Analysis of Women Safety in Indian Cities Using Machine Learning on Tweets

Salla Anisha and Srinidhi Ghankot

Dr. BV Ramana Murthy

Department of Computer Science and Engineering, Stanley College of Engineering and Technology for Women, Telangana, India

ABSTRACT

Women and girls have been experiencing a lot of violence and harassment in public places in various cities starting from stalking and leading to abuse harassment or abuse 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. This paper also focuses on how a sense of responsibility on part of Indian society can be developed by the common Indian people so that we should focus on the safety of women surrounding them. Tweets on Twitter which usually contain images and text and also written messages and quotes which focus on the safety of women in Indian cities can be used to read a message amongst the Indian Youth Culture and educate people to take strict action and punish those who harass the women. Twitter and other Twitter handles which include hashtag messages that are widely spread across the whole globe sir as a platform for women to express their views about how they feel while we go out for work or travel in a public transport and what is the state of their mind when they are surrounded by unknown men and whether these women feel safe or not?

Keywords: Machine Learning, Tweets, Python

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 traveling 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 neighbors 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 behavior 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 smoothing 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 Objective of the Project:

Results of the sentimental analysis can be used in many areas like sentiments regarding a particular brand or release of a product, analyzing public opinions on the government policies, people's thoughts on women, etc. In order to perform classification of tweets and analyze the outcome, a lot of study has been done on the data obtained by twitter. We also review some studies on machine learning in this paper and research on how to perform sentiment analysis using that domain on twitter data.

1.2 Scope of the Project:

The project scope is restricted to machine learning algorithms and models. Staring at women and passing comments can be certain types of violence and harassment and these practices, which are unacceptable, are usually normal especially on the part of urban life. Many researches that have been conducted in India shows that women have reported sexual harassment and other practices as stated above. Such studies have also shown that in popular metropolitan cities like Delhi, Pune, Chennai and Mumbai, most women feel they are unsafe when surrounded by unknown people.

1.3 Advantages

Analysis of twitter texts collection also includes the names of people and names of women who stand up against abuse, harassment and unethical behavior 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.

1.4 Disadvantages

Twitter and Instagram point and most of the people are using it to express their emotions and also their opinions about what they think about the Indian cities and Indian society. There are several methods of sentiment that can be categorized like machine learning hybrid and lexicon-based learning. Also there is another categorization presented with categories of statistical, knowledge-based and age wise differentiation approaches.

2. SOFTWARE AND HARDWARE REQUIREMENTS

Software Requirements:

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation.

The appropriation of requirements and implementation constraints gives the general overview of the project in regards to what the areas of strength and deficit are and how to tackle them.

- Python idel 3.7 version   (or)
- Anaconda 3.7   (or)
- Jupiter   (or)
- Google colab

Hardware Requirements:

Minimum hardware requirements are very dependent on the particular software being developed by a given Enthought Python / Canopy / VS Code user. Applications that need to store large arrays/objects in memory will require more RAM, whereas applications that need to perform numerous calculations or tasks more quickly will require a faster processor.

- Operating system   : windows, linux
- Processor   : minimum intel i3
- Ram   : minimum 4 gb
- Hard disk   : minimum 250gb

3. LITERATURE SURVEY

Existing System:

The concept to analyze women safety using social networking messages and by applying machine learning algorithms on it. 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 analyzing those messages we can detect which area is more unsafe for women.

Proposed System:

The proposed work uses the TWEEPY package from python to download tweets from twitter but every time INTERNET will not be available to download tweets online so we downloaded MEETOO tweets on women safety and safety inside the dataset folder. Application will read these tweets to detect women's sentiments.

We use NLTK (natural language toolkit) to remove special symbols and stop words from tweets and to make them clean.

Also, we use 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.
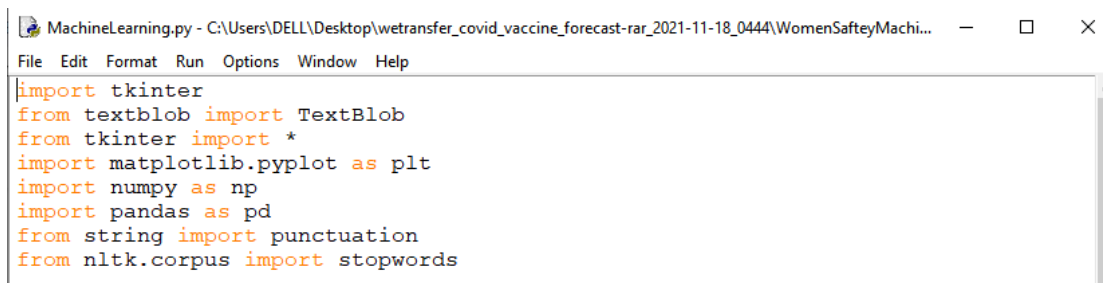
4. SYSTEM ARCHITECTURE
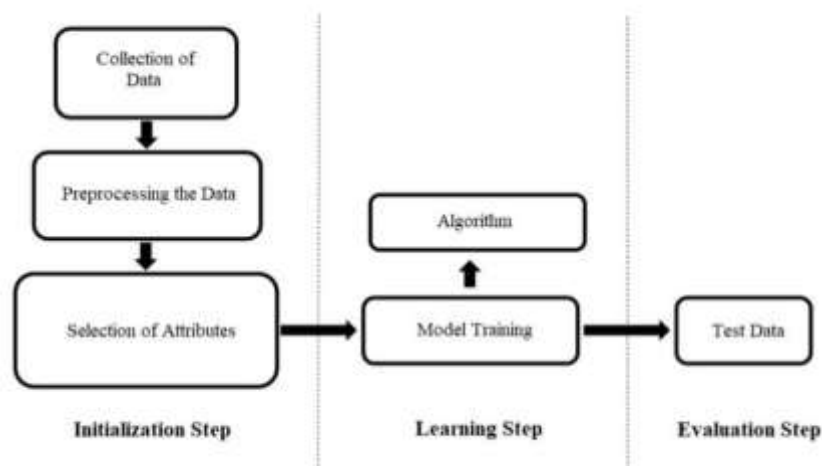
5. MODULES

- Upload dataset: Using this module we will upload dataset

- Dataset cleaning: Using this module we will find empty values in the dataset and replace them with mean or 0 values.

- Train & Test Split: Using this module we will split the dataset into two parts called training and testing. All machine learning algorithms take 80% dataset to train classifiers and 20% dataset is used to test classifier prediction accuracy.

6. RESULTS

Step 1: import libraries


```
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
```

Fig 7.1 - Importing Libraries

(where tkinter used for GUI(front-end),text blob -processing textual data, matplotlib - data visualization,pandas - data analysis and preprocessing,numpy - mathematical purpose,nltk - building python program(remove special symbols and stop words))

Step 2: Defining main function and setting the title & size of tkinter

```
main = tkinter.Tk()
main.title("Analysis of Women Safety in Indian Cities Using Machine Learning on Tweets") #designing main screen
main.geometry("1300x1200")
```

Fig 7.2 - Defining main function and setting the title & size of tkinter

Step 3: Defining the global function

```
global filename
tweets_list = []
clean_list = []
global pos, neu, neg
```

Fig 7.3 - Defining the global function

Step 4: Upload data set

```
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");
```

International Journal for Innovative Engineering and Management Research
A Peer Reviewed Open Access International Journal

www.ijiemr.org

Fig 7.4 - Upload data set

Step 5: Defining the read tweets function

```
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")
```

Fig 7.5 - Defining the read tweets function

Step 6: Defining tweet cleaning function

```
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
```

Fig 7.6.1 - Defining tweet cleaning function

```
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")
```

Fig 7.6.2 - Defining tweet cleaning function

Step 7: Build machine learning algorithm


```
def machineLearning():
    text.delete('1.0', END)
    global pos, neu, neg
    pos = 0
    neu = 0
    neg = 0
    for i in range(len(clean_list)):
        tweet = clean_list[i]
        blob = TextBlob(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,'==============================================================\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,'==============================================================\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,'==============================================================\n')
```

Fig 7.7 - Build machine learning algorithm

Step 8: Women safety graph


```
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()
```

Fig 7.8 - Women safety graph

Step 9: Defining the button size and configuration

```
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)
```

7.9 - Defining the button size and configuration

Now double click on 'run.bat' file to run project and to get below screen

Fig 7.10 - 'Upload Tweets Dataset' Page

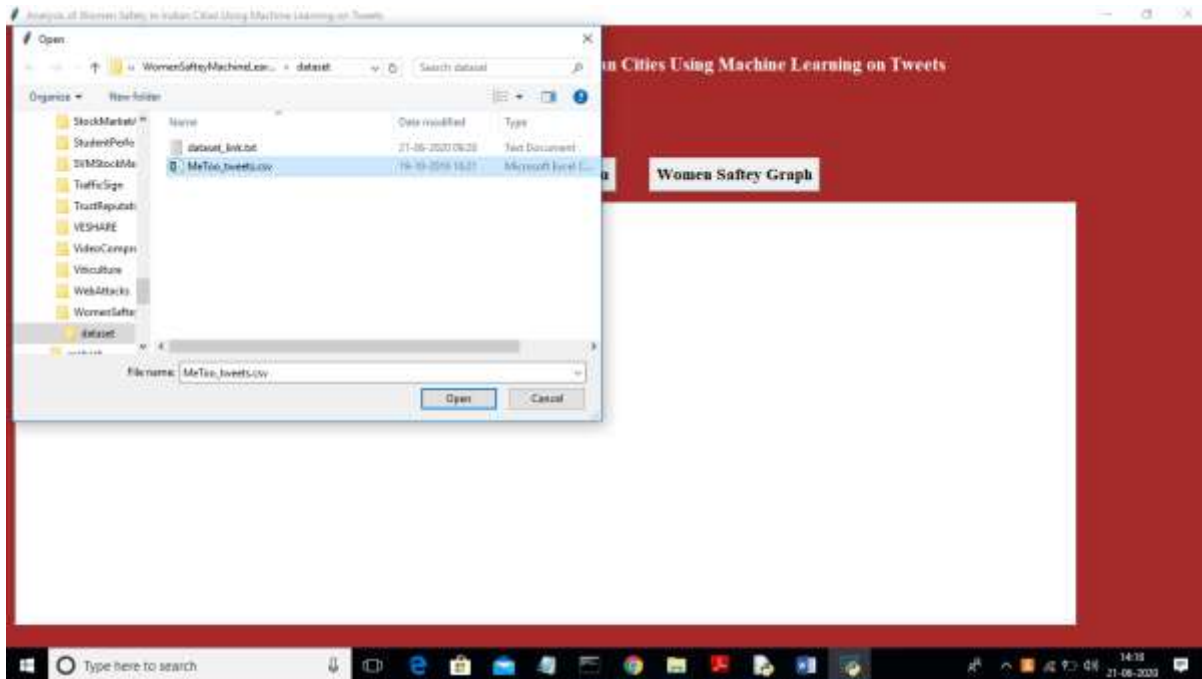In above screen click on 'Upload Tweets Dataset' button and upload tweets

Fig 7.11 - Selecting the Dataset

In above screen uploading MeeToo_tweets.csv file and then click on 'Open' button to load dataset and to get below screen

Fig 7.12 - Loading the Dataset

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

Fig 7.13 - Reading and Cleaning the Dataset

In the above screen each line represents one tweet and you can scroll down above the text area to view all tweets. In the above screen we can see all tweets containing special symbols and stop words and to clean those tweets click on 'Tweets Cleaning' button.
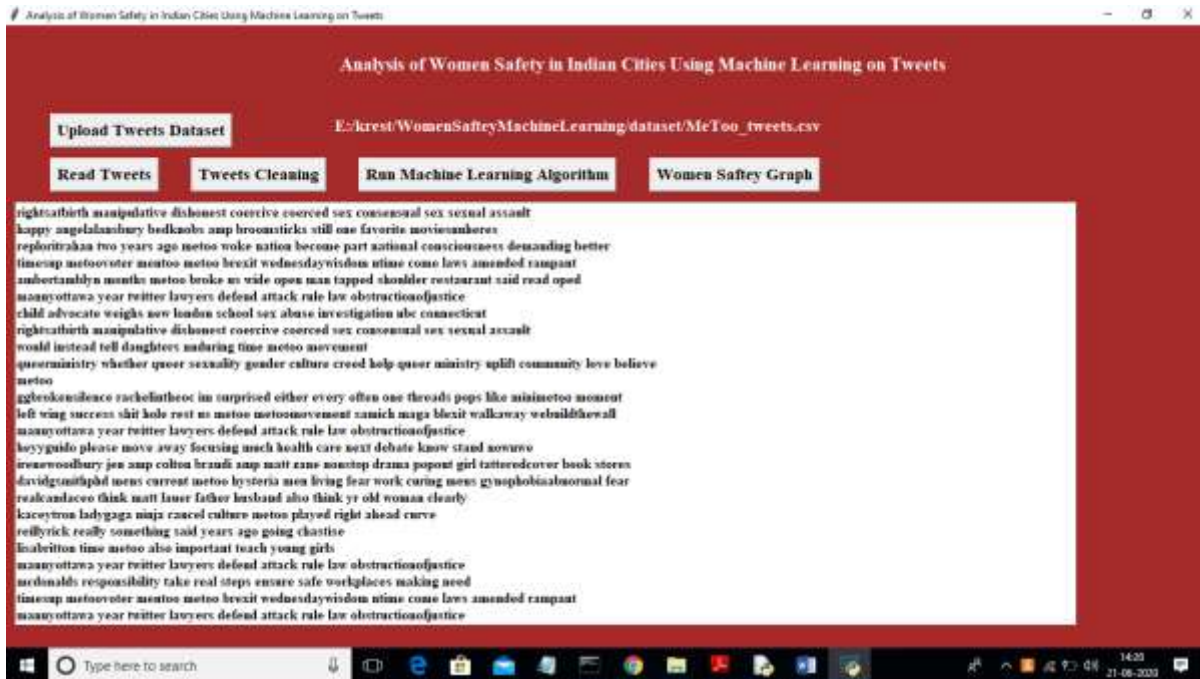
Fig 7.14 - Dataset Cleaned

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 7.15 - Polarity Scores

In the above screen each tweet has tweet text and then displaying tweets sentiments with polarity score. Scroll down above the text area to see all tweets. Now click on the 'Women Safety Graph' button to get the results below and by seeing that result, the user can easily understand whether the area is safe or not. If the area is safe then more peoples will express either positive or neutral tweets and if not safe then more peoples will discuss negative tweets.
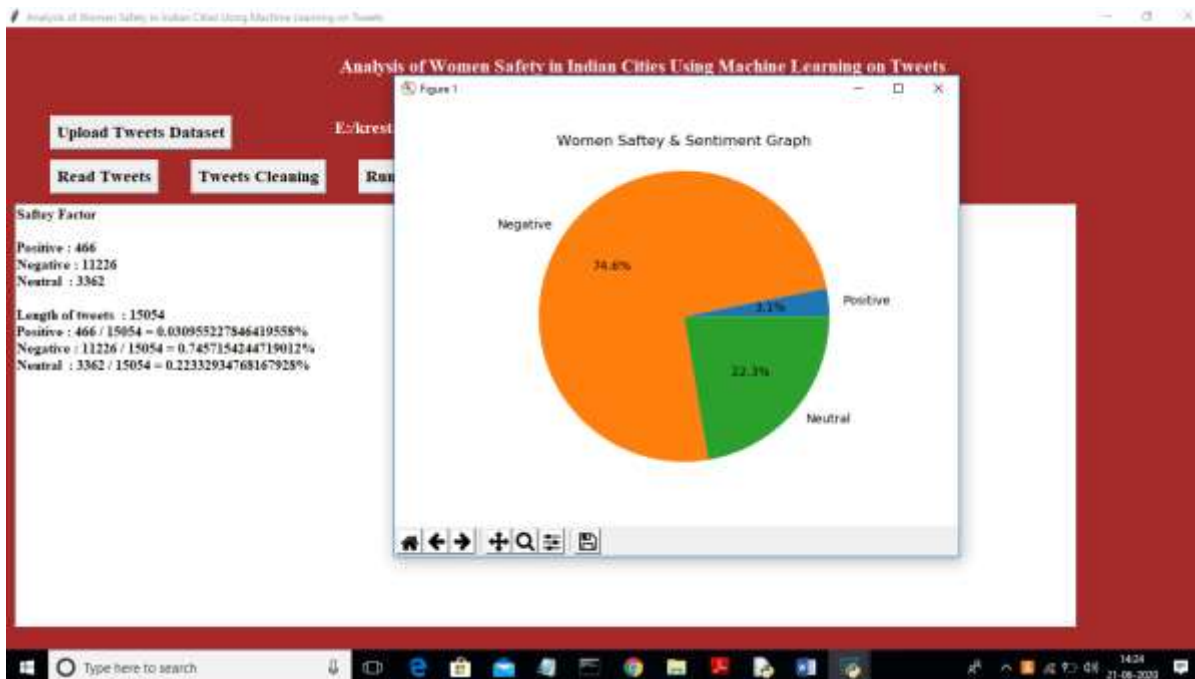
Fig 7.16 - Women Safety & Sentiment Graph

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

7. CONCLUSION

Throughout the research paper we have discussed various machine learning algorithms that can help us to organize and analyze the huge amount of Twitter data obtained including millions of tweets and text messages shared every day. These machine learning algorithms are very effective and useful when it comes to analyzing large amounts of data including the SPC algorithm and linear algebraic Factor Model approaches which help to further categorize the data into meaningful groups. Support vector machines is yet another form of machine learning algorithm that is very popular in extracting Useful information from Twitter and getting an idea about the status of women safety in Indian cities.

Future Enhancement:

For the future enhancement, we can extend to apply these machine learning algorithms on different social media platforms like facebook and instagram also since in our project only twitter is considered. Present ideology which is

proposed can be integrated with the twitter application interface to reach a larger extent and apply sentimental analysis on millions of tweets to provide more safety.

8. REFERENCES

Journals:

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.

Textbooks:

1. Programming Python, Mark Lutz
2. Head First Python, Paul Barry
3. Core Python Programming, R. Nageswara Rao
4. Learning with Python, Allen B. Downey

Websites:

- https://www.w3schools.com/python/
- https://www.tutorialspoint.com/python/index.htm
- https://www.javatpoint.com/python-tutorial
- https://www.learnpython.org/
- https://www.pythontutorial.net/