CPTS 315
Course Project

Reddit Ban Predictor

Introduction:

Reddit is a platform that is made up of many different communities. Each community, or sub, designed to cater to their own needs, but is also required to follow Reddit's own terms of service. One or more moderators are put in charge of each sub; tasked with the expectation that they will hold users accountable for breaking either their own rules, or the rules of the platform. These rules, however, are often subjective, and a variety of factors can influence if a moderator

On the platform, there exist several quarantined subs; a sub-reddit that is hidden from the general public and only accessible by those who are already subscribed to it.

> "The purpose of quarantining a community is to prevent its content from being accidentally viewed by those who do not knowingly wish to do so, or viewed without appropriate context. Quarantined subreddits and their subscribers are still fully obliged to abide by Reddit's Content Policy and remain subject to enforcement measures" - https://www.reddithelp.com/en/categories/rules-reporting/account-and-community-restrictions/quarantined-subreddits

Quarantine is the last step before a sub is banned. Certainly, quarantine is appropriate in many cases, but Reddit is often accused of being hypocritical of what they ban. Those who claim this say they are more likely to target those who are of a certain political leaning. I hope to be able to use my results to find evidence for or against this.

I am working from a set of data from kaggle that includes posts from several different sub reddit's. Ideally, I would be able to have a classifier for each individual sub reddit to be more accurate. I had thought to have a variable for the sub reddit in the classifier, but there are well over a million sub reddit's and I would never feasibly be able to gather test data from all of them, so in the end I have decided to go for a more generalized approach. I will do my testing with https://www.removeddit.com/ so that I can view deleted threads. In this way, the final classifier was able to correctly predict if a thread was deleted or not based on the title 65.7% of the time. But on an input of pure deleted threads, it only correctly predicted 10% of the time

Data Mining Task

Input Data:

For this project. The input data consists of just over 17k individual reddit posts that includes the following information for each post in a csv file:

Id -Unique identifier

Title -Title

Score - Score

Author - Post's author

author_flair_text -Author's falir

removed_by - Who removed

total_awards_received - Total of awards

Awarders - Awarders received

created_utc - Created at (UTC)

full_link - Link of post

num_comments - Number of comments

over_18 - True if NSFW

For the purposes of this project, we will be focusing on the “removed_by” and “Title” columns with a single perceptron algorithm. Originally, I planned on using more of the identifiers, but certain subreddits, especially those that are quarantined, do not allow for flairs or awards. The other categories may even go against the purpose of this classifier. To have the most objective sense of if a post should be banned. Categories such as num_comments, Author, and especially total_awards_received could skew the results.

Questions:

Before I started, I already had many questions I wanted to answer. First and foremost, of course, is whether a binary classifier can be made that accurately predicts if a reddit post will be removed. Beyond that, what defines “accurately”? How can I use such a classifier to determine if there is a bias in reddit moderators between subs? How will I determine a control group to test against the more ‘unseemly’ subs? Will I be adding my own bias by hand picking the subs I am going to test against? How am I going to

get the data needed for these tests? Will a data set with so few examples of removed posts create a good classifier?

Challenges:

The implementation of the perceptron algorithm was my main challenge. I had gotten quite far in my own implementation before deciding to use Sclearn. I was hesitant to use it at first because I felt that it would take me more time to learn this module than it would to write it myself from scratch, but after several failed and inefficient attempts, I changed my mind.

The problem with the reddit dataset was that it was not in the correct format to be accepted by Sclearn's built in perceptron algorithm, so even though I didn't need to build the algorithm itself, I had a lot of pre-processing before I could start. Also, doing all of my coding through kaggle was a challenge and a mistake. As I tried to increase the size of my input in the perceptron function, I would run out of space or an unspecified error would halt my progress.

My biggest problem was that I could not find a dataset for the specific sub Reddits that I planned on testing on. On the website removeddit.com, where I had planned on gathering posts that had been removed for subs that are under quarantine, I found that there were no instances of removed threads to be found. This unfortunately forced me to change my initial plan of finding a bias in the moderation of reddit into simply predicting if a post will result in a ban.

Technical Approach

To evaluate the data, I used a perceptron algorithm to create a binary classifier to determine if a reddit post will result in a ban based only on the title. The classifier is based on an array of strings taken from the “title” section of the data, vocabulary. Each title was split into its component strings and appended to this array only if they were composed of ascii values, and they were not articles – stop words that were taken from another data set.

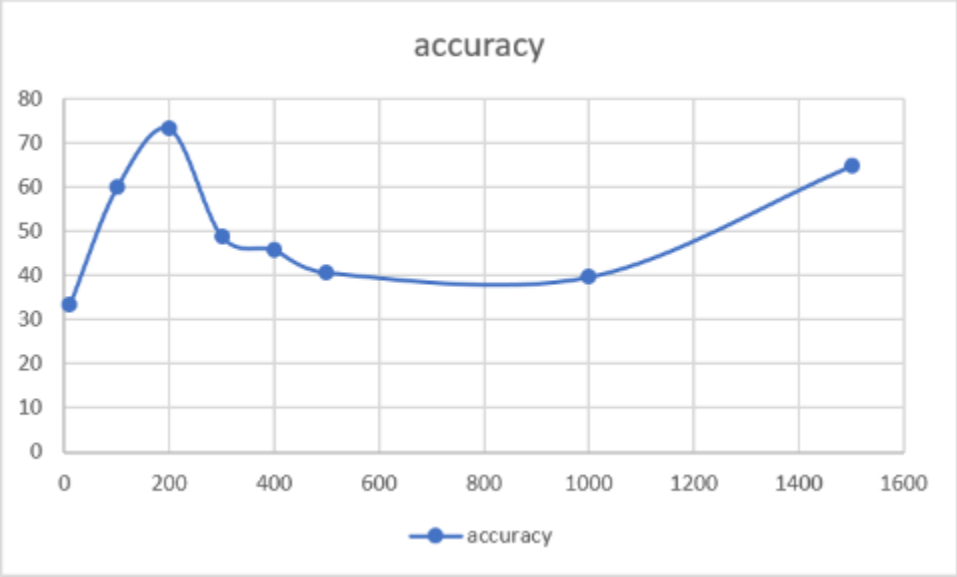
Using the data on “removed_by”, I converted all values to a Bool where it is True if the post was removed by a moderator, reddit, or the user, and the value set to False if it was not removed.

Using the original data for “title” along with this python list, I created another list that would be used for the training by converting each element of title to its corresponding position in vocabulary. After this I begin testing with different input sizes in a perceptron

Evaluation Methodology

My planned evaluation was unsuccessful. As I already mentioned, I was unable to gather any data needed to evaluate a bias based on a difference between sub-reddits as I could not find a single post that has been deleted on removeddit.com. I attribute this to the fact that posts being removed is extremely rare.

Because of this I simply split the training data into two sets to test the classifier. This way I was able to use different data for training and testing even though it came from the same set. My initial idea was to use the entire data set, but after running for well over 24 hours, I decided to abandon that plan and work with something more manageable.



The graph above shows the accuracy of my classifier using different data sizes. The accuracy appears to go up around 200, I believe due to over-fitting. I settled on n=1500 for the final classifier due to the limitations of my chosen environment. Using removeddit, I ran the top 10 posts from r/all that had been deleted, and only one was successfully classified with this classifier.

Lessons Learned

My goals for this project went largely unfulfilled, but it wasn't completely unsuccessful. I unable to test what had hoped to, the bias in Reddit's moderation, but my classifier performed much better than I had expected it to at 64.7%. However, it is obvious that the false negative rate is much higher than the false positive which is the only reason it appears to have such a high success rate. Only being able to classify 1/10 posts that end up deleted is not helpful in a real-world situation. I would need to improve the classifier, first by increasing the number of training data, but also applying other ml techniques. I stated earlier that including other elements like awards and author would be negative to the algorithm, building a multi-layer perceptron that includes elements such as these may turn out to be beneficial.

Acknowledgements

English stopwords

http://xpo6.com/list-of-english-stop-words/

Scikit learn

https://scikit-learn.org/stable/

Reddit quarantine

https://www.reddithelp.com/en/categories/rules-reporting/account-and-community-restrictions/quarantined-subreddits

Deleted reddit posts

https://www.removeddit.com/

Kaggle data-set

https://www.kaggle.com/unanimad/dataisbeautiful