Data Glacier

Your Deep Learning Partner

Final Report

Hate Speech Detection using Transformers

Deep Learning

https://github.com/kkalyagina/NLP_Project

Agenda

- Executive Summary
- Problem Statement
- Business understanding
- Approach
- Data understanding
- EDA
- Model Building and Training
- Performance Evaluation
- Model Prediction
- Model Deployment
- Results

Data Glacier
Your Deep Learning Partner

Group Name: individually
Name: Kristina Kaliagina
Email: kalyagina.kristina@gmail.com
Country: Russia
College/Company: Graduated from the University “Higher School of Economics”
Specialization: NLP

Problem description

The task of the project is to classify tweets from Twitter, namely, it is necessary to create a model that will help determine whether a particular tweet belongs to such a type of speech as hate speech. Hate Speech, in simple terms, is offensive language directed at individuals or groups based on their affiliation, interests, and characteristics, such as their religion, nationality, race, color, origin, gender, or other identity factor.

The task is quite difficult because of the inherent complexity of natural language constructs - different forms of hatred, different types of goals, different ways of representing the same meaning.

Business understanding

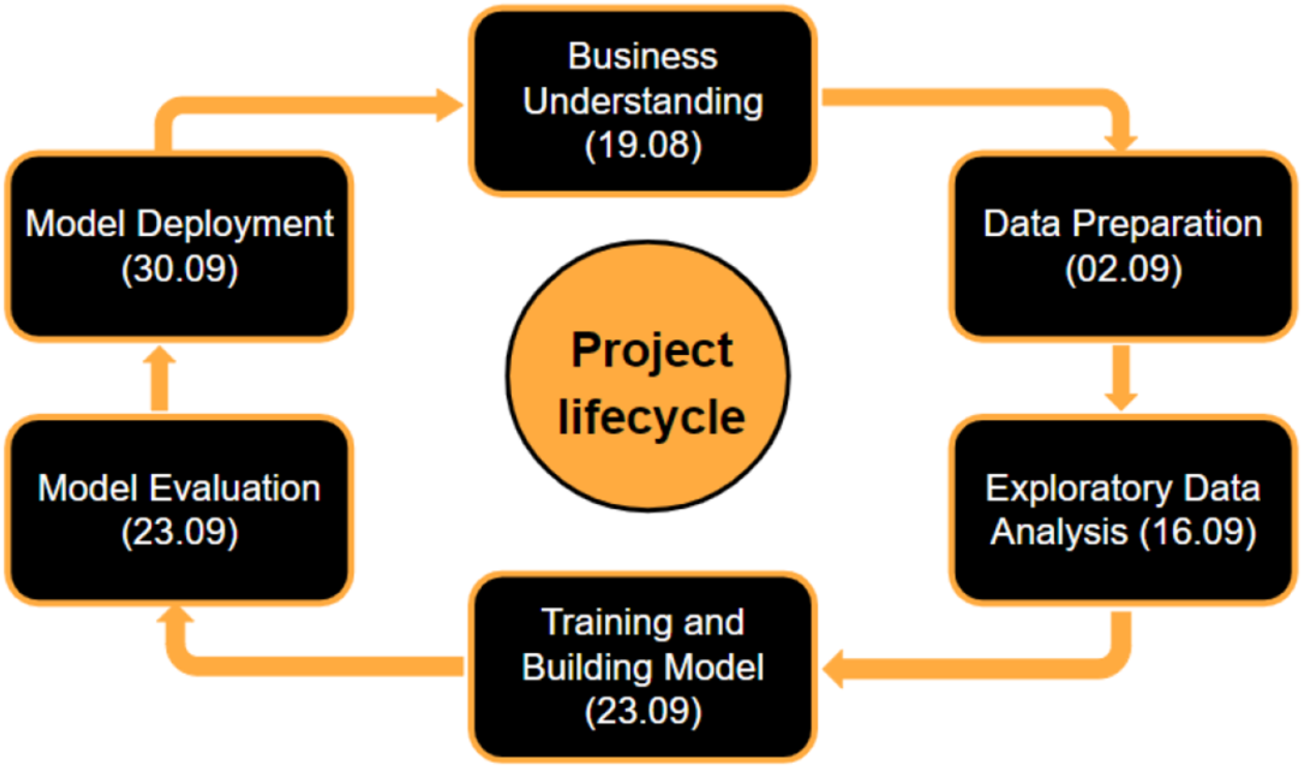
Social media has become the main driver of social change in global society. The consequences of events taking place in one corner of the world are reflected around the globe in different regions. This is because the vast amount of data generated on these platforms reaches the far corners of the world in the blink of an eye. The developers of these platforms face numerous challenges to make cyberspace as inclusive and healthy as possible. However, in recent years, the phenomena of offensive speech and hate speech have been spreading with greater force. Despite manual efforts, the scale of this problem is so huge that it cannot be solved with coordinated teams. In fact, an automated technique needs to be developed that detects and removes offensive and hateful comments before their harmful effects materialize.

The detection of such hate speech is important for the analysis of public sentiment. User groups in relation to another group, as well as to prevent illegal actions. It's also useful to filter tweets before content recommendations or explore AI chatbots on tweets.

Data Glacier

Your Deep Learning Partner

Approach



Data Glacier

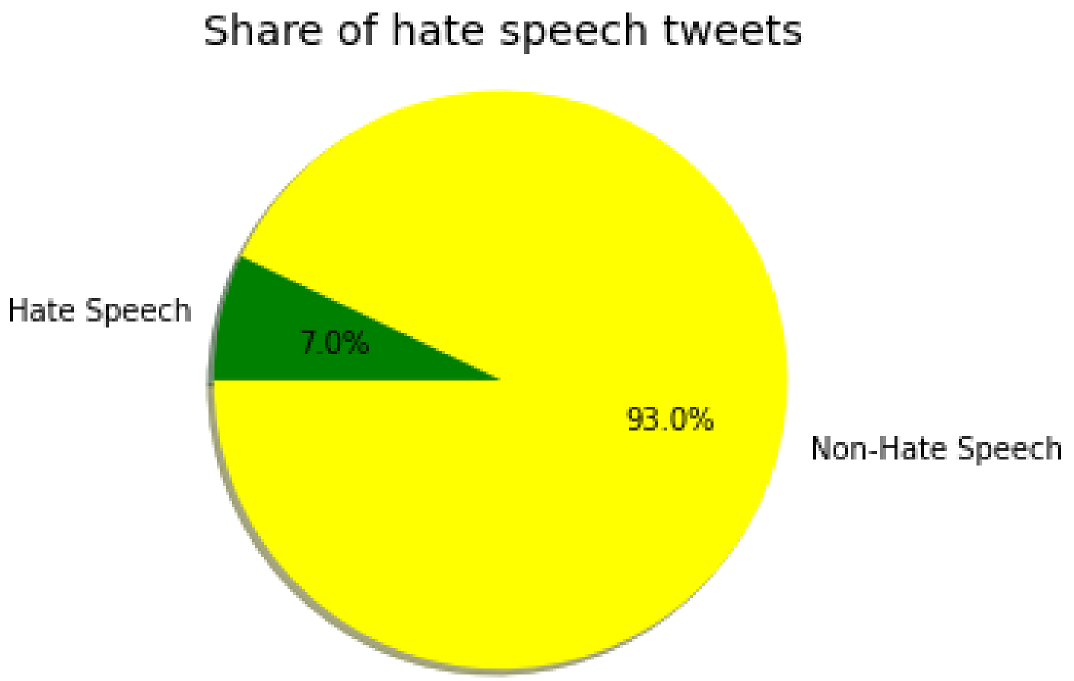
Your Deep Learning Partner

Data understanding

To solve the problem of determining hate speech, data was taken in the form of tweets from the well-known Twitter platform. The data consists of a training and test set. In total, these are 49159 observations, 31961 and 17197, respectively.

Data problems:

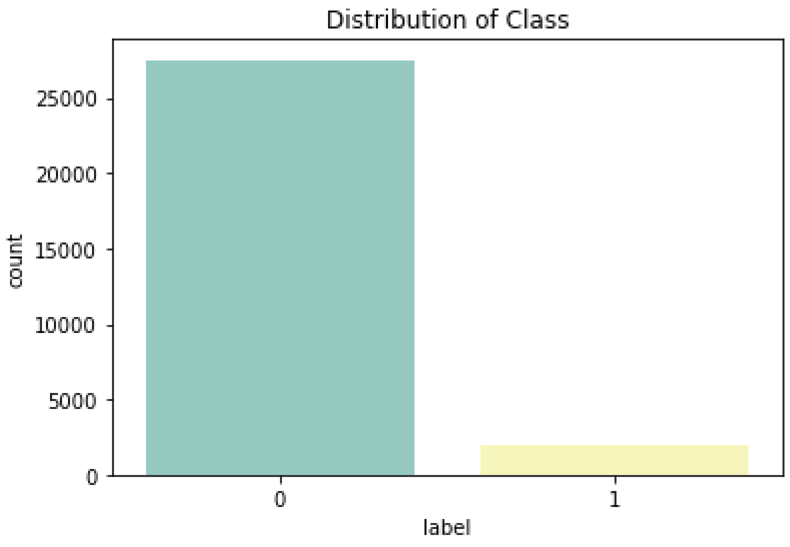
- Unbalanced classes
- Duplicate retweets
- Passes, extra spaces
- Lots of punctuation
- @usernames and #hashtags
- Slang
- Interjections
- Greek symbols

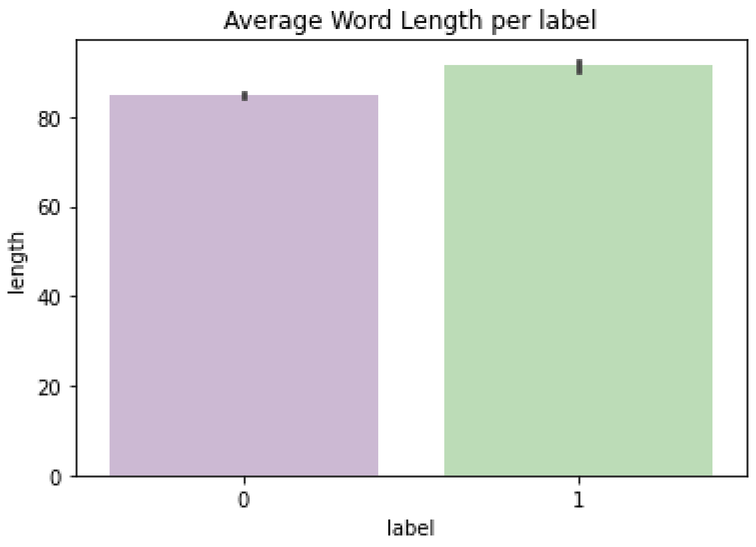


Data Glacier
Your Deep Learning Partner

Exploratory Data Analysis

There are more non hate speach than hate speech twitters. The level of hate speech tweets is 7% of the total number of observations.



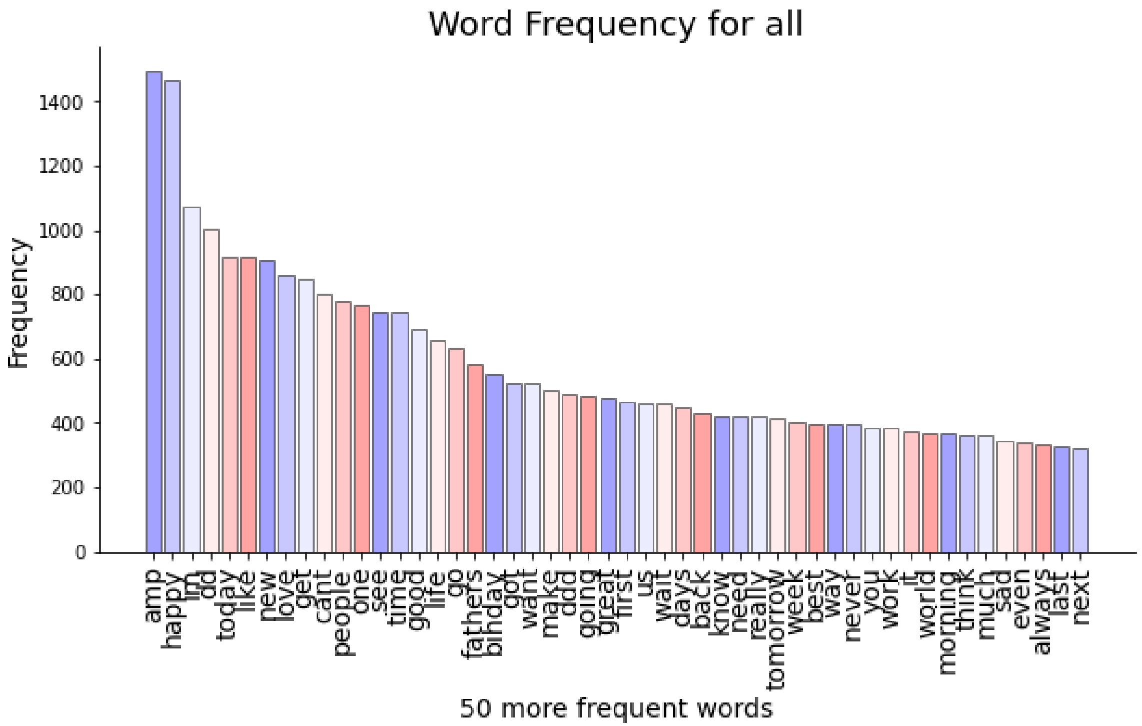


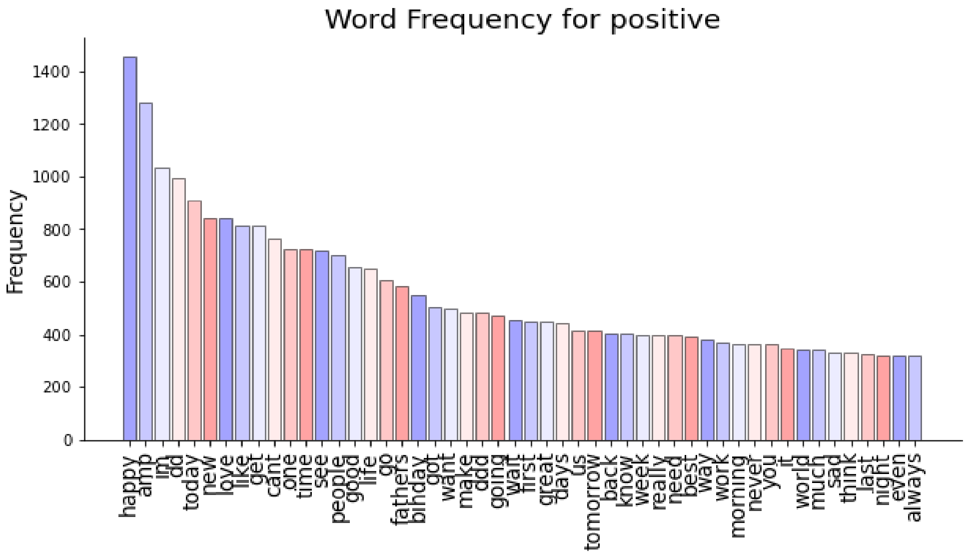
The average length of words in tweets containing hate speech is slightly longer than in regular tweets.

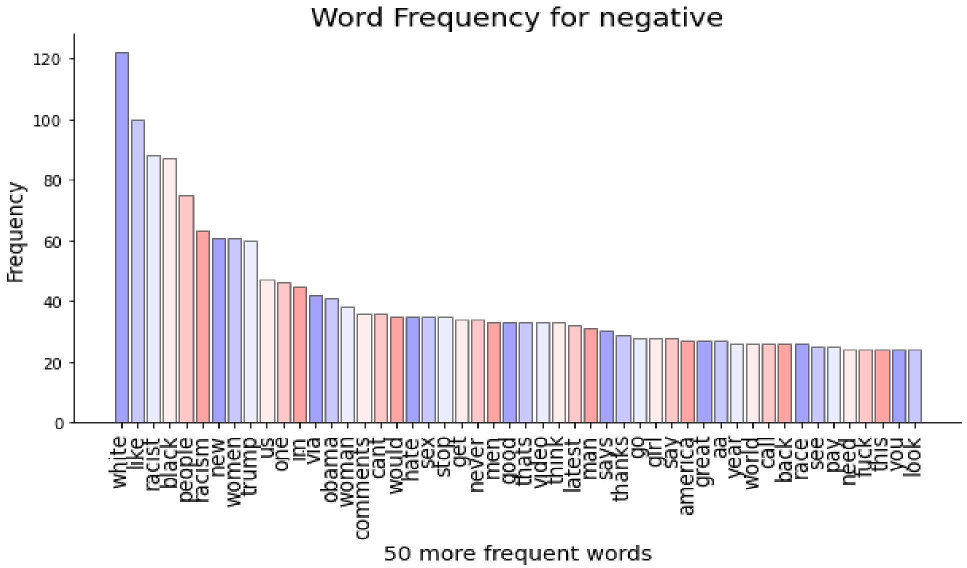
Data Glacier

Your Deep Learning Partner

Word frequency



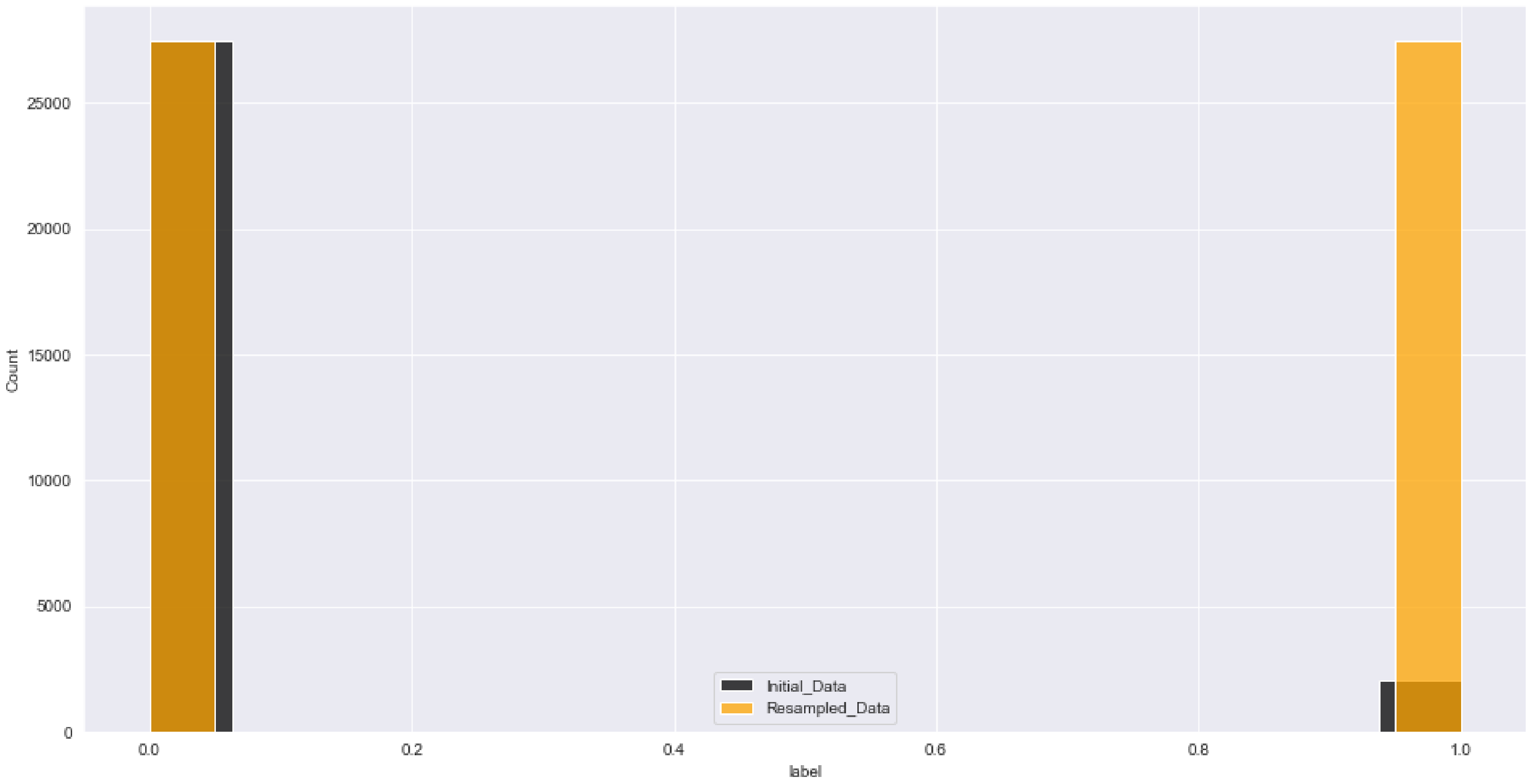




Data Glacier

Unbalanced classes

There is an imbalance of classes in the dataset, so we managed to restore the balance with the help of resampling. Since when building a model, such a ratio can spoil our results.

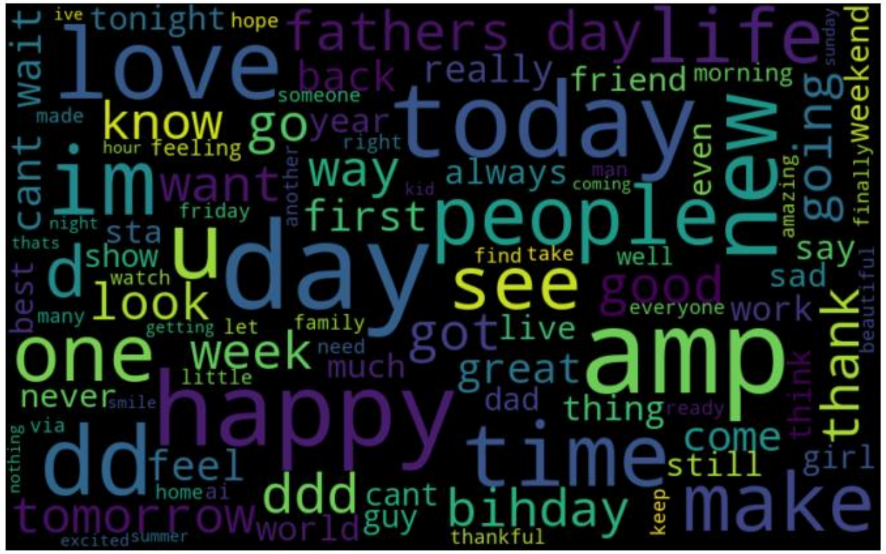


Data Glacier

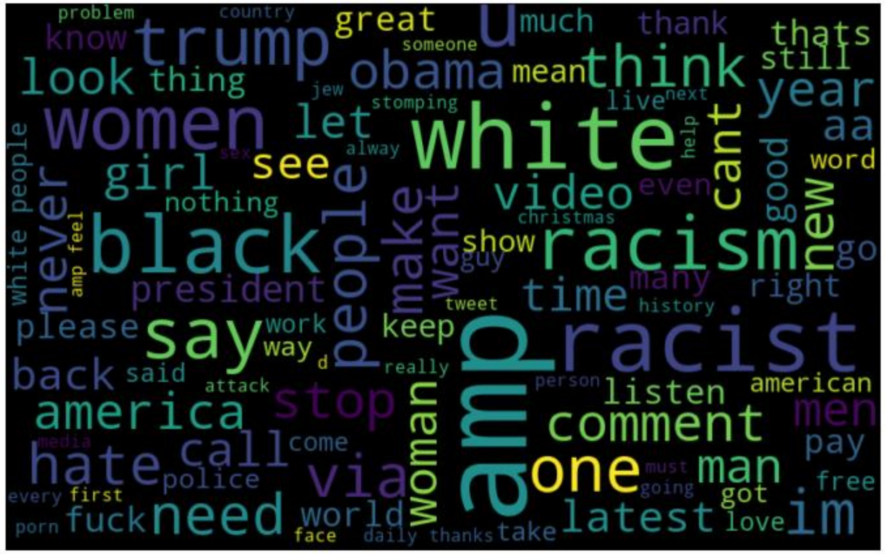
Your Deep Learning Partner

Wordcloud for tweets before resampling

Non-hate speech



Hate speech



We can see that word's common in positive comments are: love, happy, friend, life, today, day, thank, time, see, new, people, one, i'm, fathers day, good and so on

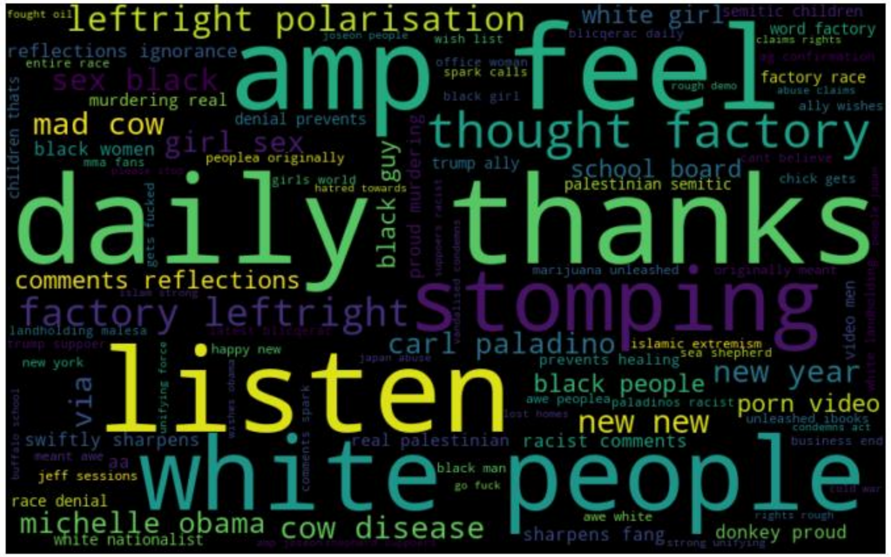
We can see that word's common in hate comments are: trump, hate, white, black, racist, racism, allahsoil, obama, women, never, america, stop and so on

Data Glacier

Wordcloud for tweets after resampling

Non-hate speech

Hate speech



There are almost the same positive words as in the situation before the resample

The last image observes words and people related to religion, politics and sex

Data Glacier

Model Building and Training

```
model.summary()
```

```
Model: "model_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_2 (InputLayer)        [(None, 24)]              0

 embedding_1 (Embedding)     (None, 24, 10)            219280

 transformer_block (Transfor  (None, 24, 10)           1235
 merBlock)

 global_average_pooling1d (G  (None, 10)               0
 lobalAveragePooling1D)

 dropout_2 (Dropout)         (None, 10)                0

 dense_3 (Dense)             (None, 20)                220

 dropout_3 (Dropout)         (None, 20)                0

 dense_4 (Dense)             (None, 1)                 21

=================================================================
Total params: 220,756
Trainable params: 220,756
Non-trainable params: 0
_________________________________________________________________
```

Transformer class:

```
class TransformerBlock(layers.Layer):
    def __init__(self, embed_dim, num_heads, ff_dim, rate=0.1):
        super(TransformerBlock, self).__init__()
        self.att = layers.MultiHeadAttention(num_heads=num_heads, key_dim=embed_dim)
        self.ffn = keras.Sequential(
            [layers.Dense(ff_dim, activation="relu"), layers.Dense(embed_dim),]
        )
        self.layernorm1 = layers.LayerNormalization(epsilon=1e-6)
        self.layernorm2 = layers.LayerNormalization(epsilon=1e-6)
        self.dropout1 = layers.Dropout(rate)
        self.dropout2 = layers.Dropout(rate)

    def call(self, inputs, training):
        attn_output = self.att(inputs, inputs)
        attn_output = self.dropout1(attn_output, training=training)
        out1 = self.layernorm1(inputs + attn_output)
        ffn_output = self.ffn(out1)
        ffn_output = self.dropout2(ffn_output, training=training)
        return self.layernorm2(out1 + ffn_output)
```

Compiling Model:

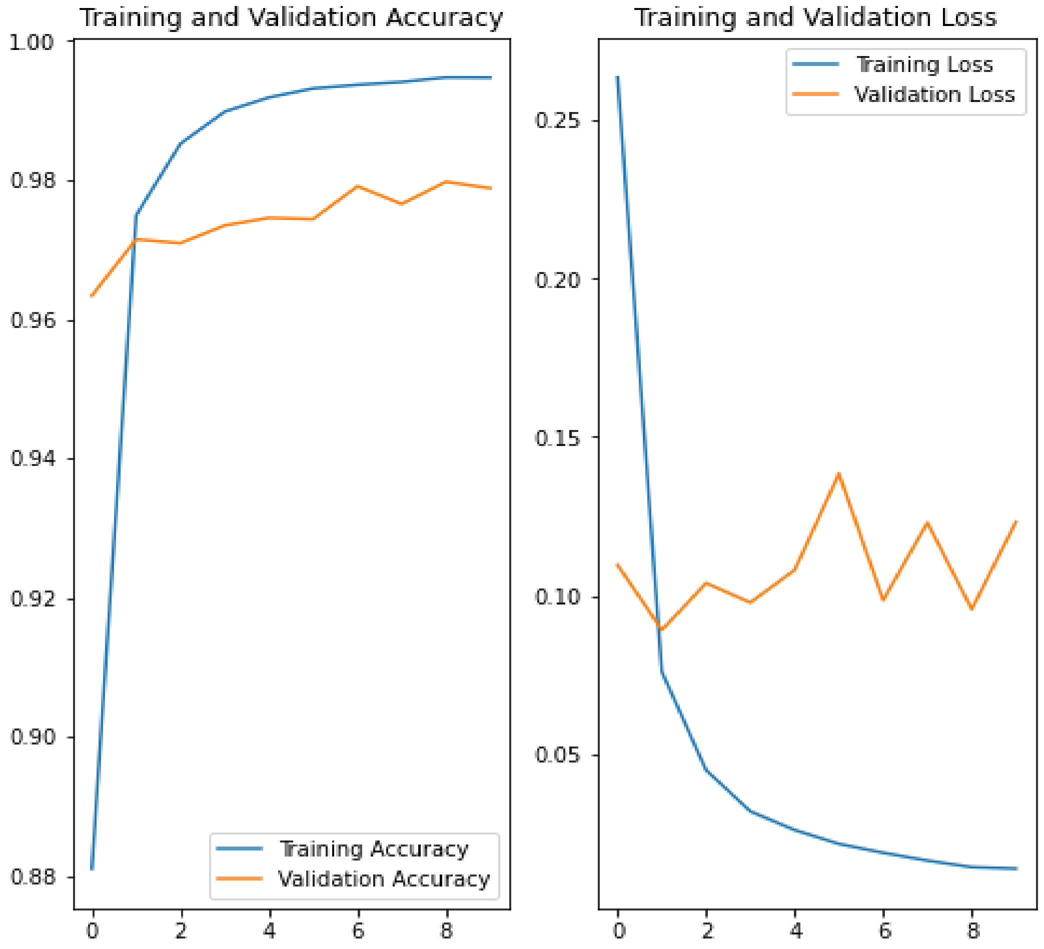
Loss: Binary Crossentropy
Optimizer: Adam
Metrics: Accuracy

Data Glacier
Your Deep Learning Partner

Performance Evaluation.
Visualizing results of the training

	Accuracy
train_set (0.80)	0.9947
validation_set (0.20)	0.9788

Validation performance is better than training performance, right from the start to the end of execution.



Data Glacier

Your Deep Learning Partner

Model Prediction

Examples of tweets	Accuracy	Prediction
'loving each other every day’	3.516552e-05 (~0.02)	Non-Hate Speech
'how many more innocent people have to die while ceain politicians choose to ignore the hate and refuse to even discuss gun control '	1.770338e-05 (~0.01)	Non-Hate Speech
‘#black lives matter is a group of #black extremist !’	0.9999895 (~1)	Hate Speech

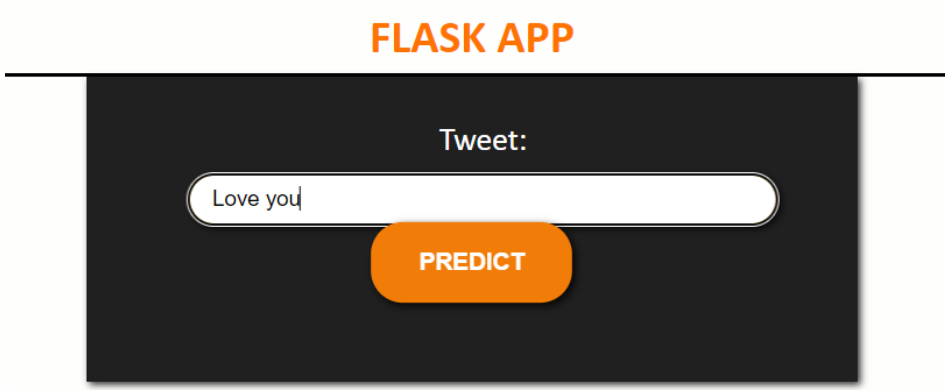
Data Glacier

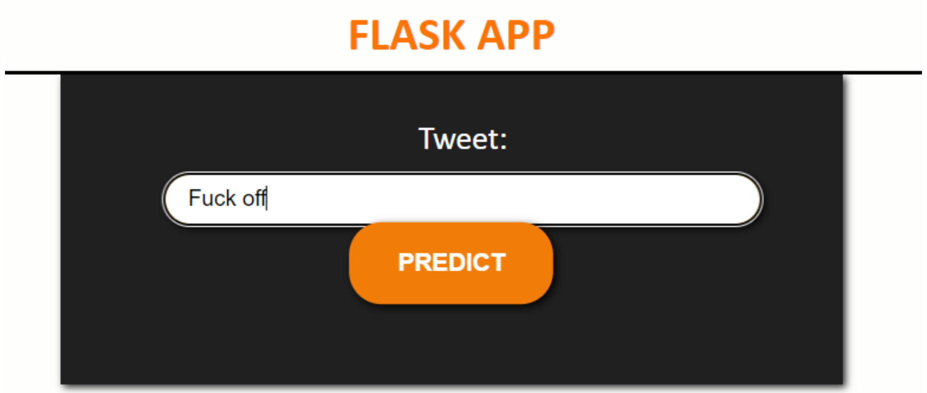
Your Deep Learning Partner

Result of Model:
The result shows the likelihood of users' tweet being related to Hate Speech.

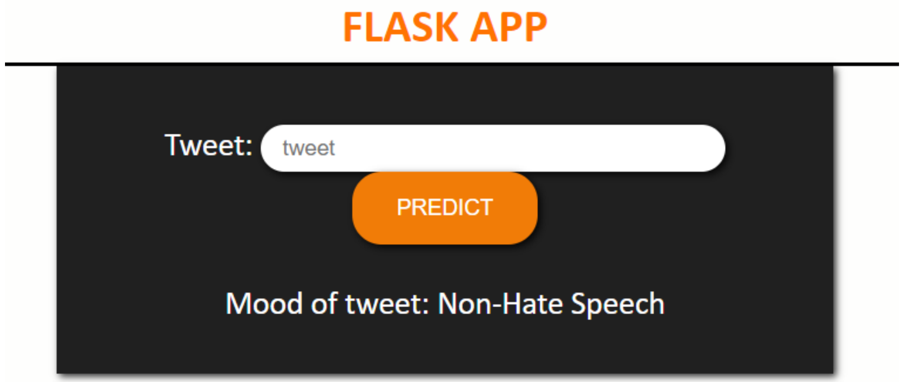
Model Deployment

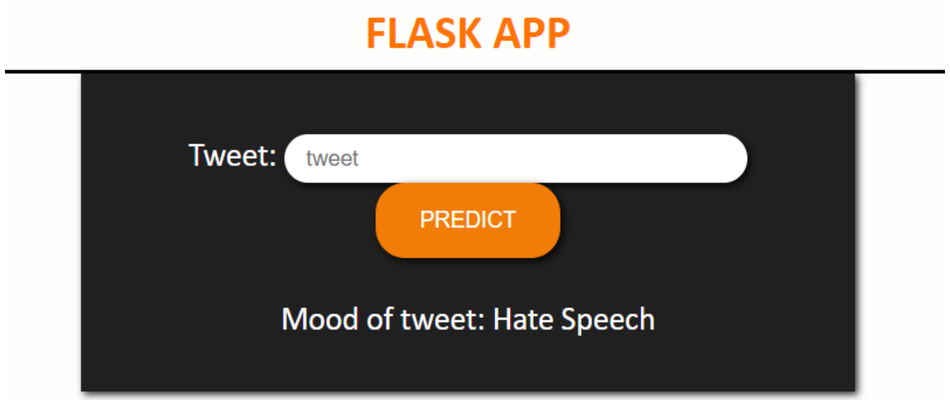
INPUT:





OUTPUT:





Data Glacier

Result of Model Deployment:
The result shows the mood of the tweets, whether it belongs to the concept of hate speech or not.

Results

- The data was prepared using the re library
- The data was balanced by resampling
- Created Transformer class
- Keras Model was build and compiled
- No overfitting
- Get ~98% validation accuracy
- Model deployment using Flask

- **Result of model** shows the likelihood of users' tweet being related to Hate Speech
- **Result of Model Deployment** shows the label of classification (Hate speech or Non-Hate Speech)

Data Glacier

Your Deep Learning Partner

Data Glacier

Your Deep Learning Partner

Thank you for attention!