TWITTER HATE SPEECH DETECTION USING MACHINE LEARNING

A PROJECT REPORT

submitted By

AVANI S GIRISH

TVE21MCA-2027

to

the APJ Abdul Kalam Technological University
in partial fullfilment of the requirements for the award of the degree

of

Master of Computer Applications

Department of Computer Applications
College of Engineering
Trivandrum-695016

NOVEMBER 2022

Declaration

I undersigned hereby declare that the project report titled ”Twitter Hate Speech Detection Using Machine Learning” submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Smt. Baby Syla L, Asst.Professor. This submission represents my ideas in my words and where ideas or words of others have been included. I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity as directed in the ethics policy of the college and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and/or University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title.

Place : Trivandrum **AVANI S GIRISH**
Date : 18/11/2022

DEPARTMENT OF COMPUTER APPLICATIONS

COLLEGE OF ENGINEERING
TRIVANDRUM

CERTIFICATE

This is to certify that the report entitled **TWITTER HATE SPEECH DETECTION** submitted by **AVANI S GIRISH** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications is a bonafide record of the project work carried out by her under my guidance and supervision. This report in any form has not been submitted to any University or Institute for any purpose.

Internal Supervisor External Supervisor

Head of the Dept

Acknowledgement

First and for most I thank **GOD** almighty and to my parents for the success of this project. I owe a sincere gratitude and heart full thanks to everyone who shared their precious time and knowledge for the successful completion of my project.

I am extremely thankful to **Dr Suresh Babu V**, Principal, College of Engineering Trivandrum for providing me with the best facilities and atmosphere which was necessary for the successful completion of this project.

I am extremely grateful to **Dr. Deepa S S**, HOD, Dept of Computer Applications, for providing me with best facilities and atmosphere for the creative work guidance and encouragement.

I express our sincere thanks to **Smt. Baby Syla L**, Asst. Professor, Department of Computer Applications, College of Engineering Trivandrum for her valuable guidance, support and advice that aided in the successful completion of my project.

I profusely thank other Asst. Professors in the department and all other staffs of CET, for their guidance and inspirations throughout my course of study.

I owe my thanks to my friends and all others who have directly or indirectly helped me in the successful completion of this project. No words can express my humble gratitude to my beloved parents and relatives who have been guiding me in all walks of my journey.

Avani S Girish

Abstract

With the rapid growth of social networks and microblogging websites, communication between people from different cultural and psychological backgrounds has become more direct, resulting in more and more "cyber"conflicts between these people. Consequently, hate speech is used more and more, to the point where it has become a serious problem invading these open spaces. Hate speech refers to the use of aggressive, violent or offensive language, targeting a specific group of people sharing a common property, whether this property is their gender (i.e., sexism), their ethnic group or race (i.e., racism) or their believes and religion. While most of the online social networks and microblogging websites forbid the use of hate speech, the size of these networks and websites makes it almost impossible to control all of their content. Therefore, arises the necessity to detect such speech automatically and filter any content that presents hateful language or language inciting to hatred. In this project, an approach to detect hate expressions on Twitter is proposed. The approach is based on unigrams and patterns that are automatically collected from the training set.The dataset for the same is taken from www.kaggle.com.These patterns and unigrams are later used as features to train the machine learning algorithm.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Hate speech is a particular form of offensive language where the person using it is basing his opinion either on segregative, racist or extremist background or on stereotypes.With the spread of internet, and the growth of online social networks, this problem becomes even more serious, since the interactions between people became indirect, and people's speech tends to be more aggressive when they feel physically safer.Critics of hate speech argue not only that it causes psychological harm to its victims, and physical harm when it incites violence, but also that it undermines the social equality of its victims. That is particularly true, they claim, because the social groups that are commonly the targets of hate speech have historically suffered from social marginalization and oppression. Hate speech therefore poses a challenge for modern liberal societies, which are committed to both freedom of expression and social equality.

This being the case, hate speech is considered a world-wide problem that many countries and organizations have been standing up against. In this work, different sets of features including writing patterns and hate speech unigrams IS Proposed. use these features together to perform the classification of texts collected from Twitter (i.e., tweets) into three classes: "Clean' and 'Hateful.'

Chapter 2

Problem Definition and Motivation

Hate speech is a particular form of offensive language where the person using it is basing his opinion either on segregative, racist or extremist background or on stereotypes.With the spread of internet, and the growth of online social networks, this problem becomes even more serious, since the interactions between people became indirect, and people's speech tends to be more aggressive when they feel physically safer.

The major motivation behind choosing the project was the drawbacks of the existing system.It is not always Possible to manually detect those hate tweets all the time.Here comes the need of an autmoated system which can detect the hate speech at the very instance they are produced.Few drawbacks of existing system:

- The accuracy of existing system is less than 90 percentage.
- There is no existing system consisting of dealing with daily life hatred speeches in twitter.

Chapter 3

Literature Review

3.1 3.1 Automatic Classification of Sexism in Social Networks: An Empirical Study on Twitter Data

Published in: 2020 2nd International Conference on Cybernetics and Intelligent System (ICORIS).The Date of Conference was 27-28 October 2020 and the conference Location was Manado, Indonesia.Publisher was IEEE

3.2 3.2 Evaluating Machine Learning Techniques for Detecting Offensive and Hate Speech in South African Tweets

Published in: IEEE Access (Volume: 8).Date of Publication was 20 January 2020.Publisher was IEEE

3.3 3.3 Affect Intensity Analysis of Dark Web Forums

Publisher was IEEE Published in 2007 IEEE Intelligence and Security Informatics Date of Conference was 23-24 May 2007.Conference Location was New Brunswick, NJ, USA

Chapter 4

Requirement Analysis

4.1 Purpose

Hate speeches can be really vulnerable to the society.It is not always possible to manually remove those tweets.Here comes the need of an automated system.Here through this project,a system which can detect and categorise the tweet into hate and clean tweets is introduced.

4.2 Overall Description

Hate speech has directly elevated to physical world outcomes such as violence and crime. As social media platforms have been shown to provide unobtrusive measures, especially on topics that may be unlikely to be reported through other means (e.g. sensitive topics like discrimination), this medium also offers opportunity to further dissect issues like discrimination to understand antecedents, exact types/targets of discrimination.Given the increase in online hate speech, it has become increasingly important to examine the ways in which harmful speech influences physical world attitudes and behaviors, such as hate crimes.In the project speeches are classified into hate and clean speech so that such speeches can be figured out.

4.2.1 Product Functions

- Feature extraction using NLP
- Develope a user interface

- connect the UI with the model.

4.2.2 Hardware Requirements

- Processor : Intel Core i3
- Storage : 512 GB Hard Disk space
- Memory : 4 GB RAM

4.2.3 Software Requirements

- Operating System : Linux/Windows
- Platform : Python
- Librarie used : nltk, pandas, matplotlib, numpy, sklearn,

4.3 Functional Requirements

The functional requirements includes all the activities or processes that should be achieved by the proposed system. It includes

- **nltk:** It's the best available platform for developing programs that uses human language data. It gives us an easy interface to more than 50 corpora and lexical resources like WordNet. It also includes various text processing libraries for tokenisation, stemming, classification, semantic reasoning, parsing and tagging. It's the most easiest and efficient way for handling data in terms of human language.
- **sklearn:** sk learn (formerly sci-kit learn and sometimes called sk learn) is a machine learning library can be used in python programming language. By using this library, we can implement various regression, classification and clustering algorithms such as random forest, support vector machine, k-means and DBSCAN. And the sk learn library is built in a way that it can work with various scientific and numeric libraries of python such as scipy and numpy.

- **matplotlib:** It's used for the visualisation of data in python programming language. It's implemented to work with the wider scipy stack and it's built on numpy arrays. It's a multi platform data visualization technique. It was developed in 2002 by John Hunter. Visualization is the most efficient way to understand the data. Using this library, we can represent our data in various plots such as line, bar, histogram, scatter etc.

4.3.1 Quality Requirements

- Scalability : The software will meet all of the functional requirements.

- Maintainability : The system should be maintainable. It should keep backups to atone for system failures, and should log its activities periodically.

- Reliability : The acceptable threshold for down-time should be large as possible. i.e. mean time between failures should be large as possible. And if the system is broken, time required to get the system backup again should be minimum.

- Availability: This system is easily available as the core equiments in building the sofware is easily obtained.

- High- Functionality: This system is highly functional in all environment since, They are highly adaptable.

Chapter 5

Design And Implementation

The proposed system is used to destect tweets automatically by using a pre-trained model. The model is trained using Linear Regression upon the features extracted by natural language processing.

5.1 Overall Design

The proposed system follows client server architecture. That is the automated hate tweet detection system has a client part and a server part as well. The client part is used by the user to input the tweet which is to be evaluated. The input is passed to server and the evaluated result is given back to the client. The server side is developed in Python and the client side is built using HTML and Python.

5.1.1 System Design

The system is web based. The input is taken from the user through a web page and the input is passed to the python program running in the server side. The server program perform tasks such as pre processing and feature extraction on the input data. The results of these processes are used to evaluate the input using the pre trained model.

The model is created using the data obtained from Kaggle.com. They provide approximately 30,000 tweets . These tweets are divided into 2 sets - clean and hate speech.

5.1.2 Methodology

There are two parts in this project. The first part is the creation of the model and the second one is the creation of user program which will work with the pre-trained model.

The main process of the this project is the creation of the trained model. The major steps in the model creation Feature extraction, training, testing and model evaluation. The major steps in the model creation are mentioned below.

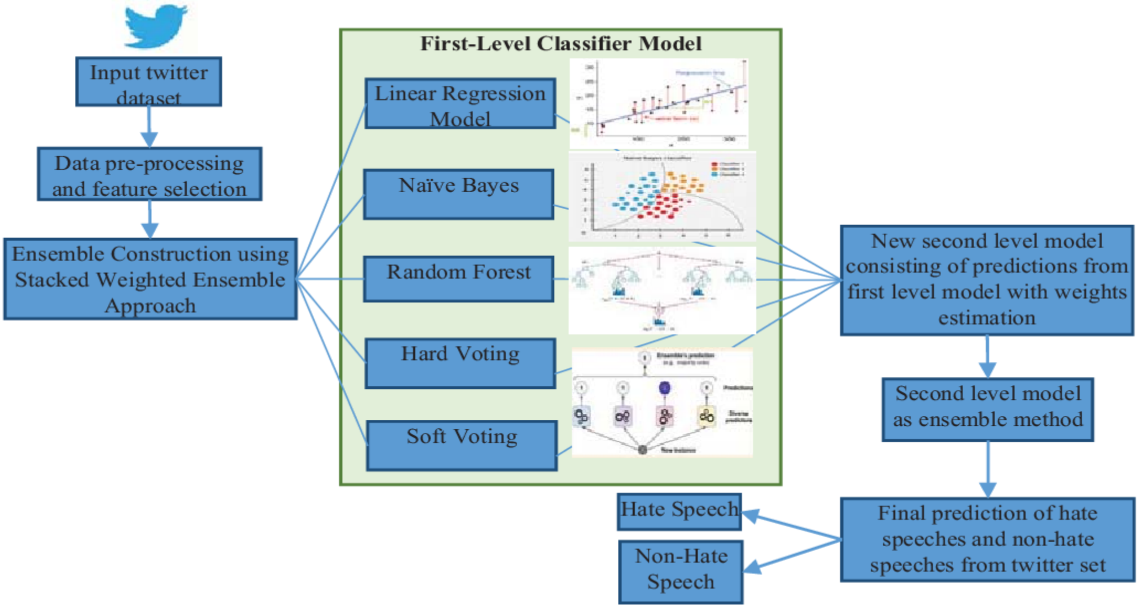

Input twitter dataset
Data pre-processing and feature selection
Ensemble Construction using Stacked Weighted Ensemble Approach
First-Level Classifier Model
Linear Regression Model
Naïve Bayes
Random Forest
Hard Voting
Soft Voting
New second level model consisting of predictions from first level model with weights estimation
Second level model as ensemble method
Final prediction of hate speeches and non-hate speeches from twitter set
Hate Speech
Non-Hate Speech

Figure 5.1: Architecture of model Creation

- **feature extraction:**Feature extraction is the most important part of any machine learning task and so is the case with us. Here we specify the linguistic features of the tweets and they are extracted using natural language processing. We extract the features such as word count, sentence count etc. as the features

- **Training:**Linear Regression is the technique used for the learning of the system.It's one of the widely used supervised learning techniques It uses the labelled dataset along with the extracted features to generate the model. A portion of our dataset is used for training. The remaining portion is for testing the model.

- **Testing:** In testing phase we test the generated model with the remaining portion of the dataset. The data set is fed into the generated model and their results are recorded for the next stage which is the model evaluation and error analysis.

- **Model evaluation and error analysis:** The results of the testing data along with their original values are used for the error calculation. Various statistical measures can be adopted for calculating the efficiency of the model. The various measures are accuracy, precision, recall, kappa score etc. If the results of these quantitative analyses are acceptable, then we move forward with the generated model. If the results are poor, the model is to be regenerated with more features so that the best result is obtained.

The second part of the project is to build the user interface. The user interface is build using HTML and Python. This is the part of project which deals with the user. The input text is fed into the server through this. And the results returned are also displayed in the user program. The interface is built in a way such that it is easy and understandable for the person who uses it. For that we uses responsible HTML designs which uses Bootsstrap, CSS and JavaScript also to provide the better user experience. Python Flask is also used for the development of user interface.

5.2 Data Flow Diagram

DFD is one of the graphical representation techniques used in a project to show the flow of the data through a project. DFD helps us to obtain an idea about the input, output, and process involved. The things absent in a DFD are control flow, decision rules, and loops. It can be described as a representation of functions, processes that capture, manipulate, store, and distribute data between a system and the surrounding and between the components of the system. The visual representation helps for good communication.

It shows the journey of the data and how will it be stored in the last. It does not provide details about the process timings or if the process shall have a parallel or sequential operation. It is very different from a traditional flow chart or a UML that shows the control flow or the data flow.

In level 0 the basic data flow of the application is showcased. It does not show the flow of data much deeper. It will be evaluated in the higher levels of Data Flow Diagram. The Data Flow Diagram of Automated essay scoring system is shown below.

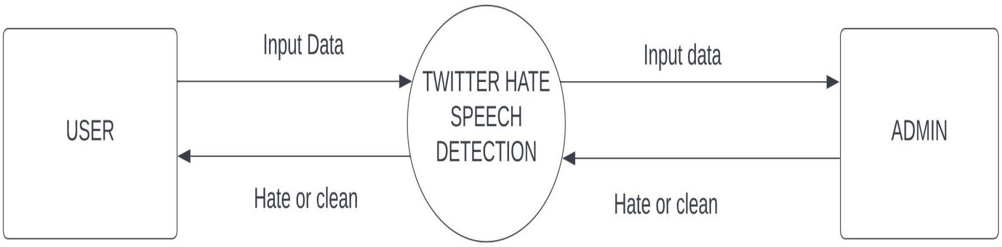
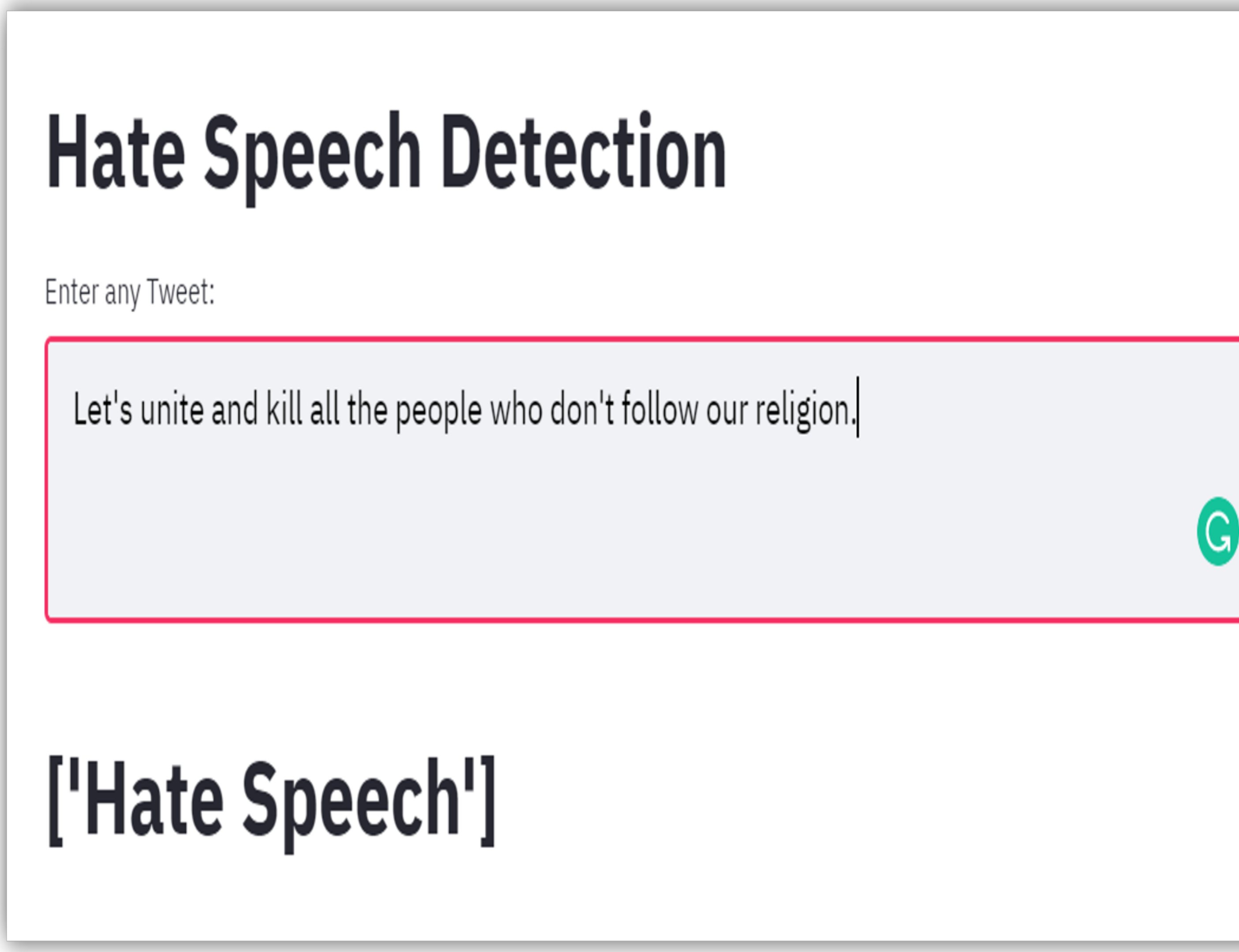


Figure 5.2: Level 0 DFD

The diagram shows Level 0 Data flow diagram of the Twitter hate speech detection. As the diagram indicates there is a user part and an admin one. The input of the project is the tweet by the user and which is given to the Automated hate speech detection system. Then the tweet is passed to the admin part . The categorisation of tweets into clean and hate is occurred in the admin side. Then output is passed back to the user through the application. This is how the data flows through the application. Since there is no database in the application, the data is not stored anywhere. The data is lost afterwards.

5.3 Screenshots of user interface



Hate Speech Detection

Enter any Tweet:

Let's unite and kill all the people who don't follow our religion.

['Hate Speech']

Figure 5.3: input

Chapter 6

Coding

Algorithm 1 Algorithm for Creating the model:

```
Split the data set into training data set and testing dataset. 80% of the dataset is used for training and the remaining 20% is used for testing to obtain better result.
The training dataset is used for the preprocessing stage and the preprocessed data is further used to extract the features. The linguistic features of the input essay is obtained using natural language processing libraries.
The extracted features are then analysed to find out which features has the higher influence on the results. Then the features with high dependency on the result of essay is used to create the model.
The model is created using Linear Regression with the selected features. The linear regression model evaluates how much the dependent values depend upon the independent values.
The testing dataset is feeded into the created model and their results are noted down.
The result of testing dataset evaluated using the created model is then compared with the actual values of the testing dataset to evaluate the efficiency of the model. Various statistical measures such as Accuracy, Precision, Recall, Kappa score etc. can be used to evaluate the model.
Further tuning is performed upon the created model to improve the efficiency of the model.
```

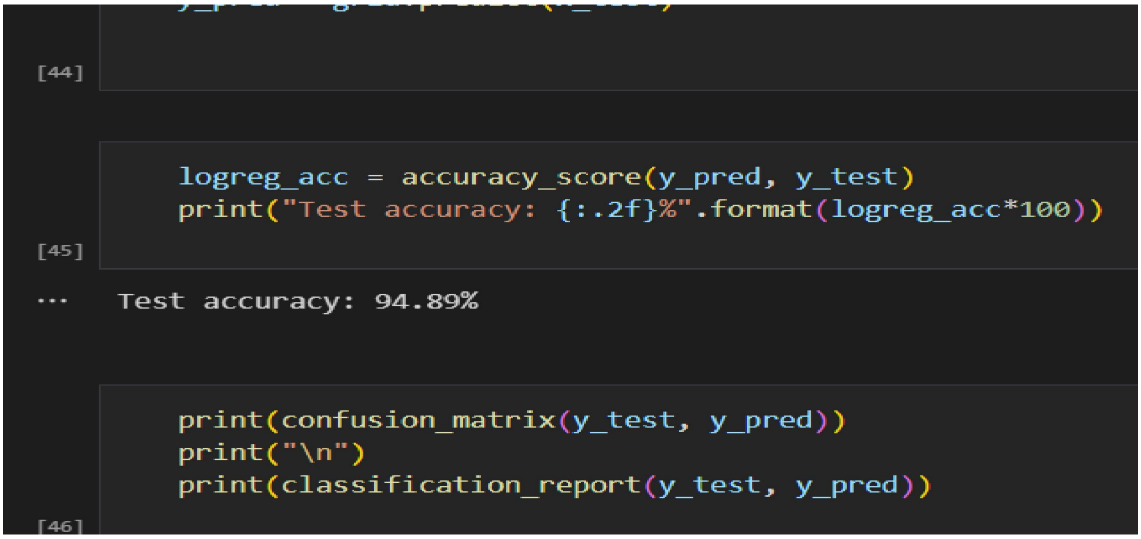
Algorithm 2 Algorithm for web Application and essay evaluation:

```
1: Read the input tweet from the user through the user interface.
2: On button click the value in the web page is passed to the server program for the detection of hate tweet.
3: From the server program, access the input tweet and perform the preprocessing tasks on it.
4: The preprocessed tweet is used to extract the required features from it using natural language processing libraries.
5: Using the pretrained model, we evaluate the input tweet using the extracted linguistic features.
6: The score is evaluated by the results of the model and the score is passed to the web page.
7: The score is displayed in the web application.
```

Chapter 7

Accuracy

The percentage of accurate predictions for the test results is known as accuracy in ML.The accuracy of a ML model is a metric for determining which model is the best at distinguishing associations and trends between variables in a dataset based on the input, or training data. The more a model can generalize to 'unseen' data, the more forecasts and ideas it can provide, and therefore the more market value it can provide.



```
[44]

    logreg_acc = accuracy_score(y_pred, y_test)
    print("Test accuracy: {:.2f}%".format(logreg_acc*100))
[45]

...  Test accuracy: 94.89%

    print(confusion_matrix(y_test, y_pred))
    print("\n")
    print(classification_report(y_test, y_pred))
[46]
```

Figure 7.1: Accuracy of proposed system

Chapter 8

Testing and Implementation

8.1 Testing and various types of testing used.

Once a software is developed, the major activity is to test whether the actual results match with the experimental results. This process is called testing. It's used to make sure that the developed system is defect free. The main aim of testing is to find the errors and missing operations by executing the program. It also ensure that all of the objectivs of the project are met by the developer. The objective of testing is not only to evaluate the bugs in the created software but also finding the ways to improve the efficiency, usability and accuracy of it. It aims to measure the functionality, specification and performance of a software program. Tests are performed on the created software and their results are compared with the expected documentation. When there are too much errors occurred, debugging is performed. And the result after debugging is tested again to make sure that the software is error free. The major testing processes applied to this project are unit testing, integration testing and system testing. In unit testing, our aim is to test all individual units of the software. It makes sure that all of the units of the software works as it intended. In integration testing, the combined individual units are tested to check whether it met the intended function or not. It helps us to find out the faults that may arise when the units are combined. In system testing the entire software is tested to make sure that it satisfies all of the requirements. The tables shown below describes the testing process occurred during the development of this project "Twitter hate speech detection". This defines the various steps took to create the project error free.

8.1.1 Unit Testing

Text Cases and Result

Sl No	Procedures	Expected result	Actual result	Pass or Fail
1	create the user interface	To load the web page with required fields	Same as expected	Pass
2	pre-processing	clean the dataset for feature extraction	same as expected	Pass
3	extract features from dataset	extract various features from dataset and store it in a csv file	csv file generated	Pass
4	training and testing of model	create the model and store it in a pickle file	pickle file generated	Pass
5	prediction	predict the result accurately	same as expected.	Pass
6	python server program	set up a python flask server to run the program	Same as expected	Pass

Table 8.1: Unit test cases and results

8.1.2 Integration Testing

Text Cases and Result

Sl No	Procedures	Expected result	Actual result	Pass or Fail
1	load the user interface from python	the user interface is loaded when we run the flask program	Same as expected	Pass
2	pass input tweet from web page to server	To pass the input essay entered by the user to the python program to and receive it there.	Same as expected	Pass
3	Hate Tweet Detection	load the previously generated pickle file to the server and predict the result with it and extracted features.	Same as expected	Pass
4	display results	pass the result to web page and display it there	Same as expected	Pass

Table 8.2: Integration cases and result

8.1.3 System Testing

Text Cases and Result

Sl No	Procedures	Expected result	Actual result	Pass or Fail
1	to run python server	Server program executed successfully, hence the entire program worked without any crash	Same as expected	Pass
2	Tweet detection	allow user to input tweet and output generated according to the input tweet.	Same as expected	Pass

Table 8.3: System test cases and results

Chapter 9

Results and Discussion

The main aim of the project was to detect hate speeches from twitter with a machine learning model. And it is observed that the system performs all the functionalities as expected.

9.1 Advantages and Limitations

The proposed system is a machine learning model to evaluate the input tweet and detect. The proposed system posses more advantages over the existing system. The proposed system save a huge amount of time. Like every other system, this system also have it's own disadvantages. But they are negligible while comparing with the advantages and they can be overcame in future.

9.1.1 Advantages

- Automatic tools and approaches could accelerate the reviewing process or allocate the human resource to the posts that require close human examination.
- It'll help remove the speeches that doesn't deserve attention.
- can be used by cyber crime protection authority
- This project can make sure to sort out hate speech and punish the source

9.1.2 Limitations

- subtleties in language, differing definitions on what constitutes hate speech, and
- Another remaining challenge is that automatic hate speech detection is a closed-loop system; individuals are aware that it is happening, and actively try to evade detection
- limitations of data availability for training and testing of these systems.

Chapter 10

Conclusion and Future Scope

As hate speech continues to be a societal problem, the need for automatic hate speech detection systems becomes more apparent. We presented the current approaches for this task as well as a new system that achieves reasonable accuracy. We also proposed a new approach that can outperform existing systems at this task, with the added benefit of improved interpretability. Given all the challenges that remain, there is a need for more research on this problem, including both technical and practical matters.

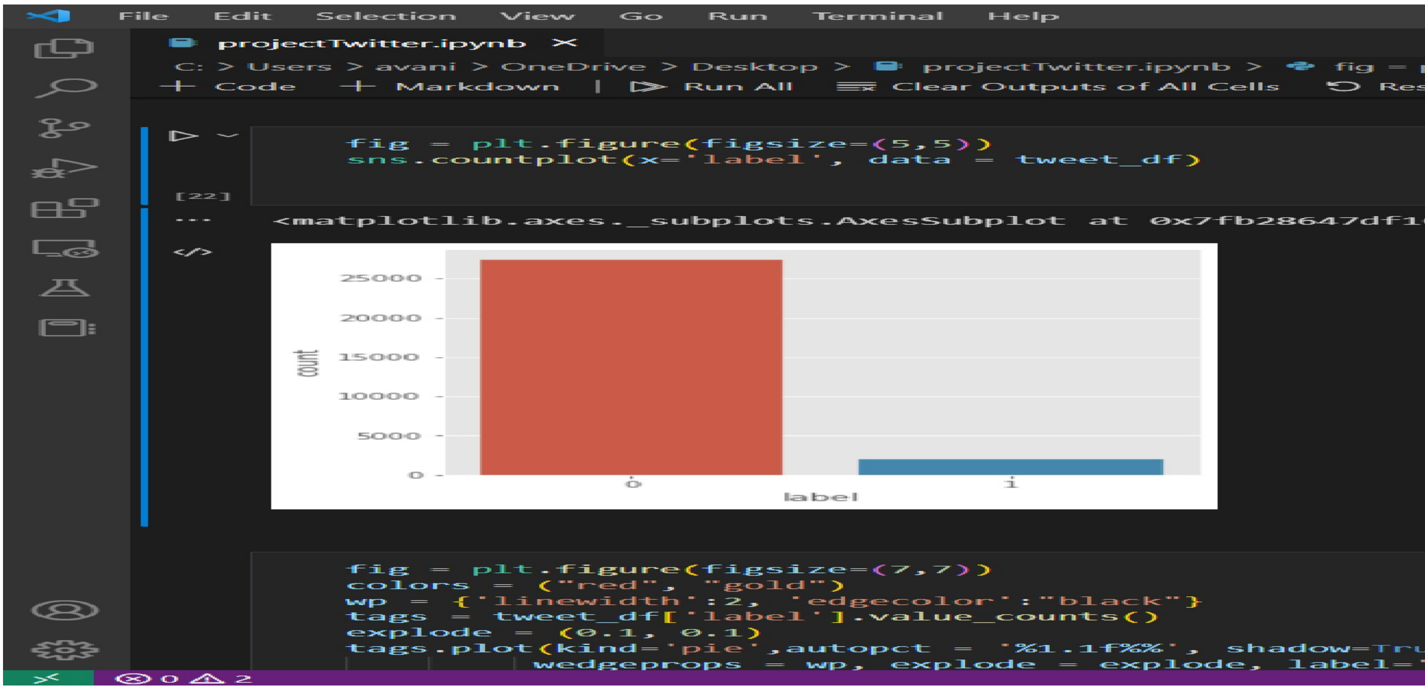
The results obtained by the created model seems encouraging and can be improved in future. The rate of errors in the machine learning model is very minimum . The majority of the project was built in python. It uses a flask server to connect to the user interface built using HTML, JavaScript and CSS. The project was built with the help of various python libraries such ass nltk, sklearn, pandas, numpy etc.

The future scope of this particular machine learning model can be extended to multiple dimensions. This project can be incorporated with the cyber security portal to recognise hate speeches fastly. This project can be incorporated with different social media platforms to remove the hate speech at the very instant they are produced.

Chapter 11

Appendix

11.1 Screenshot

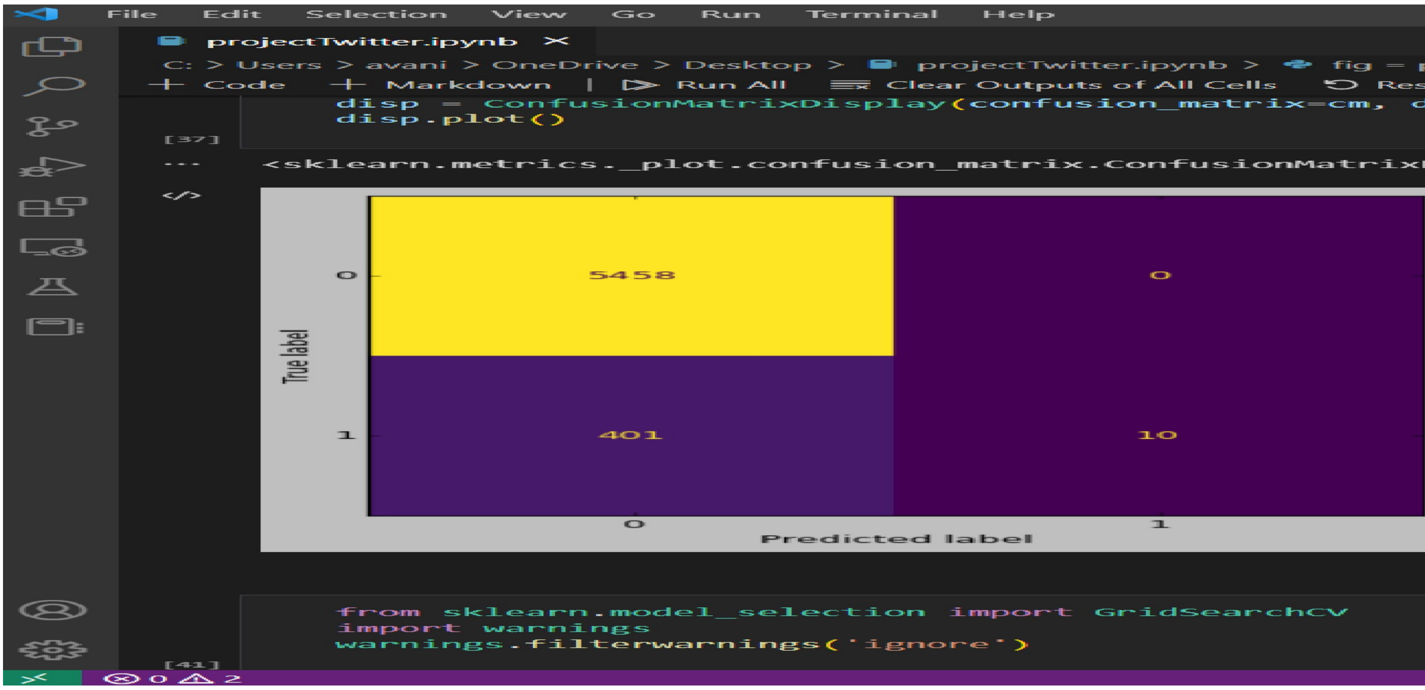


```
fig = plt.figure(figsize=(5,5))
sns.countplot(x='label', data = tweet_df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb28647df1
```

```
fig = plt.figure(figsize=(7,7))
colors = ("red", "gold")
wp = {'linewidth':2, 'edgecolor':"black"}
tags = tweet_df['label'].value_counts()
explode = (0.1, 0.1)
tags.plot(kind='pie',autopct = '%1.1f%%', shadow=Tru
            wedgeprops = wp, explode = explode, label='
```

Figure 11.1: visualisation

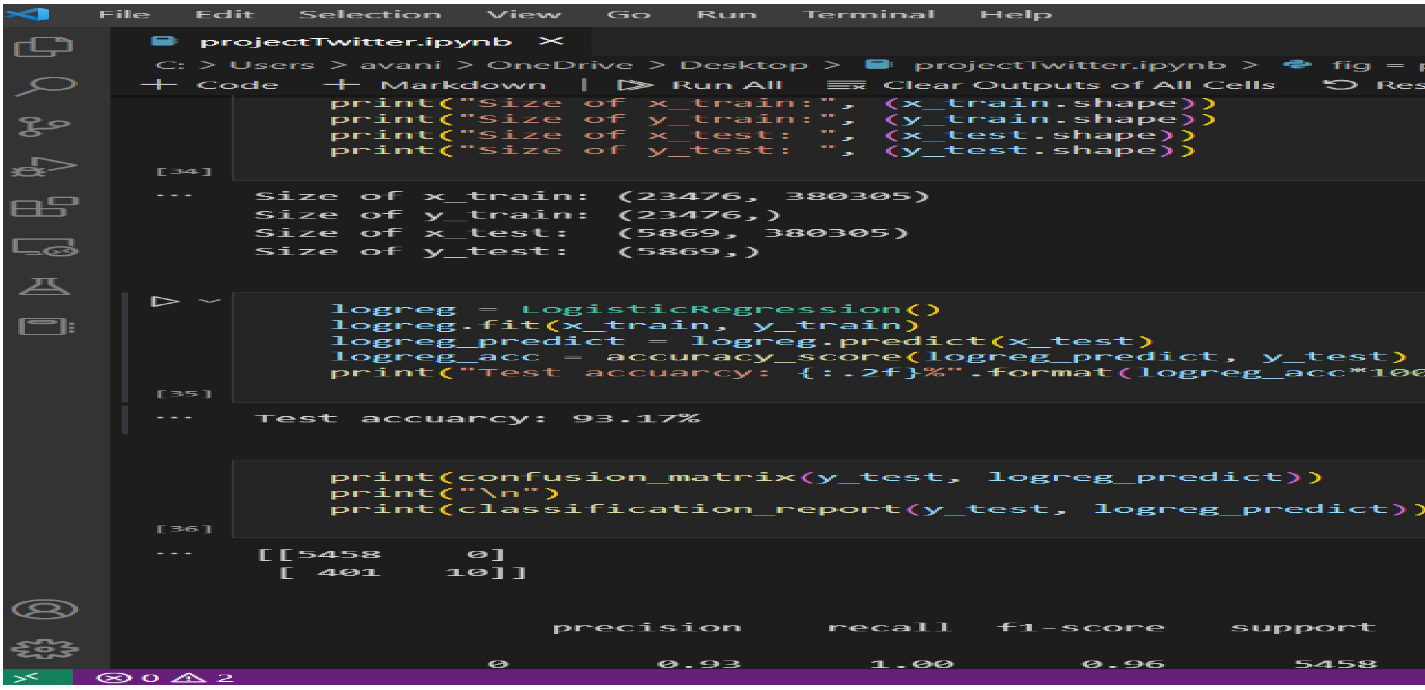


```
disp = ConfusionMatrixDisplay(confusion_matrix=cm, d
disp.plot()
```

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrix
```

```
from sklearn.model_selection import GridSearchCV
import warnings
warnings.filterwarnings('ignore')
```

Figure 11.2: confusion metrix



```
print("Size of x_train:", (x_train.shape))
print("Size of y_train:", (y_train.shape))
print("Size of x_test: ", (x_test.shape))
print("Size of y_test: ", (y_test.shape))
```

```
Size of x_train: (23476, 380305)
Size of y_train: (23476,)
Size of x_test:  (5869, 380305)
Size of y_test:  (5869,)
```

```
logreg = LogisticRegression()
logreg.fit(x_train, y_train)
logreg_predict = logreg.predict(x_test)
logreg_acc = accuracy_score(logreg_predict, y_test)
print("Test accuarcy: {:.2f}%".format(logreg_acc*100
```

```
Test accuarcy: 93.17%
```

```
print(confusion_matrix(y_test, logreg_predict))
print("\n")
print(classification_report(y_test, logreg_predict))
```

```
[[5458     0]
 [ 401    10]]

              precision    recall  f1-score   support

           0       0.93      1.00      0.96      5458
```

Figure 11.3: Linear regression algorithm

Bibliography

[1] Scikit learn: https://scikit-learn.org

[2] Flask: https://www.flasproject.com/

[3] Fortuna P, Nunes S (2018) A survey on automatic detection of hate speech in text. ACM Comput Surv 51:1–30. https://doi.org/10.1145/3232676

[4] tieglitz S, Mirbabaie M, Ross B, Neuberger C (2018) Social media analytics—challenges in topic discovery, data collection, and data preparation. Int J Inf Manage, pp 156–168