**CHAPTER-1**

**INTRODUCTION**

There are certain types of harassment and Violence that are very aggressive including staring and passing comments and these unacceptable practices are usually seen as a normal part of the urban life. There have been several studies that have been conducted in cities across India and women report similar type of sexual harassment and passing off comments by other unknown people. The study that was conducted across most popular Metropolitan cities of India including Delhi, Mumbai and Pune, it was shown that 60 % of the women feel unsafe while going out to work or while travelling in public transport.

Women have the right to the city which means that they can go freely whenever they want whether it be too an Educational Institute, or any other place women want to go. But women feel that they are unsafe in places like malls, shopping malls on their way to their job location because of the several unknown Eyes body shaming and harassing these women point Safety or lack of concrete consequences in the life of women is the main reason of harassment of girls. There are instances when the harassment of girls was done by their neighbours while they were on the way to school or there was a lack of safety that created a sense of fear in the minds of small girls who throughout their lifetime suffer due to that one instance that happened in their lives where they were forced to do something unacceptable or was sexually harassed by one of their own neighbor or any other unknown person.

Safest cities approach women safety from a perspective of women rights to the affect the city without fear of violence or sexual harassment. Rather than imposing restrictions on women that society usually imposes it is the duty of society to imprecise the need of protection of women and also recognizes that women and girls also have a right same as men have to be safe in the City.

Analysis of twitter texts collection also includes the name of people and name of women who stand up against sexual harassment and unethical behaviour of men in Indian cities which make them uncomfortable to walk freely. The data set that was obtained through Twitter about the status of women safety in Indian society was for the processed through machine learning algorithms for the purpose of smoothening the data by removing zero values and using Laplace and porter’s theory is to developer method of analyzation of data and remove retweet and redundant data from the data set that is obtained so that a clear and original view of safety status of women in Indian society is obtained.

**1.1 Twitter Analysis**

A huge number of clients utilize online media like tweeter to communicate their feelings, assumptions, sentiments, and suppositions for the world to peruse. These tweets can be effectively extricated and can be exposed to an extremity trial of the expressions utilizing profound figuring out how to decide the rating of lady's wellbeing specifically area. A large number of clients utilize web-based media like tweeter to communicate their feelings, slants, sentiments, and assessments for the world to peruse.

These tweets can be handily removed and can be exposed to an extremity trial of the expressions utilizing profound figuring out how to decide the rating of lady's security specifically territory. We carried out API from tweeter to gather every one of the tweets on tweeter which were labelled under lady harassment or lady wellbeing hashtags or contained words like lady harassments and lady security. In the wake of gathering all the tweets, all tweets considered are isolated in datasets which were then prepared for polarities and isolated by our calculation.

**1.2 Sentimental Analysis**

Sentiment analysis is the process of extracting the sentiment behind any sentence or statement. It can be called as a classification technique which is used to obtain the opinion from tweet. This opinion is useful in formulating a sentiment which can further be used to achieve sentiment classification. Sentiments are personal to the topic and thus we need to decide what kind of specifications is formulated out of it. Person performing the sentimental analysis wants to find the class of entities of the tweets using the programming model. The dimension of the sentimental class is an important factor in order to decide the efficiency of the algorithm. For instance, there can be two class sentimental classification of tweets – Positive and Negative or there can be three class classification – Positive, Negative and Neutral.Approaches of sentimental analysis can be broadly differentiated into two types – machine learning based and lexicon learning based. Machine learning approach includes the process of extraction of features, programming model training using dataset of features. Whereas lexicon learning based approach uses the vocabulary and scoring method to detect opinions. In this paper, we use machine learning approach. Collection of data, pre-processing the data, extraction of features, choosing base features, detection of sentiments and classification of sentiments using machine learning approaches or simple computations are the basic steps to perform sentimental analysis.

**1.3 Motivation**:

Now-a-days almost all peoples are using social networking sites to express their feelings and if any women feel unsafe in any area then she will express negative words in her post/tweets/messages and by analysing those messages we can detect which area is more unsafe for women’s.

**1.4 Problem Statement:**

In propose paper author is analysing social media tweets to detect women’s safety but the main problem is social media can be used by malicious users who will give fake ratings to worst products to boom their sale or can write fake tweets to raise finger at country safety towards Indian women.

**1.5 Objective:**

Women and girls have been experiencing a lot of violence and harassment in public places in various cities starting from stalking and leading to sexual harassment or sexual assault. This research paper basically focuses on the role of social media in promoting the safety of women in Indian cities with special reference to the role of social media websites and applications including Twitter platform Facebook and Instagram.

**1.6 Scope of work:**

There are certain types of harassment and Violence that are very aggressive including staring and passing comments and these unacceptable practices are usually seen as a normal part of the urban life. There have been several studies that have been conducted in cities across India and women report similar type of sexual harassment and passing off comments by other unknown people. The study that was conducted across most popular Metropolitan cities of India including Delhi, Mumbai and Pune, it was shown that 60 % of the women feel unsafe while going out to work or while travelling in public transport.

**1.7 Applications:**

Women Apps

**1.8 Organization of Report:**

The rest of the report is organized into 5 chapters. After this introductory chapter, the next chapter-2 describes about the survey of the existing system. This establishes a context of the research conducted by the researchers up until now in the field of Analysis of Women Safety using **Sentimental analysis**

Chapter-3 describes the proposed system. This starts with the introduction of the dataset, the models that have been used in the report. Then it covers the architecture of the proposed system. Describes the process and the algorithms used, the details of the software used for the research work. It also describes the evaluation parameters used for this study.

Chapter-4 shows the experiment and the results. It appears the confusion network of each model and the comparison graph. This helps us to identify which model is the most efficient for the stock market trend prediction using ML and DL algorithms. .

Chapter-5 gives a conclusion about the result of all the models in this research paper and gives suggestions about which model to use when. It gives a new direction of future work.

**CHAPTER-2**

**LITERATURE SURVEY**

**2.1 Survey Of Existing System:**

**1. Sentiment analysis of top colleges in India using Twitter data:**

In today's world, opinions and reviews accessible to us are one of the most critical factors in formulating our views and influencing the success of a brand, product or service. With the advent and growth of social media in the world, stakeholders often take to expressing their opinions on popular social media, namely Twitter. While Twitter data is extremely informative, it presents a challenge for analysis because of its humongous and disorganized nature. This paper is a thorough effort to dive into the novel domain of performing sentiment analysis of people's opinions regarding top colleges in India. Besides taking additional preprocessing measures like the expansion of net lingo and removal of duplicate tweets, a probabilistic model based on Bayes' theorem was used for spelling correction, which is overlooked in other research studies. This paper also highlights a comparison between the results obtained by exploiting the following machine learning algorithms: Naïve Bayes and Support Vector Machine and an Artificial Neural Network model: Multilayer Perceptron. Furthermore, a contrast has been presented between four different kernels of SVM: RBF, linear, polynomial and sigmoid.

**2. Study of Twitter sentiment analysis using machine learning algorithms on Python**

Twitter is a platform widely used by people to express their opinions and display sentiments on different occasions. Sentiment analysis is an approach to analyze data and retrieve sentiment that it embodies. Twitter sentiment analysis is an application of sentiment analysis on data from Twitter (tweets), in order to extract sentiments conveyed by the user. In the past decades, the research in this field has consistently grown. The reason behind this is the challenging format of the tweets which makes the processing difficult. The tweet format is very small which generates a whole new dimension of problems like use of slang, abbreviations etc. In this paper, we aim to review some papers regarding research in sentiment analysis on Twitter, describing the methodologies adopted and models applied, along with describing a generalized Python based approach.

**3. Determining the sentiment of opinions**

Identifying sentiments (the affective parts of opinions) is a challenging problem. We present a system that, given a topic, automatically finds the people who hold opinions about that topic and the sentiment of each opinion. The system contains a module for determining word sentiment and another for combining sentiments within a sentence. We experiment with various models of classifying and combining sentiment at word and sentence levels, with promising results.

**4. sentiment detection on twitter from biased and noisy data**

In this paper, we propose an approach to automatically detect sentiments on Twitter messages (tweets) that explores some characteristics of how tweets are written and meta-information of the words that compose these messages. Moreover, we leverage sources of noisy labels as our training data. These noisy labels were provided by a few sentiment detection websites over twitter data. In our experiments, we show that since our features are able to capture a more abstract representation of tweets, our solution is more effective than previous ones and also more robust regarding biased and noisy data, which is the kind of data provided by these sources.

**5. Classifying sentiment in microblogs: is brevity an advantage.**

Microblogs as a new textual domain offer a unique proposition for sentiment analysis. Their short document length suggests any sentiment they contain is compact and explicit. However, this short length coupled with their noisy nature can pose difficulties for standard machine learning document representations. In this work we examine the hypothesis that it is easier to classify the sentiment in these short form documents than in longer form documents. Surprisingly, we find classifying sentiment in microblogs easier than in blogs and make a number of observations pertaining to the challenge of supervised learning for sentiment analysis in microblogs.

**6. Sentiment analysis on twitter data**

We examine sentiment analysis on Twitter data. The contributions of this paper are: (1) We introduce POS-specific prior polarity features. (2) We explore the use of a tree kernel to obviate the need for tedious feature engineering. The new features (in conjunction with previously proposed features) and the tree kernel perform approximately at the same level, both outperforming the state-of-the-art baseline.

**7. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking**

Discriminative reranking is one method for constructing high-performance statistical parsers (Collins, 2000). A discriminative reranker requires a source of candidate parses for each sentence. This paper describes a simple yet novel method for constructing sets of 50-best parses based on a coarse-to-fine generative parser (Charniak, 2000). This method generates 50-best lists that are of substantially higher quality than previously obtainable. We used these parses as the input to a MaxEnt reranker (Johnson et al., 1999; Riezler et al., 2002) that selects the best parse from the set of parses for each sentence, obtaining an f-score of 91.0% on sentences of length 100 or less.

**2.2 Research Contribution:**

In propose work author using TWEEPY package from python to download tweets from twitter but every time INTERNET will not available to download tweets online so we downloaded MEETOO tweets on women safety and safe inside dataset folder. Application will read this tweets to detect women’s sentiments.

Author using NLTK (natural language tool kit) to remove special symbols and stop words from tweets and to make them clean.

Author using TEXTBLOB corpora package and dictionary to count positive, negative and neutral polarity and the tweets which has polarity value less than 0 will consider as negative as and greater than 0 and less than 0.5 will consider as neutral and polarity greater than 0.5 will consider as positive.

**CHAPTER-3**

**PROPOSED SYSTEM**

This research paper basically focuses on the role of social media in promoting the safety of women in Indian cities with special reference to the role of social media websites and applications including Twitter platform Facebook and Instagram.

In propose work author using TWEEPY package from python to download tweets from twitter but every time INTERNET will not available to download tweets online so we downloaded MEETOO tweets on women safety and safe inside dataset folder. Application will read the tweets to detect women’s sentiments.

Author using NLTK (natural language tool kit) to remove special symbols and stop words from tweets and to make them clean

**3.1 Algorithms:**

In this study, we use **Sentimental analysis,** TFIDF (term frequency inverse frequency document) algorithm, and Decision Tree algorithm

**3.1.1 Sentimental analysis**

**Step 1:** Start

**Step 2:** load the dataset

**Step 3:** Clean tweets by removing the stop words and noise like repetitive letters.

**Step 4:** For each word, compare it with positive sentiments and negative sentiments word dictionary and then increment positive count or negative count of the overall phrase.

**Step 5:** Finally, based on the positive count & negative count, we can get result percentage about sentiment to decide the polarity which is categorized in Positive, Negative and Neutral.

**3.1.2 TF-IDF (term frequency-inverse document frequency)**

It is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. This is done by multiplying two metrics: how many times a word appears in a document, and the inverse document frequency of the word across a set of documents. Which will convert all tweets words into numeric vector which will contains average frequency of each tweet words.

**3.1.3 Decision Tree algorithm**

The purpose is to make a model which is able to predict a target value by learning easy decision rules formed from the data features. There are some advantages of using this method like being easy to interpret and understand or Able to work out problems with multi-outputs. Decision Tree is a common supervised learning approach employed for both regression and classification problems. The goal of technique is forecasting a target by using easy decision rules shaped from the dataset and related features. Being easy to interpret or able to solve problems with different outputs are two advantages of using this model; on the contrary, constructing over-complex trees that cause overfitting is a typical disadvantage.

**3.2 Requirement Specifications:**

**HARDWARE REQUIREMENTS:**

System : Pentium i5.

Hard Disk : 500 GB.

Monitor : 15 LED

Input Devices : Keyboard, Mouse

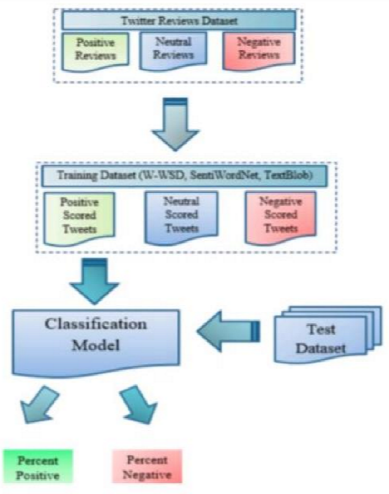
RAM : 4 GB

**SOFTWARE REQUIREMENTS:**

Operating system : Windows 10.

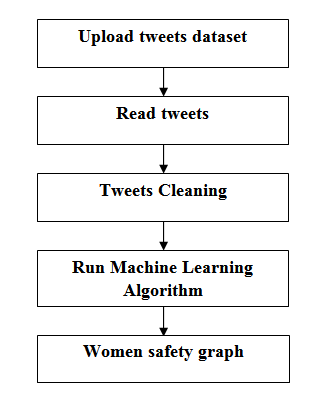
Coding Language : Python 3.7.0

**3.3 Architecture/Framework**:



**Fig 1: Architecture Diagram**

**3.4 Algorithm and Process Design:**

****

**Fig 2:Process Diagram**

**Upload tweets dataset:** using this module we will upload tweets dataset

**Read tweets:** using this module we will see all the tweets in dataset and we can see all tweets contains special symbols and stop words.

**Tweets Cleaning:** Using this module we will remove or clean the special symbols and stop words.

**Run Machine Learning Algorithm:** using this module we will see tweet text and then displaying tweets sentiments with polarity score

**Women safety graph:** using this module we will see the result in graph

**3.5 Functional Requirements:**

Functional requirements are product features that developers must implement to enable the users to achieve their goals. They define the basic system behavior under specific conditions.

**Packages:**

● **tkinter:** tkinter is the standard GUI library for Python. tkinter module helps in creating GUI applications in a fast and easy way.

● **Matplotlib:** Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib consists of several plots like line, bar, scatter etc.

● **Numpy:** NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices.

● **Pandas:** Pandas is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively it is used for Fast and efficient for manipulating and analyzing data.

**Dataset:**

To implement this project we are using Tweets dataset called MeToo\_tweets dataset.

The datasets is download from

<https://www.kaggle.com>

**3.6 Non-Functional Requirements:**

A non-functional requirement defines are the performance attribute of a software system. Types of non-functional requirements are Scalability Capacity, Availability, Reliability, Recoverability, Data Integrity, etc.

**CHAPTER-4**

**SYSTEM DESIGN**

System design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. System design could be seen as the application of systems theory to product development. System design is therefore the process of defining and developing system to specify requirements of the user.

System Analysis is the process that decomposes a system into its component pieces for the purpose of defining how well those components interact to accomplish the set requirements.

The purpose of the System Design Process is to provide sufficient detailed data and information about the system and its system element to enable the implementation consistent with architectural entities as defined in models and views of the system architecture.

**System Design Process:**

1. Initialize design definition
2. Establish design characteristics
3. Assess alternatives for obtaining system elements
4. Manage the design

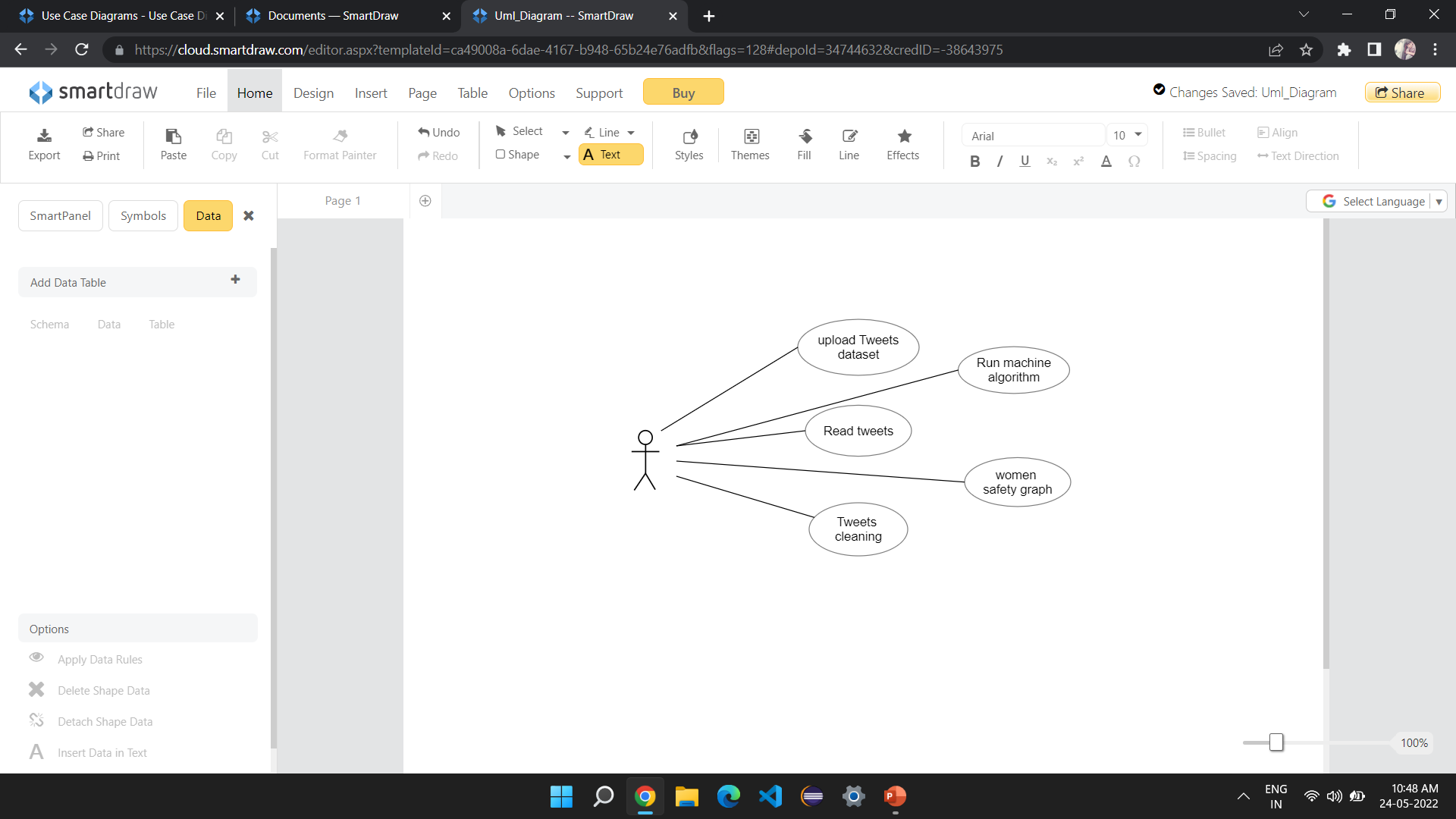
**4.1 UML DIAGRAMS**

Unified Modeling Language (UML) is a general purpose modeling language**.** The main aim of UML is to define a standard way to visualize the way a system has been designed. It is quite similar to blueprints used in other fields of engineering.

UML is not a programming language, it is rather a visual language. We use UML diagram to portray the behavior and structure of a system. UML helps software engineers, businessmen and system architects with modeling, design and analysis.

UML is linked with object oriented design and analysis. UML makes the use of elements and forms association between them to form diagrams. Diagram is UML can be broadly classified as:

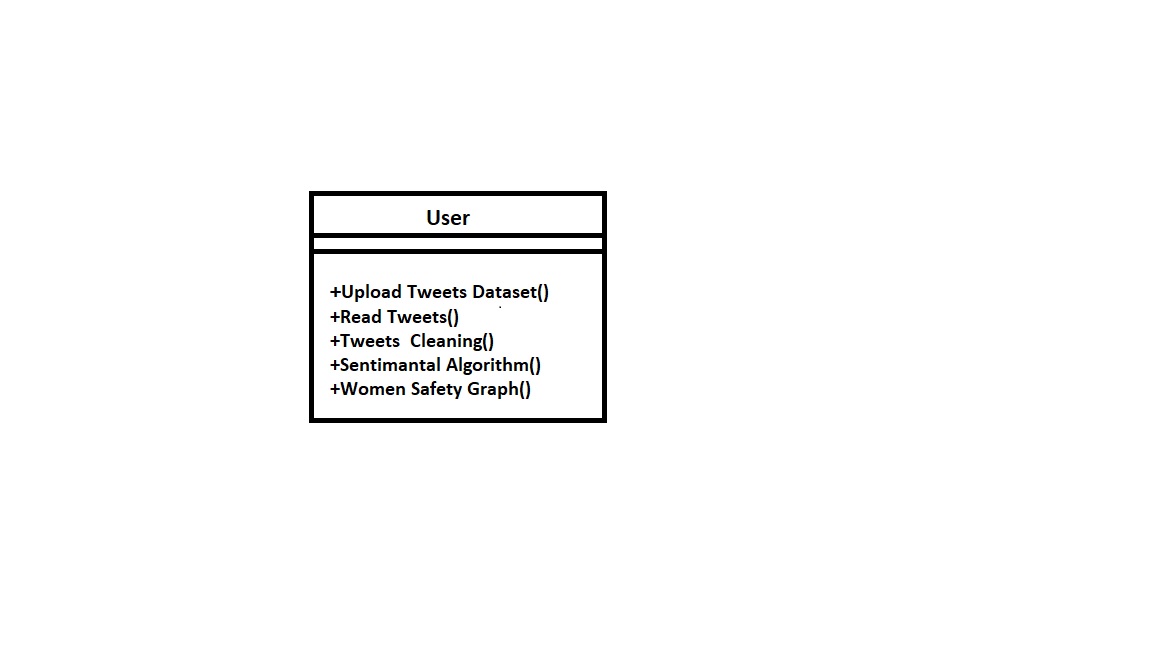
**4.1.1Usecase Diagram**

A use case diagram at its simplest is a representation of a user interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the difference types types of users of a system and the difference use case and will often be accompanied by other types of diagram as well. The use case are represented by either circle or ellipses.

**Fig 3: Use Case Diagram for project**

**4.1.2 Class Diagram**

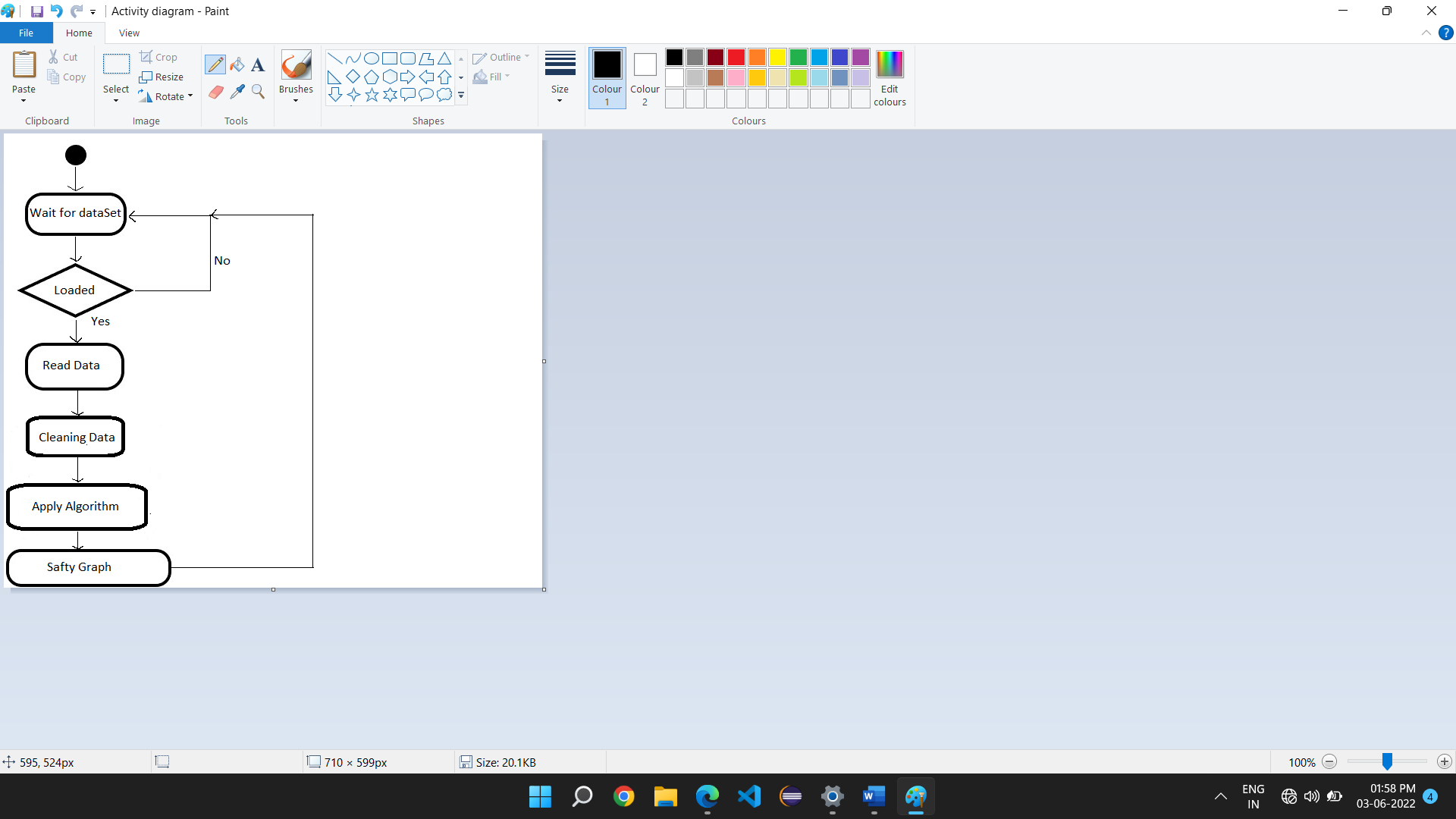
A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describe the structure of a system by showing the system classes, their attributes, operations and relationships among objects.



**Fig 4: Class Diagram**

**4.1.3 Activity Diagram**

Activity diagram is defined as a UML diagram that focuses on the execution and flow of the behavior of a system instead of implantation. It is also called object oriented flowchart. Activity diagram consists of activities that are made up of actions which apply to behavioral Modelling technology.

****

**Fig 5 : Activity Diagram**

**CHAPTER-5**

**IMPLEMENTATION**

**5.1 Data collection**

In propose work author using TWEEPY package from python to download tweets from twitter but every time INTERNET will not available to download tweets online so we downloaded MEETOO tweets on women safety and safe inside dataset folder. Application will read this tweets to detect women’s sentiments.

**5.2 Evaluation Parameters:**

**1. Data extraction:**

The first stage in sentiment analysis is to get data from social media sites such as Twitter. This aids in the extraction of the tweet message, but this message also contains additional information such as tweet likes, dislikes, and comments.

**2. Text Cleaning:**

Once the data has been extracted as datasets from the social media resource, it has to sent to the classifier. Before the analysis, the classifier cleans the dataset by removing unnecessary data such as stop words and emoticons to ensure that non-textual stuff is recognized and removed.

**3. Sentimental Analysis:**

The data is ready for sentiment analysis after the classifier cleans the dataset. Sentiment analysis employs a variety of techniques, including machine learning, Lexicon-based learning, and hybrid learning. Other techniques, such as Nero-Linguistic Programming and NLP(Natural Language Process), are also available. A machine learning approach involves training a dataset and then testing that dataset. The classifier needs both training and testing data to complete the algorithm. The sentiment classifier's efficiency is determined by using the testing data. The training dataset is not used in Lexicon-based leaning. This method employs a built-in vocabulary that includes words connected with human emotions. In order to increase the effectiveness of the classifier, the third approach, Hybrid learning, combines both machine learning and lexical learning

**4. Sentimental Classification:**

This is the point at which the dataset is ready for classification. For subjectivity, each sentence of the tweet will be evaluated and an opinion developed accordingly. Subjective expression sentences are kept, but objective expression sentences are discarded. At various levels of sentiment analysis, techniques like as metagrams, Negation and Lemmas etc., are applied. Positive and negative emotions can be broadly divided into two categories. At this stage of sentimental analysis, each of the retained subjective statements is rated as excellent, bad, like, dislike, or positive and negative.

**5.** **TEXTBLOB**

Author using TEXTBLOB corpora package and dictionary to count positive, negative and neutral polarity and the tweets which has polarity value less than 0 will consider as negative as and greater than 0 and less than 0.5 will consider as neutral and polarity greater than 0.5 will consider as positive.

**6.** **Output Result:** Sentiment analysis is most essential for extracting the usable and relevant information from raw data. After the method has been performed, the results of the analysis can be shown using several graph forms.

**5.3 Implementation Code:**

import tkinter

from textblob import TextBlob

from tkinter import \*

import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

from string import punctuation

from nltk.corpus import stopwords

from nltk.stem import PorterStemmer

from nltk.tokenize import word\_tokenize

import pickle

main = tkinter.Tk()

main.title("Analysis of Women Safety in Indian Cities Using Machine Learning on Tweets") #designing main screen

main.geometry("1300x1200")

global filename

tweets\_list = []

clean\_list = []

global pos, neu, neg

global decision\_tree, tfidf

#declaring NLTK word stemmer and stop word list

SW\_array = set(stopwords.words('english'))

word\_stemmer = PorterStemmer()

#function to remove stop words

def removeStopWords(data):

content = []

for i in range(len(data)):

if data[i] not in SW\_array and len(data[i]) > 2:

content.append(word\_stemmer.stem(data[i].strip()))

content = ' '.join(content)

return content

with open('model/dt.txt', 'rb') as file:

decision\_tree = pickle.load(file)

file.close()

with open('model/tfidf.txt', 'rb') as file:

tfidf = pickle.load(file)

file.close()

def tweetCleaning(doc):

tokens = doc.split()

table = str.maketrans('', '', punctuation)

tokens = [w.translate(table) for w in tokens]

tokens = [word for word in tokens if word.isalpha()]

stop\_words = set(stopwords.words('english'))

tokens = [w for w in tokens if not w in stop\_words]

tokens = [word for word in tokens if len(word) > 1]

tokens = ' '.join(tokens) #here upto for word based

return tokens

def upload(): #function to upload tweeter profile

global filename

filename = filedialog.askopenfilename(initialdir="dataset")

pathlabel.config(text=filename)

text.delete('1.0', END)

text.insert(END,filename+" loaded\n");

def read():

text.delete('1.0', END)

tweets\_list.clear()

train = pd.read\_csv(filename,encoding='iso-8859-1')

for i in range(len(train)):

tweet = train.get\_value(i, 'Text')

tweets\_list.append(tweet)

text.insert(END,tweet+"\n")

text.insert(END,"\n\nTotal tweets found in dataset is : "+str(len(tweets\_list))+"\n\n\n")

def clean():

text.delete('1.0', END)

clean\_list.clear()

for i in range(len(tweets\_list)):

tweet = tweets\_list[i]

tweet = tweet.strip("\n")

tweet = tweet.strip()

tweet = tweetCleaning(tweet.lower())

clean\_list.append(tweet)

text.insert(END,tweet+"\n")

text.insert(END,"\n\nTotal tweets found in dataset is : "+str(len(clean\_list))+"\n\n\n")

def checkTweet(tweet):

global decision\_tree, tfidf

data = tweet.strip("\n").strip().lower()

data = data.replace("\n"," ")

data = re.sub(r'[^a-zA-Z\s]+', '', data)

data = removeStopWords(data.split(" "))

data = tfidf.transform([data])

predict = decision\_tree.predict(data)

result = None

if predict[0] == 0:

result = "Genuine Tweet"

if predict[0] == 1:

result = "Fake Tweet"

return result

def machineLearning():

text.delete('1.0', END)

global pos, neu, neg

pos = 0

neu = 0

neg = 0

for i in range(0,100):

tweet = clean\_list[i]

blob = TextBlob(tweet)

tweet\_type = checkTweet(tweet)

if blob.polarity <= 0.2:

neg = neg + 1

text.insert(END,tweet+"\n")

text.insert(END,"Predicted Sentiment : NEGATIVE\n")

text.insert(END,"Polarity Score : "+str(blob.polarity)+"\n")

text.insert(END,"Tweet Predicted As : "+tweet\_type+"\n")

text.insert(END,'====================================================================================\n')

if blob.polarity > 0.2 and blob.polarity <= 0.5:

neu = neu + 1

text.insert(END,tweet+"\n")

text.insert(END,"Predicted Sentiment : NEUTRAL\n")

text.insert(END,"Polarity Score : "+str(blob.polarity)+"\n")

text.insert(END,"Tweet Predicted As : "+tweet\_type+"\n")

text.insert(END,'====================================================================================\n')

if blob.polarity > 0.5:

pos = pos + 1

text.insert(END,tweet+"\n")

text.insert(END,"Predicted Sentiment : POSITIVE\n")

text.insert(END,"Polarity Score : "+str(blob.polarity)+"\n")

text.insert(END,"Tweet Predicted As : "+tweet\_type+"\n")

text.insert(END,'====================================================================================\n')

text.update\_idletasks()

def graph():

label\_X = []

category\_X = []

text.delete('1.0', END)

text.insert(END,"Saftey Factor\n\n")

text.insert(END,'Positive : '+str(pos)+"\n")

text.insert(END,'Negative : '+str(neg)+"\n")

text.insert(END,'Neutral : '+str(neu)+"\n\n")

text.insert(END,'Length of tweets : '+str(len(clean\_list))+"\n")

text.insert(END,'Positive : '+str(pos)+' / '+ str(len(clean\_list))+' = '+str(pos/len(clean\_list))+'%\n')

text.insert(END,'Negative : '+str(neg)+' / '+ str(len(clean\_list))+' = '+str(neg/len(clean\_list))+'%\n')

text.insert(END,'Neutral : '+str(neu)+' / '+ str(len(clean\_list))+' = '+str(neu/len(clean\_list))+'%\n')

label\_X.append('Positive')

label\_X.append('Negative')

label\_X.append('Neutral')

category\_X.append(pos)

category\_X.append(neg)

category\_X.append(neu)

plt.pie(category\_X,labels=label\_X,autopct='%1.1f%%')

plt.title('Women Saftey & Sentiment Graph')

plt.axis('equal')

plt.show()

font = ('times', 16, 'bold')

title = Label(main, text='Analysis of Women Safety in Indian Cities Using Machine Learning on Tweets')

title.config(bg='brown', fg='white')

title.config(font=font)

title.config(height=3, width=120)

title.place(x=0,y=5)

font1 = ('times', 14, 'bold')

uploadButton = Button(main, text="Upload Tweets Dataset", command=upload)

uploadButton.place(x=50,y=100)

uploadButton.config(font=font1)

pathlabel = Label(main)

pathlabel.config(bg='brown', fg='white')

pathlabel.config(font=font1)

pathlabel.place(x=370,y=100)

readButton = Button(main, text="Read Tweets", command=read)

readButton.place(x=50,y=150)

readButton.config(font=font1)

cleanButton = Button(main, text="Tweets Cleaning", command=clean)

cleanButton.place(x=210,y=150)

cleanButton.config(font=font1)

mlButton = Button(main, text="Run Machine Learning Algorithm", command=machineLearning)

mlButton.place(x=400,y=150)

mlButton.config(font=font1)

graphButton = Button(main, text="Women Saftey Graph", command=graph)

graphButton.place(x=730,y=150)

graphButton.config(font=font1)

font1 = ('times', 12, 'bold')

text=Text(main,height=25,width=150)

scroll=Scrollbar(text)

text.configure(yscrollcommand=scroll.set)

text.place(x=10,y=200)

text.config(font=font1)

main.config(bg='brown')

main.mainloop()

**CHAPTER-6**

**TESTING**

**6.1 General:**

Software Testing is the process used to help identify the correctness, completeness, security and quality of developed computer software. Testing is a process of technical investigation, performed on behalf of stakeholders, that is intended to reveal quality-related information about the product with respect to the context in which it is intended to operate. In general, software engineers distinguish software faults from software failures. Our project is tested with the following testing methodologies.

**Developing Methodologies:** The test process begins by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process verifies that the application meets the requirements specified in the system requirements document and is bug free. The following are the considerations used to develop the framework for developing the test methodologies.

**Acquire and study the test strategy:** A team very familiar with the business risks associated with the software normally develops test strategy, the test team develops tactics. Thus, the test team needs to acquire and study the test strategy. The test tactics are analyzed and studied for finding our various test factors, risks and effects. The risk involved in our project is implementing the encoding of the image. So, the proper knowledge about the testing strategies should be gained in order to avoid such high-level risks.

**Determine the type of development project:** The type of the development refers to the platform or methodology for developing the project. As it is been a simulation project we go for the prototyping. The prototypes are simply predefined structure or model, which can be used for further modeling. By using the prototypes we can modify the existing module of the application for some other specific operations. Here the test tactics is to verify that all the tools are used properly and to test functionality.

**Determine the type of software system:** The type of software system relates to the type of processing which will be encountered by the system. In this project, the software system we prefer to use is python. We have choosen python for its portability and its support to graphics & multimedia specifically for image processing.

**Determine the scope of the software system:** The scope of the project refers to the overall activities or operation to be included into the system being tested. The scope of the new system varies from that of the existing once.

**Mode of testing:** It is necessary to select the test mode in which the testing method to be carried out. The two different modes are manual and automated tool. The real time projects need frequent interactions. So, it is impossible to carry out the testing process by means of automated tool. Our project uses manual testing.

**Unit test technique:** This phase examines the techniques, assessment and management of unit testing and analysis.

Testing and analysis strategies are categorized according to whether they goal is functional or structural or combination of these. It will assist a software engineer to define, conduct and evaluate unit tests and to assess new unit test techniques.

**6.2 System Testing :**

Once the entire system has been built then it has to be tested against the “System Specification” to check if it delivers the features required. It is still developer focused, although specialist developers known as systems testers are normally employed to do it. In essence system, Testing is not about checking the system as a whole. In effect it is one giant component. System testing can involve a number of specialist types of test to see if all the functional and nonfunctional requirements have been met.

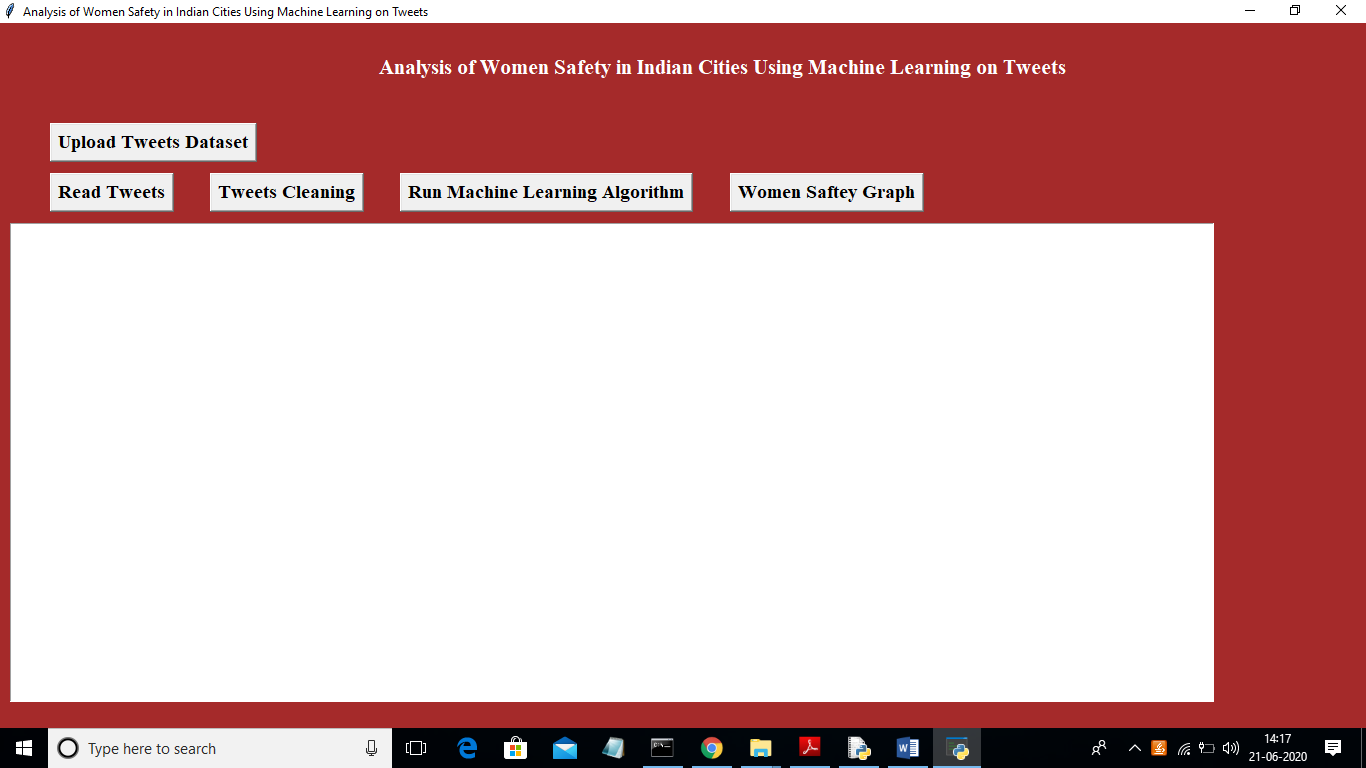
**6.3 Regression Testing:**

This involves assurance that all aspects of an application system remain functional after testing. The introduction of change is the cause of problems in previously tested segments. It is retesting unchanged segments of the application system. It normally involves rerunning tests that have been previously executed to ensure that the same results can be achieved currently when the segments were last tested.

**CHAPTER-7**

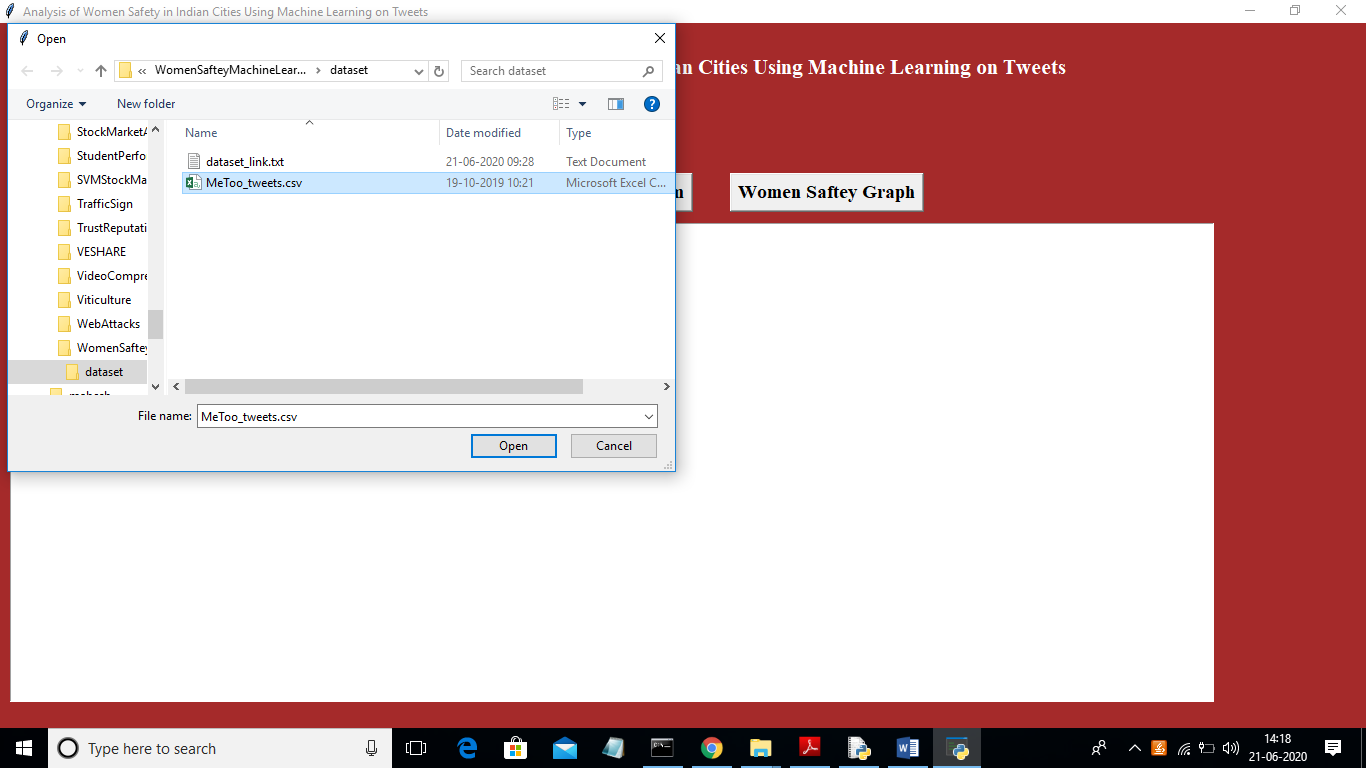
**OUTCOME**

Initial Output screen with 5 modules to perform respective operations:



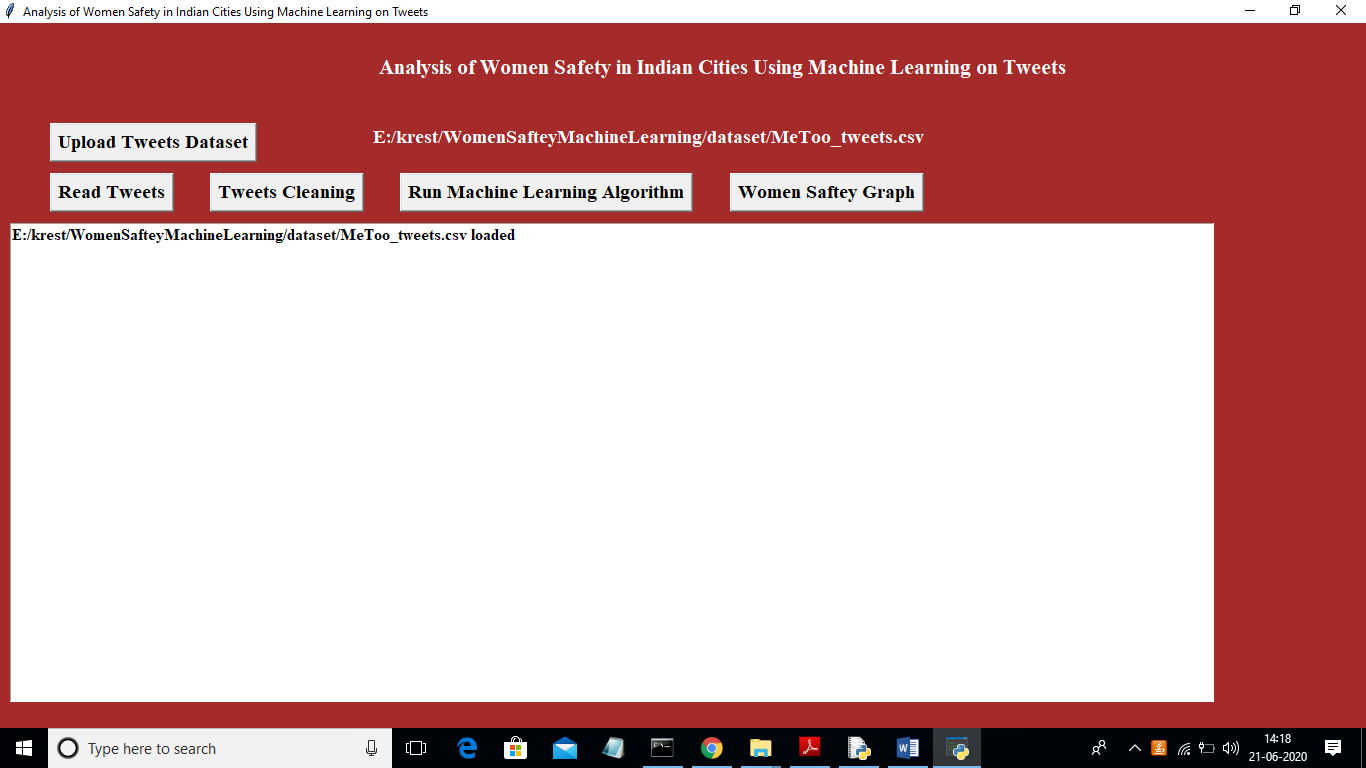
**Fig 6: Result Screen**

In above screen click on ‘Upload Tweets Dataset’ button and upload tweets.



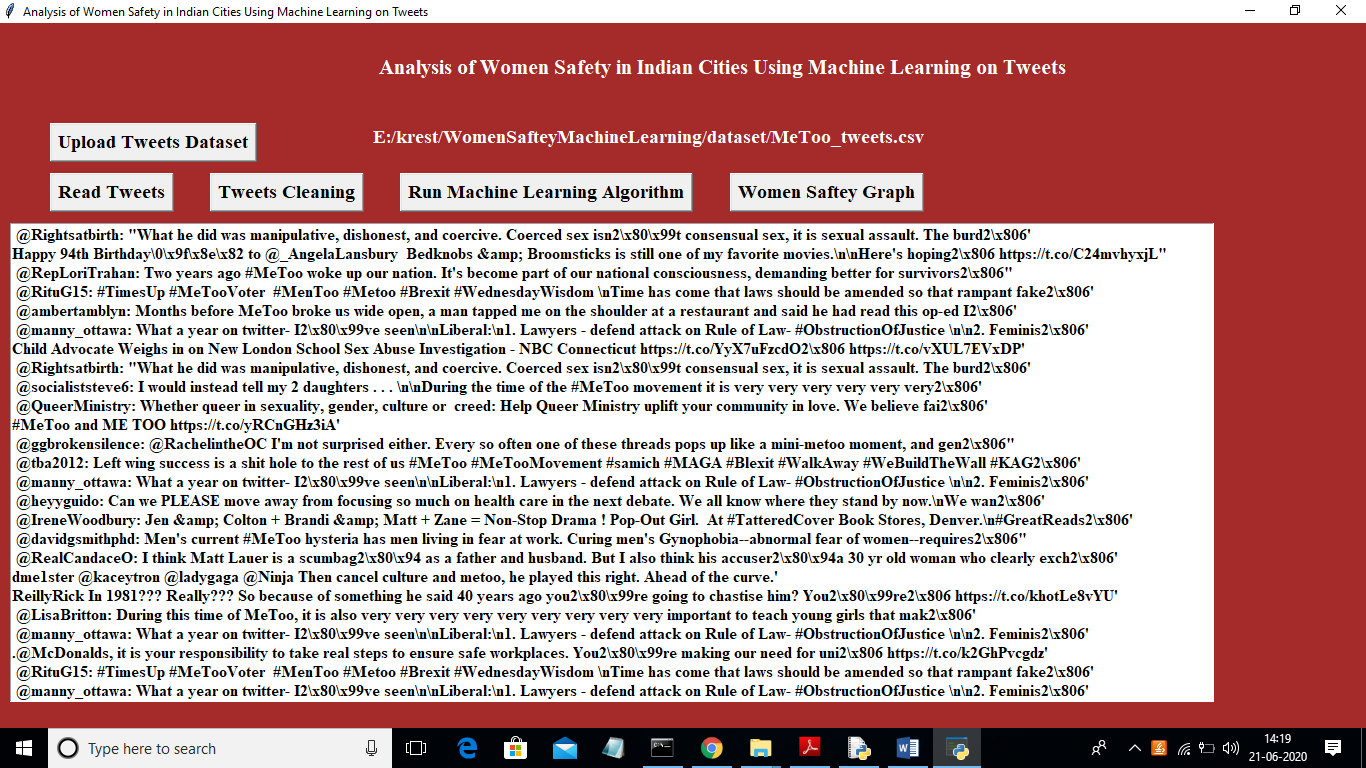
**Fig 7: Upload Dataset**

In above screen uploading MeeToo\_tweets.csv file and then click on ‘Open’ button to load dataset and to get below screen.



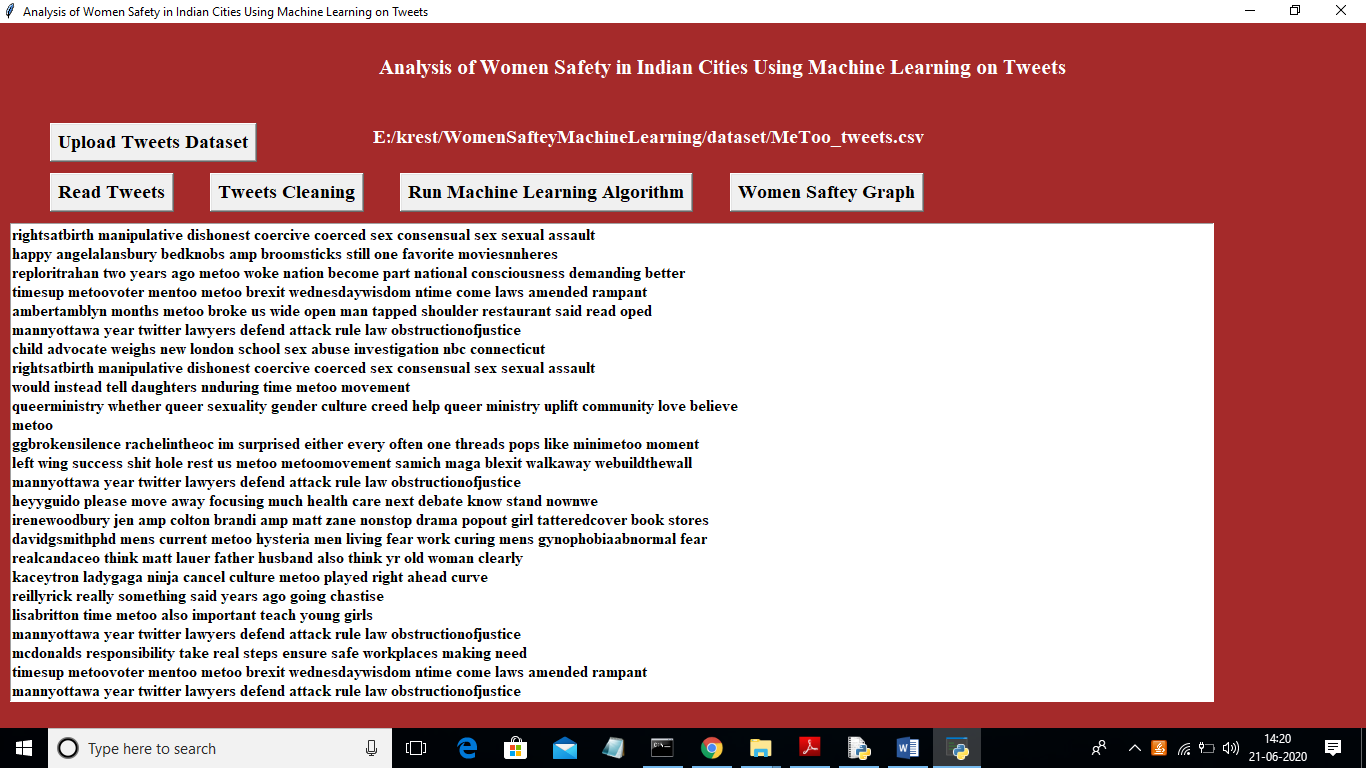
**Fig 8: LOADED THE DATASET**

In above screen tweets dataset loaded and now click on ‘Read Tweets’ button to read tweets from dataset.



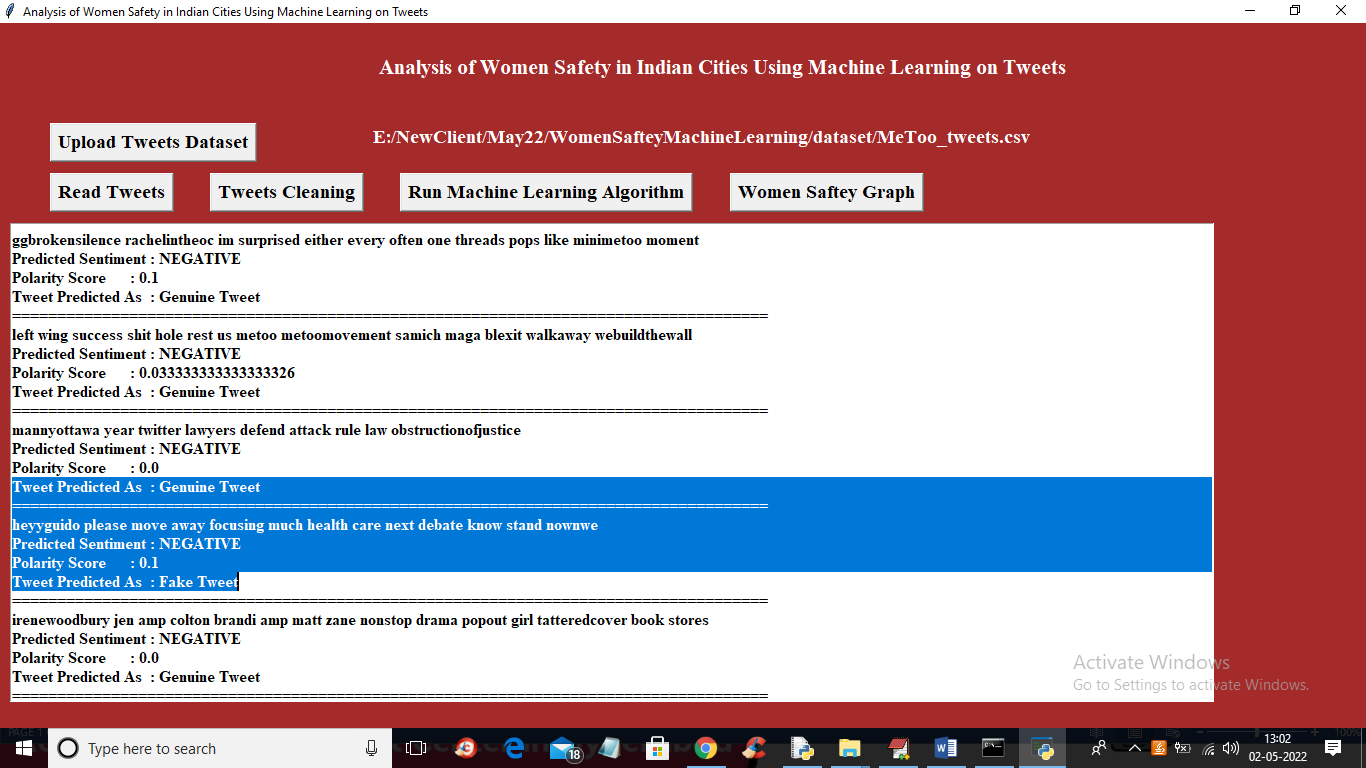
**Fig 9: Read Dataset**

In above screen each line represents one tweet and you can scroll down above screen text area to view all tweets. In above screen we can see all tweets contains special symbols and stop words and to clean those tweets click on ‘Tweets Cleaning’ button.



**Fig 10: Cleaning Dataset**

In above screen we can see all special symbols and stop words remove from tweets and only clean words are there and now click on ‘Run Machine Learning Algorithm’ button to predict sentiments from tweets.



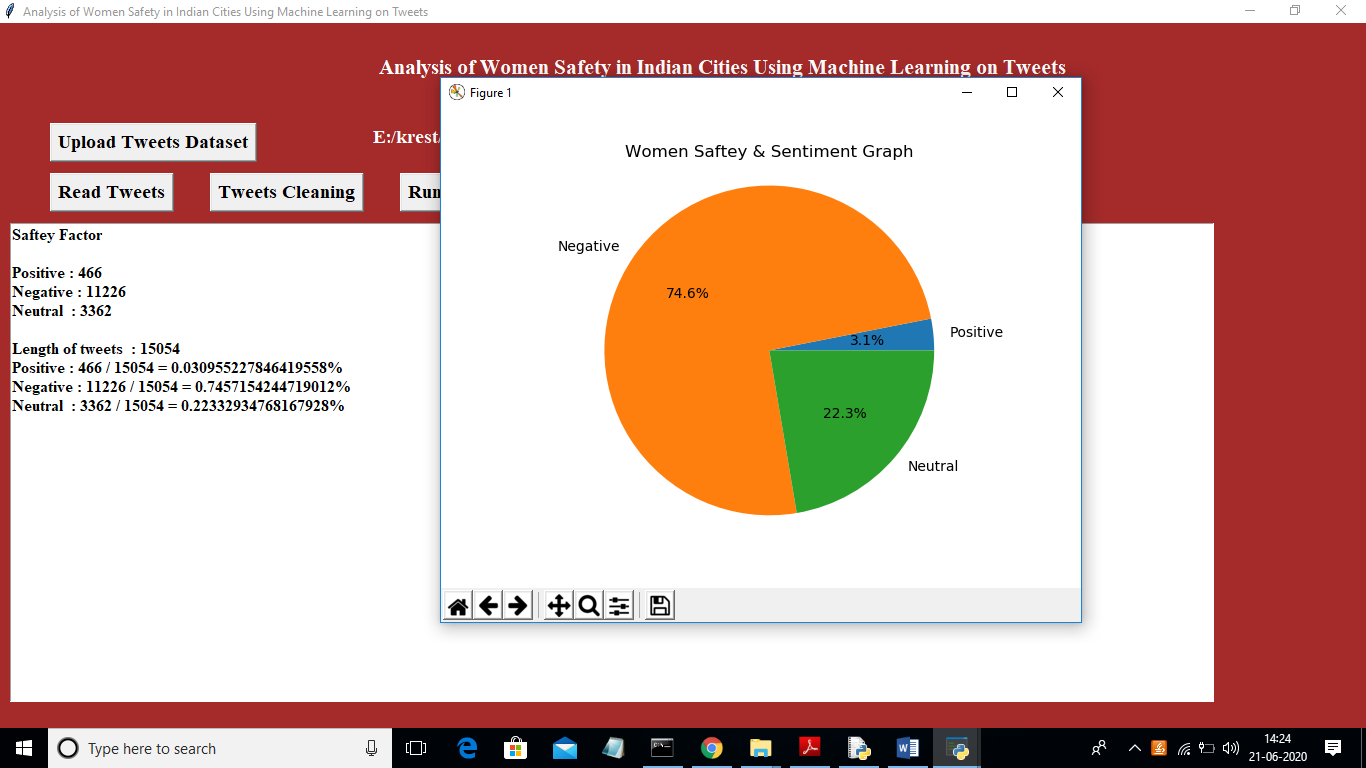
**Fig 11: Apply Algorithm**

In above screen we are displaying tweets with sentiment and its authenticity as FAKE or GENUINE by using decision tree algorithm.

we have added TFIDF (term frequency inverse frequency document) algorithm which will convert all tweets words into numeric vector which will contains average frequency of each tweet words.

This TFIDF vector will be input to Decision Tree algorithm which will predict weather given tweet is FAKE or GENUINE. Decision Tree algorithm is already trained on FAKE and REAL words so it can easily predict authenticity of each tweet.

In above screen each tweet having tweet text and then displaying tweets sentiments with polarity score. Scroll down above text area to see all tweets. Now click on ‘Women Saftey Graph’ button to get below results and by seeing that result user can easily understand whether area is safe or not. If area is safe then more peoples will express either positive or neutral tweets and if not safe then more peoples will discuss negative tweets.



**Fig 12: Women safety & Sentiment Graph**

In above screen 0.74 multiply by 100 will give 74% which means 74% peoples are talking negative and area is not safe and only 22 and 3% peoples are talking positive and neutral.

**CONCLUSION**

In propose paper author is analysing social media tweets to detect women’s safety but the main problem is social media can be used by malicious users who will give fake ratings to worst products to boom their sale or can write fake tweets to raise finger at country safety towards Indian women and to overcome from this problem we have modified propose work with two algorithms which will check weather given tweets are fake or genuine. So by getting tweets authenticity peoples will comes to real conclusion on women safety. In extension work we have added TFIDF (term frequency inverse frequency document) algorithm which will convert all tweets words into numeric vector which will contains average frequency of each tweet words. This TFIDF vector will be input to Decision Tree algorithm which will predict weather given tweet is FAKE or GENUINE. Decision Tree algorithm is already trained on FAKE and REAL words so it can easily predict authenticity of each tweet.

**REFERENCE**

[1] Agarwal, Apoorv, Fadi Biadsy, and Kathleen R. Mckeown. "Contextual phrase-level polarity analysis using lexical affect scoring and syntactic n-grams." Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 2009.

[2] Barbosa, Luciano, and Junlan Feng. "Robust sentiment detection on twitter from biased and noisy data." Proceedings of the 23rd international conference on computational linguistics: posters. Association for Computational Linguistics, 2010.

[3] Bermingham, Adam, and Alan F. Smeaton. "Classifying sentiment in microblogs: is brevity an advantage?." Proceedings of the 19th ACM international conference on Information and knowledge management. ACM, 2010.

[4] Gamon, Michael. "Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis." Proceedings of the 20th international conference on Computational Linguistics. Association for Computational Linguistics, 2004.

[5] Kim, Soo-Min, and Eduard Hovy. "Determining the sentiment of opinions." Proceedings of the 20th international conference on Computational Linguistics. Association for Computational Linguistics, 2004.

[6] Klein, Dan, and Christopher D. Manning. "Accurate unlexicalized parsing." Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1. Association for Computational Linguistics, 2003..

[7] Charniak, Eugene, and Mark Johnson. "Coarse-to-fine n-best parsing and MaxEnt discriminative reranking." Proceedings of the 43rd annual meeting on association for computational linguistics. Association for Computational Linguistics, 2005.

[8] Gupta, B., Negi, M., Vishwakarma, K., Rawat, G., & Badhani, P. (2017). Study of Twitter sentiment analysis using machine learning algorithms on Python. International Journal of Computer Applications, 165(9), 0975-8887.

[9] Sahayak, V., Shete, V., & Pathan, A. (2015). Sentiment analysis on twitter data. International Journal of Innovative Research in Advanced Engineering (IJIRAE), 2(1), 178-183.

[10] Mamgain, N., Mehta, E., Mittal, A., & Bhatt, G. (2016, March). Sentiment analysis of top colleges in India using Twitter data. In Computational Techniques in Information and Communication Technologies (ICCTICT), 2016 International Conference on (pp. 525-530). IEEE.