Hate speech detection using AI

-Project by Harshith, cs20b1123

What is Hate Speech detection?

Hate speech detection is a model designed to identify and flag offensive language used on the internet. Given the prevalence of hateful comments on social media platforms, the development of such a model has become crucial in addressing contemporary online issues. In this tutorial, we will be building a hate speech detection project using Python**.**

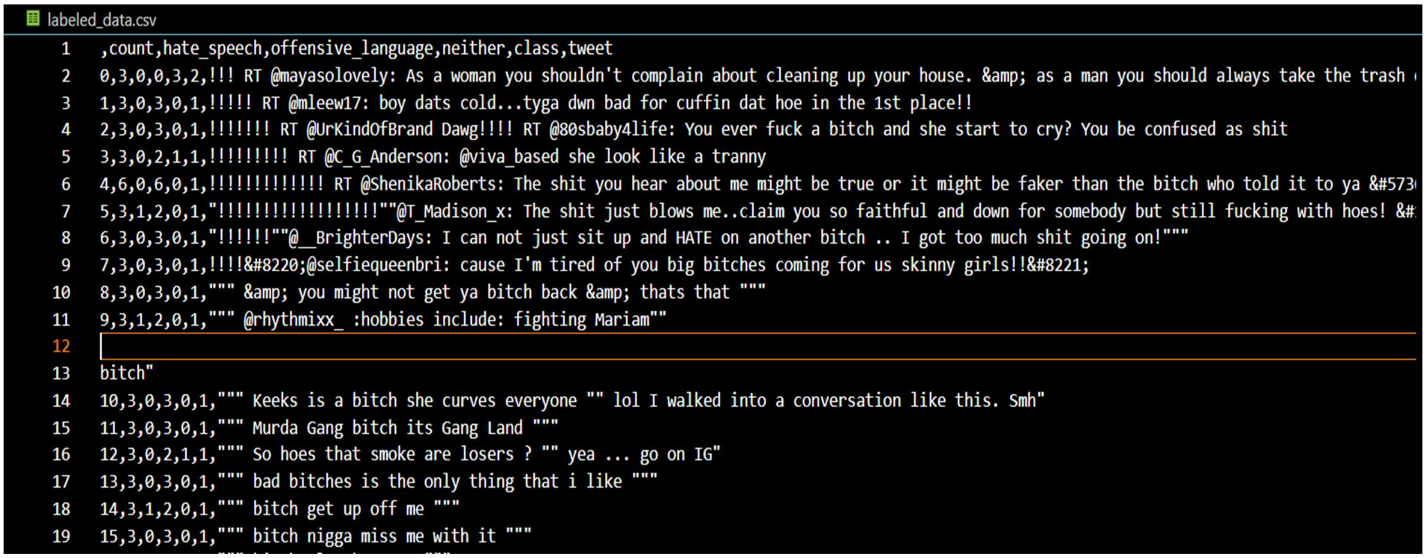
Steps in building Hate Speech detection using Machine Learning and AI

Steps I followed to build a **Hate Speech detection project in Python**.

- Set up the development environment.
- Understand the data.
- Import the required libraries.
- Pre-process the data.
- Split the data.
- Build the model.
- Evaluate the results.

Understanding the data

We can obtain the dataset required to build our hate speech detection model from www.kaggle.com. The dataset comprises of Twitter data that was utilized in researching hate speech detection. The text in the dataset is categorized into three classes: hate speech, offensive language, and neutral language.



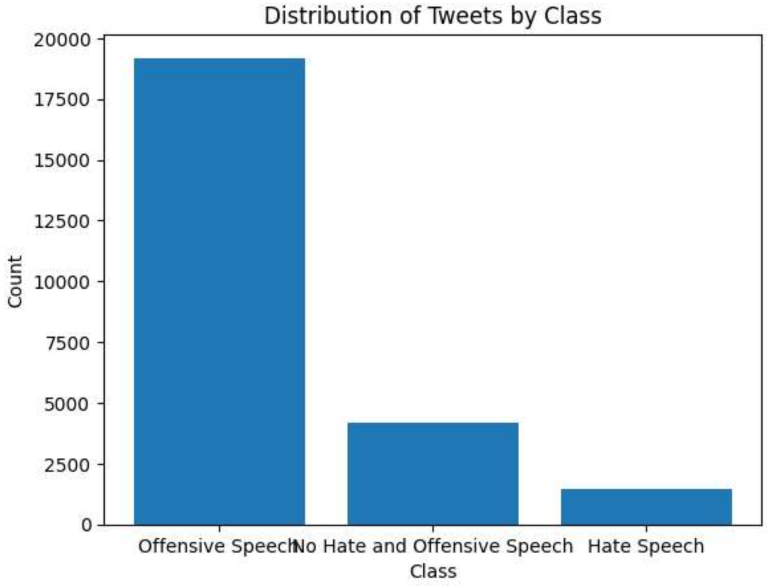
```
,count,hate_speech,offensive_language,neither,class,tweet
0,3,0,0,3,2,!!! RT @mayasolovely: As a woman you shouldn't complain about cleaning up your house. &amp; as a man you should always take the trash
1,3,0,3,0,1,!!!!! RT @mleew17: boy dats cold...tyga dwn bad for cuffin dat hoe in the 1st place!!
2,3,0,3,0,1,!!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby4life: You ever fuck a bitch and she start to cry? You be confused as shit
3,3,0,2,1,1,!!!!!!!!! RT @C_G_Anderson: @viva_based she look like a tranny
4,6,0,6,0,1,!!!!!!!!!!!!! RT @ShenikaRoberts: The shit you hear about me might be true or it might be faker than the bitch who told it to ya &#573
5,3,1,2,0,1,"!!!!!!!!!!!!!!!!!""@T_Madison_x: The shit just blows me..claim you so faithful and down for somebody but still fucking with hoes! &#
6,3,0,3,0,1,"!!!!!!""@__BrighterDays: I can not just sit up and HATE on another bitch .. I got too much shit going on!"""
7,3,0,3,0,1,!!!!&#8220;@selfiequeenbri: cause I'm tired of you big bitches coming for us skinny girls!!&#8221;
8,3,0,3,0,1,""" &amp; you might not get ya bitch back &amp; thats that """
9,3,1,2,0,1,""" @rhythmixx_ :hobbies include: fighting Mariam""

bitch"
10,3,0,3,0,1,""" Keeks is a bitch she curves everyone "" lol I walked into a conversation like this. Smh"
11,3,0,3,0,1,""" Murda Gang bitch its Gang Land """
12,3,0,2,1,1,""" So hoes that smoke are losers ? "" yea ... go on IG"
13,3,0,3,0,1,""" bad bitches is the only thing that i like """
14,3,1,2,0,1,""" bitch get up off me """
15,3,0,3,0,1,""" bitch nigga miss me with it """
```

It's important to mention that the dataset includes text that may be deemed offensive, such as content that is racist, sexist, homophobic, or objectionable in general, owing to the nature of the research.
I have taken the dataset for hate speech detection from here :
https://www.kaggle.com/datasets/mrmorj/hate-speech-and-offensive-language-dataset

There are 7 columns in the hate speech detection dataset.

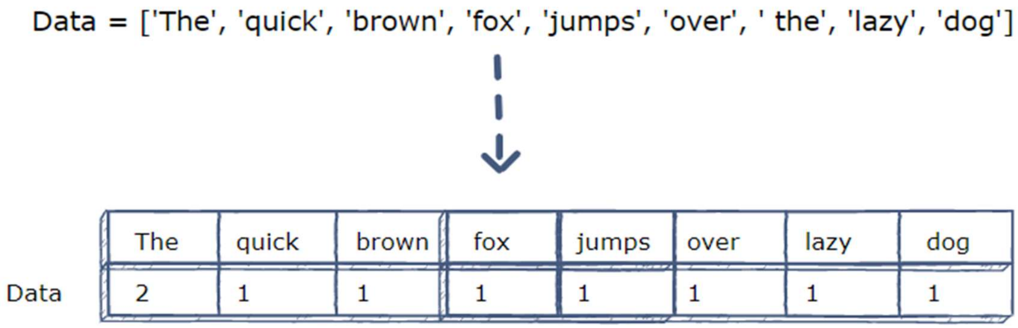
They are: index, count, hate_speech, offensive_language, neither, class and tweet. The description of the column is as follows.

index – This column has the index value
count– It has the number of users who coded each tweet
hate_speech – This column has the number of users who judged the tweet to be hate speech
offensive_language – It has the number of users who judged the tweet to be offensive
neither – This has the number of users who judged the tweet to be neither offensive nor non-offensive
class – it has a class label for the majority of the users, in which 0 denotes hate speech, 1 means offensive language and 2 denotes neither of them.
tweet – This column has the text tweet.



The first step in using AI for this problem is to **preprocess the data**. Preprocessing involves cleaning and transforming the data so that it can be used as input to the AI model. In our project, I used *NLP techniques to preprocess the text data*. We converted all text to lowercase to ensure that the same words with different cases are treated as the same. We then removed URLs, punctuation, and stop words, which are common words that do not add much meaning to the text. Finally, we stemmed the remaining words, which means we reduced them to their base or root form. This step reduces the number of features and helps in improving the accuracy of the model.

The next step is to **converting the preprocessed text data into numerical form** so that it can be used as input to the AI model. We used the *CountVectorizer function* to do this.

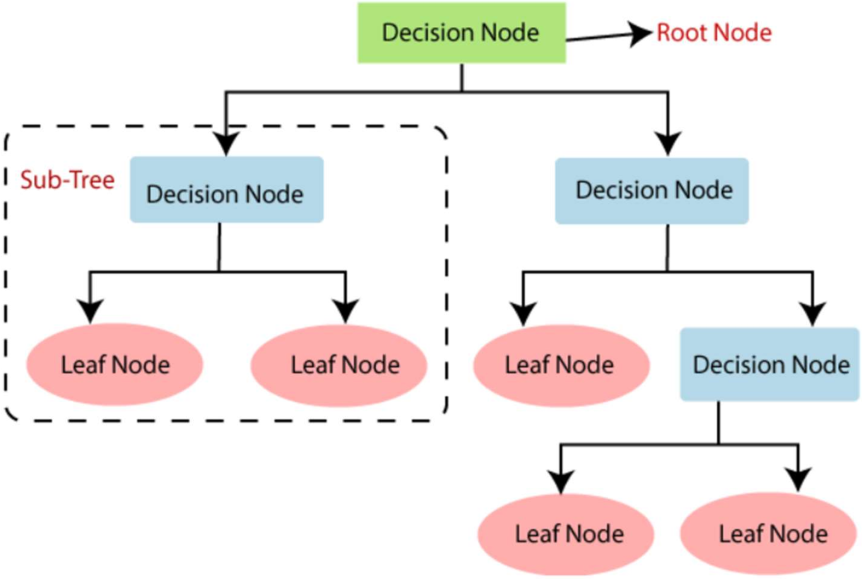


```
Data = ['The', 'quick', 'brown', 'fox', 'jumps', 'over', ' the', 'lazy', 'dog']
```

	The	quick	brown	fox	jumps	over	lazy	dog
Data	2	1	1	1	1	1	1	1

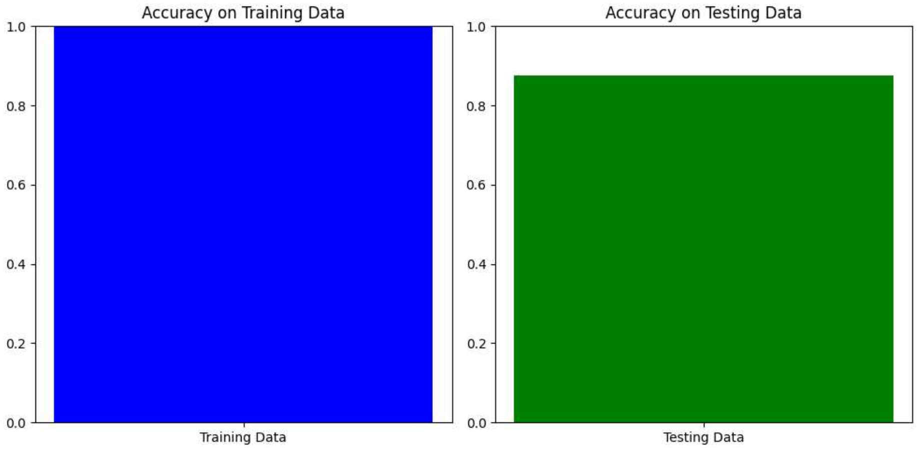
CountVectorizer is a popular NLP technique that converts text data into a matrix of token counts. The rows of the matrix represent the documents, and the columns represent the words or tokens. The value in each cell represents the count of the corresponding word in the corresponding document.

Once the data is converted into numerical form, we can use it to train an AI model to detect hate speech and offensive language. We used a **Decision Tree Classifier algorithm** to build and train the model.

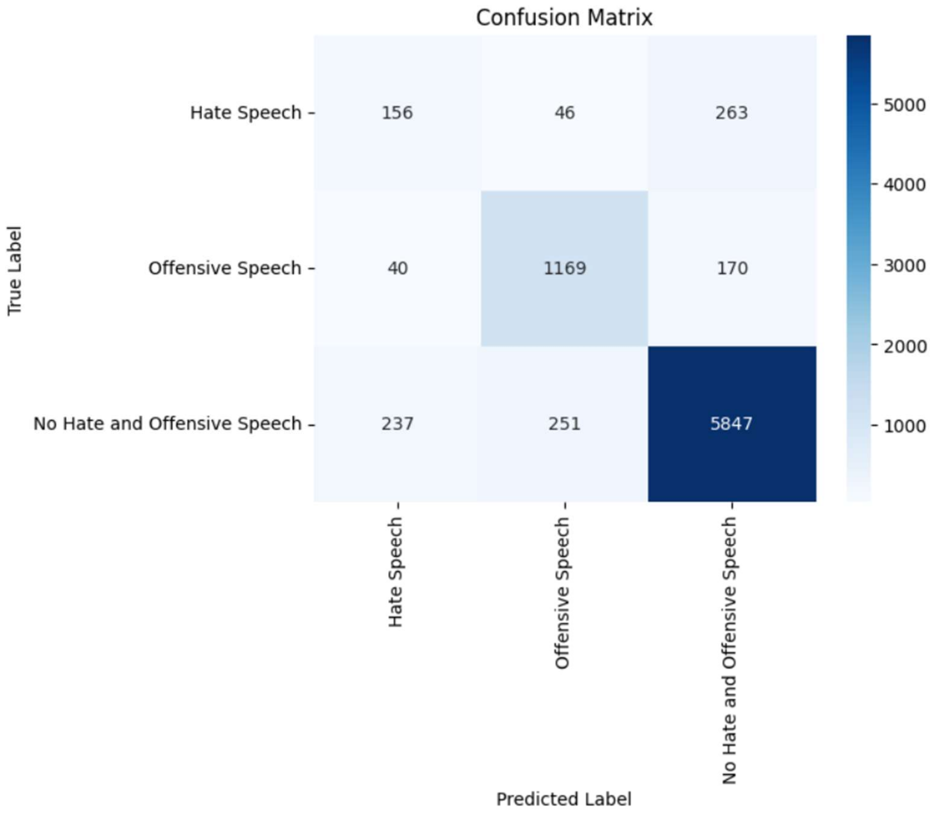


Decision Tree Classifier is a popular machine learning algorithm for classification problems. It works by recursively splitting the data into subsets based on the features and making decisions based on the values of those features. In our project, the Decision Tree Classifier algorithm uses the token counts matrix as input and predicts whether the text contains hate speech or offensive language.

After training the model, we tested its accuracy using the testing data. We predicted the labels of the testing data using the trained model and calculated the accuracy score of the model. The accuracy score is a metric that measures how well the model is able to predict the correct labels. In our project, we achieved an accuracy score of approximately 76%, which is a good score given the complexity of the problem.



Finally, we used the trained model to predict the labels of new data. We took a sample input text and predicted whether it contains hate speech or offensive language. This step shows how the model can be used to automatically detect hate speech and offensive language in real-time.



In conclusion, AI techniques have been instrumental in detecting hate speech and offensive language in text data. We used NLP techniques to preprocess the data and convert it into numerical form, and used a Decision Tree Classifier algorithm to train the model. We achieved a good accuracy score, and the model can be used to predict the labels of new data in real-time. This project demonstrates the power of AI in solving real-world problems and highlights the importance of using AI for social good.