ISyE 6740 – Spring 2021

Final Report

Team Member Names: Katie Barthelson (903546171), Elizabeth Yates (903651544), Daniel Tylutki (903547563)

Project Title: Classification for Reddit Bot Detection

Problem Statement

While the internet has brought about many significant benefits to society by giving individuals the power of influence on a global audience through easy information sharing, it has also been used as platform for malicious activity such as disinformation campaigns or harassment by bad actors, including national governments. A common tool employed by bad actors on the internet are social media bots, which have become a common feature of online conversation. For the purposes of this project, bots are defined as completely or partially automated social media accounts. While some bots are easy to detect or declare themselves outright, bots that covertly mimic human behavior or seek to pose as real individuals are more difficult to identify, especially for users who are just reading social media posts for news or entertainment, and not actively on the lookout for bots or even aware of their presence. Some bots are used for legitimate or helpful purposes such as answering questions or assisting online shoppers, but automated accounts are also frequently used for more nefarious purposes like spreading misinformation or harassment. For the purposes of this paper, the focus is on so-called “bad” bots. A 2018 study by Efthimion et. al at Southern Methodist University characterizes bad bots based on the transparency with which they disclose their identity. Thus, “bad” bots are accounts which work to influence others while maintaining their illicit legitimacy as a real human.

Bots have the capability to spread misinformation, affect political outcomes, and cause panic. Identifying social media accounts that are utilized as automated accounts for bad actors and removing or otherwise publicly identifying such accounts may improve the online landscape and help to reduce the spread of false information that can do real damage to the public like convincing people that they will be harmed by a vaccine that in reality could actually save their life.

The goal is to build a model that can identify bots with significant accuracy so that the public can be made aware of these bots as they operate. By applying the model to active social media activity, there is potential to warn social media users of illegitimate accounts and disinformation as it is happening and prevent its spread. To achieve this goal, the analysis will attempt to answer several important questions: What are some features that can be utilized to identify fake accounts? How can we work to explore and ideally improve current practices? How difficult is it to distinguish between a bot and a real human user?

Data Source

In order to find labeled data, several open-source data repositories and articles were explored. The data chosen was a set of pre-identified Reddit bots curated by Jason Skowronski and provided through an article he wrote on the blogging site Medium. The labeled data set he created contained 267,036 comments from “393 known bots plus 167 more from the BotWatch subreddit.” Not all of the bots were “bad bots”. Many of them are openly declared bot accounts which are used to post helpful information or assist in moderating a subreddit, which is a specific forum on Reddit about a particular topic. These types of bots are registered with and authorized by Reddit to operate. However, there were also bots in the data set that were created for malicious purposes. Either way, the data satisfied the requirement for a multitude of bot-made comments which would be our target class. The original data set did not have comments by normal users to train a model to differentiate between the two classes, bots and humans (also called authentic or normal users throughout this report). In order to obtain data for normal users, this needed a data collection methodology that was unlikely to scoop up comments made by bots. We decided to compile a list of twenty different subreddits which were unlikely to contain automated activity. Since malicious Reddit bots are usually created to spread misinformation, troll, or harass users in popular subreddits focused on politics and social issues, data collection was focused on collecting data from subreddits that were (1) not too popular, (2) not related to a controversial topics like politics, business, and social issues that would be prime targets for influence campaigns, and (3) not meme-focused or primarily filled with "karma-seeking" content where users are trying to get as many upvotes and awards from other users as possible. Therefore, the list contained mild subreddits such as r/books, r/vegetablegardening, r/gaming, and r/painting where there was likely to be mostly authentic conversation between real users. Once the list was created, random comments were collected from the data using the Reddit API until we had comments from about 1,000 unique users who we reasonably believed to be authentic accounts. This gave us about two times as many human users compared to bot users. The final step in the data collection was then to use our list of normal user accounts and retrieve up to the last 1,000 comments posted by each of these users. This yielded hundreds of thousands of comments by human users. We then merged this data with our pre-existing data of labeled bots. After preprocessing, which included removing duplicates and records containing missing values, we had a clean data set comprised of 572,642 comments by normal users (approximately 75% of the data) and 189,249 comments by bots.

Methodology

The methodology for building a bot detection model included preprocessing, feature enhancement, data exploration, feature selection, model training, and evaluation. A number of models were trained and evaluated including: Decision Tree, Naive Bayes, K-Nearest Neighbors, Logistic Regression, SVM, Kernel SVM, Neural Network. While it was not expected that linear classifiers would perform well, they were included as a comparison to other methods.

Preprocessing

Multiple preprocessing methods were used on the data to prepare it for analysis. Duplicate records were dropped along with all features that contained all null values. These fields were "banned_by" and "num_reports" which were likely null because the information is only to be used by Reddit employees and therefore not delivered by the API. Records were removed if they contained any null values, but there were only 1,005 records like this which is a relatively small amount compared to the overall size of the data. This included values that were only whitespace. Type conversion was performed to ensure that fields contain only strings and that all float fields are converted to integers. Extra fields such as author were also removed as they have little analytical value or could be uniquely identifying. Emojis were replaced with text representations so they would be easier to conduct text analysis on (e.g., the rocket-ship emoji becomes ":rocketship:"). Uncommon characters or escape characters were replaced with accepted equivalents to prevent difficulties when text processing. Additionally, posts were checked if the author's account was deleted or if comment text has been removed.

Feature Enhancement

Features were enhanced by creating calculated features related to the behavior of each user in the data set, such as the calculating the average character length of each user’s posts and the similarity score of a user’s posts using Levenshtein Distance Ratio. This will expand the number of numerical features that our model can use to predict the class of each social media account.

- author_%_is_submitter (float): the proportion of the author's comments that are posted on their own submission
- author_avg_num_comments (float): the average number of comments that a user receives on their comments
- author_%_no_follow (float): the proportion of the author's comments where nofollow attribute is set to True
- author_%_gilded (float): the proportion of the author's comments that are gilded
- author_avg_score (float): the average score of the author's comments
- author_%_over_18 (float): the proportion of comments by the author that are marked NSFW
- author_avg_ups (float): the average number of up-votes that the user gets per comment
- author_avg_downs (float): the average number of down-votes that the user gets per comment
- author_%_controversiality (float): the proportion of the author's comments that are controversial
- author_%_quarantine (float): the proportion of the author's comments that have been quarantined
- sentiment (int): the predicted sentiment of the comment. 1 == Positive, 0 == Neutral, and -1 == Negative.
- author_avg_sentiment (float): the average sentiment of the user's recent comments
- author_avg_comment_similarity (float): the average similarity score of the user's recent comments

Data Exploration

Exploratory data analysis was performed to gain an understanding of the patterns and distributions of each feature. Understanding each feature helped to arrive at a model hypothesis and helped identify features that are indicative of bot behavior. Overall, the dataset has 189,249 datapoints from bots and 572,642 regular users. 87% of bot-labeled comments came from users with no followers compared to 64% for non-bot labeled comments. Bot-labeled postings also commonly came from authors with an average of 0 comments on their postings. For comment similarity, comments from bots had a high similarity score close to 1, which implies repetitive postings.

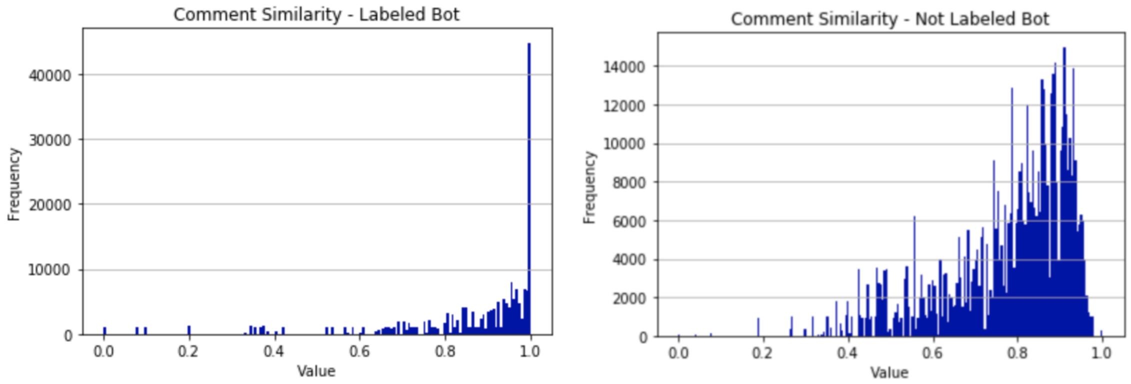

Figure 1: Comment Similarity

Regular users tended to have a higher average number of comments received on their comments, indicating that users tend to engage less in comments made by bots.

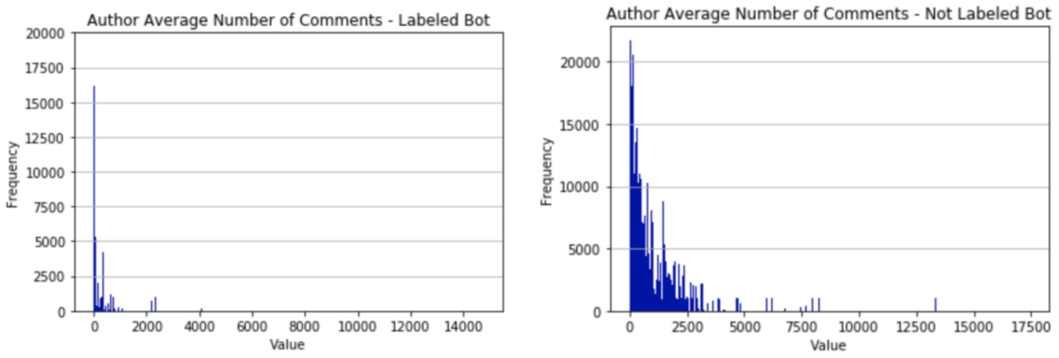

Figure 2: Average Number of Comments

For vulgarity, bots had a slightly smaller percentage of postings labeled as NSFW (2%) compared to regular user postings (3%). Both bots and regular user postings had roughly 2% of postings labeled controversial. Given how close and small these percentages are, it was hypothesized that “over_18” and “controversiality” are not significant features. However, the proportion of the author's comments that are controversial (“author_%_controversiality”) is concentrated around 0 for bots, so it could be hypothesized that this is a useful feature. This could indicate that either most posting from bots are not controversial or that most bot accounts have too few postings in general to achieve a higher percentage.

Feature Selection

Feature selection was conducted to identify the most important features in predicting the classification of each social media account. Lasso and Elastic Net regression was performed, resulting in an optimal alpha of 0.2. When an alpha level of 0 is chosen, the model’s best performance is attained by including all features. As the penalization, alpha, increases, $\sum|\beta i|$ is pulled towards zero, with the less important parameters being pulled to zero earlier. With an alpha of 0.2, initial assumptions can be made to suggest that our optimal model will likely include most, but not all, of our features. The following graphs represent the tested alphas in comparison to the mean squared errors (MSE), as well as the LASSO Path representing the rate at which each coefficient entered the model. The MSE is calculated by taking the difference between the model’s prediction and the ground truth, which is then squared and averaged out over the entire dataset. The following plots provide a visual representation of our optimal alpha as a result of the lowest MSE's using both Lasso and Elastic Net. Thus, the optimal alpha used to penalize and select important features is correlated with the optimal MSE scores.

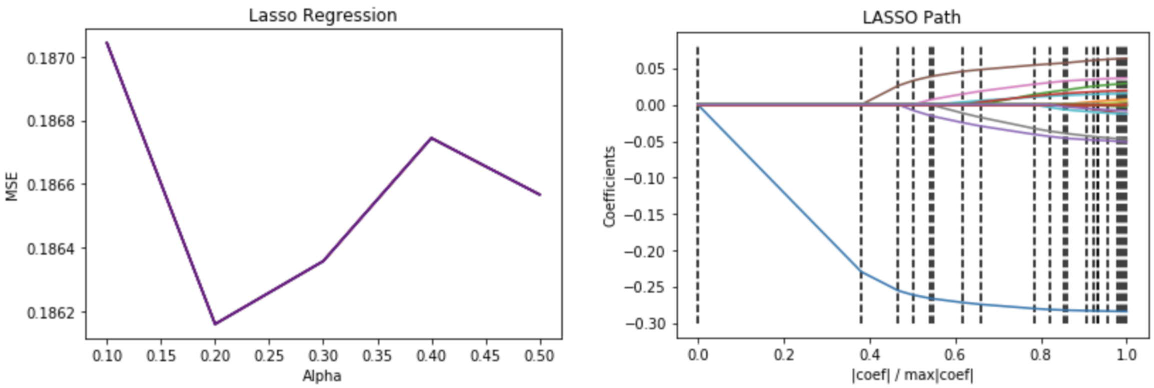

Figure 3: Lasso Regression for Feature Selection

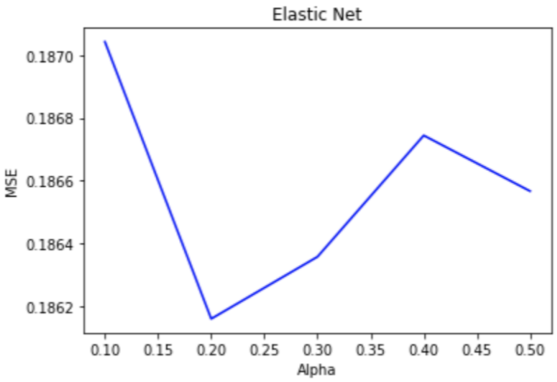

Figure 4: Elastic Net

Several models and techniques were used to conduct feature selection including but not limited to correlation matrices, chi-squared, and ensemble methods. These approaches were analyzed and compared in order to adequately decide upon the most important features with which to train our models.

According to correlation analysis, which is most commonly done using the Pearson correlation, the following 3 independent variables are found to be the most highly correlated (above 0.5) with the output variable: author_verfied, author_%_no_follow, and created_utc.

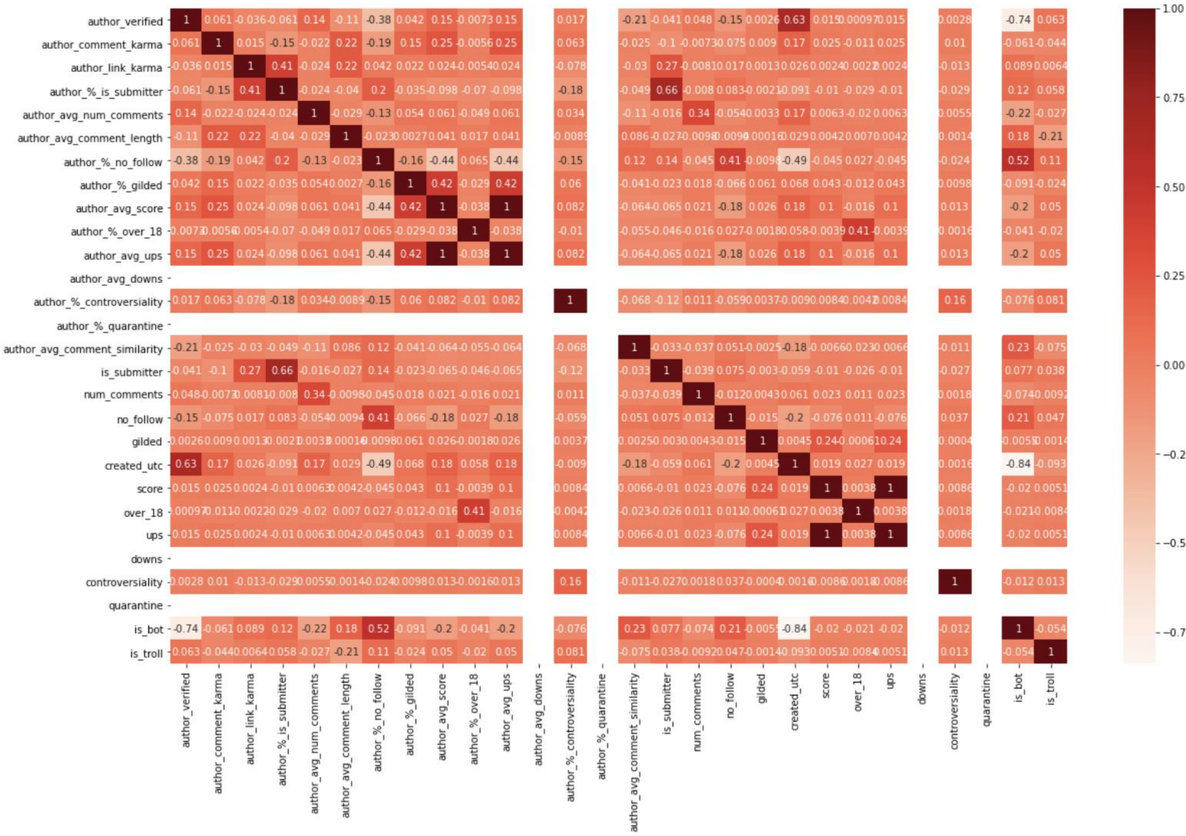

Figure 5: Feature Correlation Analysis

However, additional analysis using Lasso revealed the following 15 important variables and eliminated the other 11 variables. At the risk of under/over-fitting and after receiving similar results from chi-squared and ensemble methods, our team chose to include the features selected via the Lasso method. The final features selected are shown below.

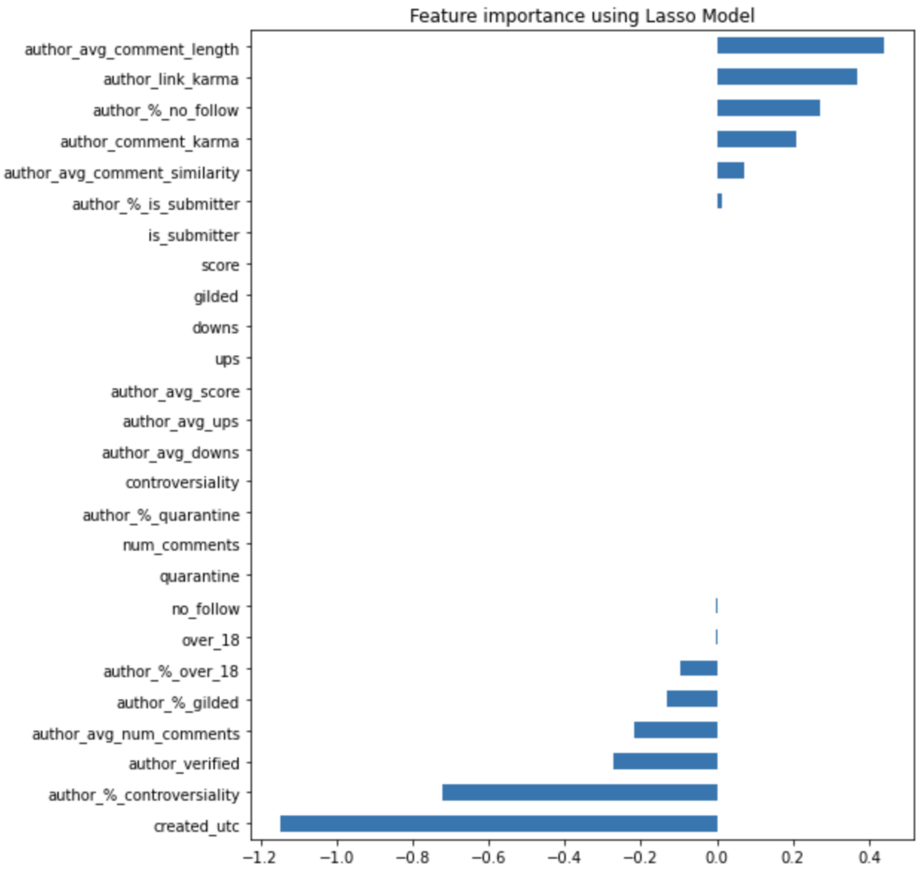

Figure 6: Feature Importance Using Lasso

The 15 chosen features were included in our final models which were tested and evaluated in the following section.

Evaluation and Final Results

To evaluate the effectiveness of the models, each model will be trained and tested on the same labeled data set containing both bot and human posts. A confusion matrix was created to calculate and report the overall accuracy of each model along with the misclassification rate.

Overall, linear classifiers performed the worst. Each classifier's accuracy is reported in the tables below using all features as well as using only the most important features. The false negative rate is an important evaluation criterion because in a real-world application of the model, it is important to prevent "accusing" legitimate human users of being bots.

Using all features, some of the models may be overfit.

Model	Decision Tree	Naive Bayes	K-Nearest Neighbors	Logistic Regression	SVM	Kernel SVM	Neural Network
Accuracy	94.22%	49.51%	99.73%	92.03%	25.11%	75.06%	97.01%
False Negatives	2842	1129	206	4983	208	37996	2676

Figure 7: Model Performance using all Features

Using the selected features, some of the models have lower accuracy and more false negatives. However, these models would likely capture more variance in reddit postings.

Model	Decision Tree	Naive Bayes	K-Nearest Neighbors	Logistic Regression	SVM	Kernel SVM	Neural Network
Accuracy	93.79%	88.61%	99.99%	91.46%	25.82%	25.61%	95%
False Negatives	4056	3382	9	4548	407	44	3632

Figure 8: Model Performance using Selected Features

The model selected from this evaluation was a neural network. However, the decision tree, and logistic regression models could also be used. Naïve Bayes, SVM, and Kernel SVM are not recommended due to low accuracy or significant differences from the all-feature model. Multiple KNN models were attempted but they tended to have very high accuracy. This implies that the KNN model is overfit.

In summary, the features that were indicative of bot behavior were the proportion of the author's comments that are posted on their own submission (author_%_is_submitter), the average number of comments that a user receives on their comments (author_avg_num_comments), the proportion of the author's comments where nofollow attribute is set to True (author_%_no_follow), the proportion of the author's comments that are gilded (author_%_gilded), the proportion of comments by the author that are marked NSFW (author_%_over_18), the proportion of the author's comments that are controversial (author_%_controversiality), and the average similarity score of the user's recent comments (author_avg_comment_similarity). This implies that bots can be identified by repetitive postings, a low number of followers, average comment length, the average author karma (upvotes) on links and comments, and the amount of engagement they receive on their postings. It is likely that real users can identify bots, which can help to mitigate false accounts.

Project Work Breakdown

Each team member is contributed approximately equally to the project. The team has been met biweekly to discuss project ideas, direction, and assignment of work. For this proposal, each team member contributed the initial version of a section and all team members have reviewed and provided improvements to the entirety of the paper. For coding-related tasks, the project was split up among the members. One person handled data collection and preprocessing, another handled data exploration, and another worked on feature selection and model training.

References

Bot or Not: An end-to-end data analysis in Python. (n.d.). Retrieved May 2, 2021, from http://www.erinshellman.com/bot-or-not/

Skowronski, J. (2019, July 30). Identifying trolls and bots on Reddit with machine learning (part 2). Retrieved March 22, 2021, from https://towardsdatascience.com/identifying-trolls-and-bots-on-reddit-with-machine-learning-709da5970af1
HYPERLINK "https://towardsdatascience.com/identifying-trolls-and-bots-on-reddit-with-machine-learning-709da5970af1"

Scikit-learn: Machine learning in Python—Scikit-learn 0.24.2 documentation. (n.d.). Retrieved May 2, 2021, from https://scikit-learn.org/stable/

Shetye, A. (2019, February 12). *Feature Selection with sklearn and Pandas*. Medium. https://towardsdatascience.com/feature-selection-with-pandas-e3690ad8504b

Jeff Reback, Wes McKinney, jbrockmendel, Joris Van den Bossche, Tom Augspurger, Phillip Cloud, … Mortada Mehyar. (2020, March 18). *pandas-dev/pandas: Pandas 1.0.3 (Version v1.0.3)*. Zenodo. http://doi.org/10.5281/zenodo.3715232

Harris, C.R., Millman, K.J., van der Walt, S.J. et al. *Array programming with NumPy*. Nature 585, 357–362 (2020). DOI: 0.1038/s41586-020-2649-2. (Publisher link).

Taehoon Kim and Kevin Wurster. *emoji (Version 1.2.0)*. https://pypi.org/project/emoji/.