A

Mini Project

On

ANALYSIS OF WOMEN SAFETY IN INDIAN CITIES USING MACHINE LEARNING ON TWEETS

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

VELDHANDI KAVYA (207R1A05P6)

DONGARI ABHILASH VARMA (207R1A05K8)

DHARAVATH HANUMU (207R1A05K7)

Under the guidance of

Dr. D.T.V. DHARMAJEE RAO

Professor of CSE & Dean Academics

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi

Recognized Under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya(V),

Medchal Road, Hyderabad-501401.

2020 – 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled **" ANALYSIS OF WOMEN SAFETY IN INDIAN CITIES USING MACHINE LEARNING ON TWEETS "** being submitted by **V. KAVYA (207R1A05P6) , D. ABHILASH VARMA (207R1A05K8)** & **D. HANUMU (207R1A05K7)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under my guidance and supervision during the year 2023-24.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr. D.T.V. Dharmajee Rao
(Professor of CSE & Dean Academics)
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
HOD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on ___________________

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard my guide **Dr. D.T.V. Dharmajee Rao**, Professor of Computer Science and Engineering & Dean Academics for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by her shall carry us long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC**) Dr. K. Maheswari , Dr. J. Narasimha Rao, G. Vinesh Shanker, Ms. Shilpa** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are all thankful to **Dr. K. Srujan Raju** , Head of the Department of Computer Science and Engineering for providing encouragement and support for completing this Project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course this of project. We also express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

VELDHANDI KAVYA	**(207R1A05P6)**
DONGARI ABHILASH VARMA	**(207R1A05K8)**
DHARAVATH HANUMU	**(207R1A05K7)**

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 the common Indian people so that we should focus on the safety of women surrounding them. Tweets on Twitter which usually contains 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 hash tag 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.

Social media platforms like Twitter, Facebook, and Instagram can help promote women's safety in Indian cities. The content of tweets includes text, images, messages, and quotes that focus on women's safety in Indian cities and encourage people to take action against harassment and assault.

LIST OF FIGURES / TABLES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1. INTRODUCTION

1. INTRODUCTION

1.1 PROJECT SCOPE

Tweets on twitter which usually contains 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.

For the imminent enrichment, one can encompass to smear these algorithms on the diverse societal websites like Facebook application and Instagram application. Subsequently, scheme contracts with tweets, present philosophy which is projected can be unified with the application request interface to influence higher possibility and spread on the algorithm used on loads of tweets that delivers added protection.

1.2 PROJECT PURPOSE

Analyzing women’s safety in indian cities using machine learning on tweets serves a crucial purpose on multiple fronts. First and foremost, it empowers us with data-driven insights into the prevailing issues surrounding women’s safety , offering a quantitative understanding of the problem’s scale and nature. Furthermore, sentiment analysis enables the quantification of public sentiments and emotions, providing a valuable metric to gauge the sense of safety among concerns are most pronounced, offering actionable information for law enforcement and policymaker.

1.3 PROJECT FEATURES

Twitter gives an excellent feature for women to express their views about what they feel while they travel or go out. We can categorize their opinions by using tweets. Hence by reviewing these classified tweets, we can identify the places which are less safety for women.

2. SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

SYSTEM ANALYSIS

The system analysis for the "Analysis of Women's Safety in Indian Cities Using Machine Learning on Tweets" project is a multifaceted endeavor that encompasses a series of crucial components. It begins with the meticulous collection and preprocessing of a vast dataset of tweets pertaining to women's safety, ensuring data integrity and privacy. Subsequently, advanced natural language processing techniques are deployed to conduct sentiment analysis and topic modeling, revealing sentiments and key themes in the tweets. Machine learning models play a pivotal role in predicting sentiments, identifying hate speech, and forecasting spikes in safety concerns. These findings are then translated into compelling visualizations and reports, making the information accessible and actionable for policymakers and stakeholders. Ethical considerations, user interaction mechanisms, and feedback loops ensure responsible and continuous improvement, while resource management and scalability guarantee the system's efficiency and adaptability.

2.1 PROBLEM DEFINITION

This project aims to use machine learning and sentiment analysis on tweets to understand and address women's safety concerns in Indian cities, providing data-driven insights for improved policy and intervention strategies. Key challenges include handling privacy issues in social media data and ensuring the ethical and effective translation of analysis findings into actionable measures.

2.2 EXISTING SYSTEM

People often express their views freely on social media about what they feel about the Indian society and the politicians that claim that Indian cities are safe for women. On social media websites people can freely Express their view point and women can share their experiences where they have faced abuse harassment or where we would have fight back against the abuse harassment that was imposed on them .

Of standing up against abuse harassment further motivates other women data on the same social media website or application like Twitter.It is a common practice to extract the information from the data that is available on social networking through procedures of data extraction, data analysis and data interpretation methods. The accuracy of the Twitter analysis and prediction can be obtained by the use of behavioral analysis on the basis of social networks.

2.2.1 DISADVANTAGES OF THE EXISTING SYSTEM

- Data Bias
- Language and Vernacular
- Contextual Understanding
- Privacy Concerns
- Sparse Data
- Limited to Digital Reporting
- Lack of Ground Truth Data

2.3 PROPOSED SYSTEM

This project uses the Tweepy package in Python to download tweets from Twitter related to women's safety. As internet connectivity may not always be available, the project also includes the MEETOO tweets on women safety and a safe inside dataset, which the application reads to detect women's sentiments.

To clean the tweets, the project uses the NLTK (Natural Language Tool Kit) to remove special symbols and stop words.

To determine the polarity of the tweets, the project uses the (NLP)TextBlob corpora package and dictionary. Tweets with polarity values less than 0 are considered negative, those with values greater than 0 and les than 0.5 are considered neutral, and those with polarity values greater than 0.5 are considered positive. polarity values will be calculated using a lexicon-based approach.

2.3.1 ADVANTAGES OF PROPOSED SYSTEM

- Identification of key advocates and influencers in the fight for women's safety through the inclusion of their names in the Twitter data
- Twitter data provides a large and diverse dataset to identify patterns and trends related to women's safety in Indian society.
- Twitter data can help inform policy and intervention efforts to address the issue.
- Real-time Twitter data can identify emerging issues and track changes over time.
- The data set that was obtained through Twitter about the status of women safety in Indian society.

2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and a business proposalis put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This isto ensure that the proposed system is not a burden to the company.

Three key considerations involved in the feasibility analysis are :

- ECONOMIC FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria ensure that effort is concentrated on a project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminaryinvestigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

- Processor : Intel core i5
- RAM : 8.00 GB
- Hard Disk : 40 GB
- Key Board : Standard Keyboard
- Mouse : Two or Three Button mouse
- Monitor : HP Laptop 15inch
- CPU : 1GHZ

2.5.2 SOFTWARE REQUIREMENTS

Software Requirements specifies the logical characteristics of each interface and software components of the system.

The following are some software requirements:

- Operating System : windows 11
- Coding language : python with Django
- Back End : MYSQL

3.ARCHITECTURE

3.ARCHITECTURE

3.1 PROJECT ARCHITECTURE

This project Architecture shows the procedure followed for classification , starting from input to final prediction.

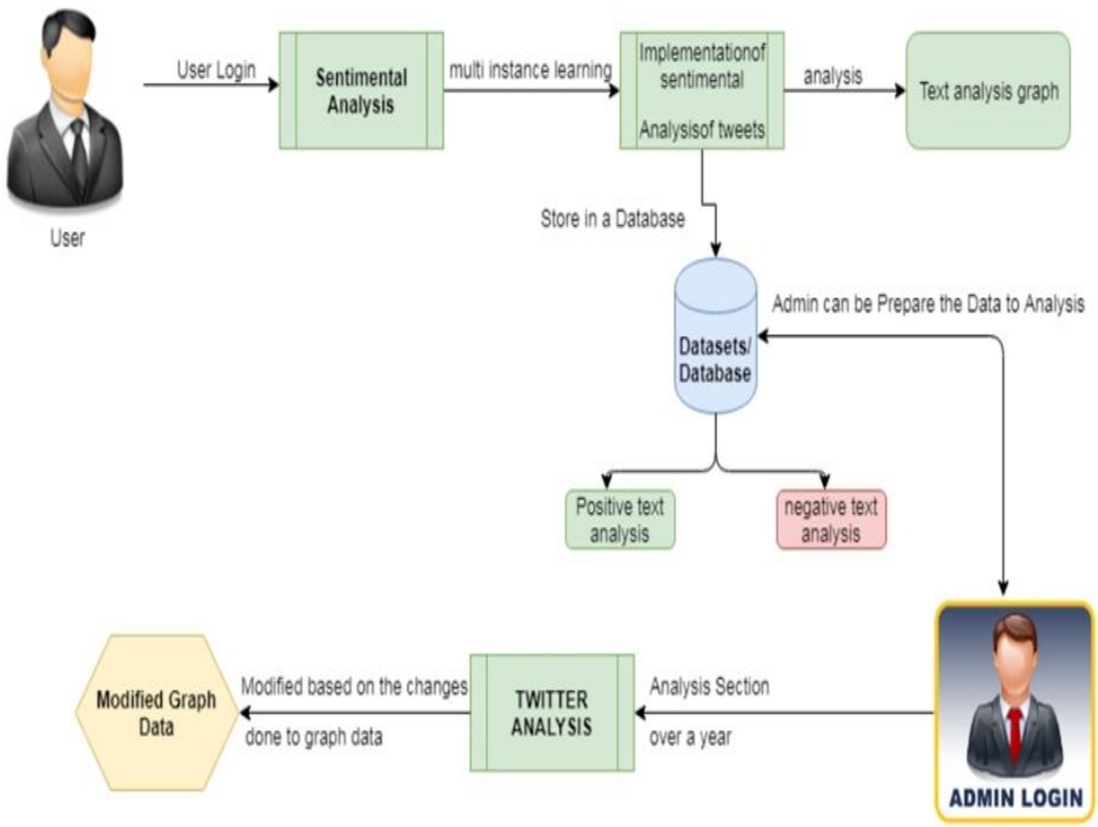


Figure 3.1: Project Architecture of Analysis of women safety in Indian cities using machine learning on tweets

3.2 DESCRIPTION

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 tweets about safety of women and stories of standing up against sexual harassment further motivates other women data on the same social media website or application like Twitter. 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 . 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.

3.3 USE CASE DIAGRAM

In the use case diagram, we have basically one actor who is the user in the trained model. A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has. The use cases are represented by either circles or ellipses.The actors are often shown as stick figures.

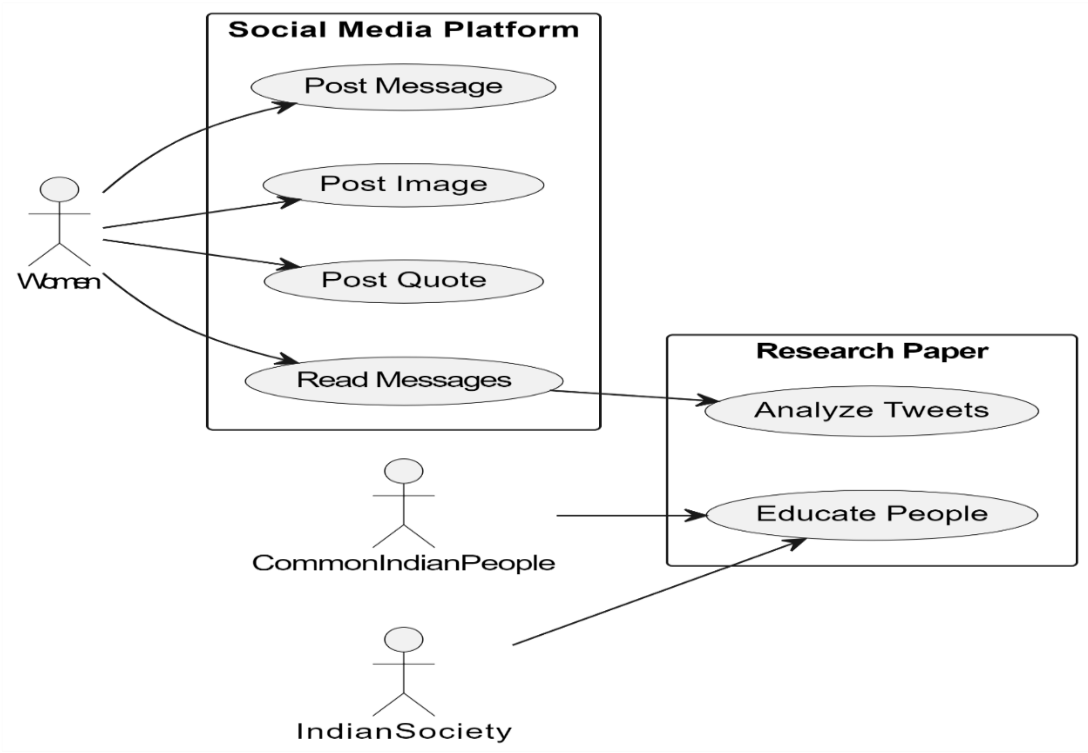


Figure 3.2 : Use Case Diagram of Analysis of women safety in Indian cities Using machine learning on tweets

3.4 CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

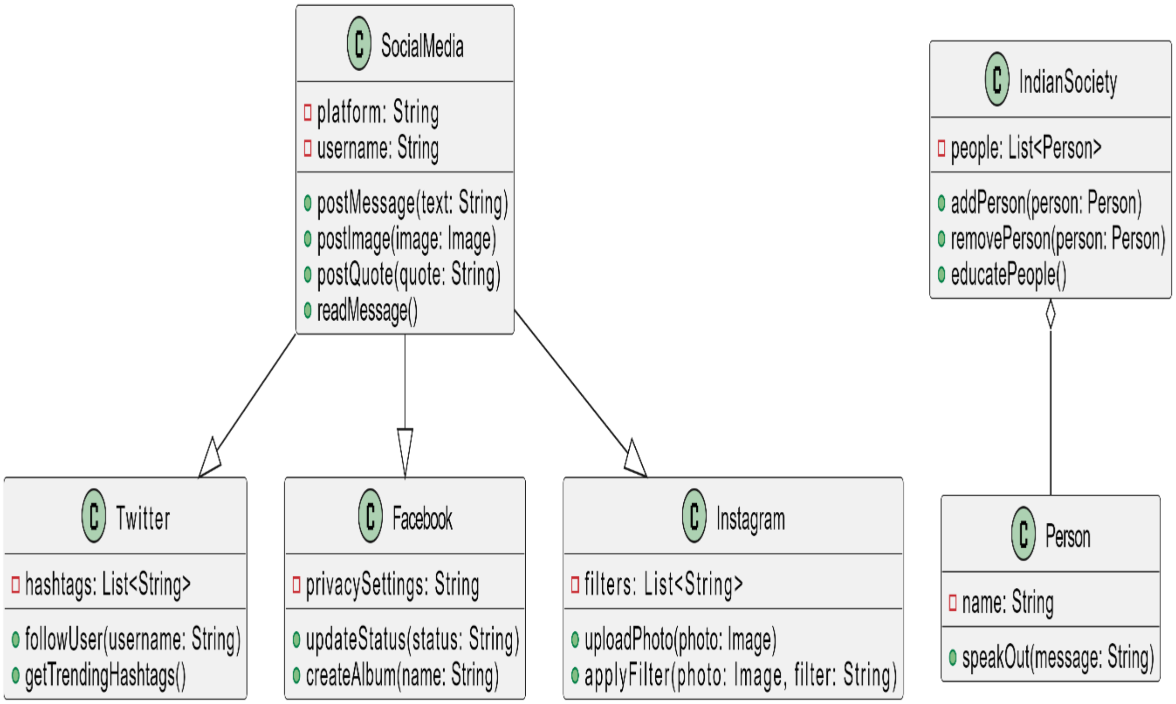


Figure 3.3: Class Diagram of Analysis of women safety in Indian cities using machine learning on tweets

3.5 SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

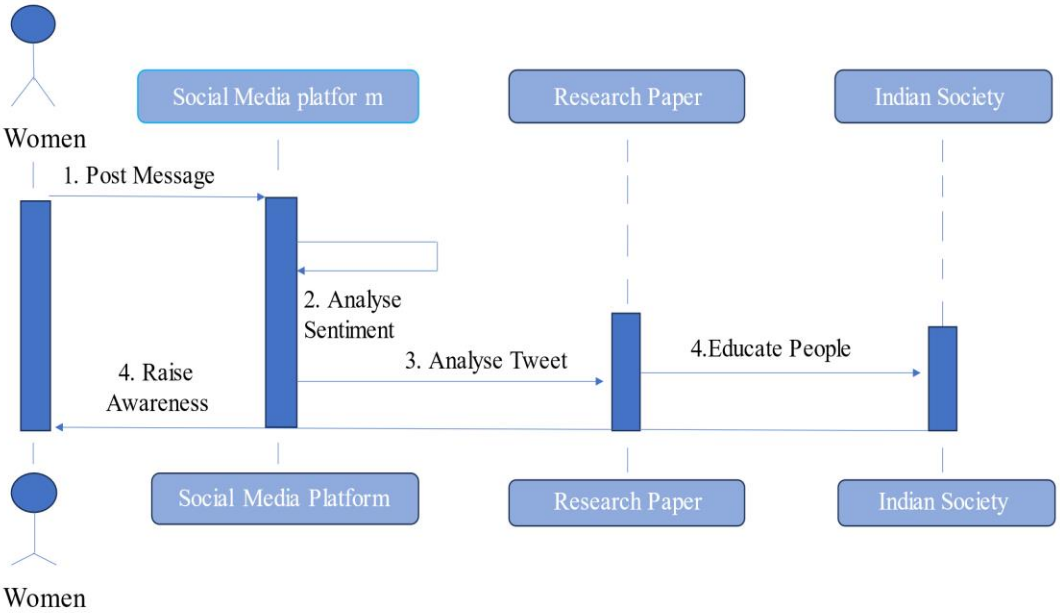


Figure 3.4: Sequence Diagram of Analysis of women safety in Indian cities using machine learning on tweets

3.6 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. They can also include elements showing the flow of data between activities through one or more data stores.

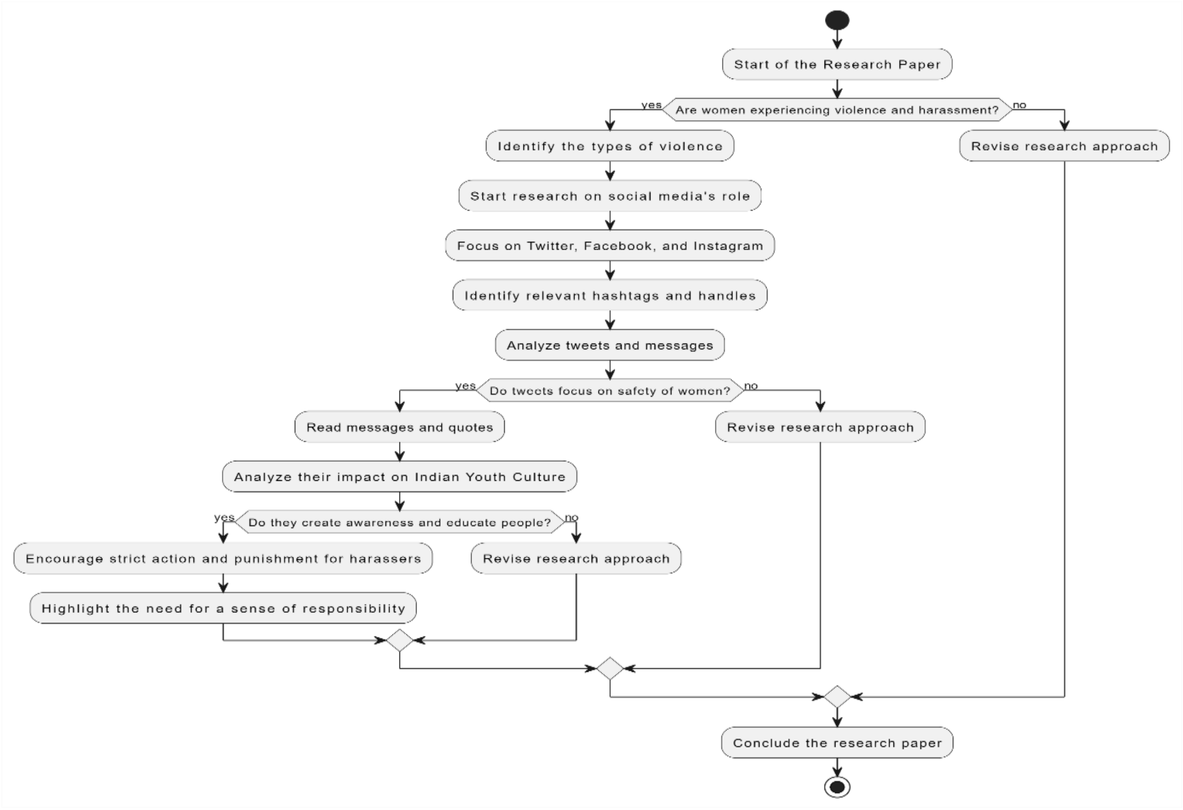

Start of the Research Paper
yes
Are women experiencing violence and harassment?
no
Identify the types of violence
Revise research approach
Start research on social media's role
Focus on Twitter, Facebook, and Instagram
Identify relevant hashtags and handles
Analyze tweets and messages
yes
Do tweets focus on safety of women?
no
Read messages and quotes
Revise research approach
Analyze their impact on Indian Youth Culture
yes
Do they create awareness and educate people?
no
Encourage strict action and punishment for harassers
Revise research approach
Highlight the need for a sense of responsibility
Conclude the research paper

Figure 3.5: Activity Diagram of Analysis of women safety in Indian cities using machine learning on tweets

4.IMPLEMENTATION

4.1 SAMPLE CODE

```
#AnalysisOf_womenSafety.inl
<?xml version="1.0" encoding="UTF-8"?>
<module type="PYTHON_MODULE" version="4">
<component name="NewModuleRootManager">
<content url="file://$MODULE_DIR$" />
<orderEntry type="inheritedJdk" />
<orderEntry type="sourceFolder" forTests="false" />
</component>
<component name="TestRunnerService">
<option name="PROJECT_TEST_RUNNER" value="Unittests" />
</component>
</module>
#misc.xml
<?xml version="1.0" encoding="UTF-8"?>
<project version="4">
<component name="ProjectRootManager" version="2" project-jdk-name="Python 3.6
(AnalysisOf_WomenSafety)" project-jdk-type="Python SDK" />
</project>

#modules.xml
<?xml version="1.0" encoding="UTF-8"?>
<project version="4">
<component name="ProjectModuleManager">
<modules>
<module fileurl="file://$PROJECT_DIR$/.idea/AnalysisOf_WomenSafety.iml"
filepath="$PROJECT_DIR$/.idea/AnalysisOf_WomenSafety.iml" />
</modules>
</component>
</project>

#workspace.xml
```

```
<?xml version="1.0" encoding="UTF-8"?>
<project version="4">
<component name="ChangeListManager">
<list default="true" id="d1892533-137e-4403-9fc5-13444ef1e120" name="Default" comment="" />
<option name="EXCLUDED_CONVERTED_TO_IGNORED" value="true" />
<option name="TRACKING_ENABLED" value="true" />
<option name="SHOW_DIALOG" value="false" />
<option name="HIGHLIGHT_CONFLICTS" value="true" />
<option name="HIGHLIGHT_NON_ACTIVE_CHANGELIST" value="false" />
<option name="LAST_RESOLUTION" value="IGNORE" />
</component>
<component name="FileEditorManager">
<leaf SIDE_TABS_SIZE_LIMIT_KEY="300">
<file leaf-file-name="views.py" pinned="false" current-in-tab="false">
<entry file="file://$PROJECT_DIR$/womensafety/Remote_User/views.py">
<provider selected="true" editor-type-id="text-editor">
<state relative-caret-position="241">
<caret line="19" column="40" lean-forward="false" selection-start-line="19" selection-start-column="29" selection-end-line="19" selection-end-column="40" />
 <folding>
 <element signature="e#0#34#0" expanded="true" />
 </folding>
 </state>
</provider>
```

```
<folding />
</state>
</provider>
</entry>
</file>
<file leaf-file-name="login.html" pinned="false" current-in-tab="false">
<entry file="file://$PROJECT_DIR$/womensafety/Template/htmls/RUser/login.html">
<provider selected="true" editor-type-id="text-editor">
<state relative-caret-position="49">
<caret line="60" column="42" lean-forward="false" selection-start-line="60" selection-start-column="38" selection-end-line="60" selection-end-column="42" />
<folding />
</state>
</provider>
</entry>
</file>
<file leaf-file-name="Register1.html" pinned="false" current-in-tab="false">
<entry
file="file://$PROJECT_DIR$/womensafety/Template/htmls/RUser/Register1.html">
<provider selected="true" editor-type-id="text-editor">
<state relative-caret-position="136">
<caret line="62" column="39" lean-forward="false" selection-start-line="62" selection-start-column="39" selection-end-line="62" selection-end-column="39" />
 <folding />
 </state>
 </provider>
 </entry>
 </file>
 <file leaf-file-name="forms.py" pinned="false" current-in-tab="false">
 <entry file="file://$PROJECT_DIR$/womensafety/Remote_User/forms.py">
 <provider selected="true" editor-type-id="text-editor">
 <state relative-caret-position="34">
 <caret line="2" column="23" lean-forward="false" selection-start-line="2" selection-start-column="17" selection-end-line="2" selection-end-column="23" />
 <folding />
  </state>
  </provider>
  </entry>
  </file>
  <file leaf-file-name="ViewAllTweets.html" pinned="false" current-in-tab="false">
 <entry
file="file://$PROJECT_DIR$/womensafety/Template/htmls/RUser/ViewAllTweets.html"
```

```
'Bangalore'),
(13, 'Ramesh', 'Ramesh@gmail.com', 'Ramesh', '9535866270', 'India', 'Karnataka',
'Bangalore'),
(14, 'tmksmanju', 'tmksmanju13@gmail.com', 'tmksmanju', '9535866270', 'India',
'Karnataka', 'Bangalore'),
(15, 'Gopal', 'Gopal123@gmail.com', 'Gopal', '9535866270', 'India', 'Karnataka',
'Bangalore'),
(16, 'Kokila', 'Kokila123@gmail.com', 'Kokila', '9535866270', 'India', 'Karnataka',
'Bangalore');

-- --------------------------------------------------------

--
-- Table structure for table `remote_user_review_model`
--

CREATE TABLE IF NOT EXISTS `remote_user_review_model` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `uname` varchar(100) NOT NULL,
  `ureview` varchar(100) NOT NULL,
  `sanalysis` varchar(100) NOT NULL,
  `dt` varchar(300) NOT NULL,
  `tname` varchar(250) NOT NULL,
  `suggestion` varchar(250) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;

--
-- Dumping data for table `remote_user_review_model`
--

INSERT INTO `remote_user_review_model` (`id`, `uname`, `ureview`, `sanalysis`, `dt`,
`tname`, `suggestion`) VALUES
(1, 'Gopal', 'The Delhi Govet hast to proper step for this bad activities against women.',
'negative', '2019-12-23 12:08:24.569335', 'Sexual_assaults', 'Really it is wort '),
(2, 'Kumar', 'There is excellent safety for women in Mumbai', 'positive', '2019-12-23
13:38:35.092812', 'Sexual_assaults', 'Want to create better law against this'),
(3, 'Ashok', 'There is nice safety for women in Kolkata', 'positive', '2019-12-23
13:43:29.278359', 'Women_Safety', 'no feedback'),
(4, 'Ramesh', 'Security is ok but want to improve more security in Chennai.', 'neutral',
'2019-12-23 13:47:28.783242', 'Women Security', 'better to improve security'),
(5, 'Manjunath', 'It is better security in chennai for women', 'positive', '2019-12-23
15:06:26.344765', 'Sexual Problems', 'better to improve security');

-- ---------------------------------------------------------
-- Table structure for table `remote_user_usertweets_model
```

```
CREATE TABLE IF NOT EXISTS `remote_user_usertweets_model` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `tdesc` varchar(500) NOT NULL,
  `topics` varchar(300) NOT NULL,
  `sanalysis` varchar(300) NOT NULL,
  `senderstatus` varchar(300) NOT NULL,
  `ratings` int(11) NOT NULL,
  `userId_id` int(11) NOT NULL,
  `usefulcounts` int(11) NOT NULL,
  `uses` varchar(100) NOT NULL,
  `tname` varchar(50) NOT NULL,
  `dislikes` int(11) NOT NULL,
  `uname` varchar(50) NOT NULL,
  `names` varchar(50) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `Remote_user_clientpo_userId_id_ab97a689_fk_Client_Si` (`userId_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;

--
-- Dumping data for table `remote_user_usertweets_model`
--

INSERT INTO `remote_user_usertweets_model` (`id`, `tdesc`, `topics`, `sanalysis`,
`senderstatus`, `ratings`, `userId_id`, `usefulcounts`, `uses`, `tname`, `dislikes`, `uname`,
`names`) VALUES
(13, 'The #Sexual_assaults are very worst in Delhi', 'Sexual_assaults', 'negative', 'process',
6, 13, 0, 'to know about women safety', 'Sexual_assaults', -11, 'Ramesh', 'Delhi'),
(14, 'The Sexual Assaults are very common everywhere..What do u think about
Mumbai.', '', 'neutral', 'process', 9, 10, 0, 'to know about women safety', 'Sexual_assaults',
-6, 'Maran', 'Mumbai'),
(15, 'I thing there is good safety for women in Kolkata', '', 'positive', 'process', 6, 14, 0, 'to
know about women safety', 'Women_Safety', -2, 'tmksmanju', 'Kolkata'),
(16, 'We are very happy to say that there is more safety on women', '', 'positive', 'process',
4, 11, 0, 'to know about women safety', 'Women Security', -1, 'Raghu', 'Chennai'),
(17, 'The women safety is considered the major problems in Capital Cities of India', '',
'neutral', 'process', 3, 16, 0, 'to know about women safety', 'Sexual Problems', -1, 'Kokila',
'Chennai');

--
-- Constraints for dumped tables
--

--
-- Constraints for table `auth_group_permissions`
```

```
--
ALTER TABLE `auth_group_permissions`
  ADD CONSTRAINT `auth_group_permissions_group_id_b120cbf9_fk_auth_group_id`
FOREIGN KEY (`group_id`) REFERENCES `auth_group` (`id`),
  ADD CONSTRAINT `auth_group_permissio_permission_id_84c5c92e_fk_auth_perm`
FOREIGN KEY (`permission_id`) REFERENCES `auth_permission` (`id`);

--
-- Constraints for table `auth_permission`
--
ALTER TABLE `auth_permission`
  ADD CONSTRAINT `auth_permission_content_type_id_2f476e4b_fk_django_co`
FOREIGN KEY (`content_type_id`) REFERENCES `django_content_type` (`id`);

--
-- Constraints for table `auth_user_groups`
--
ALTER TABLE `auth_user_groups`
  ADD CONSTRAINT `auth_user_groups_group_id_97559544_fk_auth_group_id`
FOREIGN KEY (`group_id`) REFERENCES `auth_group` (`id`),
  ADD CONSTRAINT `auth_user_groups_user_id_6a12ed8b_fk_auth_user_id`
FOREIGN KEY (`user_id`) REFERENCES `auth_user` (`id`);
-- Constraints for table `auth_user_user_permissions`
--
ALTER TABLE `auth_user_user_permissions`
  ADD CONSTRAINT
`auth_user_user_permissions_user_id_a95ead1b_fk_auth_user_id` FOREIGN KEY
(`user_id`) REFERENCES `auth_user` (`id`),
  ADD CONSTRAINT `auth_user_user_permi_permission_id_1fbb5f2c_fk_auth_perm`
FOREIGN KEY (`permission_id`) REFERENCES `auth_permission` (`id`);

--
-- Constraints for table `django_admin_log`
--
ALTER TABLE `django_admin_log`
  ADD CONSTRAINT `django_admin_log_content_type_id_c4bce8eb_fk_django_co`
FOREIGN KEY (`content_type_id`) REFERENCES `django_content_type` (`id`),
  ADD CONSTRAINT `django_admin_log_user_id_c564eba6_fk_auth_user_id`
FOREIGN KEY (`user_id`) REFERENCES `auth_user` (`id`);

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION
*/;
```

```
#0001_initial.py
# Generated by Django 2.0.5 on 2019-04-23 07:01

from django.db import migrations, models


class Migration(migrations.Migration):

initial = True

dependencies = [
    ]

  operations = [
  migrations.CreateModel(
  name='ClientRegister_Model',
  fields=[
  ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False,
verbose_name='ID')),
            ('username', models.CharField(max_length=30)),
            ('email', models.EmailField(max_length=30)),
            ('password', models.CharField(max_length=10)),
            ('phoneno', models.IntegerField()),
            ('country', models.CharField(max_length=30)),
            ('state', models.CharField(max_length=30)),
            ('city', models.CharField(max_length=30)),
        ],
      ),
    ]
#0002_usertweets_model.py
from django.db import migrations, models
import django.db.models.deletion


class Migration(migrations.Migration):

    dependencies = [
        ('Remote_User', '0001_initial'),
    ]

    operations = [
    migrations.CreateModel(
     name='usertweets_Model',
     fields=[
     ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False,
verbose_name='ID')),
            ('tdesc', models.CharField(max_length=300)),
            ('uname', models.CharField(max_length=300)),
            ('topics', models.CharField(max_length=300)),
            ('sanalysis', models.CharField(max_length=300)),
            ('senderstatus', models.CharField(default='process', max_length=300)),
            ('ratings', models.IntegerField(default=0)),
            ('userId',
models.ForeignKey(on_delete=django.db.models.deletion.CASCADE,
to='Remote_User.ClientRegister_Model')),
         ],
      ),
   ]
#0003_usertweets_model_usefulcounts.py
from django.db import migrations, models


class Migration(migrations.Migration):

 dependencies = [
      ('Remote_User', '0002_usertweets_model'),
   ]

 operations = [
 migrations.AddField(
  model_name='usertweets_model',
  name='uname',
  field=models.IntegerField(default=0),
      ),
   ]
#0004_auto_20190429_1027.py

from django.db import migrations, models


class Migration(migrations.Migration):
dependencies = [
      ('Remote_User', '0003_usertweets_model_usefulcounts'),
   ]

 operations = [
 migrations.AddField(
 model_name='usertweets_model',
 name='uses',
 field=models.CharField(default='', max_length=100), preserve_default=False,
      ),
        migrations.AddField(
            model_name='clientposts_model',
            name='tname',
            field=models.CharField(default='', max_length=50),
            preserve_default=False,
        ),
    ]
#0005_usertweets_model_dislikes.py

from django.db import migrations, models


class Migration(migrations.Migration):
dependencies = [
        ('Remote_User', '0004_auto_20190429_1027'),
    ]

operations = [
        migrations.AddField(
            model_name='usertweets_model',
            name='dislikes',
            field=models.IntegerField(default=0),
        ),
    ]
#0006_review_model.py

from django.db import migrations, models


class Migration(migrations.Migration):

dependencies = [
        ('Remote_User', '0005_usertweets_model_dislikes'),
    ]
operations = [
migrations.CreateModel(
name='review_Model',
fields=[

                ('uname', models.CharField(max_length=100)),
                ('ureview', models.CharField(max_length=100)),
                ('tname', models.CharField(max_length=300)),
                ('suggestion', models.CharField(max_length=300)),
               ('dt', models.CharField(max_length=300)),
            ('sanalysis', models.CharField(max_length=300)),
          ],
        ),
    ]

#0007_usertweets_model_names.py

from django.db import migrations, models

class Migration(migrations.Migration):

dependencies = [
        ('Remote_User', '0006_review_model'),
    ]

 operations = [
  migrations.AddField(
  model_name='usertweets_model',
   name='uname',
   field=models.CharField(default='', max_length=50),
   preserve_default=False,
        ),
    ]

#admin.py

from django.contrib import admin

#apps.py

from django.apps import AppConfig


class ClientSiteConfig(AppConfig):
name = 'Remote_User'

#forms.py

from django import forms

from Remote_User.models import ClientRegister_Model


class ClientRegister_Form(forms.ModelForm):

password = forms.CharField(widget=forms.PasswordInput())
email = forms.EmailField(required=True)

class Meta:
```

```
model = ClientRegister_Model
fields = ("username","email","password","phoneno","country","state","city")
#models.py

from django.db import models

# Create your models here.
from django.db.models import CASCADE


class ClientRegister_Model(models.Model):
username = models.CharField(max_length=30)
email = models.EmailField(max_length=30)
password = models.CharField(max_length=10)
phoneno = models.CharField(max_length=10)
country = models.CharField(max_length=30)
 state = models.CharField(max_length=30)
 city = models.CharField(max_length=30)
 class usertweets_Model(models.Model):
 userId = models.ForeignKey(ClientRegister_Model, on_delete=CASCADE)
 uname = models.CharField(max_length=300)

 tname= models.CharField(max_length=500)
 uses = models.CharField(max_length=100)
 tdesc = models.CharField(max_length=500)
 topics = models.CharField(max_length=300)
 sanalysis = models.CharField(max_length=300)
 names= models.CharField(max_length=300)
 senderstatus = models.CharField(default="process", max_length=300 )
 ratings = models.IntegerField(default=0)
 usefulcounts = models.IntegerField(default=0)
 dislikes = models.IntegerField(default=0)

 class review_Model(models.Model):
 uname = models.CharField(max_length=100)
 ureview = models.CharField(max_length=100)
 sanalysis = models.CharField(max_length=100)
 dt= models.CharField(max_length=300)
 tname= models.CharField(max_length=300)
 suggestion = models.CharField(max_length=300)
 #tests.py
```

```
from django.test import TestCase

#views.py

from django.db.models import Count
from django.shortcuts import render, redirect, get_object_or_404
import datetime

# Create your views here.
from Remote_User.models import
review_Model,ClientRegister_Model,usertweets_Model


def login(request):
if request.method == "POST" and 'submit1' in request.POST:

username = request.POST.get('username')
password = request.POST.get('password')
try:

enter = ClientRegister_Model.objects.get(username=username, password=password)
request.session["userid"] = enter.id

return redirect('CreateTweet')
except:
pass


return render(request,'RUser/login.html')
def Register1(request):


 if request.method == "POST":
 username = request.POST.get('username')
 email = request.POST.get('email')
 password = request.POST.get('password')
 phoneno = request.POST.get('phoneno')
 country = request.POST.get('country')
  state = request.POST.get('state')
  city = request.POST.get('city')
  ClientRegister_Model.objects.create(username=username, email=email,
password=password, phoneno=phoneno, country=country, state=state, city=cityreturn
render(request, 'RUser/Register1.html')
```

```
 else:

  return render(request,'RUser/Register1.html')


  def ViewYourProfile(request):
   userid = request.session['userid']
   obj = ClientRegister_Model.objects.get(id= userid)
   return render(request,'RUser/ViewYourProfile.html',{'object':obj})

  def Review(request,pk):
   userid = request.session['userid']
   userObj = ClientRegister_Model.objects.get(id=userid)
   username = userObj.username

   objs = usertweets_Model.objects.get(id=pk)
   tname = objs.tname

   datetime_object = datetime.datetime.now()

   result = ''
   pos = []
   neg = []
   oth = []
   se = 'se'
   if request.method == "POST":
   uname = request.POST.get('uname')
   tname1 = request.POST.get('tname')
   suggestion1 = request.POST.get('suggestion')
   cmd = request.POST.get('review')


    if '#' in cmd:
    startingpoint = cmd.find('#')
     a = cmd[startingpoint:]
     endingPoint = a.find(' ')
     title = a[0:endingPoint]
     result = title[1:]
     # return redirect('')

     for f in cmd.split():
     if f in ('good', 'nice', 'better', 'best', 'excellent', 'extraordinary', 'happy', 'won', 'love',
'greate',):
      pos.append(f)
      elif f in ('worst', 'waste', 'poor', 'error', 'imporve', 'bad'):
```

```
 neg.append(f)
 else:
 oth.append(f)
  if len(pos) > len(neg):
  se = 'positive'
  elif len(neg) > len(pos):
  se = 'negative'
  else:
  se = 'neutral'
  review_Model.objects.create(uname=uname ,
ureview=cmd,sanalysis=se,dt=datetime_object,tname=tname1 ,suggestion=suggestion1)

  return render(request,'RUser/Review.html', {'objc':username,'objc1':tname,'result': result,
'se': se})

 def CreateTweet(request):
 userid = request.session['userid']
 userObj = ClientRegister_Model.objects.get(id=userid)
 userid = userObj.username

 result = ''
 pos = []
 neg = []
 oth = []
 se = 'se'
 if request.method == "POST":
 uname = request.POST.get('uname')
 tname = request.POST.get('tname')
 uses = request.POST.get('uses')
 cmd = request.POST.get('tdesc')
 mcity = request.POST.get('mcity')

 if '#' in cmd:
 startingpoint = cmd.find('#')
 a = cmd[startingpoint:]
 endingPoint = a.find(' ')
 title = a[0:endingPoint]
 result = title[1:]
 # return redirect('')

 for f in cmd.split():
  if f in ('good', 'nice', 'better', 'best', 'excellent', 'extraordinary', 'happy', 'won',
'love','greate'):
  pos.append(f)
  elif f in ('worst', 'waste', 'poor', 'error', 'imporve', 'bad'):
   neg.append(f)
   else:
  oth.append(f)
  if len(pos) > len(neg):
  se = 'positive'
  elif len(neg) > len(pos):
  se = 'negative'
  else:
  se = 'neutral'
  usertweets_Model.objects.create(userId=userObj,uname=uname ,tname=tname
,uses=uses, tdesc=cmd, topics=result, sanalysis=se,
  senderstatus='process',names=mcity)

  return render(request,'RUser/CreateTweet.html', {'objc':userid,'result': result, 'se': se})

 def ViewAllTweets(request):
 userid = request.session['userid']
 obj = usertweets_Model.objects.all()

 return render(request,'RUser/ViewAllTweets.html',{'list_objects': obj})

 def Viewreviews(request):

 obj = review_Model.objects.all()

 return render(request,'RUser/Viewreviews.html',{'list_objects': obj})

 def ratings(request,pk):
 vott1, vott, neg = 0, 0, 0
 objs = usertweets_Model.objects.get(id=pk)
 unid = objs.id
 vot_count = usertweets_Model.objects.all().filter(id=unid)
 for t in vot_count:
 vott = t.ratings
 vott1 = vott + 1
  obj = get_object_or_404(usertweets_Model, id=unid)
  obj.ratings = vott1
  obj.save(update_fields=["ratings"])
  return redirect('ViewAllTweets')

  return render(request,'RUser/ratings.html',{'objs':vott1})
```

```
def dislikes(request,pk):
vott1, vott, neg = 0, 0, 0
objs = usertweets_Model.objects.get(id=pk)
unid = objs.id
vot_count = usertweets_Model.objects.all().filter(id=unid)
for t in vot_count:
vott = t.dislikes
vott1 = vott - 1
 obj = get_object_or_404(usertweets_Model, id=unid)
 obj.dislikes = vott1
 obj.save(update_fields=["dislikes"])
 return redirect('ViewAllTweets')
 return render(request,'RUser/dislikes.html',{'objs':vott1})

 def ViewTrending(request):
 topic =
usertweets_Model.objects.values('topics').annotate(dcount=Count('topics')).order_by('-
dcount')
return render(request, 'RUser/ViewTrending.html', {'objects': topic})
```

#charts.html

```
{% extends 'TServer/design1.html' %}
{% block researchblock %}

<style>
 body{
background-color: #FFFFFF;
 }
 .menu table{
  width:100%;
   text-align:center;

    }

  .menu table td:hover{
   background:rgb(0,0,0);
    }

  .menu table td{
  background: #584b4f;
```

```
.menu table,.menu table th,.menu table td {
 border: ;
 border-collapse: collapse;
 }

 .menu table th,.menu table  td {
  padding: 15px;
   }
  .topic h1{
   color:white;
    padding:2px;
    text-align:center;
    border-style:none;
    height:100px;
    width:1330px;
     float:left;
     }
    #chartContainer{
     width: 900px;
     margin-left: 365px;
     margin-top: -354px;
      position: fixed;
     }
    .sidemenu{
     margin-top: 118px;
     }
    </style>
    <script>
     window.onload = function() {
     var chart = new CanvasJS.Chart("chartContainer", {

      animationEnabled: true,

      title: {
      text: ""
      },
      data: [{
      {% if chart_type == "line" %}
      type: "line",
     {% elif chart_type == "pie" %}
     type: "pie",

     {% elif chart_type == "spline" %}
     type: "spline",
{% endif %}
startAngle: 240,
yValueFormatString: "##0.00\"%\"",
indexLabel: "{label} {y}",
dataPoints: [
{% for o in form %}
{y: {{o.dcount}}, label: "{{o.names}}"},
{% endfor %}
]
}]
});
chart.render();

}

</script>
<body>
<div class="topic">



<div class="sidemenu">
 <h3 style="color:black;margin-left:60px;margin-top:80px"><a href="{% url 'charts' 'pie'
%}" >PIE CHAERT</a></h3>

 <h3 style="color:white;margin-left:60px;margin-top:100px;"><a href="{% url 'charts'
'line' %}">LINE CHART</a></h3>
 <h3 style="color:white;margin-left:60px;margin-top:100px;"><a href="{% url 'charts'
'spline' %}"></a></h3>
</div>
<div id="chartContainer" class="graph"></div>
<script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>
</div>

</body>
{% endblock %}
```

#design1.html

```
<!DOCTYPE html>

<html lang="en">
<head>
<meta charset="UTF-8">
<title>Tweet Server</title>
<link href="https://fonts.googleapis.com/css?family=Russo+One" rel="stylesheet">
```

```
<style>
body{
 background-size:cover;
 font-family: 'Russo One', sans-serif;
 background-color: #FFFFFF;
 }
 h1{
 color:white;
 }
.topnav {
overflow: hidden;
 background-color: #812;
}
.topnav a {
float: left;
color: #FFFFFF;
text-align: center;
padding: 14px 16px;
text-decoration: none;
font-size: 17px;
}

.topnav a:hover {
 background-color: #ddd;
 color: black;
 }

.topnav a.active {
background-color: #8e4fd1;
color: white;
}
.style1 {
color: #FF0000;
font-weight: bold;
}
</style>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"></head>
<body>

<h1 class="style1">Analysis of Women Safety in Indian Cities Using Machine Learning
on Tweets</h1>

<div  class="topnav">
```

```
 <td><a class="active" href="{% url 'Viewalltweets' %}">VIEW ALL
TWEETS</a></td>
 <a href="{% url 'View_Senti_Reviews' %}">VIEW ALL SENTIMENTS ON
REVIEWS</a>
<td bgcolor="#00FF00"><a href="{% url 'charts' 'line' %}">VIEW LIKES(SAFETY)
RESULTS</a></td>

<td><a href="{% url 'dislikeschart' 'bar' %}">VIEW DISLIKE(NO SAFETY)
RESULTS</a></td>
<td>  <a href="{% url 'viewallclients' %}">VIEW ALL REMOTE USERS</a></td>
<td>  <a href="{% url 'View_Senti_Analysis' %}">VIEW SENTIMENT ANALYSIS ON
TWEETS</a></td>
 <td><a  href="{% url 'ViewTrendings' %}">VIEW TRENDING NEWS </a>
 <td><a  href="{% url 'View_User_Reviews' %}">VIEW ALL USERS REVIEWS </a>
 <td><a href="{% url 'login'  %}">LOGOUT</a></td>
 </div>
 <div class="mainholder">
{% block researchblock %}
{% endblock %}
</div>
</body>
</html>
```

#dislikeschart.html

```
{% extends 'TServer/design1.html' %}
{% block researchblock %}

<style>
 body{
 background-color: #FFFFFF;
 }
 .menu table{
  width:100%;
  text-align:center;

   }

  .menu table td:hover{
  background:rgb(0,0,0);
    }

   .menu table td{
  background: #584b4f;
```

```
    }

    .menu table,.menu table th,.menu table td {
    border: ;
     border-collapse: collapse;
    }

    .menu table th,.menu table  td {
    padding: 15px;
    }
    .topic h1{
     color:white;
     padding:2px;
     text-align:center;
     border-style:none;
     height:100px;
     width:1330px;
     float:left;
    }
    #chartContainer{
    width: 900px;
    margin-left: 365px;
    margin-top: -354px;
    position: fixed;
    }
 .sidemenu{
margin-top: 118px;
 }
  </style>

  <script>
  window.onload = function() {

  var chart = new CanvasJS.Chart("chartContainer", {
  animationEnabled: true,
  title: {
  text: ""
   },
  data: [{
  {% if dislike_chart == "bar" %}
   type: "bar",
  {% elif dislike_chart == "pie" %}
  type: "pie",
  {% elif dislike_chart == "spline" %}
type: "spline",
{% endif %}
startAngle: 240,
yValueFormatString: "##0.00\"%\"",
indexLabel: "{label} {y}",
dataPoints: [
{% for o in form %}
{y: {{o.dcount}}, label: "{{o.names}}"},
{% endfor %}
]
}]
});
chart.render();

}

</script>
<body>
<div class="topic">
<div class="sidemenu">
   <h3 style="color:black;margin-left:60px;margin-top:80px"><a href="{% url
'dislikeschart' 'pie' %}" >PIE CHAERT</a></h3>
   <h3 style="color:white;margin-left:60px;margin-top:100px;"><a href="{% url
'dislikeschart' 'bar' %}">BAR CHART</a></h3>
   <h3 style="color:white;margin-left:60px;margin-top:100px;"><a href="{% url
'dislikeschart' 'coloumn' %}"></a></h3>
</div>
<div id="chartContainer" class="graph"></div>
<script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>
</div>

</body>
{% endblock %}
```

#tweetserverlogin.html

```
<link href="//maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
rel="stylesheet" id="bootstrap-css">

<!DOCTYPE html>

<html lang="en">
```

```
<title>Login</title>

<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">

<head>

<link rel="icon" href="images/icon.png" type="image/x-icon" />

<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Righteous" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Fredoka+One" rel="stylesheet">

<style>
body {background-color:#eee;}
.container-fluid {padding:50px;}
.container{background-color:white;padding:50px;   }
#title{font-family: 'Fredoka One', cursive;
}
.text-uppercase{
font-family: 'Righteous', cursive;

}
.style1 {
color: #FF0000;
font-weight: bold;
}
 .style2 {
font-family: 'Righteous', cursive;
color: #FF0000;
font-weight: bold;
}
  </style>
</head>
<body>
<div class="container-fluid">
<div class="container">
<h2 class="text-center style1" id="title">Analysis of Women Safety in Indian Cities
Using Machine Learning on Tweets</h2>
 <p class="text-center">
```

```
                                        <span class="style1"><small id="passwordHelpInline" class="text-muted">Women, Safety,Sexual Harassment,Hash tag,Sentimental Analysis.. </small></span> </p>
 <hr>
<div class="row">
<div class="col-md-2">
<!-------null------>
</div>
<div class="col-md-5">
 <form method="POST" role="form">
<fieldset>
<p class="style2"> Login Tweet Server: </p>
<div class="form-group">
<input type="text" name="admin"  class="form-control input-lg" placeholder="User Name" required>
</div>
<div class="form-group">
<input type="password" name="password"     class="form-control input-lg" placeholder="Password" required>
</div>
<div>
<input type="submit" name="submit1"  class="btn btn-md" value="Login Tweet Server">
</div>
 </fieldset>
<br><br>
button class="btn btn-lg "><a href="{% url 'login' %}">User Login</a></button>
</form>
</div>
</div>
</div>

</div>
          </body>

</html>
```

#view_senti_analysis.html

```
{% extends 'TServer/design1.html' %}
{% block researchblock %}

<link rel="icon" href="images/icon.png" type="image/x-icon" />

<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Righteous" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Fredoka+One" rel="stylesheet">

<style>
body {background-color:#FFFFFF;}
.container-fluid {padding:50px;}
.container{background-color:white;padding:50px;   }
#title{font-family: 'Fredoka One', cursive;
}
.text-uppercase{
font-family: 'Righteous', cursive;

}
input{
font-family:'Russo One', sans-serif;
font-size:15px;
}

.style1 {
color: #FF0000;
font-weight: bold;
}
 .style2 {color: #FFFF00}
  </style>

<body>
<div class="container-fluid">
<div class="container">

<div class="row">
<div class="col-md-5">
 <form role="form" method="POST" >
{% csrf_token %}
```

```
 <span class="style1">SELECT CITY NAME::</span>
 <select name="mcity">
<option>--- Select ---</option>
 <option>Delhi</option>
  <option>Mumbai</option>
   <option>Kolkata</option>
   <option>Chennai</option>
   </select>

   <input name="submit" type="submit" class="style1">
   </form>
    <br>
     <br>

   <form role="form" method="POST" >
  {% csrf_token %}
<fieldset>
<p class="text-uppercase pull-center style1">VIEW SENTIMENT ANALYSIS ON
CLIENT POSTS !!! </p>
  <hr>
 <div class="userdetails">
  <table border="1">
  <tr>
 <th bgcolor="#FF0000"><span class="style2">USER NAME</span></th>
  <th bgcolor="#FF0000"><span class="style2">TWEET NAME</span></th>
  <th bgcolor="#FF0000"><span class="style2">TWEET DESC</span></th>
  <th bgcolor="#FF0000"><span class="style2">USES</span></th>
   <th bgcolor="#FF0000"><span class="style2">TWEET SENTIMENT
ANALYSIS</span></th>
   <th bgcolor="#FF0000"><span class="style2">CITY NAME</span></th>
    </tr>
    {% for i in objects %       }

 <tr>
<td bgcolor="#FFFF00" style="color:black" ><span
class="style1">{{i.uname}}</span></td>
<td bgcolor="#FFFF00" style="color:black" ><span
class="style1">{{i.tname}}</span></td>
 <td bgcolor="#FFFF00" style="color:red"><span
class="style1">{{i.tdesc}}</span></td>
```

```
<td bgcolor="#FFFF00" style="color:red"><span
class="style1">{{i.names}}</span></td>
</tr>
{% endfor %}
 </table>
 </div>
 <div class="sideimage"></div>

  </fieldset>
  </form>
  </div>
  <div class="col-md-2">
  <!-------null------>
 </div>
 </div>
</div>
</div>
{% endblock %}
```

```
#view_senti_review.html

{% extends 'TServer/design1.html' %}
{% block researchblock %}

<link rel="icon" href="images/icon.png" type="image/x-icon" />

<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Righteous" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Fredoka+One" rel="stylesheet">
<style>
body {background-color:#FFFFFF;}
.container-fluid {padding:50px;}
.container{background-color:white;padding:50px;   }
#title{font-family: 'Fredoka One', cursive;
}
.text-uppercase{
```

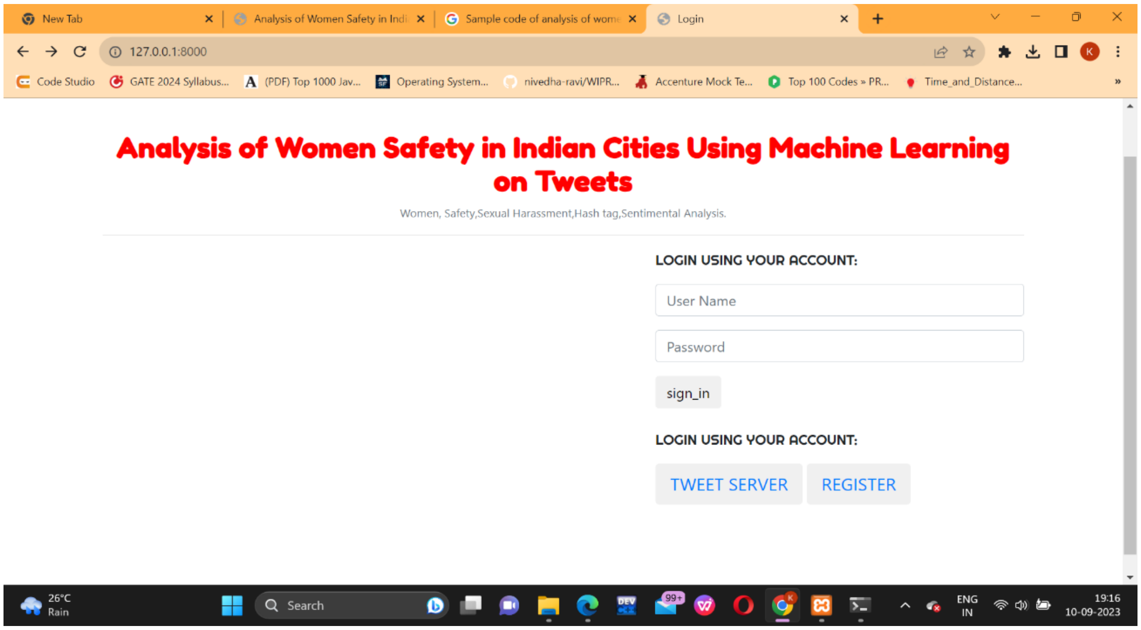
```
font-family: 'Righteous', cursive;

}
.tweettext{

   border: 2px solid yellowgreen;
   width: 1104px;
   height: 442px;
   overflow: scroll;
   background-color:;
}

.style1 {color: #FF0000}
.style3 {color: #FF0000; font-weight: bold; }
 </style>
 <body>
 <div class="container-fluid">
<div class="container">
 <div class="row">
  <div class="col-md-5">
  <form role="form" method="POST" >
 {% csrf_token %}
  <span class="style3">SELECT SENTIMENT TYPE::</span>
  <select name="stype">
  <option>--- Select ---</option>
  <option>Positive</option>
  <option>Negative</option>
  <option>Neutral</option>
  </select>
  <input name="submit" type="submit" class="style1">
  </form>
  <br>
 <br>
 <form role="form" method="POST" >
{% csrf_token %}
<fieldset>
<p class="style1 pull-center text-uppercase"><strong>VIEW ALL SENTIMENT
ANALYSIS BASED ON USERS REVIEWS !!! </strong></p>
```

5.SCREENSHOTS



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

Women, Safety,Sexual Harassment,Hash tag,Sentimental Analysis.

LOGIN USING YOUR ACCOUNT:

User Name

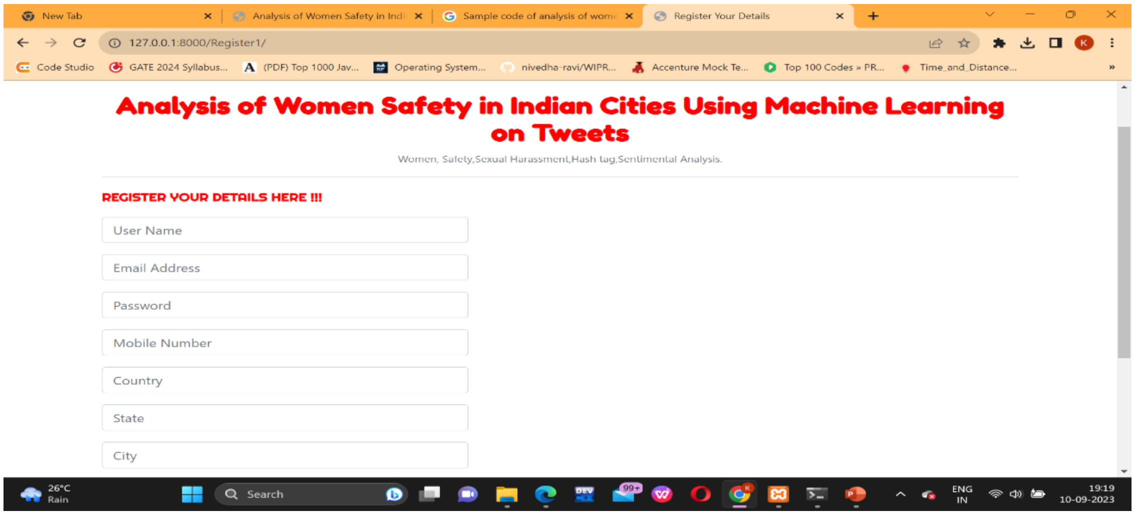
Password

sign_in

LOGIN USING YOUR ACCOUNT:

TWEET SERVER REGISTER

Screenshot 5.1: Login page



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

Women, Safety,Sexual Harassment,Hash tag,Sentimental Analysis.

REGISTER YOUR DETAILS HERE !!!

User Name

Email Address

Password

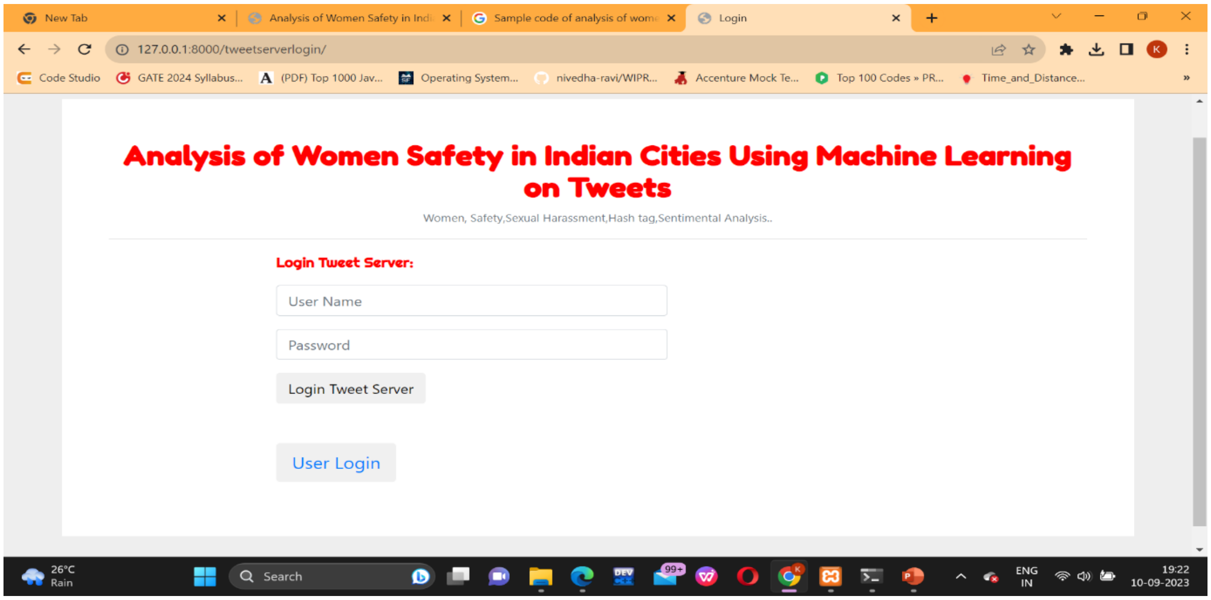
Mobile Number

Country

State

City

Screenshot 5.1: Registration Page

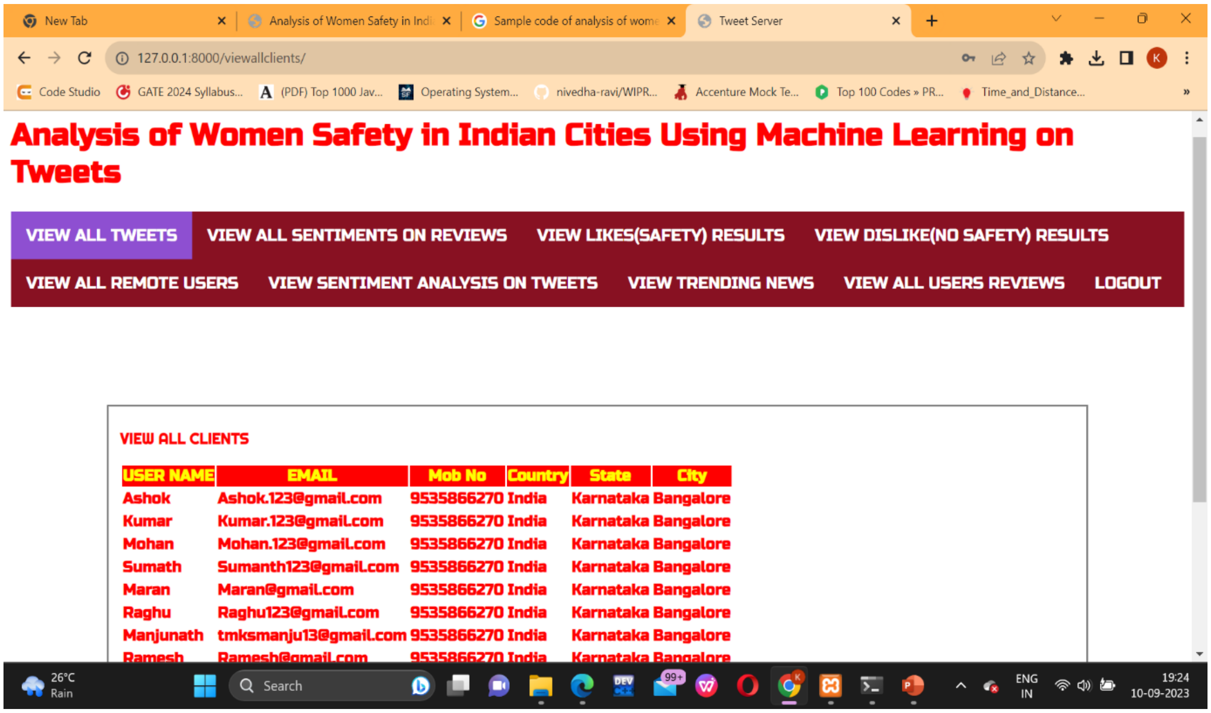


Analysis of Women Safety in Indian Cities Using Machine Learning on Tweets

Women, Safety,Sexual Harassment,Hash tag,Sentimental Analysis..

Login Tweet Server:

Screenshot 5.2 : Login Tweet Server



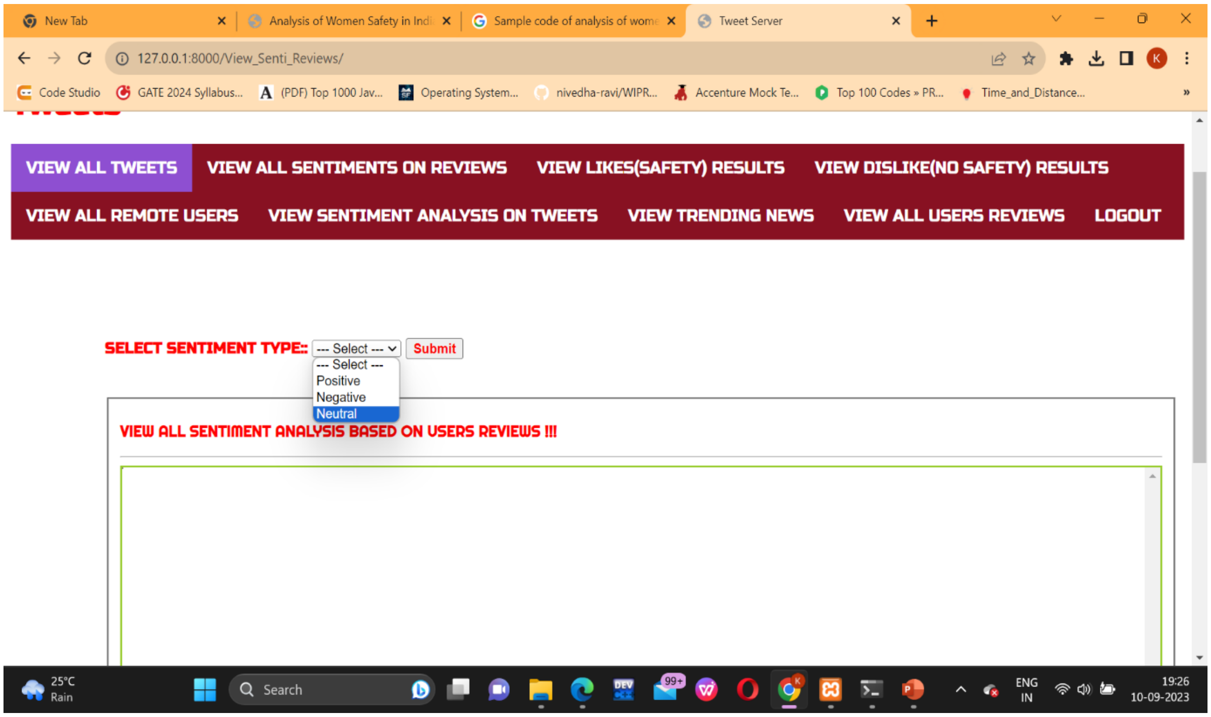
Analysis of Women Safety in Indian Cities Using Machine Learning on Tweets

127.0.0.1:8000/viewallclients/
VIEW ALL TWEETS
VIEW ALL SENTIMENTS ON REVIEWS
VIEW LIKES(SAFETY) RESULTS
VIEW DISLIKE(NO SAFETY) RESULTS
VIEW ALL REMOTE USERS
VIEW SENTIMENT ANALYSIS ON TWEETS
VIEW TRENDING NEWS
VIEW ALL USERS REVIEWS
LOGOUT

VIEW ALL CLIENTS

USER NAME	EMAIL	Mob No	Country	State	City
Ashok	Ashok.123@gmail.com	9535866270	India	Karnataka	Bangalore
Kumar	Kumar.123@gmail.com	9535866270	India	Karnataka	Bangalore
Mohan	Mohan.123@gmail.com	9535866270	India	Karnataka	Bangalore
Sumath	Sumanth123@gmail.com	9535866270	India	Karnataka	Bangalore
Maran	Maran@gmail.com	9535866270	India	Karnataka	Bangalore
Raghu	Raghu123@gmail.com	9535866270	India	Karnataka	Bangalore
Manjunath	tmksmanju13@gmail.com	9535866270	India	Karnataka	Bangalore
Ramesh	Ramesh@gmail.com	9535866270	India	Karnataka	Bangalore

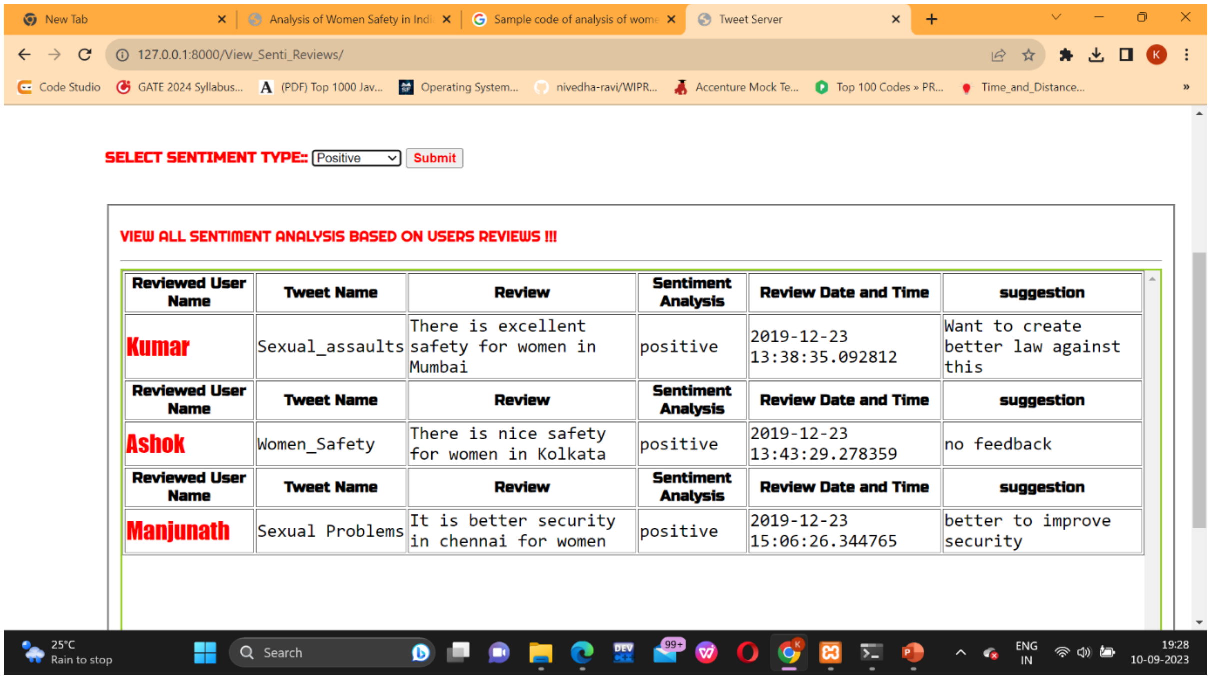
Screenshot 5.3 : View All Tweets



SELECT SENTIMENT TYPE:: --- Select --- Submit

--- Select ---
Positive
Negative
Neutral

VIEW ALL SENTIMENT ANALYSIS BASED ON USERS REVIEWS !!!

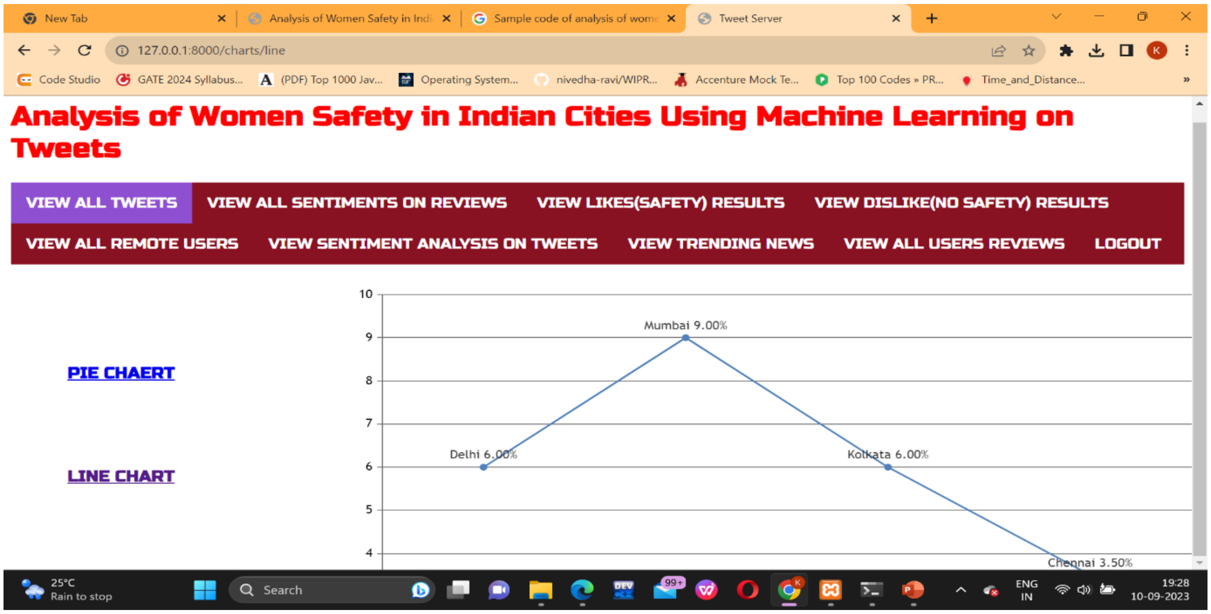


SELECT SENTIMENT TYPE:: Positive Submit

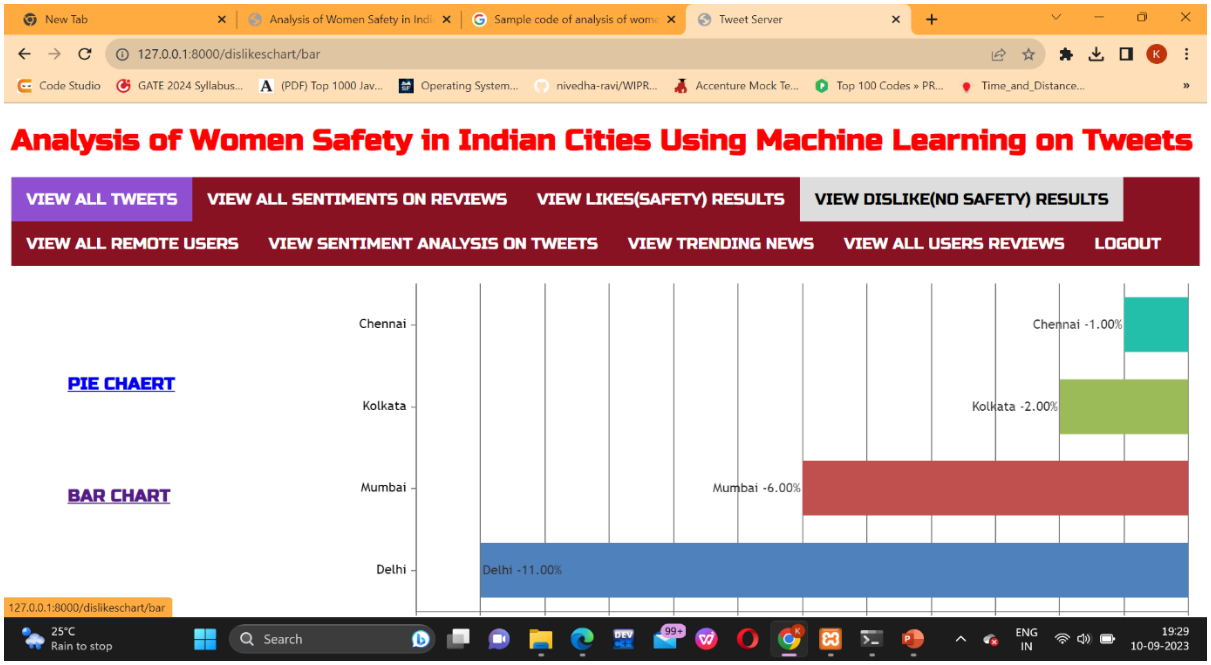
VIEW ALL SENTIMENT ANALYSIS BASED ON USERS REVIEWS !!!

Reviewed User Name	Tweet Name	Review	Sentiment Analysis	Review Date and Time	suggestion
Kumar	Sexual_assaults	There is excellent safety for women in Mumbai	positive	2019-12-23 13:38:35.092812	Want to create better law against this
Reviewed User Name	**Tweet Name**	**Review**	**Sentiment Analysis**	**Review Date and Time**	**suggestion**
Ashok	Women_Safety	There is nice safety for women in Kolkata	positive	2019-12-23 13:43:29.278359	no feedback
Reviewed User Name	**Tweet Name**	**Review**	**Sentiment Analysis**	**Review Date and Time**	**suggestion**
Manjunath	Sexual Problems	It is better security in chennai for women	positive	2019-12-23 15:06:26.344765	better to improve security

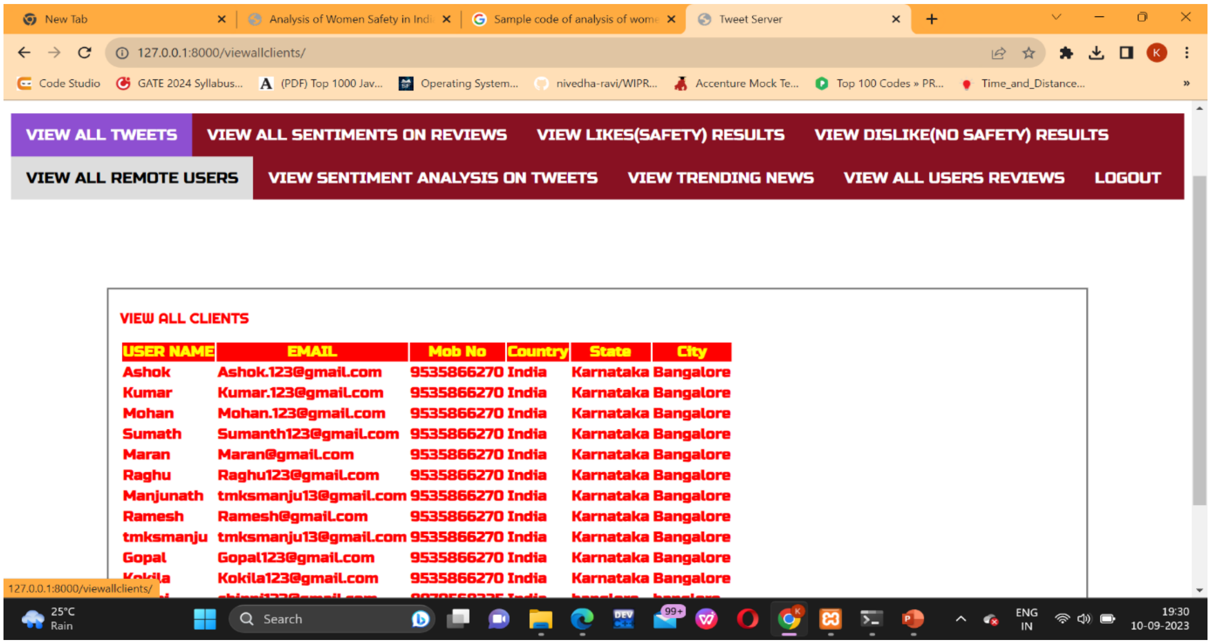
Screenshot 5.4 : View All Sentiments On Reviews



Screenshot 5.5 : View Like Results



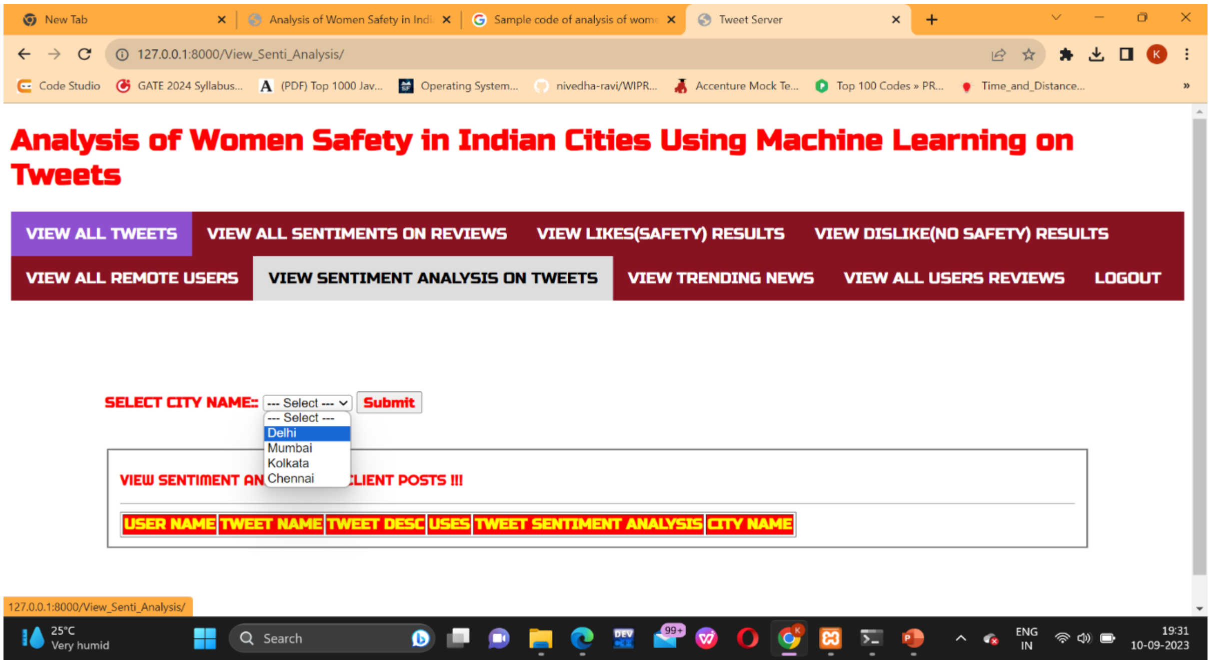
Screenshot 5.6 : View Dislike Results



VIEW ALL CLIENTS

USER NAME	EMAIL	Mob No	Country	State	City
Ashok	Ashok.123@gmail.com	9535866270	India	Karnataka	Bangalore
Kumar	Kumar.123@gmail.com	9535866270	India	Karnataka	Bangalore
Mohan	Mohan.123@gmail.com	9535866270	India	Karnataka	Bangalore
Sumath	Sumanth123@gmail.com	9535866270	India	Karnataka	Bangalore
Maran	Maran@gmail.com	9535866270	India	Karnataka	Bangalore
Raghu	Raghu123@gmail.com	9535866270	India	Karnataka	Bangalore
Manjunath	tmksmanju13@gmail.com	9535866270	India	Karnataka	Bangalore
Ramesh	Ramesh@gmail.com	9535866270	India	Karnataka	Bangalore
tmksmanju	tmksmanju13@gmail.com	9535866270	India	Karnataka	Bangalore
Gopal	Gopal123@gmail.com	9535866270	India	Karnataka	Bangalore
Kokila	Kokila123@gmail.com	9535866270	India	Karnataka	Bangalore

Screenshot 5.7 : View All Remote Users



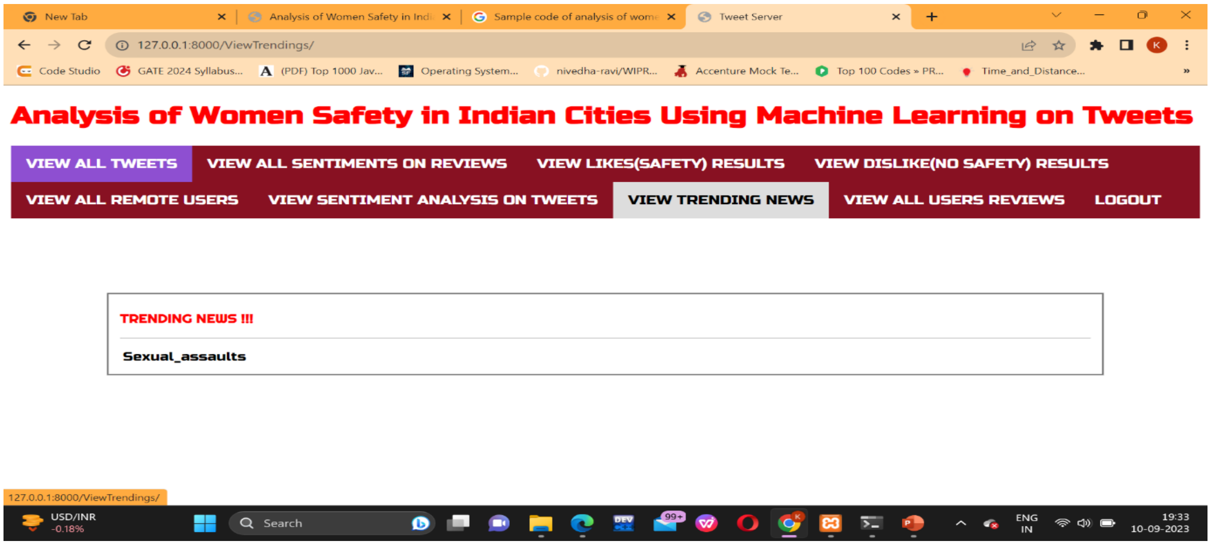
Analysis of Women Safety in Indian Cities Using Machine Learning on Tweets

SELECT CITY NAME:

VIEW SENTIMENT AN LIENT POSTS !!!

USER NAME	TWEET NAME	TWEET DESC	USES	TWEET SENTIMENT ANALYSIS	CITY NAME

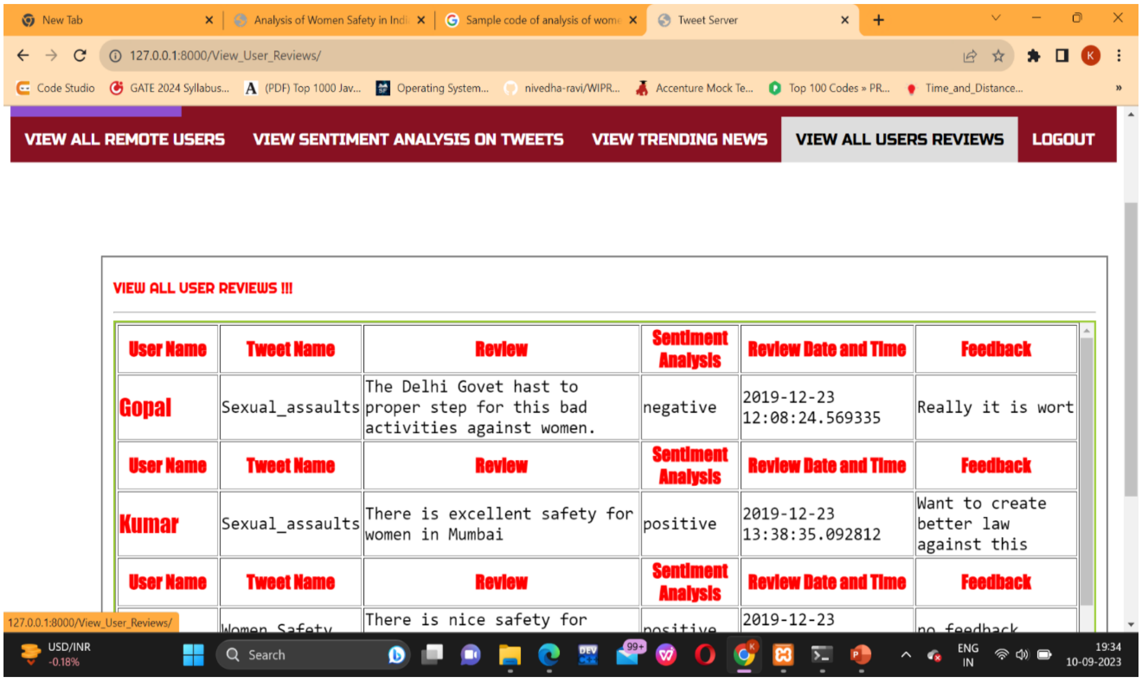
Screenshot 5.8 : View Sentiment Analysis On Tweets


New Tab
Analysis of Women Safety in Indi
Sample code of analysis of wome
Tweet Server
127.0.0.1:8000/ViewTrendings/
Code Studio
GATE 2024 Syllabus...
(PDF) Top 1000 Jav...
Operating System...
nivedha-ravi/WIPR...
Accenture Mock Te...
Top 100 Codes » PR...
Time_and_Distance...
Analysis of Women Safety in Indian Cities Using Machine Learning on Tweets
VIEW ALL TWEETS
VIEW ALL SENTIMENTS ON REVIEWS
VIEW LIKES(SAFETY) RESULTS
VIEW DISLIKE(NO SAFETY) RESULTS
VIEW ALL REMOTE USERS
VIEW SENTIMENT ANALYSIS ON TWEETS
VIEW TRENDING NEWS
VIEW ALL USERS REVIEWS
LOGOUT
127.0.0.1:8000/ViewTrendings/
USD/INR
-0.18%
Search
ENG
IN
19:33
10-09-2023

TRENDING NEWS !!!

Sexual_assaults

Screenshot 5.9 : View Trending News



VIEW ALL USER REVIEWS !!!

User Name	Tweet Name	Review	Sentiment Analysis	Review Date and Time	Feedback
Gopal	Sexual_assaults	The Delhi Govet hast to proper step for this bad activities against women.	negative	2019-12-23 12:08:24.569335	Really it is wort
User Name	Tweet Name	Review	Sentiment Analysis	Review Date and Time	Feedback
Kumar	Sexual_assaults	There is excellent safety for women in Mumbai	positive	2019-12-23 13:38:35.092812	Want to create better law against this
User Name	Tweet Name	Review	Sentiment Analysis	Review Date and Time	Feedback
	Women_Safety	There is nice safety for	positive	2019-12-23	no feedback

Screenshot 6.0 : View All Users Reviews

6.TESTING

6 .TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application . It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.

6.3 TEST CASES

6.3.1 CLASSIFICATION

Test Case ID	Test Case Name	Purpose	Input	Output
1	User Register	User gets Registered	The user gives the input in the form of data	The data gets registered in dataset
2	User Login	User gets login	The user provides the login details	Successfully logged in
3	Login tweet server	Need to login to server	Gives the server password and name	Successfully directed to server page
4	View all tweets	All tweets can be seen in server	Go through dataset	Successfully data seen in server
5	View all sentiments on reviews	Go through positive, negative , neutral reviews	Depends on selection of either of three	Successfully gives the output depends on selection
6	View like & dislike result	Analysis of tweets made by women	Go through the dataset	Like gives Linechart& dislike gives Barchart

7.CONCLUSION

7. CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

The 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 of large amount 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 the Twitter and get an idea about the status of women safety in Indian cities. Presents a global picture of various crimes against women and how their goals and motivations are structured.

7.2 FUTURE SCOPE

Analyzing women's safety in Indian cities through machine learning applied to tweets presents a promising avenue with significant future potential. Leveraging this approach, we can anticipate more sophisticated predictive models that continually refine their accuracy in identifying safety concerns and risk areas. Real-time monitoring of tweets could provide invaluable, up-to-the-minute insights into the safety landscape within cities, benefiting residents and law enforcement alike. Furthermore, this method allows for crowdsourced data to complement official crime statistics, facilitating the early detection of emerging safety trends. Implementing alert systems based on machine learning algorithms could facilitate rapid responses to incidents and proactive measures to enhance security. By identifying influential voices in the online discourse, community engagement efforts can be focused effectively, leading to positive social impact initiatives aimed at making cities safer for women. However, ethical considerations regarding privacy and bias must be a priority in the development of these systems to ensure fairness and data protection. As machine learning integrates with other data sources and gains traction, it can contribute to comprehensive solutions that address women's safety comprehensively.

8. BIBLIOGRAPHY

8. BIBLIOGRAPHY

8.1 REFERENCES

[1] 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.

[2] 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.

[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.

8.2 GITHUB LINK

https://github.com/207r1a05p6/Miniproject