CSC 240: Final Project Proposal

The Classification of Cyberbullying Tweets

By: Sammy Haq

Abstract

As technology becomes more prevalent in our society and our interactions with other people, cyberbullying becomes an increased problem. However, the internet is expansive – as a result, monitoring cyberbullying is a difficult task if only done by human workers. Therefore, methods to automatically identify cyberbullying via a more programatic approach, to at least "filter" possible cyberbullying should be examined. After classifying cyberbullying tweets given in the Dataset for Detection of Cyber-Trolls on Kaggle, we will then verify our classifier's "skill" via K-Fold Cross Validation with $k = 10$. Limitations of the classifier and further discussion will shortly follow.

1 Introduction

Interactions have taken to cyberspace. From major politicians such as the President of the United States to teenagers, the internet is the modern day wild west. Due to this widespread use, interactions that may have previously been in person can now be done online. This applies to workspace meetings such as the use of Slack, to social interactions like those on Facebook and Twitter.

Unfortunately, malicious interactions also take place on the internet, such as cyberbullying. Although these interactions can be policed by teachers and other authoritative figures in reality, policing these sort of interactions on the internet is almost non-existant. Furthermore, the internet is such an

expansive place that the implementation of human moderators on these large forums is extremely expensive and uneffective. To aid moderators in their policing of cyberbullying, developing a filter to help automatically detect the more extreme cases of cyberbullying should be considered.

2 Research Objectives

This paper proposes the examination of the Dataset for Detection of Cyber-Trolls found on Kaggle. A convolutional neural network will be implemented on this dataset in order to classify the data. Associative pattern mining is disregarded, as although it may be interesting what phrases are used to cyberbully, using a less simplisitic method such as a convolutional neural network will allow the creation of hidden attributes that may be not considered by humans, and therefore allow more expression in the model.

After successful implementation of the classifier, the classifier's ability to successfully classify (known as its "skill") will be examined via the k-fold cross validation method with a value of $k = 10$. Then, the behaviour of the classifier will be discussed and its viability in fitting the data.

3 General Approach

In total, there are 20001 items that the model will be trained upon. There are two possible classifications: 0 (which corresponds to non-cyber

aggressive) and 1 (which implies that the tweet is cyber-aggressive). These remarks are tweets hand-labeled by contributors on Dataturks.

A feed-forward neural network is an artificial neural network where the connections are non-recursive and do not allow feedback. A convolutional neural network (CNN) is a class of feed-forward artifical neural networks that are based upon a specialized version of a multi-layer perceptron network. The multi-layer perceptrons in convolutional neural networks require minimal pre-processing, or manipulation of the raw data to be trained upon. This lack of human interaction, as mentioned earlier, is a huge advantage as it does not require the creation of handmade attributes to classify upon. This allows the classifier to create its own attributes, free from human-bias.

Although convolutional neural networks are largely used in image processing, research done by Collobert and Weston (2008) show promising results regarding applying convolutional neural networks to natural language processing.

The implmentation of the convolutional neural network classifier found in the scikit-learn library will be used. The minimal data preprocessing required for this task will be done with assistance from the numpy and pandas libraries.

The scikit-learn library will be generally used for most of the testing, as well – namely, its train_test_split class and K-Fold cross validator will be used to train and test the classifier, respectively.