Data Science Coding Challenge: Twitter Sentiment Analysis

Vasco Fernandes

August 24, 2018

Contents

1 Introduction

Hate speech is an unfortunately common occurrence on the Internet. Often social media sites like Facebook and Twitter face the problem of identifying and censoring problematic posts while weighing the right to freedom of speech. The importance of detecting and moderating hate speech is evident from the strong connection between hate speech and actual hate crimes.

Early identification of users promoting hate speech could enable outreach programs that attempt to prevent an escalation from speech to action. Sites such as Twitter and Facebook have been seeking to actively combat hate speech. In spite of these reasons, NLP research on hate speech has been very limited, primarily due to the lack of a general definition of hate speech, an analysis of its demographic influences, and an investigation of the most effective features.

2 Problem Statement

The objective of this project is to find the best classification model to classify tweets.

The objective of this task is to detect hate speech in tweets. For the sake of simplicity, we say a tweet contains hate speech if it has a racist or sexist sentiment associated with it. So, the task is to classify racist or sexist tweets from other tweets.

Formally, given a training sample of tweets and labels, where label ’1’ denotes the tweet is racist/sexist and label ’0’ denotes the tweet is not racist/sexist, your objective is to predict the labels on the test dataset.

3 Data Set

3.1 Train Set

The data set provided (https://datahack.analyticsvidhya.com/contest/practice-problem-twitter-sentiment-analysis/) is composed by two different files, the train set (”*train_E6oV3lV.csv*”) and test set (”*test_tweets_anuFYb8.csv*”). The first file is composed by 31621 observations, with 3 columns (”id”, ”label” and ”tweet”), with an uneven label (or class) distribution, as can be seen in Figure 1:

3.2 Test Set

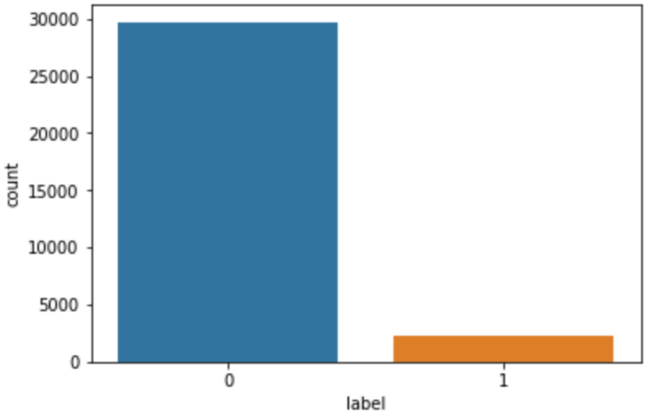
The test set is composed by 17197 observations, with only two columns (”id” and ”tweet”). Given that this challenge is an open challenge, makes sense no ”label” column, to make users submit their results.

4 Initial Feature Engineering

4.1 Pre-clean Tweet length

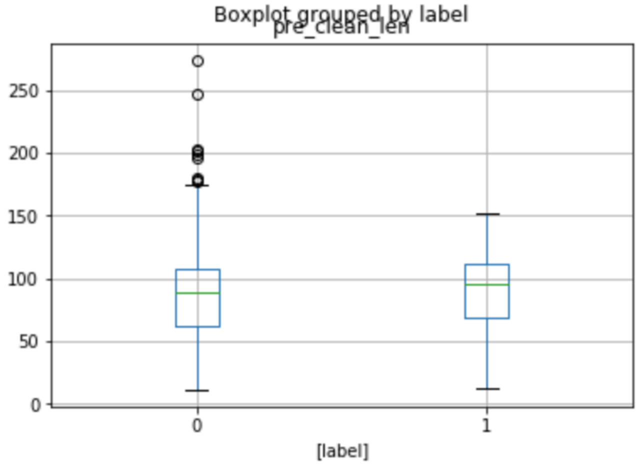
Given that in NLP tasks the most difficult challenge is the feature engineering part, and after applying algorithms out-of-box **after** cleaning and I was not able to produce good results, I tried to create my own features.

Figure 1: Class distribution



So the first hypothesis that I wanted to test was if one class of tweets has a statistically different pre-clean length that the other one. As we can see in Figure 2:

Figure 2: Pre-clean length



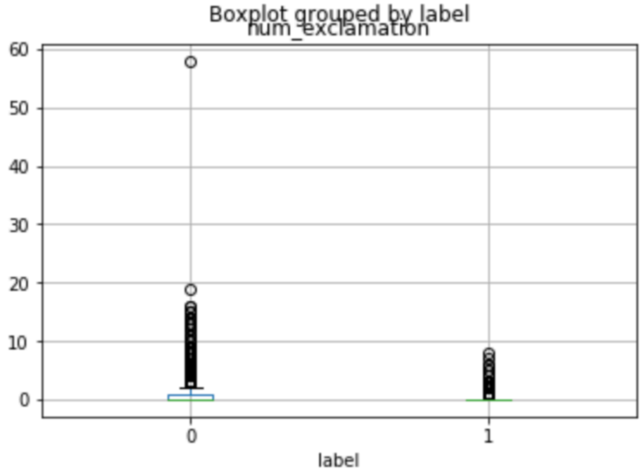
So, in terms of the median value, there is not a significant difference, but the 75^{th} percentile is somewhat deviated in class 0, comparing to class 1.

4.2 Number of exclamation marks

So following the same line of thought for the

efwefwef

Figure 3: Number of Exclamation Marks



5 Data Cleaning

6 Exploratory Data Analysis

7 Algorithms & Feature Engineering

7.1 Logistic Regression

7.2 Algorithm Comparison

7.3 Deep Learning

8 Discussion and Future Work