**Project Final**

**Report**

**for**

**Toxic Comment**

**Classification Challenge**

**Prepared by   
De Guzman, Nikko**

**Ho, Henry**

**Sarenas, Justin**

**April 30, 2018**

# **1. Project Description, Goals, Data**

## **1.1 Project Description**

In this day and age, many people have access to the internet. They can freely express their thoughts and opinions on different platforms. However, there are also people who will express their opinions in a negative way. These toxic behaviors limit people to express their opinions since they become self conscious of what they say. Many platforms do not have a way to monitor and prohibit these behaviors and as a result they limit or remove user comments.

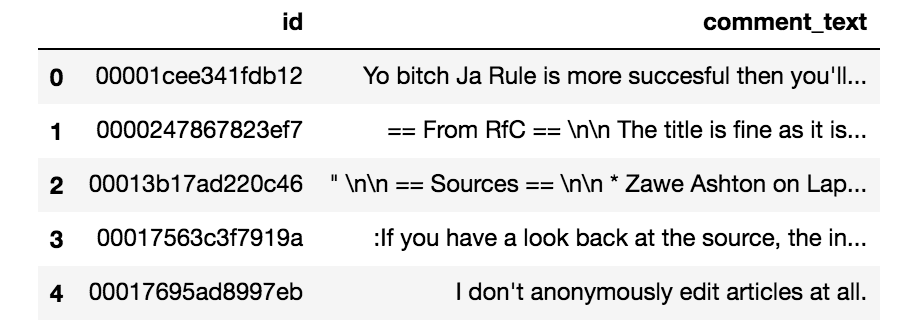
## **1.2 Goals**

The main goal of this project is to build a model to correctly identify a comment and categorize it under the correct label. The different labels are toxic, severe toxic, obscene, threat, insult and identity hate. We will be using 5 different algorithms to achieve this goal. The algorithms are Artificial Neural Network(ANN), Support Vector Machine(SVM), Random Forest, Logistic Regression and Recurrent Neural Network(RNN).

## **1.3 Data**

The data set we have is provided by Kaggle. They provided us with both a test and train file. The train file contains the id, comment text and the different categories of toxicity. The test file contains the id and comment text used to predict through the model we train. These comments were taken from the Wikipedia talk page edits.





# **2. Methods and Algorithms**

## **2.1 Count Vectorizer**

Count Vectorizer converts our comments to a matrix of token counts. By tokenizing, it lets us extract features to be used in our classifiers

## **2.2 Logistic Regression Classifier**

This algorithms uses a logistic function to measure the relationship between a categorical dependent feature and independent features by estimating a probability.

## **2.3 Random Forest Classifier**

This algorithm constructs multiple decision trees at training stage and combines all the results into a prediction. The features that the algorithm selects are all random so that the decision trees constructed would be different from each other. The advantages of using this classifier is that it resolves the problem of overfitting that the decision tree classifier has. This algorithm also runs well with big data with multiple features as well as data with missing values.

## **2.4 Long Short-Term Memory**

Long Short-Term Memory or LSTM is a type of recurrent neural network. The advantage of using LSTM is that it can remember the ordering between words in a sentence. This is very important because to understand a sentence you need the context of previous words. For this approach we used Keras with a TensorFlow backend.

## **2.5 K-Fold Cross Validation**

K-fold Cross Validation is used on supervised learning algorithms such as KNN, DT, Linear and Logistic Regression to solve the problems of overfitting of data model. The method partitions the data set into k amount of bins. A model is then trained on k-1 as training data. The model is then validated on the remaining k data. This method loops through all k bins and calculates the average. The benefits of this method is that it will not waste too much data however with big data, computations will become expensive.

We will be using 10-fold Cross Validation for our data set.

## **2.6 LinearSVC (Support Vector Machine)**

Support Vector Machine is a supervised learning model with associated learning algorithms that analyze data for classification and regression analysis. It essentially creates an optimal line which separates classes. LInearSVC is an implementation of Support Vector Machine which uses the one vs rest scheme.

## **2.7 MLP Classifier (Artificial Neural Network)**

Artificial Neural Network is a supervised learning model that approaches problems by imitating the human brain. It creates a network of neurons where a neuron takes an input and changes its state based on the input. It then proceeds to produce an output based on the input and its state. This process is done throughout the whole network of neurons where the output of one neuron becomes the input of another neuron. Multi-Layer Perception Classifier(MLP) is an implementation of Artificial Neural Network. MLP trains using backpropagation.

# **3. Results**

## **3.1 MLP Classifier (ANN)**

After implementing MLP Classifier to our training and testing set we calculated the accuracy of our data to be 0.900673 or 90%.

## **3.2 Random Forest**

After implementing Random Forest to our training and testing set, we calculated the accuracy of our data to be 0.914366 or 91%.

## **3.3 Logistic Regression**

After implementing logistic regression to our split data, we calculated the accuracy score to be:

**Without Cross Validation**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Categories | Toxic | Severe Toxic | Obscene | Threat | Insult | Identity Hate |
| Accuracy | 0.956102 | 0.989848 | 0.974463 | 0.996867 | 0.968447 | 0.990005 |

**With Cross Validation**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Categories | Toxic | Severe Toxic | Obscene | Threat | Insult | Identity Hate |
| Accuracy | 0.954354 | 0.990051 | 0.975606 | 0.996506 | 0.965807 | 0.990482 |

## 

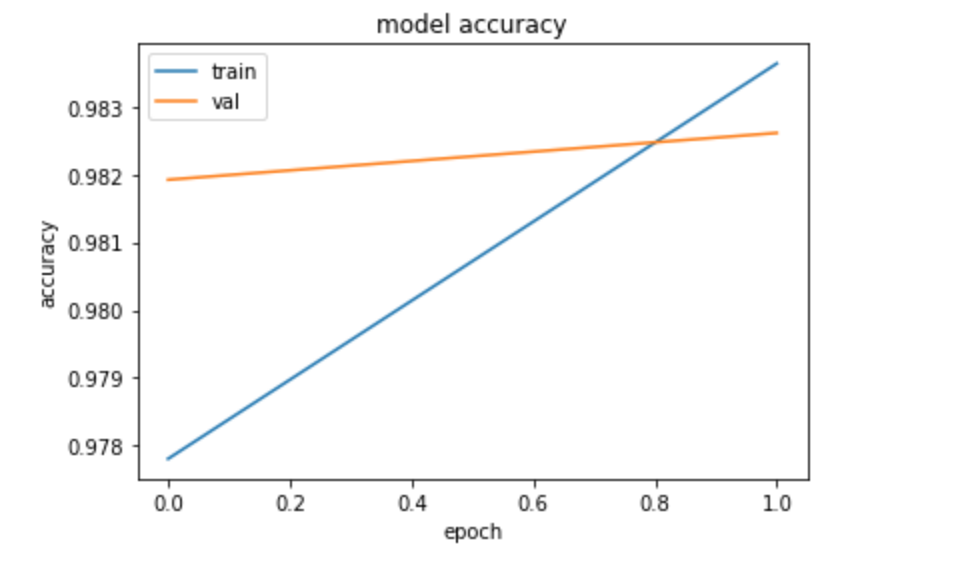
## **3.4 LinearSVC (SVM)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Categories | Toxic | Severe Toxic | Obscene | Threat | Insult | Identity Hate |
| Accuracy | 0.951997 | 0.987341 | 0.977064 | 0.996271 | 0.966097 | 0.989096 |

## 

## **3.5 Long Short-Term Memory (LSTM)**

After implementing Long Short-Term Memory to our split data, we calculated the accuracy score to be 0.9819 after 1 epoch and 0.9826 after 2 epochs which is 98.1% and 98.2% respectively.



# **4. Prediction Comparisons:**

“

|  |
| --- |
| ::You're funny. Ugly? We're dudes on computers, moron. You are quite astonishingly stupid. |

“

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Toxic | Severe Toxic | Obscene | Threat | Insult | Identity Hate |
| LogReg | 1 | 0 | 0 | 0 | 1 | 0 |
| Random Forest | 1 | 0 | 1 | 0 | 1 | 0 |
| ANN | 1 | 0 | 0 | 0 | 0 | 0 |
| SVM | 1 | 0 | 0 | 0 | 1 | 0 |

“

|  |
| --- |
| you are gay was not a valid comment because you are the gay one my child |

“

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Toxic | Severe Toxic | Obscene | Threat | Insult | Identity Hate |
| LogReg | 1 | 0 | 0 | 0 | 0 | 0 |
| Random Forest | 1 | 0 | 0 | 0 | 1 | 1 |
| ANN | 1 | 0 | 1 | 0 | 1 | 0 |
| SVM | 1 | 0 | 0 | 0 | 0 | 0 |

“(we hate america and we are going to bomb the shit out of there cities, Queza;s will rain)”

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Toxic | Severe Toxic | Obscene | Threat | Insult | Identity Hate |
| LogReg | 1 | 0 | 0 | 0 | 0 | 0 |
| Random Forest | 1 | 0 | 1 | 0 | 1 | 0 |
| ANN | 1 | 0 | 0 | 0 | 0 | 0 |
| SVM | 1 | 0 | 1 | 0 | 0 | 0 |

# **5. Conclusion**

The accuracy of our classifiers ranged in the 90%, which is a higher accuracy range than what we expected.

To improve accuracy we believe we have to do a better job at preprocessing the data. That includes creating regex cases. There is a lot of punctuations and capitalizations that should be taken into consideration when evaluating toxic comments. In the online world people get very creative in finding ways to display toxicity. Another way we can improve our accuracy is upgrading to better hardware. Due to hardware limitations, performance issues were encountered, especially when running neural networks like ANN and LSTM. There is also visualization techniques that we did not utilize that may improve our analysis. We tried using Tableau software but our machines could not handle the large data set. Better optimized algorithms that were not learned in class may better handle text analysis such as Naive Bayes.