# **Group name:** Arjohi

**Group members:**

**Name:** Arda Baris Basaran

**Email:** ardabarisbasaran@hotmail.com

**Country:** Turkey

**College/Company:** Middle East Technical University

**Specialization:** NLP

**Name:** Jose Luis Castañeda Terrones

**Email:** joseluiscastanedat@gmail.com

**Country:** Perú

**College/Company:** IFT-UNESP (São Paulo)

**Specialization:** NLP

**Name:** Hiten Chadha

**Email:** hitenchadha1995@gmail.com

**Country:** Denmark

**College/Company:** Technical University of Denmark

**Specialization:** NLP

**GitHub repository link:**
https://github.com/JoseLuisCastanedaT/dataglacier-week7-13.git

**Problem description & Business understanding:**

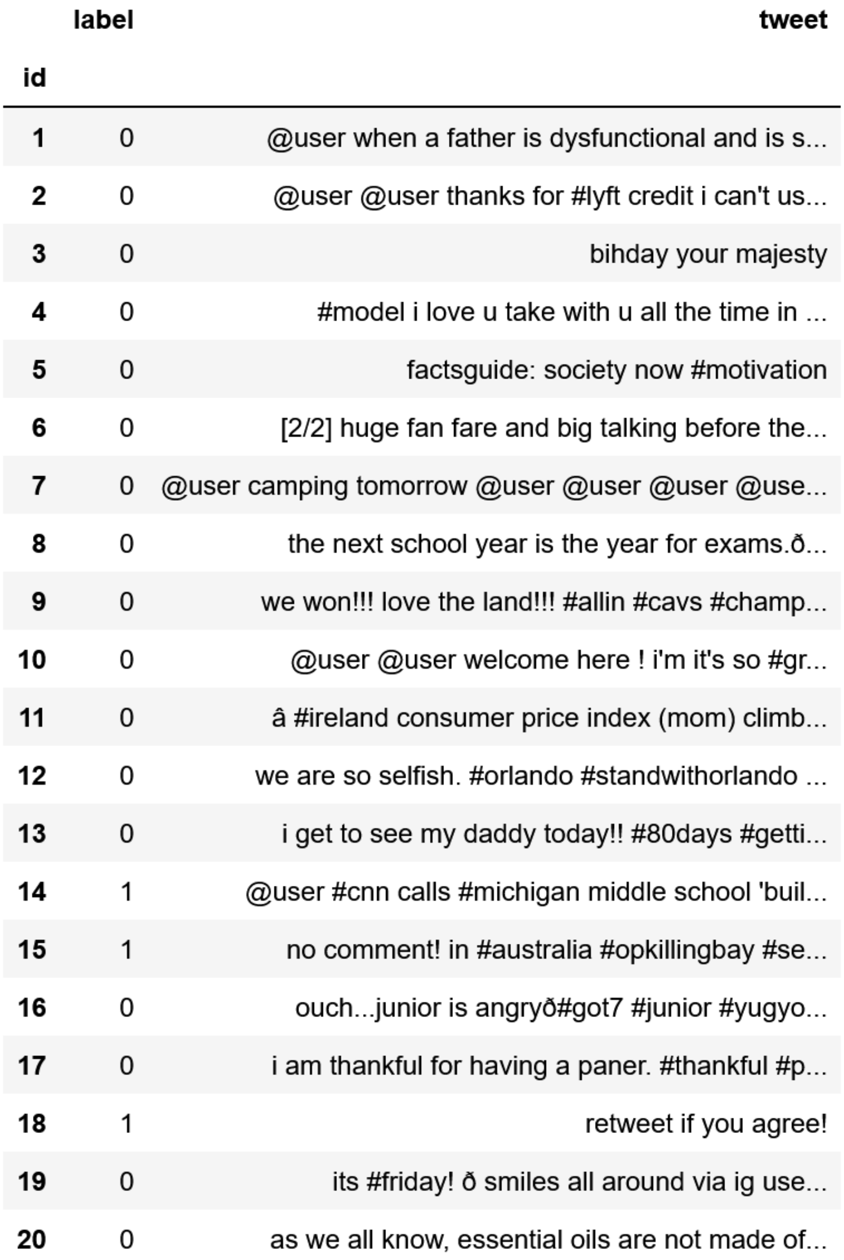
The term hate speech is understood as any type of verbal, written or behavioural communication that attacks or uses derogatory or discriminatory language against a person or group based on what they are, in other words, based on their religion, ethnicity, nationality, race, colour, ancestry, sex or another identity factor. In this problem, We will take you through a hate speech detection model with Machine Learning and Python.

Hate Speech Detection is generally a task of sentiment classification. So for training, a model that can classify hate speech from a certain piece of text can be achieved by training it on a data that is generally used to classify sentiments. So for the task of hate speech detection model, We will use the Twitter tweets to identify tweets containing Hate speech.

We will analyze a dataset CSV file from Kaggle containing 31,935 tweets. The dataset is heavily skewed with 93% of tweets or 29,720 tweets containing non-hate labeled Twitter data and 7% or 2,242 tweets containing hate-labeled Twitter data. We will try different classification algorithms after the preprocessing and data cleaning steps.

**Type of data:**

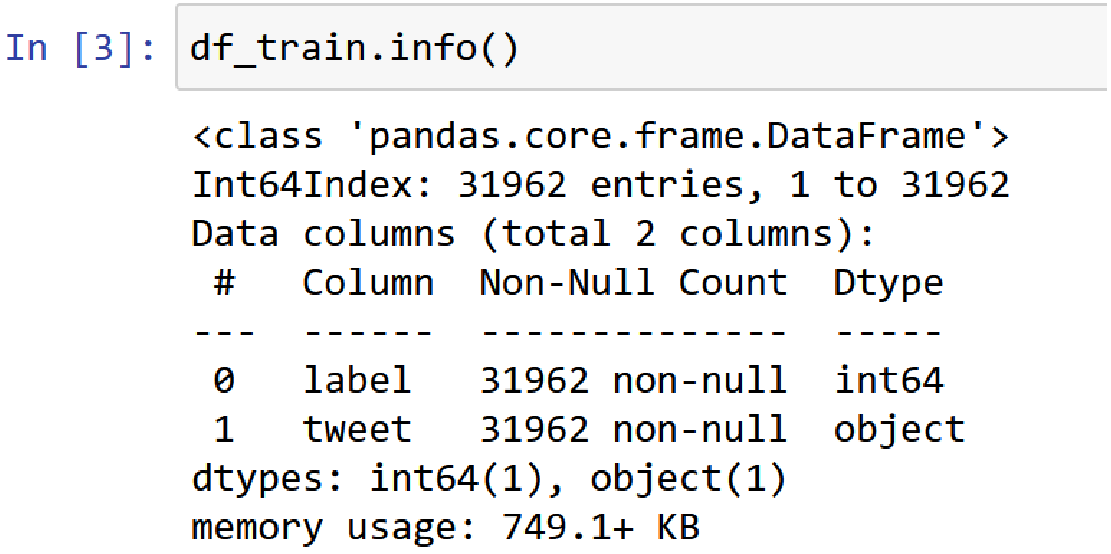
- 1 boolean column (1 representing hate speech tweet and 0 non-hate speech tweet)
- 1 string column (the tweet itself)
- 1 numerical column (index column, representing the id)



|  | label | tweet |
|---|---|---|
| id |  |  |
| 1 | 0 | @user when a father is dysfunctional and is s... |
| 2 | 0 | @user @user thanks for #lyft credit i can't us... |
| 3 | 0 | bihday your majesty |
| 4 | 0 | #model i love u take with u all the time in ... |
| 5 | 0 | factsguide: society now #motivation |
| 6 | 0 | [2/2] huge fan fare and big talking before the... |
| 7 | 0 | @user camping tomorrow @user @user @user @use... |
| 8 | 0 | the next school year is the year for exams.ð... |
| 9 | 0 | we won!!! love the land!!! #allin #cavs #champ... |
| 10 | 0 | @user @user welcome here ! i'm it's so #gr... |
| 11 | 0 | â #ireland consumer price index (mom) climb... |
| 12 | 0 | we are so selfish. #orlando #standwithorlando ... |
| 13 | 0 | i get to see my daddy today!! #80days #getti... |
| 14 | 1 | @user #cnn calls #michigan middle school 'buil... |
| 15 | 1 | no comment! in #australia #opkillingbay #se... |
| 16 | 0 | ouch...junior is angryð#got7 #junior #yugyo... |
| 17 | 0 | i am thankful for having a paner. #thankful #p... |
| 18 | 1 | retweet if you agree! |
| 19 | 0 | its #friday! ð smiles all around via ig use... |
| 20 | 0 | as we all know, essential oils are not made of... |

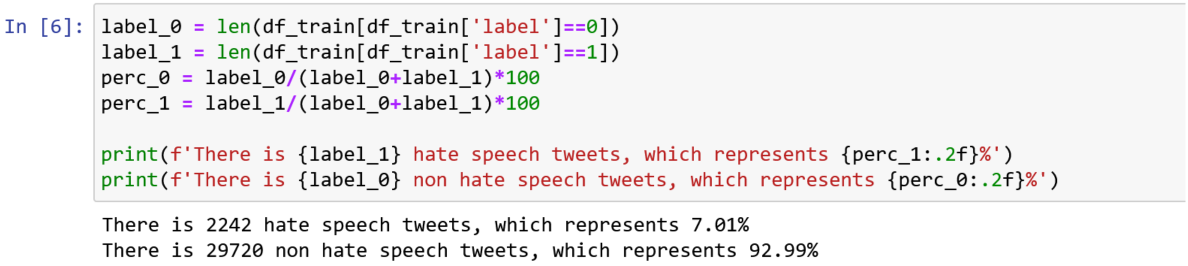
**Problems in the data:**

- There is no NA values



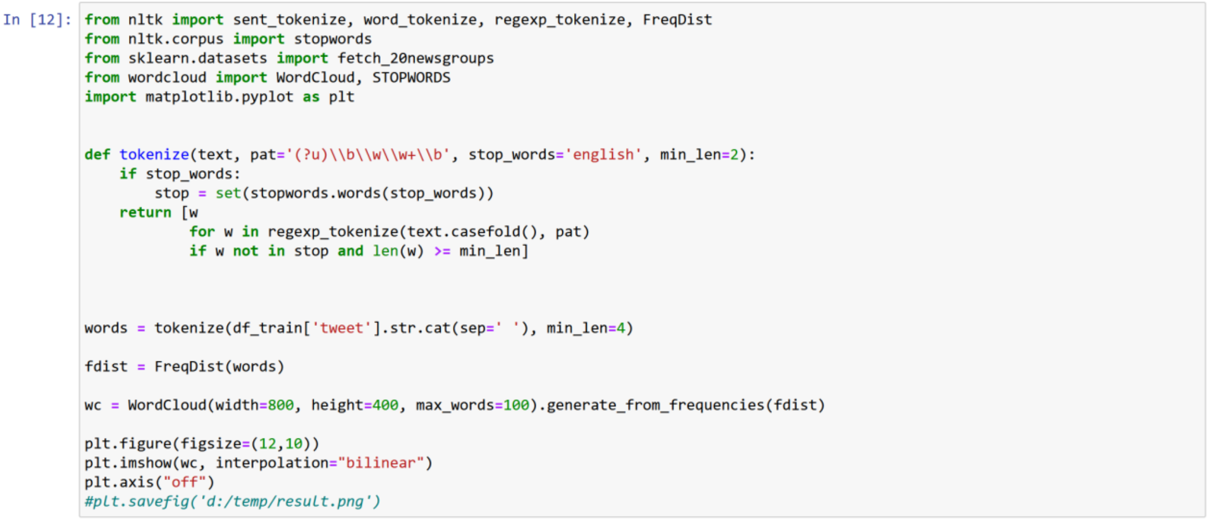
```
In [3]: df_train.info()

        <class 'pandas.core.frame.DataFrame'>
        Int64Index: 31962 entries, 1 to 31962
        Data columns (total 2 columns):
         #   Column  Non-Null Count  Dtype 
        ---  ------  --------------  ----- 
         0   label   31962 non-null  int64 
         1   tweet   31962 non-null  object
        dtypes: int64(1), object(1)
        memory usage: 749.1+ KB
```

- The dataset is heavily skewed



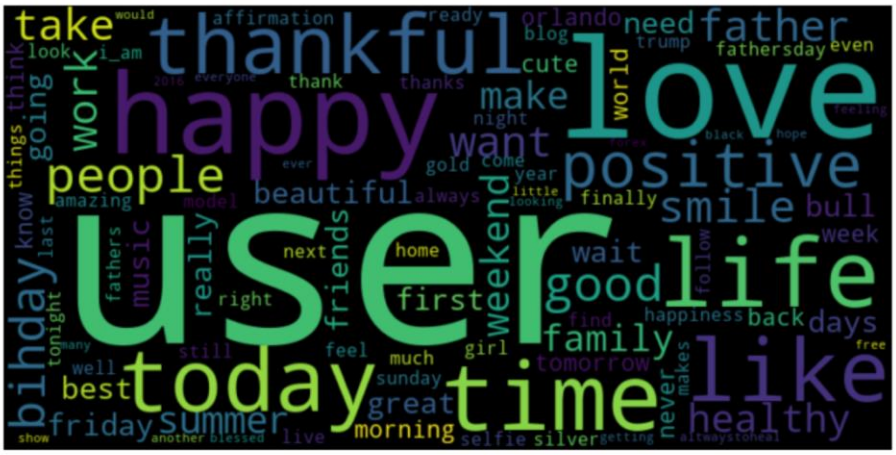
```
In [6]: label_0 = len(df_train[df_train['label']==0])
        label_1 = len(df_train[df_train['label']==1])
        perc_0 = label_0/(label_0+label_1)*100
        perc_1 = label_1/(label_0+label_1)*100

        print(f'There is {label_1} hate speech tweets, which represents {perc_1:.2f}%')
        print(f'There is {label_0} non hate speech tweets, which represents {perc_0:.2f}%')

        There is 2242 hate speech tweets, which represents 7.01%
        There is 29720 non hate speech tweets, which represents 92.99%
```

- We can also notice the majority of non hate speech tweets by inspecting the most frequent words (love, happy, thankful, positive, ...):



```
In [12]: from nltk import sent_tokenize, word_tokenize, regexp_tokenize, FreqDist
         from nltk.corpus import stopwords
         from sklearn.datasets import fetch_20newsgroups
         from wordcloud import WordCloud, STOPWORDS
         import matplotlib.pyplot as plt


         def tokenize(text, pat='(?u)\\b\\w\\w+\\b', stop_words='english', min_len=2):
             if stop_words:
                 stop = set(stopwords.words(stop_words))
             return [w
                     for w in regexp_tokenize(text.casefold(), pat)
                     if w not in stop and len(w) >= min_len]



         words = tokenize(df_train['tweet'].str.cat(sep=' '), min_len=4)

         fdist = FreqDist(words)

         wc = WordCloud(width=800, height=400, max_words=100).generate_from_frequencies(fdist)

         plt.figure(figsize=(12,10))
         plt.imshow(wc, interpolation="bilinear")
         plt.axis("off")
         #plt.savefig('d:/temp/result.png')
```

```
Out[12]: (-0.5, 799.5, 399.5, -0.5)
```



(The most frequent being ‘user’, but this will be removed when cleaning and preprocessing the tweets)

**To handle the imbalanced data problem we will try different over and under-sampling techniques like:**

- Random Oversampling (equate the minority dataset to the majority dataset copying the data)
- Smote Oversampling (same as random oversampling but creating synthetic data)
- Random Undersampling (equate the majority dataset to the minority dataset)
- Nearmiss Undersampling (eliminating samples from the majority dataset but preventing information loss using KNN)