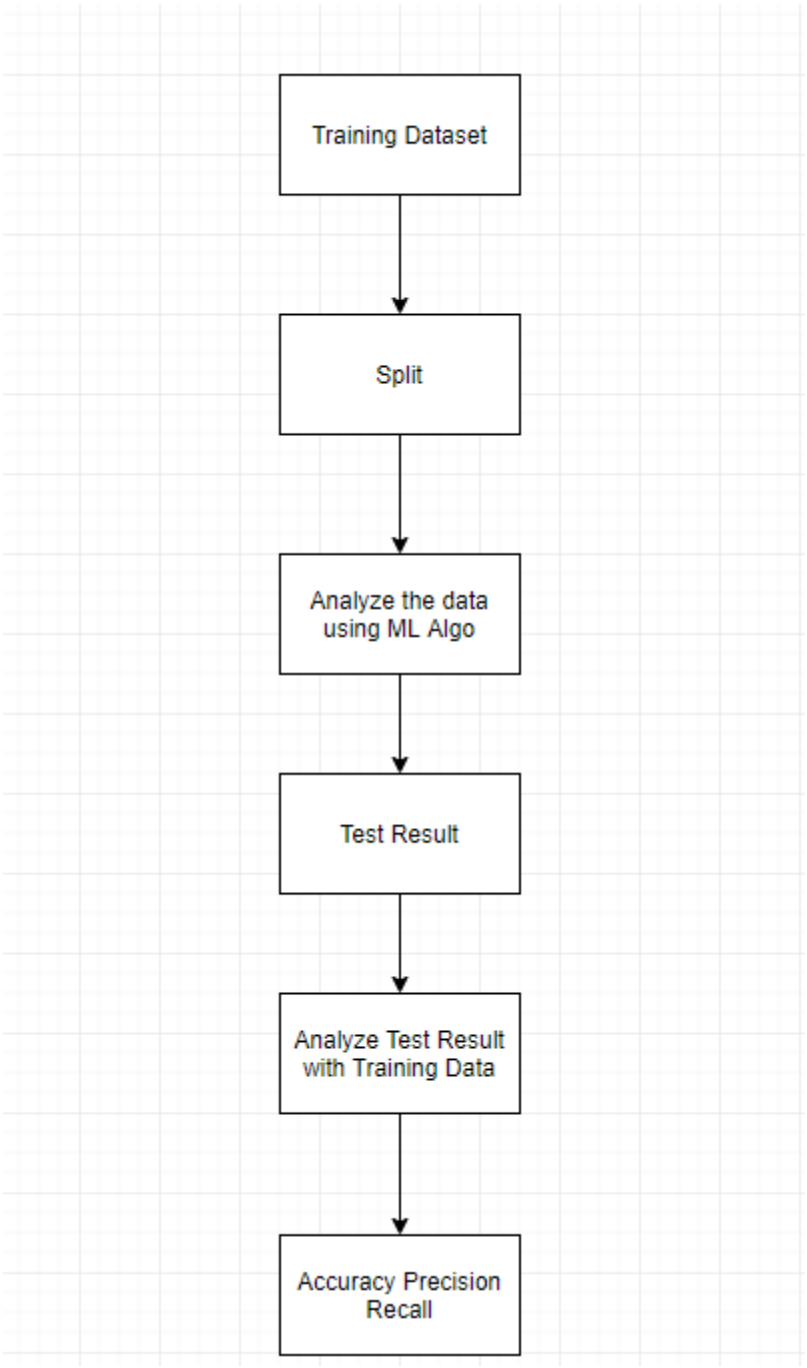

Fig. 3.2 Flowchart of the system and training system

Firstly, we have used different packages such as pandas, numpy, scikit learn. Pandas is a software library written in python programming language for data manipulation and analysis. It helos manipulating numerical tables. Numpy is a fundamental package for scientific computing with python. Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data. Scikit learn is a machine learning library for python for python programming language. It features various classifications, regression and clustering algorithm including support vector machines, random forest alogorithm, gradiant boosting, k means etc. In order to improve our system using python and anaconda, first we have to import various packages such as numpy,, pandas scikit learn. Then we have to load our training data set using pandas.
After that we have implemented sklearns cross validation function. This function splits the training data set into two parts. For our system , it separates the result section from the others for testing purposes. Then we have trained our system using the whole dataset. After the completion of training we have tested our data set using different algorithms.

Decision Tree Model:
For decision tree model we have used the decision tree classifier which is built in function in Scikit learn. Decision Tree Algorithm works in three parts: The best attribute of the dataset at the root of the tree.

- **Information Gain process:**

1. infoGain(examples, attribute, entropyOfSet)
2. gain = entropyOfSet
3. for value in attributeValues(examples, attribute):
4. sub = subset(examples, attribute, value)
5. gain = (number in sub)/(total number of examples) * entropy(sub)
6. return gain

- **Entropy pseudo code**

1. entropy(examples)
2. log2(x) = log(x)/log(2)
3. result = 0

4. handle target attributes with arbitrary labels
5. dictionary = summarizeExamples(examples, targetAttribute)
6. for key in dictionary:
7. proportion = dictionary[key]/total number of examples
8. result -= proportion * log2(proportion)
9. return result

- **After testing the data we get the following result:**

Accuracy 0.59

Precision 0.22

Recall 0.25

Random Forest

Precondition: A training set S := (x1, y1), . . . ,(xn, yn), features F, and number of trees in forest B.

1. function RandomForest(S , F)
2. H ←
3. for i 1, . . . , B do
4. S (i) ← A bootstrap sample from S
5. hi ← RandomizedTreeLearn(S (i) , F)
6. H ← H hi
7. end for
8. return H
9. end function
10. function RandomizedTreeLearn(S , F)
11. At each node:
12. f ← very small subset of F
13. Split on best feature in f
14. return The learned tree

15. end function

Result type:

Accuracy 0.61

Precision 0.26

Recall 0.26

Logistic Regression:

Octave code:

% X is original m x n matrix

a = ones(n, 1) % initial value for parameter vector

X = studentize(X) % normalize X

X = [ones(m, 1) X] % prepend all 1s column

for t = 1:100 % repeat 100 times

D = X*a - y a = a - alpha / m * X' * D % we store consecutive values of J over time t

J(t) = 1/2/m * D' * DResult[type]

Analysing the final outcome:
After training our system using the training data set, we have tested our system. The result of our system with four different algorithms. The results are given below:

Table 3.1 Accuracy from different algorithms

	Random Forest	Decision Tree	SVM	Logistic Regression
Accuracy	0.61	0.59	0.666	0.668
Precision	0.26	0.22	0.163	0.345
Recall	0.26	0.25	0.059	0.146

Here, accuracy shows how much accurate our test and training data is. High precision means that an algorithm returned substantially more relevant results than irrelevant ones, while high recall means that an algorithm returned most of the relevant results. Analyzing the result, we can clearly see Random Forest Algorithm, SVM and Logistic regression has the highest accuracy. On the other hand, Decision tree has fallen far behind in terms of accuracy. The Precision and recall of random Forest Algorithm is better other algorithms. Here, our system has measured new result against the result portion of our training data set. As we have worked on only 100 data so our accuracy was low. If we provide the system with any dataset it will give prediction according to the training data set, so in order to fulfill our goal of personnel security we can analyze our system with dataset for single person and it will give the prediction for the given dataset. After that we can measure the probability of person being suspicious or non-suspicious.

Chapter 4

Conclusion

4.1 Conclusion

The major step behind our research was to secure arrangements against different dangers, rather than undertaking security steps. Using a special personnel security model for nuclear power plants, we proposed and demonstrated a preliminary model consists of questionnaires, data acquisition through image processing and sentiment analysis subsystems and then used all the three idea to analyze data to get a result about a person to predict his emotional status. We have taken ten people to collect these four data for our research. First of all, the question set for the questionnaire part was generated by consulting only one psychiatrist. This part needs more opinions from many other professional psychiatrists who has better knowledge about criminal psychology as the output of this part plays a great role to detect negativity in a person. Again in our sentiment analysis section, due to lack of funding we chose to do web scraping rather than going with the twitter API. Twitter API could get us more authentic data direct from the twitter database. Moreover, we have worked on a small data set so our accuracy was not that high. The reason behind this was there was a lack of resource to generate an enriched data set. A large data set will able to train the system in a way which will increase our accuracy. If the accuracy level gets higher, it will be easier to predict a suspicious person more accurately and thus the authority of nuclear power plants can further take initial steps against that particular person.

4.2 Future work

The future focus of our work is to train the system with a bigger data set. The bigger the data set will be, the more perfectly will the machine be trained. In that case, the final output of the system will be higher too. We are trying to get enough funding to get high resolution cameras, to use the Twitter API etc. We also have plans to monitor employee's pulse and sweat rate during the questionnaires' to get more validate results in future. As a person who is nervous while telling lie tend to have higher pulse rate than the people who are telling the truth. Our system also has plans to secure the nuclear atomic plant field by monitoring employee's search histories, keeping track of their past life behavior and also monitor people added on their social medias. In case of work place security by monitoring daily video footage of the office to identify any kind of unusual activity of any employee on daily basis. Finally, our intention is to introduce this system not only in nuclear atomic field which requires high-security environment but also in crime investigating departments to interrogate people who are accused for crime.

References

[1] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., and Passonneau, R. (2011). Sentiment analysis of twitter data. In *Proceedings of the workshop on languages in social media*, pages 30–38. Association for Computational Linguistics.

[2] Barbosa, L. and Feng, J. (2010). Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd international conference on computational linguistics: posters*, pages 36–44. Association for Computational Linguistics.

[3] Desai, M. and Mehta, M. A. (2016). Techniques for sentiment analysis of twitter data: A comprehensive survey. In *Computing, Communication and Automation (ICCCA), 2016 International Conference on*, pages 149–154. IEEE.

[4] Guide, R. (2010). 5.71. *Cyber Security Programs for Nuclear Facilities, US Nuclear Regulatory Commission*.

[5] Kumar, A. and Sebastian, T. M. (2012). Sentiment analysis on twitter. *International Journal of Computer Science Issues (IJCSI)*, 9(4):372.

[6] Leal, S. and Vrij, A. (2008). Blinking during and after lying. *Journal of Nonverbal Behavior*, 32(4):187–194.

[7] Liu, Z. and Ai, H. (2008). Automatic eye state recognition and closed-eye photo correction. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4. IEEE.

[8] Nasri, H., Ouarda, W., and Alimi, A. M. (2016). Relidss: Novel lie detection system from speech signal. In *Computer Systems and Applications (AICCSA), 2016 IEEE/ACS 13th International Conference of*, pages 1–8. IEEE.

[9] Pimplaskar, D., Nagmode, M., and Borkar, A. (2015). Real time eye blinking detection and tracking using opencv. *technology*, 13(14):15.

[10] Remeseiro, B., Fernández, A., and Lira, M. (2015). Automatic eye blink detection using consumer web cameras. In *International Work-Conference on Artificial Neural Networks*, pages 103–114. Springer.

[11] Sheridan, M. R. and Flowers, K. A. (2010). Reaction times and deception-the lying constant. *International Journal of Psychological Studies*, 2(2):41.

[12] Sorensen, J. (2002). Safety culture: a survey of the state-of-the-art. *Reliability Engineering & System Safety*, 76(2):189–204.

[13] Soukupová, T. (2016). Eye blink detection using facial landmarks.

[14] Vrij, A., Edward, K., Roberts, K. P., and Bull, R. (2000). Detecting deceit via analysis of verbal and nonverbal behavior. *Journal of Nonverbal behavior*, 24(4):239–263.

Appendix A

Apendix

Table A.1 Sample s of positive and negative words

Positive Words	Negative Words
	2-faced
a+	2-faces
abound	abnormal
abounds	abolish
abundance	abominable
abundant	abominably
accessible	abominate
accessable	abomination
acclaim	abort
acclaimed	aborted
acclamation	aborts
accolade	abrade
accolades	abrasive
accommodate	abrupt
accomodative	abruptly
accomplish	abscond
accomplished	absence
accomplishment	absent-minded
accomplishments	absentee
accurate	absurd
accurately	absurdity
achievable	absurdly
achievement	absurdness
achievements	abuse
achievable	abused
acumen	abuses
adaptable	abusive
adaptive	abysmal
adequate	abysmally
adjustable	abyss
admirable	accidental
admirably	accost
admiration	accursed
admire	accusation
admirer	accusations
admiring	accuse
admiringly	accuses
adorable	accusing
agreeably	aggravation

Table A.2 Sample s of positive and negative words

Positive Words	Negative Words
adore	accusingly
adored	acerbate
adorer	acerbic
adoring	acerbically
adoringly	ache
adroit	ached
adroitly	aches
adulate	achy
adulation	aching
adulatory	acrid
advanced	acridly
advantage	acridness
advantageous	acrimonious
advantageously	acrimoniously
advantages	acrimony
adventuresome	adamant
adventurous	adamantly
advocate	addict
advocated	addicted
advocates	addicting
affability	addicts
affable	admonish
affably	admonisher
affectation	admonishingly
affection	admonishment
affectionate	admonition
affinity	adulterate
affirm	adulterated
affirmation	adulteration
affirmative	adulterer
affluence	adversarial
affluent	adversary
afford	adverse
affordable	adversity
affordably	afflict
affordable	affliction
agile	afflictive
agilely	affront
agility	afraid
agreeable	aggravate
agreeableness	aggravating
all-around	aggression
alluring	aggressive