



Data Science Intern at Data Glacier

Project: Hate Speech Detection using Transformers (Deep Learning)

Week 8: Deliverables

Name: Anitha Venkatachalam

Email: anitha.Venkat83@gmail.com

Country:

Specialization: Data Science

Batch Code: LISUM14

Date: 22Nov 2022

Submitted to: Data Glacier

Table of Contents:

| | |
|---------------------------------------|---|
| 1. Project Plan | 3 |
| 2. Problem Statement | 3 |
| 3. Data Collection | 3 |
| 4. Data Preprocessing | 4 |
| 4.1. Text Cleaning..... | 4 |
| 4.1.1. Lower Case..... | 4 |
| 4.1.2. Remove Punctuation..... | 4 |
| 4.1.3. Remove URL..... | 4 |
| 4.1.4. Remove @tags..... | 4 |
| 4.1.5. Remove Special Characters..... | 4 |

1. Project Plan

| Weeks | Date | plan |
|----------|----------------|---|
| Weeks 07 | Nov 14th, 2022 | Problem Statement, Data Collection, Data Report |
| Weeks 08 | Nov 21, 2022 | Data Preprocessing (Text Cleaning) |
| Weeks 09 | Nov 28, 2022 | Data Preprocessing (Preprocessing Operation + Feature Extraction) |
| Weeks 10 | Dec 5, 2022 | Building the Model |
| Weeks 11 | Dec 12, 2022 | Model Result Evaluation |
| Weeks 12 | Dec 19, 2022 | Flask Development + Heroku |
| Weeks 13 | Dec 26, 2022 | Final Submission (Report + Code + Presentation) |

2. 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, color, 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 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 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.

3. Data Collection

The Data is about Twitter hate Speech taken from Kaggle [1] which contains the 3 number of features and 31962 number of observations. Dataset using Twitter data, it was used to research hate-speech detection. The text is classified as: hate-speech, offensive language, and neither. Due to the nature of the study, it is important to note that this dataset contains text that can be considered racist, sexist, homophobic, or offensive.

Table 1: Data Information

| | |
|-------------------------------------|---------|
| 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 | 2.95 MB |

4. Data Preprocessing

In part, we explain the data preprocessing approach that we apply in the text data.

4.1 Text Cleaning

First, we clean our text because it was so messy data.

4.1.1 Lowercase

Converting a word to lower case (NLP -> nlp). Words like Racism and racism mean the same but when not converted to the lower case those two are represented as two different words in the vector space model (resulting in more dimensions). Therefore, we convert all text word into lower case letter.

4.1.2 Remove Punctuation

It is important to remove the Punctuation because is not important. Therefore, we remove that Punctuation in order to do that we use regular expression.

4.1.3 Remove URLs

In this part, we remove URLs because we are working on hate speech application which detect the hate and free speech and to get the output, we need to give only text not URLs therefore, we remove the URLs because we need only clean text input.

4.1.4 Remove @tags

In this part, we remove @tags which basically used when we mentioned someone So, it's doesn't concern to our application therefore, we remove @tags by using regular expressions.

4.1.5 Remove Special Characters

Remove Special Characters is essentially the following set of symbols [!'"#\$%&'()*+,-./:;<=>?@[^_`{|}~] which basically don't have meaning. Therefore, we remove that kind of symbols because we don't need that. In order to remove we use python isalnum method.

