**Project:** Hate Speech detection using Transformers (Deep Learning)

## Project Report


- **Group name:** Speechium
- **Names:** Richard Flores, Christos Christoforou
- **Email:** rflo147@wgu.edu, christoschristoforou97@hotmail.com
- **Country:** United States, United Kingdom
- **Specialization:** NLP
- **GitHub repo link:** https://github.com/DataDaimon/twitter-sentiment


## Problem Statement:

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.

## Business Understanding

Detection of hate speech in tweets is an important issue for businesses to consider for several reasons.

First, hate speech can be harmful and offensive to individuals and groups, and businesses have a social responsibility to address it. In addition, businesses may face legal and reputational risks if they fail to address hate speech on their platforms.

Second, businesses that operate social media platforms or engage in social media marketing may need to monitor and address hate speech to maintain the trust and loyalty of their users and customers. If a business is perceived as tolerating hate speech, it may face backlash from users and negative media attention.

Finally, businesses may also have a financial incentive to address hate speech, as it can negatively impact the user experience and drive users away from the platform.

To detect hate speech in tweets, businesses may use a combination of automated tools and human moderation. Automated tools may include machine learning algorithms that are trained to identify hate speech based on certain characteristics, such as the use of certain words or phrases. Human moderation may involve a team of moderators who review tweets and take appropriate action, such as deleting the tweet or banning the user.

It's important to note that detecting hate speech can be challenging, as it may involve complex issues of context and intent. It is also important for businesses to consider the potential for false positives and ensure that their approaches to detecting and addressing hate speech are fair and transparent.

## Project lifecycle and deadlines:

- **Week 7:** Data collection & Problem description | **Due date:** 19/12/2022
- **Week 8:** Data understanding & preprocessing | **Due date:** 26/12/2022
- **Week 9:** Data cleansing and transformation | **Due date:** 02/01/2023
- **Week 10:** Model Building & Training | **Due date:** 09/01/2023
- **Week 11:** Model Evaluation & Selection | **Due date:** 16/01/2023
- **Week 12:** Model Deployment | **Due date:** 23/01/2023
- **Week 13:** Final Report, Code, and Presentation | **Due date:** 30/01/2023

## Data Intake Report:

**Name:** Hate Speech Detection

**Report date:** 18/12/2022

**Internship Batch:** LISUM15

**Version:** 1.0

**Data intake by:** Richard Flores, Christos Christoforou

**Data storage location:** Twitter hate speech | Kaggle

**Tabular data details: train_E6oV3lV**

| | |
|---|---|
| **Total number of observations** | 31962 |
| **Total number of files** | 1 |
| **Total number of features** | 3 |
| **Base format of the file** | .csv |
| **Size of the data** | 3MB |

**Tabular data details: test_tweets_anuFYb8**

| | |
|---|---|
| **Total number of observations** | 17197 |
| **Total number of files** | 1 |
| **Total number of features** | 2 |
| **Base format of the file** | .csv |
| **Size of the data** | 1.6MB |

**Data understanding**

In this section, we familiarise ourselves with the data and try to understand them. In our case, we have 3 features in the training dataset, which are pretty much straight forward to understand. Namely, we have:

- `id’: the primary key,
- `label’: 0 for free speech, 1 for hate speech,
- `tweet’: the tweet we want to classify.

Since the `id’ column is not needed for our purpose, we will drop it. Next, we checked the shape of the dataset, the type of each feature, and if there are any null values or duplicated rows.

- Feature types were correct, `label’ : int64 and `tweet’: object,
- No null-values,
- Found and removed 2432 duplicated rows.

Finally, we checked if our dataset is balanced. In other words, whether we have about the same number or free speech and hate speech tweets or not. We found out that our dataset is highly imbalanced. To address this issue, we will later apply an oversampling or a downsampling technique to balance our dataset.

**Data cleansing and transformation completed:**

- **Standard Data Cleaning and Transformation**
  - Verify data types
  - Check for NULL values
  - Remove duplicated data
  - Check for missing data

- **NLP Specific Data Cleaning and Transformation**
  - Apply Tokenization and Lemmatization
  - Lowercase the text
  - Remove stop-words and one-letter words
  - Remove tags and other special characters
  - Remove non-ASCII characters
  - Vectorize the training data using the TfidfVectorizer