Hate speech detection using transformers

Group Name: MKYK-Hate-Speech-Detector

- Mussie Berhane:
    - berhanem2000@gmail.com
    - NLP
- Khaled Elshamy:
    - kelshamy094@gmail.com
    - NLP
- Yitong Lu:
    - guotonton@gmail.com
    - NLP
- Khalid Lawal:
    - Khalid6199lawal@gmail.com
    - NLP

Problem description:

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.

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 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:

The transformer neural network architecture has boasted significant improvements for accuracy for a variety of tasks compared to previous deep learning models. In particular, through the attention mechanism they allow for more accurate reading of text, maintaining long range dependencies between words in the documents. This architecture is simpler than other models and requires less time for training while being more parallelizable. Automated hate speech detection is an important tool in combating the spread of hate speech, and is thus of particular importance for social media companies. Here, we attempt to use the transformer for this task.

Data

I first notice that:

1) The tweets are in string format, so we will need to tokenize and encode them. There are several encoders such as the BERT Tokenizer. We will use this to begin with and iterate as the model progresses.
2) Also, we are dealing with an imbalanced dataset. Only 7% of the dataset corresponds to negative tweets, and the training set is quite small as well. We can

begin training with the data as is, then if the model performance is subpar we will experiment with generating synthetic data and under/over sampling.