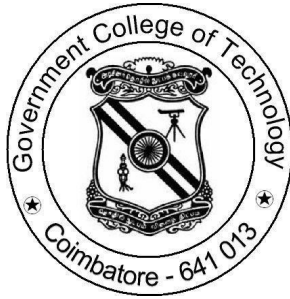


HATE SPEECH PREDICTION



PROJECT SUBMITTED IN PARTIAL
FULFILLMENT OF THE
REQUIREMENTS FOR THE AWARD
**OF THE DEGREE OF BACHELOR OF
ENGINEERING** IN COMPUTER
SCIENCE AND ENGINEERING OF
THE ANNA UNIVERSITY

MINI PROJECT WORK

Submitted by

**VISHALI.R
1917303**

**VIJAY.P
1917152**

**DEEPAN.P
1917L04**

**SENTHILNAYAGAN.S
1917L13**

2022

Under the Guidance of
Dr.K.KUMAR, M.E., Ph.D.,

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GOVERNMENT COLLEGE OF TECHNOLOGY**

(An Autonomous Institution affiliated to Anna University)

COIMBATORE - 641 013

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GOVERNMENT COLLEGE OF TECHNOLOGY**

COIMBATORE – 641 013

MINI PROJECT WORK

DEC 2022

This is to certify that this project work entitled

HATE SPEECH PREDICTION

is the bonafide record of project work done by

VISHALI R

1917303

VIJAY.P

1917152

DEEPAN.P

1917L04

SENTHILNAYAGAN

1917L13

of B.E. (COMPUTER SCIENCE AND ENGINEERING) during the year 2022 – 2023

PROJECT GUIDE

Dr.K.KUMAR M.E., Ph.D.

HEAD OF THE DEPARTMENT

Dr.J.C.MIRACLIN JOYCE PAMILA M.E, Ph.D.

Submitted for the Project Viva-Voce examination held on _____

Internal Examiner

External Examiner

ACKNOWLEDGMENTS

We express our sincere gratitude to **Dr.P.THAMARAI, Ph.D.**, Principal, Government College of Technology, Coimbatore for providing us all facilities that we needed for the completion of this project.

We whole-heartedly express our thankfulness and gratitude to **Dr.J.C.MIRACLIN JOYCE PAMILA, M.E., Ph.D.**, Professor and Head of the Department of Computer Science and Engineering, Government College of Technology, for helping us to successfully carry out this project.

Our thankfulness and gratitude to our respectable project guide **Dr.K.KUMAR, M.E.,Ph.D.**, Associate Professor(CAS) of the Department of Computer Science and Engineering who has been an immense help through the various phases of the project With his potent ideas and excellent guidance, we were able to comprehend the essential aspects involved.

We extend our sincere thanks to **Dr.S.RATHI, M.E., Ph.D.**, Professor (CAS), **Dr.T.RAJASENBAGAM, M.E.Ph.d**,AssistantProfessor, **Dr.A.MEENAKOWSHALYA M.E., Ph.D.**, Assistant Professor, **Dr.R.BHAVANI, M.E., Ph.D.**, Assistant Professor, **Dr.R.MUTHURAM, M.E., Ph.D.**, Assistant Professor, **Prof.L.SUMATHI, M.E.**, Assistant Professor for all their valuable suggestions to the completion of the project. We thank all the non-teaching staff and our friends for their cooperation towards the successful completion of the project.

We would like to dedicate the work to our parents for their constant encouragement throughout the project.

SYNOPSIS

Hate speech phenomenon has significantly increased in recent years, Due to its harmful effect on minority groups as well as on large communities, there is a pressing need for hate speech detection and filtering. Thus, there is a need for predictive machine learning models that not only detect hate speech but also help users understand when texts cross the line and become unacceptable.

Hate speech represents written or oral communication that in any way discredits a person or a group based on characteristics such as race, color, ethnicity, gender, sexual orientation, nationality, or religion . Hate speech targets disadvantaged social groups and harm them both directly and indirectly. Social networks like Twitter and Facebook, where hate speech frequently occurs, receive many critics for not doing enough to deal with it.

To create a system for predictive machine learning model i.e, Hate Speech Prediction that can predict the Hate Speech which is used to avoid the violence in the social media which can prevent an escalation from speech to action

TABLE OF CONTENTS

CHAPTER	TITLE	PAGENO
	BONAFIDE CERTIFICATE	I
	ACKNOWLEDGEMENT	II
	SYNOPSIS	III
	TABLE OF CONTENTS	IV
1	INTRODUCTION	
	1.1 OVERVIEW OF HATE SPEECH PREDICTION	1
	1.2 NEED FOR HATE SPEECH PREDICTION	2
	1.3 MOTIVATION FOR HATE SPEECH PREDICTION	2
2	REVIEW OF LITERATURE	
	2.1 PREVIOUS WORK	3
	2.2 PROPOSED SYSTEM	3
3	PROJECT DESIGN	
	3.1 PROJECT WORKFLOW	4
4	IMPLEMENTATION	
	4.1 ALGORITHM	
	4.1.1 LOGISTIC REGRESSION	6
	4.2 DATASET DESCRIPTION	8
	4.3 SOFTWARE ENVIRONMENT	11
	4.4 METHODOLOGIES	
	4.4.1 MULTILABEL VS MULTICLASS	
	CLASSIFICATION	12

	4.4.2.PIPELINE OF THE SOLUTION	13
	4.4.3.DATA COLLECTION	13
	4.4.4.DATA CLEANING	14
	4.4.5.FEATURES EXTRACTION	15
5	IMPLEMENTATION AND ANALYSIS	16
6	OPEN SOURCE CONTRIBUTION	18
7	SCREENSHOTS	20
8	CONCLUSION AND FUTURE ENHANCEMENTS	23
	REFERENCE	24

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF HATESPEECH PREDICTION

Hate speech prediction is the task of predicting if communication such as text, audio, and so on contains hatred and or encourages violence towards a person or a group of people. This is usually based on prejudice against 'protected characteristics' such as their ethnicity, gender, sexual orientation, religion, age. It can also be used to intimidate and threaten people. It can make people feel isolated, anxious, and scared. It can also lead to hate crimes. Hate speech can also damage relationships between different groups of people. Detection of hate speech is important because it can help prevent these harmful effects

The goal of Hate Speech Prediction is to predict the hate content posted in the Social media by identifying the bad word as target. The area of focus is the study of negative online behaviors language, like toxic comments (i.e. comments that are rude, disrespectful judgment, or otherwise likely to make someone leave a discussion). So far they have built a range of publicly available models served through the Perspective API. But the current models still make errors and they do not allow users to select which types of toxicity they are interested in finding

So in this project, we build a multi-headed model that is capable of detecting different types of toxicity like threats, obscenity, insults, and identity-based comments

1.2 NEED FOR HATE SPEECH PREDICTION

Hate speech prediction is nowadays highly in demand by companies because harmful comments can create a bad user experience on social media, like facebook or Twitter, or even on online games.

1.3 MOTIVATION FOR HATE SPEECH PREDICTION

The project is based on real life problem faced by many social media user If the problem is solved So far we have a range of publicly available models served through the Perspective API, including toxicity. But the current models still make errors, and they don't allow users to select which type of toxicity they're interested in finding. (E.g. some platforms may be ne with profanity, but not with other types of toxic content)

The goal of this project is to create a safer environment online. By detecting toxic comments on social media, they can be easily reported and removed. In the long term, this would allow people to better connect with each other in this increasingly digital world.

CHAPTER 2

REVIEW OF LITERATURE

2.1 PREVIOUS WORK

Toxic comment classification has been extensively studied in recent years, especially in the context of social media, where researchers have used various machine learning algorithms to classify toxic comments found on social media forums into different toxic classes

The majority of conventional machine learning algorithms are designed for classification problems with single-label. Hence, we'll use techniques to break the multi-label problem into several single-label problems, allowing us to use the existing conventional machine learning algorithms.

More research done in this field involved the prediction of hate speech whether it is hatred or not. Research done by Asogara D.C, Ngene C.C[1] Hate Speech Detection using svm and Naïve Bayes it will predict whether the comment is hate or not only. Instead of binary classification we want more classification

Research done by Abhishek Aggarwal, Atul Tiwari [4] predict the text as toxicity. While we have followed a loosely similar approach to predict the toxic comment using logistic regression from image .

2.2 PROPOSED SYSTEM

The proposed system which we have employed to classify tweets into Six different classes namely toxic ,severe toxic, obscene, insult, identity hate with either approximate probabilities. With the use of probability how much percentage of each toxicity label is identified . we are also predict toxic comment from image

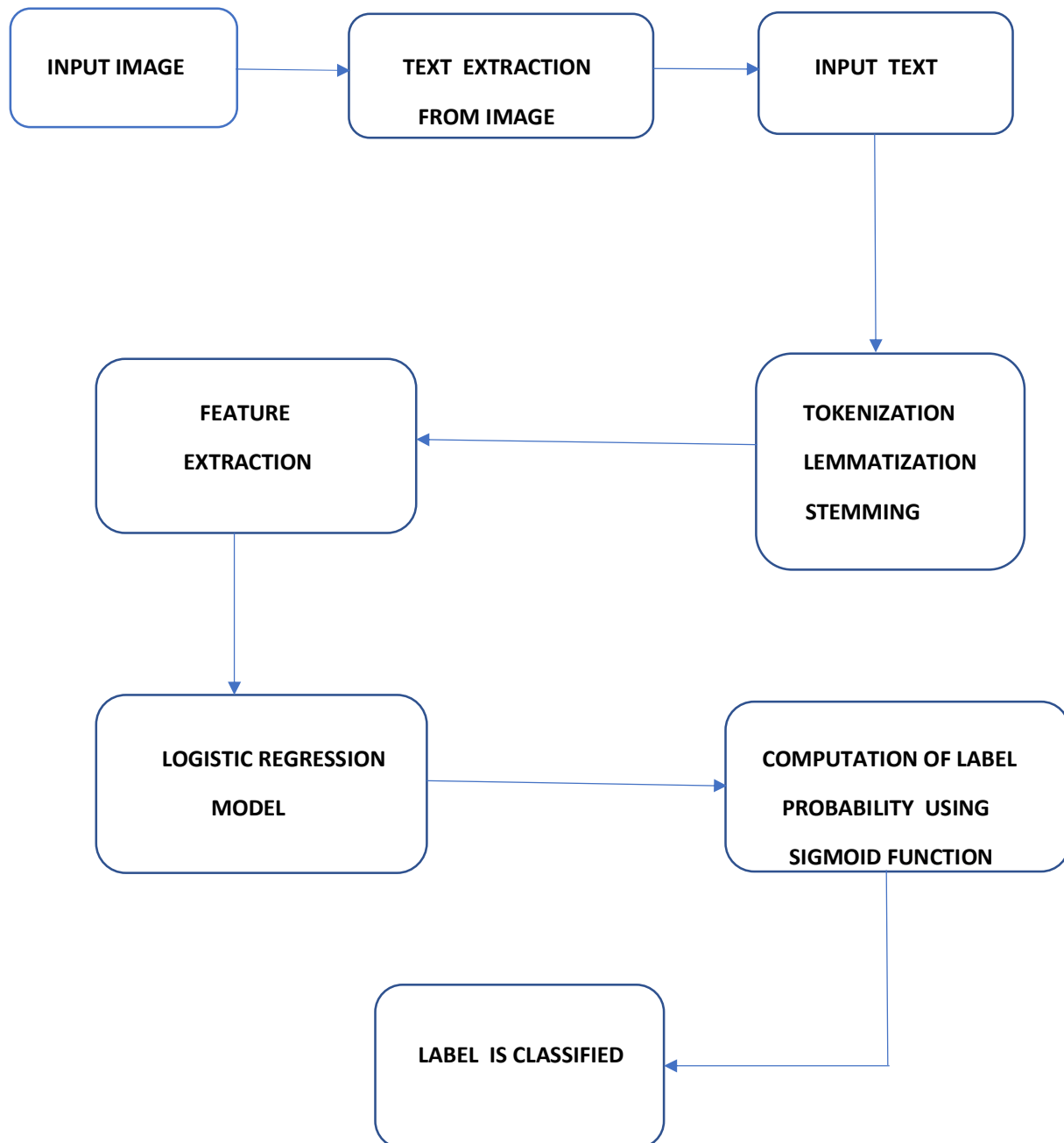
Advantages

1. By using the hate speech prediction we identify and examine the challenges faced by social media.
2. By this system we can identify whether it is insult or threat or obscene

CHAPTER 3

PROJECT DESIGN

3.1 PROJECT WORKFLOW



In social media users upload lots of images and tweet. It may contain hatred content. In our project image is given as an input to predict which type of toxicity it is. Image is taken from the social media like twitter and Facebook. The next step is extract the text from the image using tesseract. Tesseract is an optical character recognition engine. OCR systems transform a two-dimensional image of text, that could contain machine printed or handwritten text from its image representation into machine-readable raw text .

The extracted text is given to the data cleaning. Before training starts raw data should be processed. Data cleaning is used to improving the quality of data by fixing error and omissions based on certain standard practices. Data cleaning used in this project is Tokenization. It is used to split the sentence into individual word. Lemmatization and Stemming considers the context and converts the words into its meaningful base form for example in our project the word killing convert into base form kill

In this project TF-IDF vectorization is used to convert the raw data into vector representation .TF-IDF gives a product of how frequent a word is in the document multiplied by how unique the word with respect to the entire corpus of document. Word in the dataset with high TF-IDF score provide the most information about that specific document and the vectorized data is input to the logistic regression model.

Logistic regression use sigmoid function to predict the probability. It is an s shaped curve. The coefficients of the logistic regression algorithm must be estimated from the training data. sigmoid function that can take any real value number(coefficient value) and map it into a value between 0 and 1. The probability will be predicted for each 6 label.

CHAPTER 4

IMPLEMENTATION

4.1 ALGORITHM

4.1.1 LOGISTIC REGRESSION

For this mini project we have used the Logistic Regression algorithm which is a classification algorithm used to solve binary classification problems. Logistic regression is a predictive analysis algorithm and based on the concept of probability. The logistic regression classifier uses the weighted combination of the input features and passes them through a sigmoid function.

In this project the logistic regression model is trained using binary relevance method. In this method each label like toxic, severe toxic, obscene, insult, threat ,identity hate treated independently for each label a classifier is trained on input data

Below is an example logistic regression equation:

$$y = e^{(b_0 + b_1 \cdot x)} / (1 + e^{(b_0 + b_1 \cdot x)})$$

Where y is the predicted output

b0 is the bias or intercept term

b1 is the coefficient for the single input value (x).

Each column in extracted input data has an associated b coefficient value that must be learned from the training data of the model. This is done using maximum-likelihood estimation. The best coefficients would result in a model that would predict a value very close to 1

Depending on the data point, we provide a class tag (classify it as 1 or 0). The weight vector is applied, and the input vector is multiplied by it, which gives a scalable output. To get a value between 0 and 1, this output is placed into a Sigmoid function.

sigmoid function maps any value between 0 to 1. The sigmoid function calculation is based on

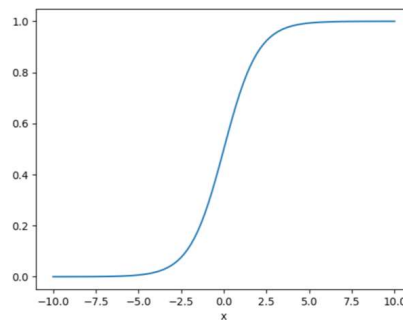


fig 4.1 – sigmoid function

$$S(x) = \frac{1}{1 + e^{-x}}$$

In above figure 4.1 shows the sigmoid function forms an S shaped graph. we use sigmoid to map predictions to probabilities.

A common approach to multi-label classification is by way of problem transformation, whereby a multi-label problem is transformed into one or more single label problems. Since we have multiple label dataset, we are dealing with Multit-label classification model.

We can approach this problem in different way, some of which used in this project are:

- **Binary Relevance:** In this labels are treated independently i.e., for each label, a classifier is trained on the input data. Since, we have six labels, we would have six different classifiers.
- **Classifier Chains:** In this, the first classifier is trained just on the input data and one label and then each next classifier is trained on the input space and all the previous classifiers in the chain. In the fig 4.2 show the classifier chain method process

X	y1
x1	0
x2	1
x3	0

Classifier 1

X	y1	y2
x1	0	1
x2	1	0
x3	0	1

Classifier 2

X	y1	y2	y3
x1	0	1	1
x2	1	0	0
x3	0	1	0

Classifier 3

X	y1	y2	y3	y4
x1	0	1	1	0
x2	1	0	0	0
x3	0	1	0	0

Classifier 4

Fig 4.2 – classifier chain

4.2 DATASET DESCRIPTION

The dataset used here is Toxic Comment Classification. Dataset is taken from the Kaggle[3]. The data-set consists of a large number of Wikipedia comments which have been labelled by human raters for toxic behaviour comments and its corresponding class labels encoded in binary format.

Train data set contains 159571 rows and 8 columns. Out of 8 columns, 6 columns are class labels and 1 column is for comment and one more column is ID to identify a row uniquely.

Data-set contains 89.83% Non-toxic comments and 10.17% Toxic comments

The types of toxicity labelled in dataset are:

- toxic
- severe toxic
- obscene
- threat
- insult
- identity hate

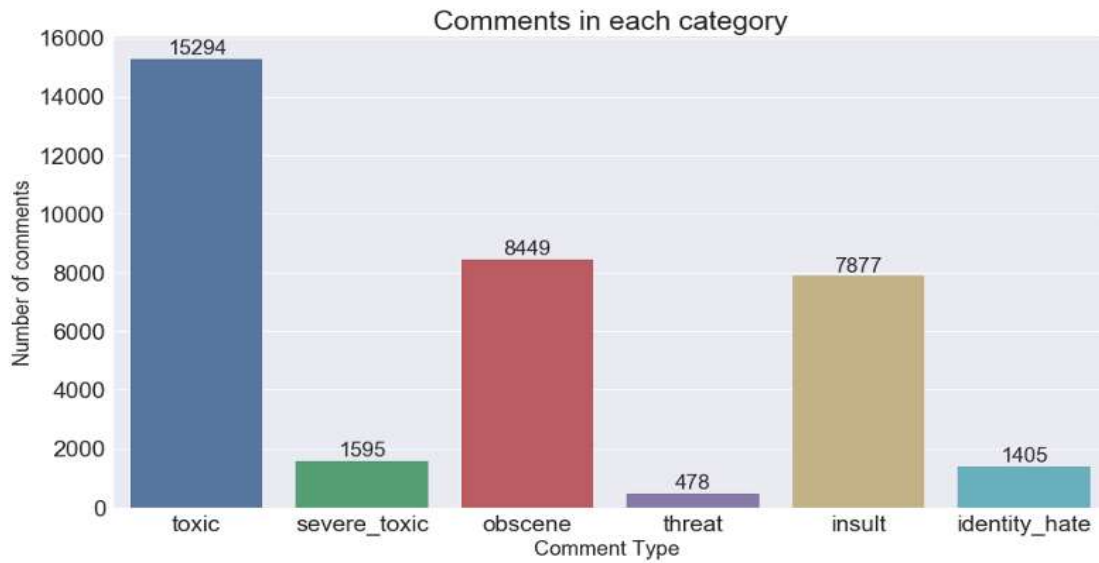


Fig 4.3 No of occurrence of each label

As shown in fig 4.3 shows that the dataset toxic comment classification contains 15294 comments are labelled as toxic, 1595 comments are labelled as severe toxic, 8449 comments are labelled as obscene, 478 comments are labelled as threat, 7877 comments are labelled as insult, 1405 comments are labels as identity hate .Remaining comments are labelled as non toxic . In fig 4.4 shows which word contains large amount of toxic frequency

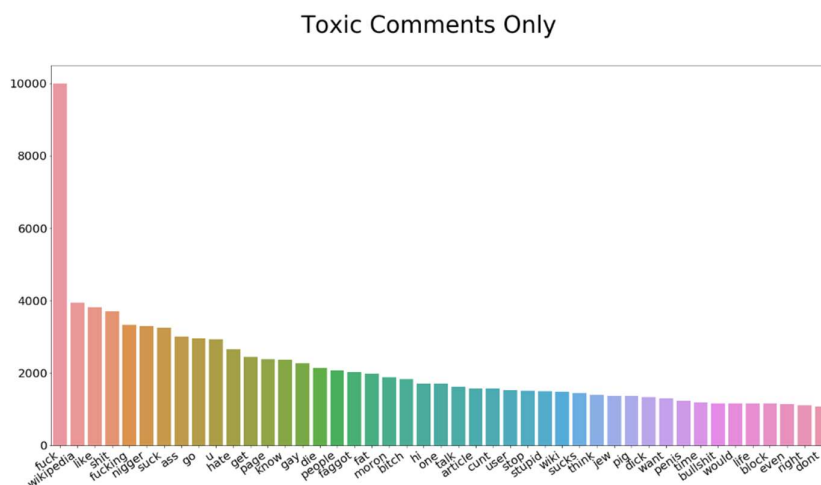


fig 4.4 Toxic comment word Frequency

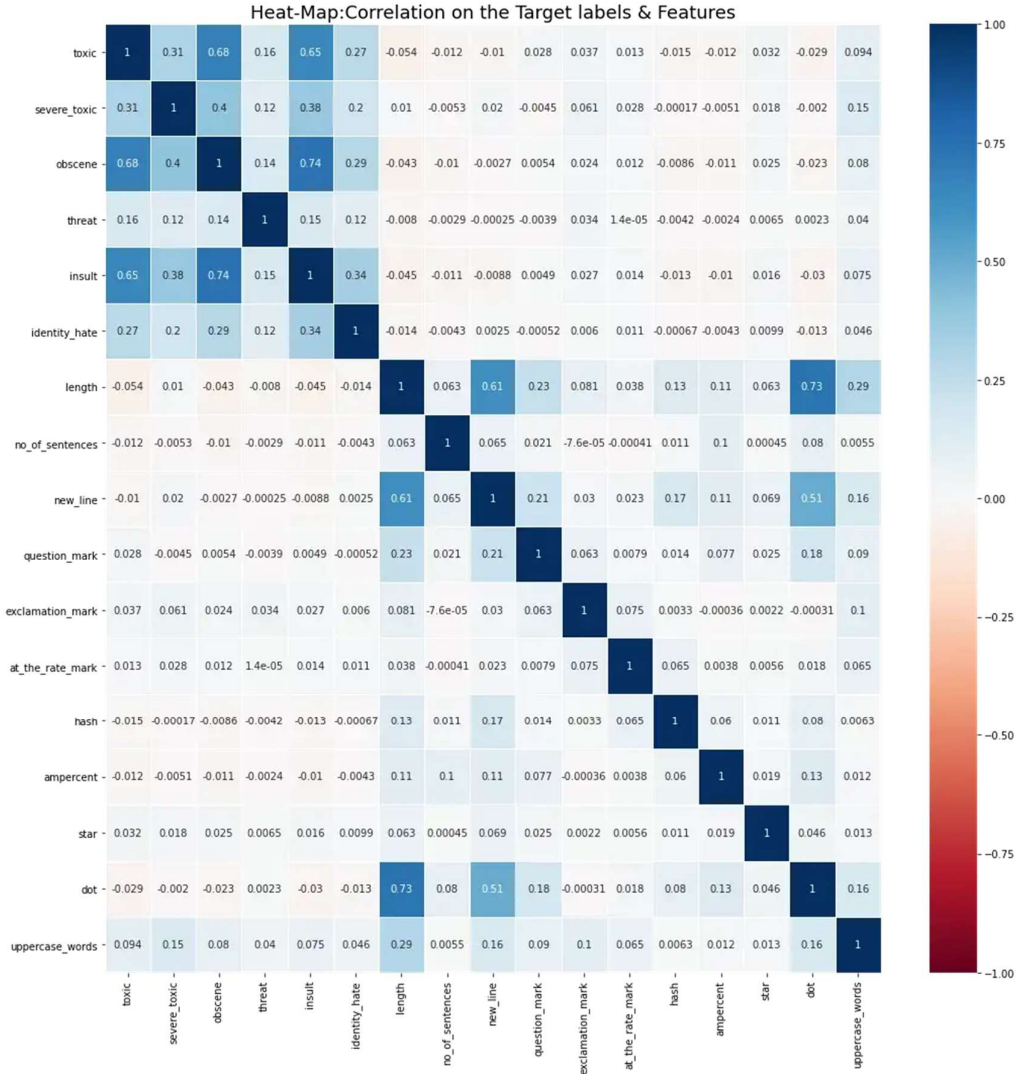


fig 4.5 correlation

From the correlation map (fig 4.5) understand that the toxic is much correlated with obscene, insult and severe toxic.

Obscene is much more correlated with insult and toxic target classes.

4.3 SOFTWARE ENVIRONMENT

The technology stack used in this project is as below fig 4.6.

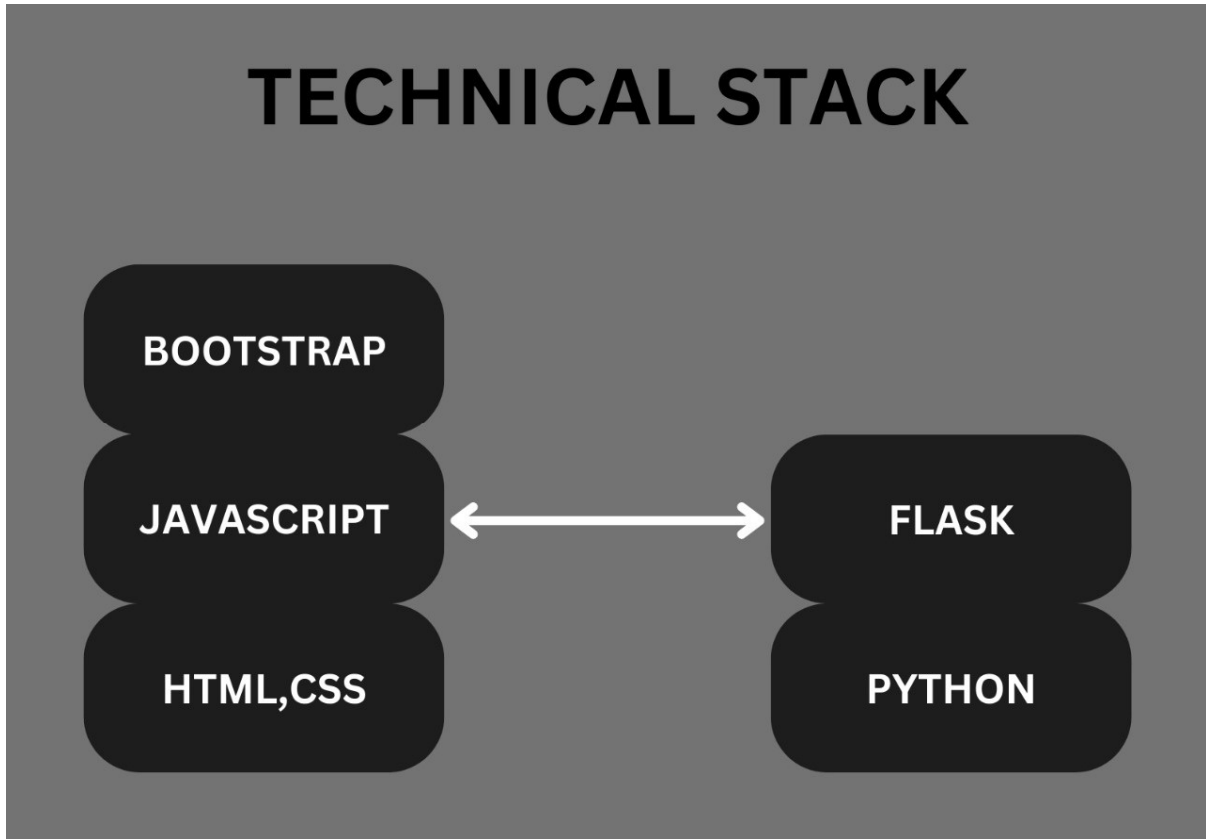


Fig 4.6. Technology Stack

Flask is used to develop this project in web page. It is considered as the basic framework and used throughout the application. The projects run in localhost in the port 5000. To style the web page we used CSS and BOOTSTRAP.

4.4 METHODOLOGIES

4.4.1. MULTILABEL VS MULTICLASS CLASSIFICATION

As the task was to figure out whether the data belongs to zero, one, or more than one category out of the six listed above, the first step before working on the problem was to distinguish between multi-label and multiclass classification.

In multi-class classification, we have one basic assumption that our data can belong to only one label out of all the labels we have. One of the approaches to solve a multilabel classification problem is Problem Transformation approach. In this approach transforming multi label classification task into single label transformation

In multi-label classification, data can belong to more than one label simultaneously. For example, in our case a comment may be toxic, obscene and insulting at the same time. It may also happen that the comment is non-toxic and hence does not belong to any of the six labels. Hence, we had a multi-label classification problem to solve. The next step was to gain some useful insights from data which would aid further problem solving.

4.4.2.Pipeline OF THE SOLUTION

The steps that we have followed to solve this problem is as below figure 4.7:

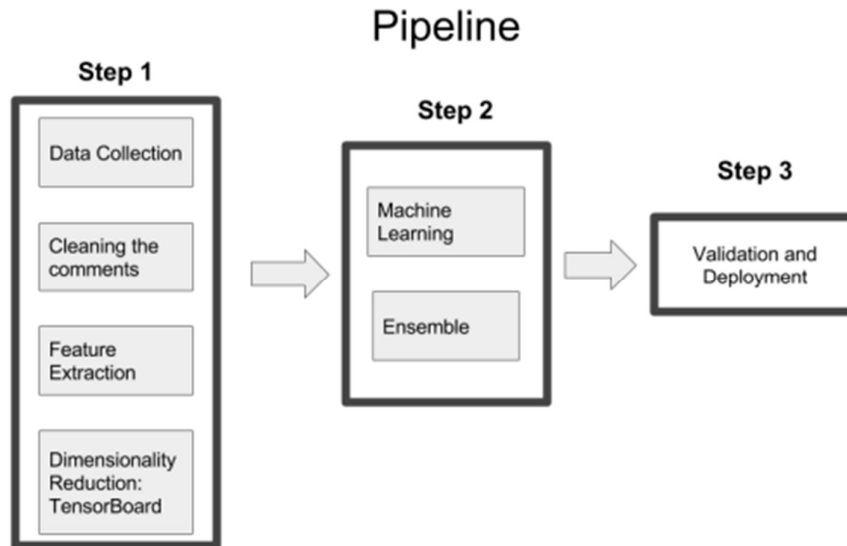


Fig 4.7 Steps to solve the problem

First step is data collecting and the collected data will be preprocessed and next step is using collected data to build the machine learning algorithm i.e Logistic regression model and the last step is validation and deployment

4.4.3.DATA COLLECTION

The data-set consists of a large number of Wikipedia comments which have been labeled by human raters for toxic behaviour.

The types of toxicity are:

- toxic
- severe toxic
- obscene
- threat
- insult
- identity hate

4.4.4.DATA CLEANING

1.Preparation for removal of punctuation marks: Imported the string library comprising all punctuation characters and appended the numeric digits to it, as those were required to be removed too.

2.Updating the list of stop words : Stop words are those words that are frequently used in both written and verbal communication and thereby do not have either a positive/negative impact on our statement. E.g. is, this, us, etc. Python has a built-in dictionary of stop words. I used the same and also appended the single letters like 'b', 'c' to it, which might be pre-existing or have generated during data preprocessing.

3.Stemming and Lemmatising : Stemming is the process of converting inflected/derived words to their word stem or the root form. Basically, a large number of similar origin words are converted to the same word. E.g. words like “killing”, “killer”, “killed” are based on “kill”. This helps in achieving the training process with a better accuracy.

Lemmatising is the process of grouping together the inflected forms of a word so they can be analysed as a single item. This is quite similar to stemming in its working but not exactly same. Lemmatising depends on correctly identifying the intended part of speech and meaning of a word in a sentence, as well as within the larger context surrounding that sentence, such as neighbouring sentences or even an entire document.

4.4.5.FEATURE EXTRACTION

Feature engineering is the process of using domain knowledge of the data to create features that make data mining algorithms work. If feature engineering is done correctly, it increases the predictive power of machine learning algorithms by creating features from raw data that facilitate the clustering process. It is the most important art in machine learning which creates the huge difference between a good model and a bad model.

For this project, we tried to use built-in feature extractors for this approach of classification such as:

- **TF-IDF Vectorizer** : The TF-IDF Vectorizer (Frequency-inverse document frequency vectorizer), is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. In practice, we imported the class `TfidfVectorizer` from `sklearn.feature extraction.text`.

CHAPTER 5

IMPLEMENTATION AND ANALYSIS

For experimentation and analysis purposes, we used 3 toxicity class for the example

1.Thread

INPUT : “ I will kill you “

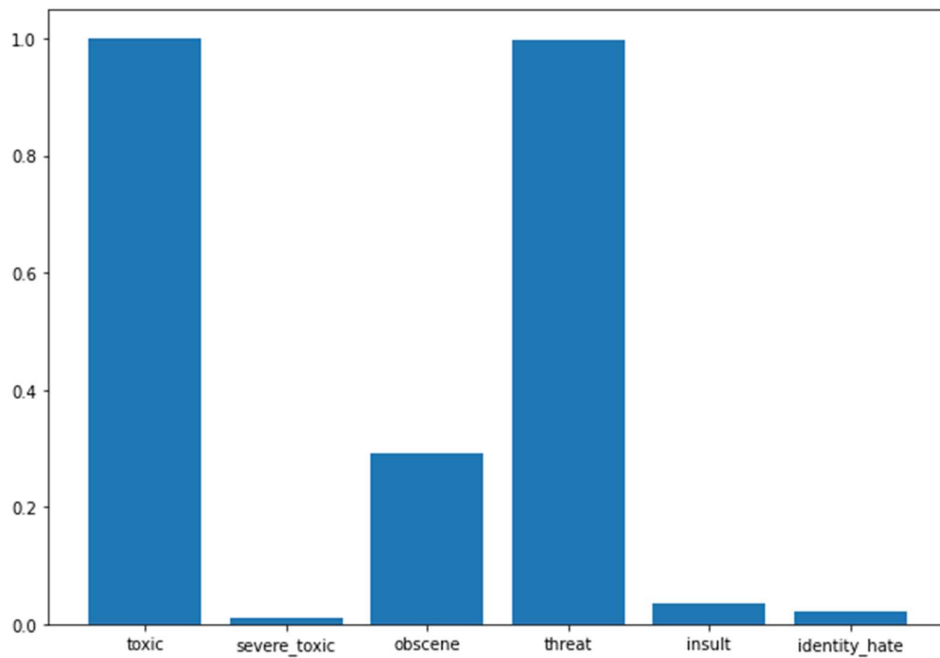


fig 5.1

The input “I will kill you “ is given to the trained model .After vectorized the input it checked with each label classifier using classifier chain method first it pass through the toxic label and input will check with already trained model it will return the predicted value and it will pass through the sigmoid function to predict the probability likewise it will predict each label probability As shown in figure 5.1 it shows that probability of toxic and threat is higher level so that input contains toxic and threat

2.INSULT

INPUT : “look you are stupid you don't know anything “

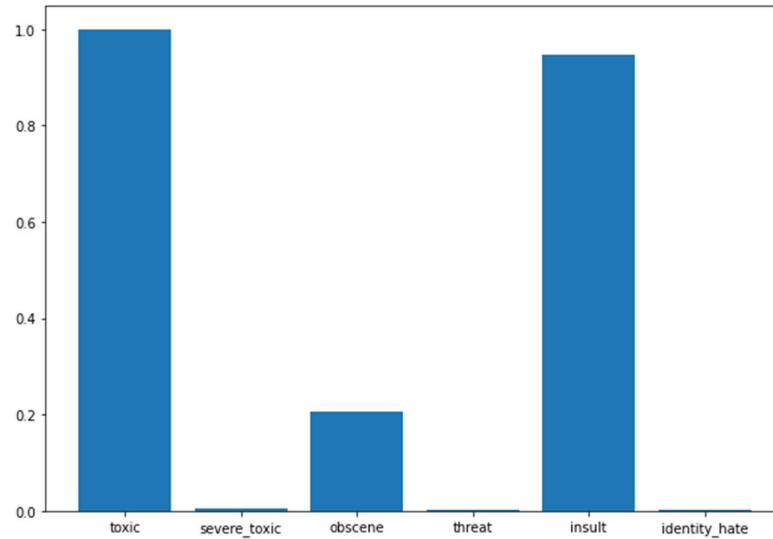


fig 5.2 – Insult graph

3.NON TOXIC

INPUT : “you look great today“

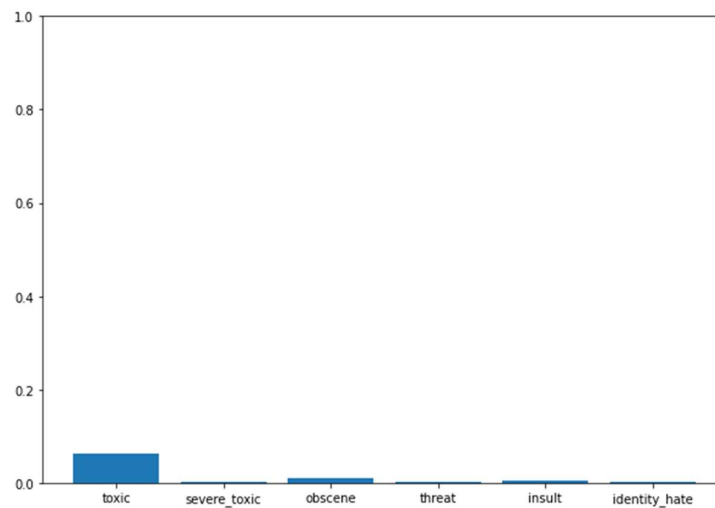


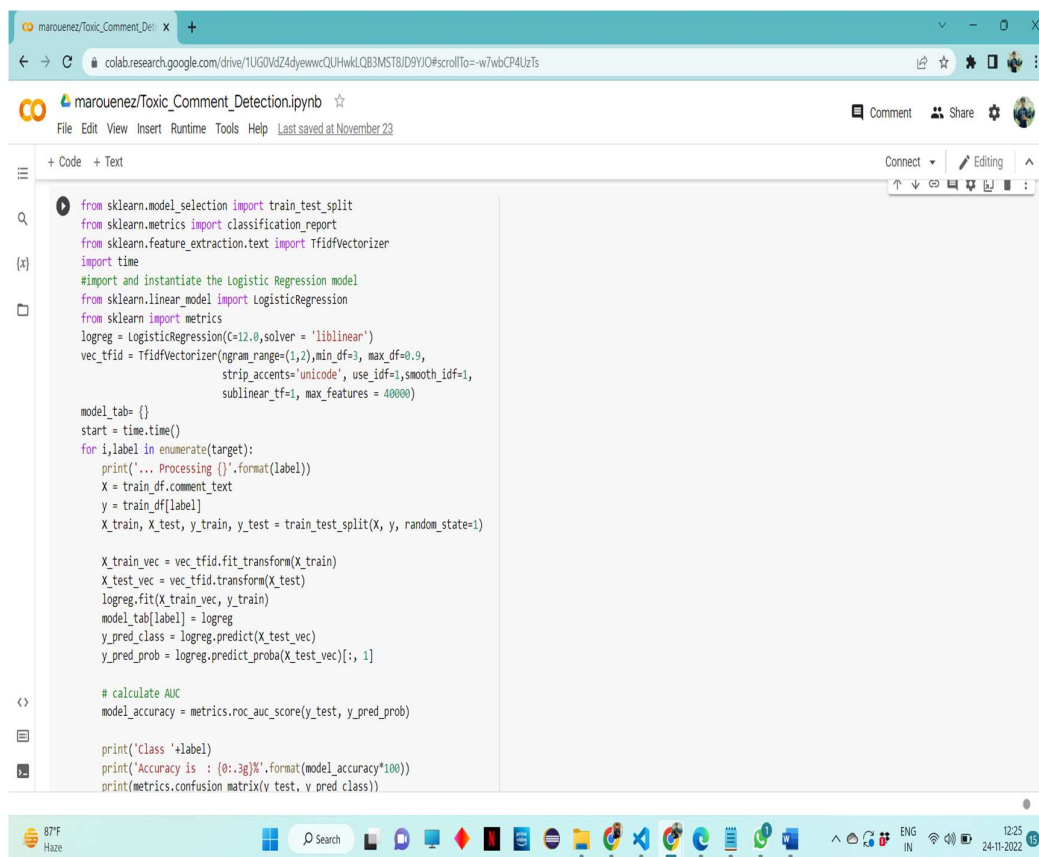
fig 5.3 – Non toxic graph

As mentioned above process likewise the input “look you are stupid you don't know anything” has higher probability level of toxic fig 5.2 and insult .In figure 5.3 no label has higher probability the threshold value is 0.4 probability so take as non toxic

CHAPTER 6

OPEN SOURCE CONTRIBUTION

After successful completion of our project, we published our model for our project. We have named our model as “Hate Speech Prediction”. If anyone wants to predict hate speech from the given image or text , they can use this model for the prediction. Our model can be used for predicting hate context by following the syntax given figure 6.1 and figure 6.2



```
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.feature_extraction.text import TfidfVectorizer
import time
# import and instantiate the Logistic Regression model
from sklearn.linear_model import LogisticRegression
from sklearn import metrics
logreg = LogisticRegression(C=12.0, solver = 'liblinear')
vec_tfidf = TfidfVectorizer(ngram_range=(1,2), min_df=3, max_df=0.9,
                           strip_accents='unicode', use_idf=1, smooth_idf=1,
                           sublinear_tf=1, max_features = 40000)

model_tab= {}
start = time.time()
for i,label in enumerate(target):
    print('... Processing {}'.format(label))
    X = train_df.comment_text
    y = train_df[label]
    X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=1)

    X_train_vec = vec_tfidf.fit_transform(X_train)
    X_test_vec = vec_tfidf.transform(X_test)
    logreg.fit(X_train_vec, y_train)
    model_tab[label] = logreg
    y_pred_class = logreg.predict(X_test_vec)
    y_pred_prob = logreg.predict_proba(X_test_vec)[:, 1]

# calculate AUC
model_accuracy = metrics.roc_auc_score(y_test, y_pred_prob)

print('Class '+label)
print('Accuracy is : {:.3g}%'.format(model_accuracy*100))
print(metrics.confusion_matrix(y_test, y_pred_class))
```

Fig 6.1

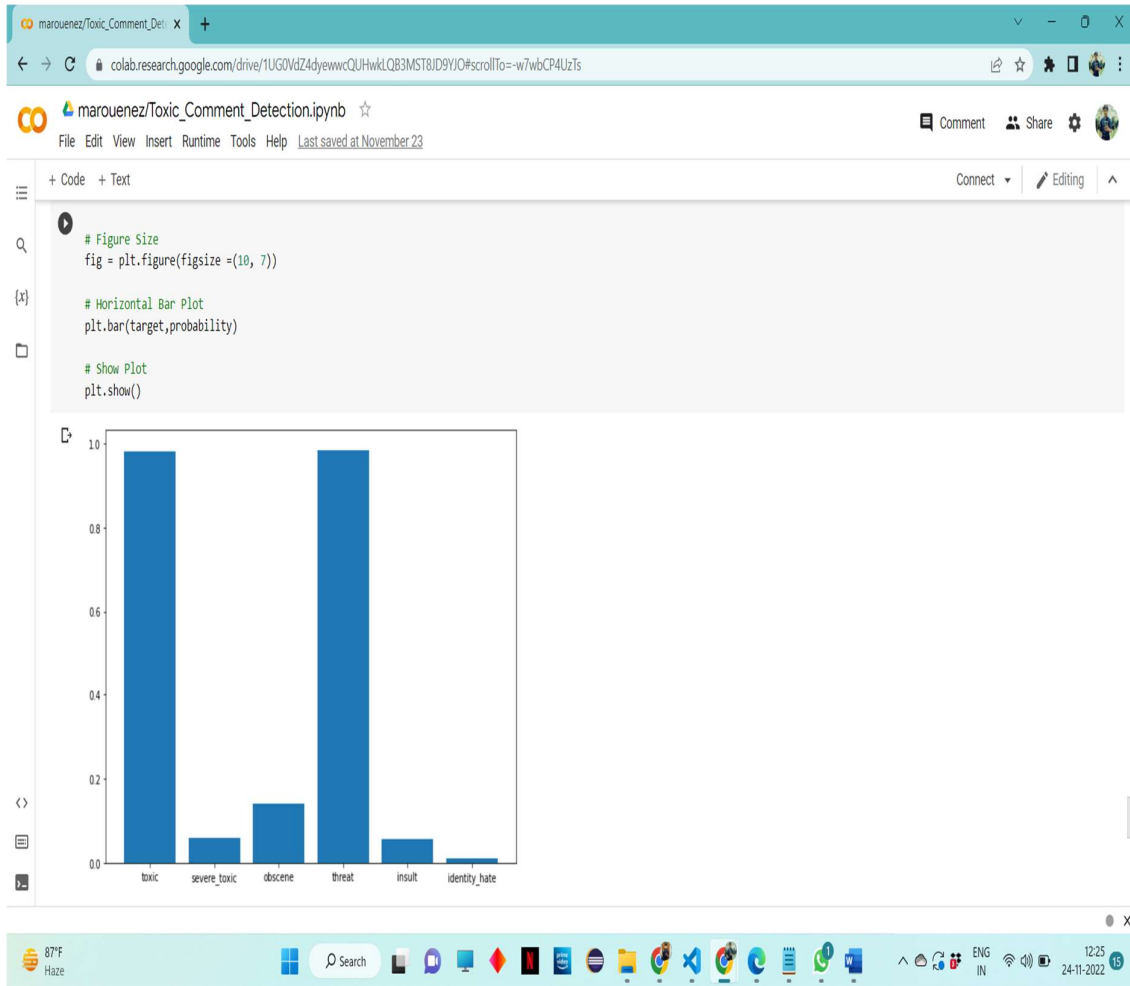


Fig 6.2

CHAPTER 7

SCREENSHOTS

The project has been implemented as a web application with all the proper modules. Here are the different screenshots of our application.

HOME PAGE

In our project home page (figure 7.1) contains register module and login module and about module it will show briefly explain what is hate speech prediction. upload the image by clicking the choose file upload button. The text extracted from the uploaded image will be shown in uploaded page (figure 7.2) and text extracted page (figure 7.3)

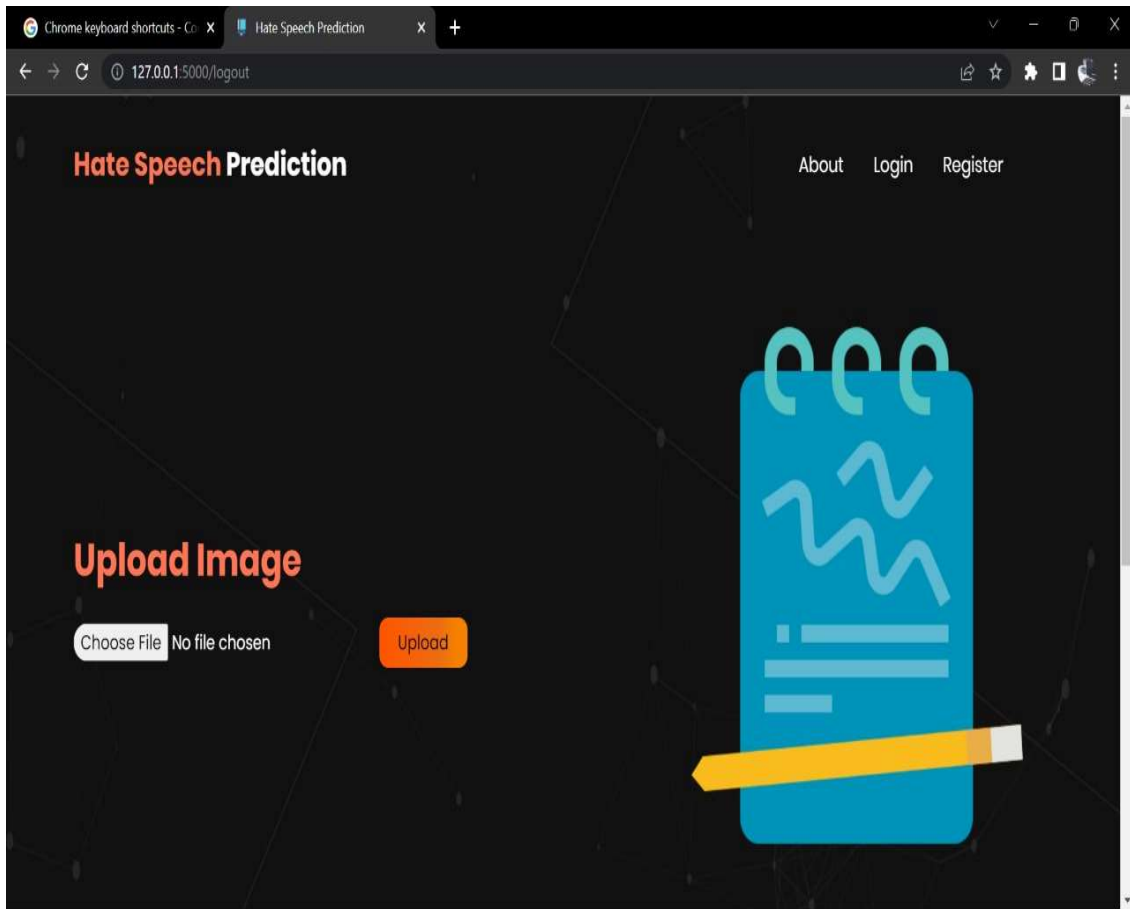


Fig 7.1 Home page

IMAGE UPLOADING PAGE

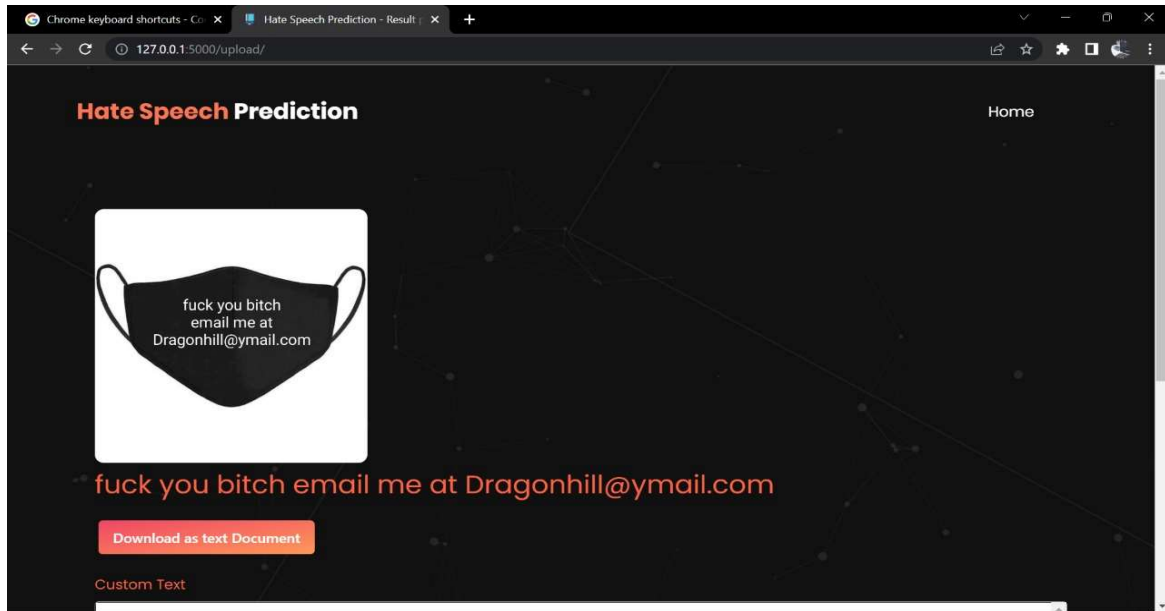


Fig 7.2 Image Uploading page

TEXT EXTRACTION PAGE

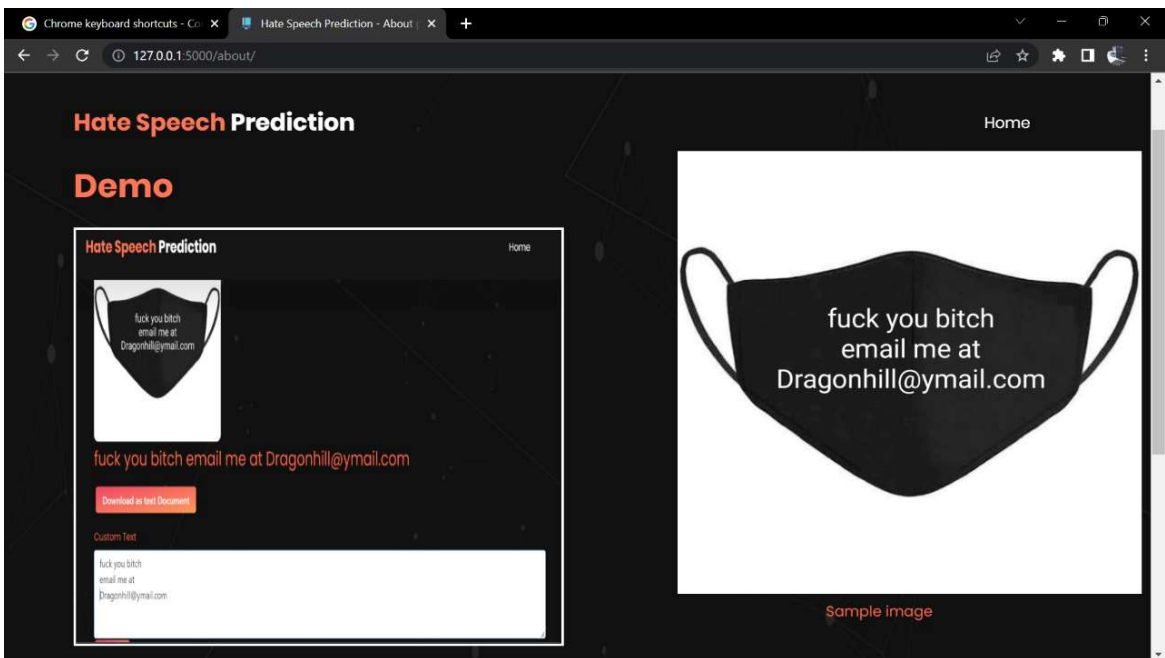


Fig 7.3 custom input Page

HATE SPEECH PREDICTION PAGE

The extracted text will be preprocessed and fit into the model logistic regression model. It will be check with each label and predicted value will be return to the sigmoid function. The function map the predicted value into probability if the probability is higher than 0.4 it will be shown in the output page .The output page shown in figure 7.4

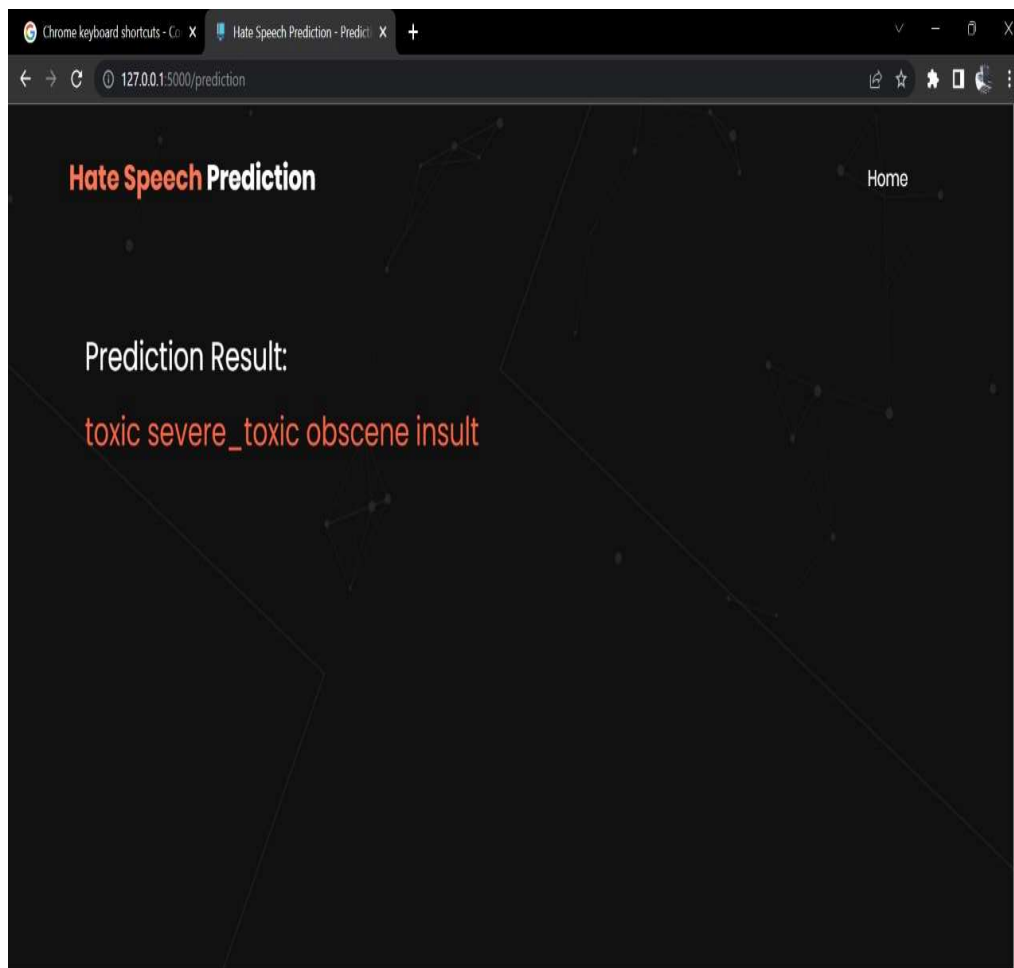


Fig . 7.4 output Page

CHAPTER 8

CONCLUSION

We present the successful approach to assessment of prediction uncertainty in hate speech classification. We demonstrate that reliability of predictions and errors of the models can be comprehensively visualized. As persons spreading hate speech might be banned, penalized, or monitored not to put their threats into actions, prediction uncertainty is an important component of decision making and can help humans observers avoid false positives and false negatives. We are getting 96% accuracy by using logistic regression algorithm model

FUTURE ENHANCEMENTS

Deploy the code on the internet as a Web Application so that the users reviews and the label(toxic or non toxic) specified by users are stored in database which can be used to train the model on in order to maintain the continuous learning.

Our future plan is to enhance the Deep Learning approach by using other type of word embeddings like FastText because it helps to handle misspelled words and that would add some accuracy for sure to the solution.

REFERENCES

[1] Asogara D.C,Ngene C.C, "Hate speech classification Using SVM and Naïve Bayes" IOSR Journal of Mobile Computing & Application (IOSR-JMCA) e- ISSN: 2394-0050, P-ISSN: 2394-0042.Volume 9, Issue 1 (Jan. – Feb. 2022), PP 27-34
www.iosrjournals.org

[2] Nina sevani, Iwan A.Soenandi , Adianto "Detection of Hate Speech by Employing support vector Machine with Word2Vec Model ",@IEEE Paper 2021

[3] Kaggle: Toxic Comment Classification Challenge

[4] Pallam Ravi, Hari Narayana Batta, Greeshma S, Shaik Yasee Toxic CommentClassification International Journal of Trend in Scientific Research and Development (IJTSRD) Volume: 3 | Issue: 4 | May-Jun 2019