Reddit Bot Classifier

Brian Norlander

November 2018

Contents

List of Figures

List of Tables

Abstract

This research investigates the problem of bots in online forums, more specifically, Russian bots on Reddit. To do this I used a list of accounts verified to be Russian bots that Reddit published in April 2018 to perform supervised classification and other data analysis. I was able to create an effective classifier that identified accounts as normal users or bots with very high accuracy, recall and precision. Although my study focused on Russian bots on Reddit, I believe this method can be used in a more general way across the internet. The detection of bots and malicious users in online forums is an important issue and that is only increasing in its commonality and sophistication.

1 Introduction

In recent years the use of social media has greatly increased with platforms such as Facebook, Twitter and Reddit. This has created a cheap, far-reaching way of spreading fake news, bias and propaganda to many people online. These online platforms produce a large amount of textual data that can be scraped, stored and processed for analysis. In my research I specifically looked at Reddit to see if I could use their readily available textual and user data to classify an account as a normal user or a bot.

In April 2018, Reddit CEO Steve Huffman released Reddit's 2017 transparency report identifying 944 accounts as Russian bots (report found here). These accounts have been flagged by Reddit for suspicion that they were of Russian Internet Research Agency origin. Most of the accounts were banned prior to the 2016 US election, however in the spirit of transparency Reddit has decided to keep the information for these accounts public. In my research I used these accounts as the ground truth for what a bot is. Using their comment and post history combined with their account data I attempted to create a user classifier.

1.1 Motivation

Many internet forums fear that users among them are attempting to influence their discussions in a purposeful way. This can be paid employees of a company promoting a product or disparaging a competitor, employees of a political campaign promoting a candidate or smearing an opponent, or a member of a foreign government spreading propaganda at home or abroad. It is a fear among many that the internet is being weaponized into a powerful tool that can manipulate the masses.

Although often not illegal, the accounts that spread false information or bias employ a variety of tactics. Some of these tactics include concern trolling - heavy caution at a new, promising lead, misdirection - exaggerated claims without evidence, and painting their opponents as lunatics or bigots. In most online forums there is little to no real-world consequences to spreading false information. This often makes it difficult for normal users to know if the post or comment they are reading is coming from a legitimate user or if it is coming from a troll who has an agenda. It is not feasible for a user to inspect each the legitimacy of each comment by searching through the posting account's history. This problem causes many people to read, believe and be influenced by false information online.

An important definition that is needed to be made is the term **bot**. When I use the word bot, I am not necessarily referring to an automated account that generates responses in some script but instead a human user that is manually crafting individual comments and posts.

1.2 Social Media Platforms - Reddit

Trolling and bots are a widespread problem across many social media platforms on the internet. Some platforms, such as Facebook, even allow users to pay to create targeted ads intended to influence other users. The first reason why I chose Reddit for my study is that the list of Russian accounts is readily available and has been confirmed by Reddit. This allowed me to be confident in the legitimacy of my ground truth data. Second, Reddit has a very structured way that users interact with each other and create content. All the content is user generated and partitioned into different categories (subreddits) and stays on the internet forever unless deleted by Reddit or the original poster. This data is easier and more straight forward to mine than other platforms.

Reddit, often called "The Front Page of the Internet", is a massive collection of smaller forums, known as **subreddits**, which have content for one specific topic such as politics, baseball or python. Within each subreddit users can create posts which can then be commented on. A crucial aspect of Reddit is its upvote and downvote system. The visibility of a post in a subreddit and a comment depends on the number of upvotes minus its number of downvotes it has. Each user can upvote or downvote each post or comment only once. In theory this system will filter out unrelated content to a subreddit as well as low quality posts and comments. This system leads many to believe that Reddit is freer from outside influence than other social media platforms because the community can conduct self-policing by down voting bad content, unlike Facebook or Twitter.

On Reddit the term **karma** refers to the accumulated amount of points a user has for each comment and post that they have made. Each user has **comment karma** and **link karma**. Comment karma is the total sum of points for all their comments and link karma is the total sum of points for all of their posts. The term **cake day** refers to the date that the account was created, the accounts birthday.

1.3 Goal

The goal of my research is to classify accounts as either normal users or bots on Reddit. I took several approaches such as analyzing the accounts comments, posts, which subreddits an account posted and commented on and the accounts meta data. Analyzing a single comment to determine whether is it from a normal user or a bot is difficult. The English language is so large and complex that a single comment often will not provide much insight to whether the comment was a bot or not. Because of this my approach also incorporated subreddit analysis to learn the patterns of the posters.

One of the difficulties in classifying users as bots or normal accounts is that their tactics and rhetoric quickly change. For example, a set of bots operating in the 2016 presidential campaign would likely not have the same tactics when operating in a later election cycle. News cycles, topics of discussion and tactics change very quickly change very quickly. Because of this I made sure to compare the bot data with normal user account data from the same time frame.

My aim is to effectively classify accounts as a bot or normal user using the account's posts, comments, and other data. If I can create a classifier for the 944 accounts from 2015 to 2018 then it should be possible to be able to reproduce a classifier for data from a different time frame on a different platform. The end goal of this would be a real-time detector of bots like how spam filtering is done with emails. Ideally, this detection mechanism could be used in different platforms other than Reddit as well and also be adaptable to different time periods.

2 Theory

In this section I will explain what type of machine learning I used in my classification. Many of the important terms I use later in my paper will also be defined in this section. Then I will then describe how I transformed the raw data into feature vectors that I could feed into the machine learning algorithms.

2.1 Machine Learning

The two primary categories of machine learning are **supervised learning** and **unsupervised learning**. Below I will explain what each is and why I chose supervised learning over unsupervised.

In supervised learning we learn a function that maps data to labels through a set of correctly labeled data known as the **ground truth**. This requires a human to act a guide to train the algorithm with data whose output is already known so that when it sees new data it can predict its label. For example, in my research the data would be each individual post and its label would be either normal or bot. In this example we learn a function that determines whether a specific post came from a normal user or a bot. Below is an example of a supervised learning algorithm that learned a function to separate two categories of data. Now if that algorithm sees a new data point it will classify it based on which side of the line it is on. Of course, this example has only two dimensions whereas analyzing posts and comments has as many dimensions as there are unique words in a list of comments or posts.

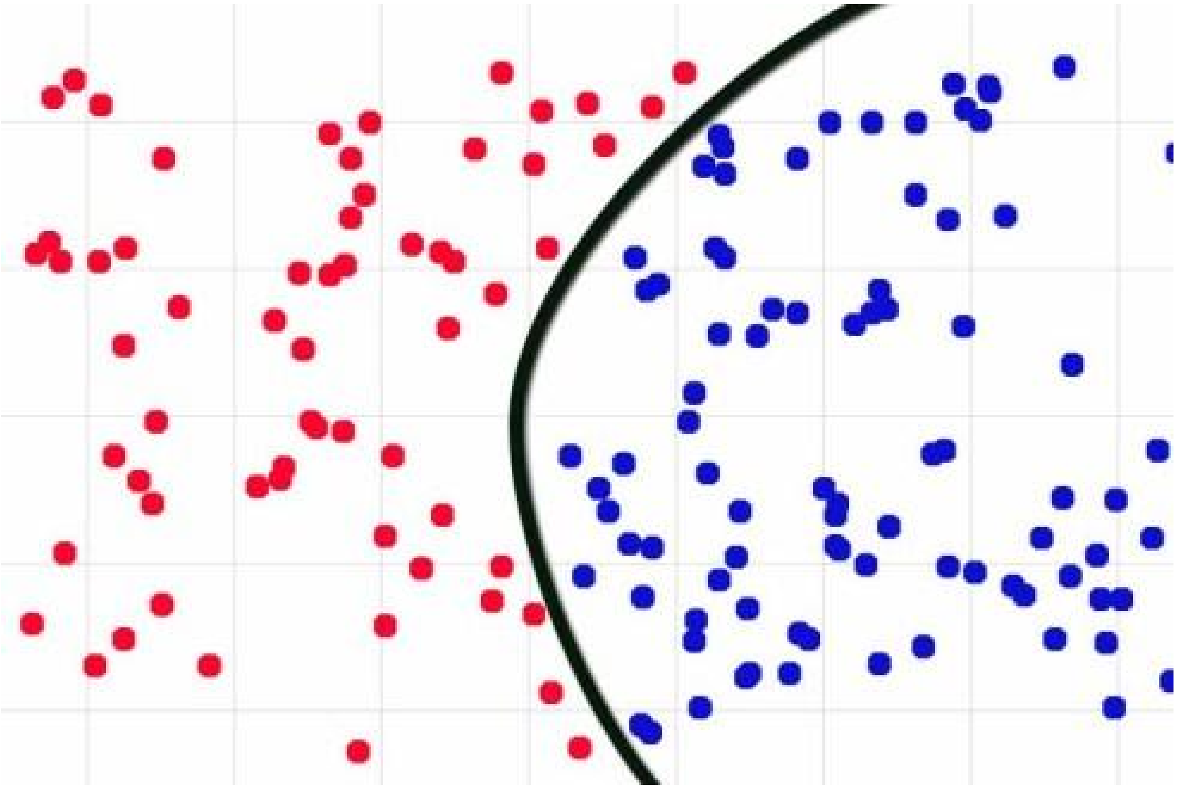


Figure 1: Supervised Learning Example

The goal of unsupervised learning is to derive structure within the data without any human guidance. This category of algorithms relies on a collection of unlabeled data. Clustering and association analysis algorithms are common for unsupervised learning.

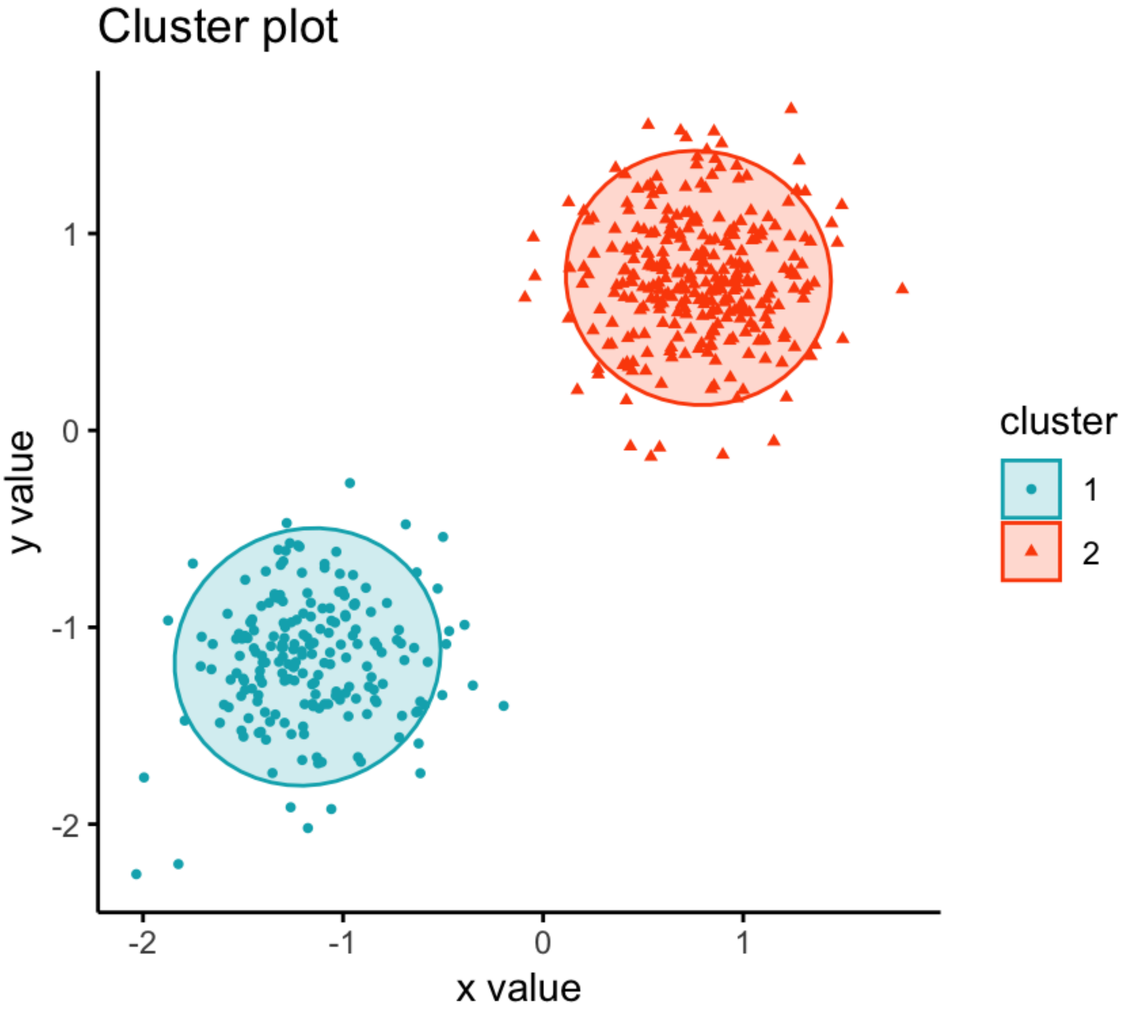


Figure 2: Unsupervised Learning Example

For my research I chose to use supervised learning. Unsupervised learning could potentially be valuable in identifying clusters of users without any ground truth, but with the 73M submissions, and 725M comments [1] and the amount of diversity between users in different subreddits this task would be difficult and extremely computationally expensive. I will use the 944 Russian bot accounts to train my classification function. Without a list such as this conducting supervised learning would not work. The alternative to a published list would be trying to identify users by hand which would be prone to a large amount of error and bias.

2.2 Text Transformation

Converting the raw text into feature vectors in which the classification algorithms can be run requires a few steps. The same exact process was done to comments and posts so without loss of generality I will just explain the process done to comments. The text of each comment will be referred to as a **document** and each comment will also have a corresponding **label**, i.e. **normal** or

bot.

First we convert each comment into a **bag of words**. To do this we tokenize the raw text of a comment by splitting the string by it's whitespace, i.e. "These are my first comments!" is turned into the array ["These", "are", "my", "first", "comments", "!"]. Each token in this array is then converted to all lowercase and punctuation is removed which creates the following array: ["these", "are", "my", "first", "comments"]. Next we remove **stop words**, which are very common words that only provide structure to the sentences grammar such as "a", "the", "is", etc. Finally, we have the bag of words: ["first", "comments"]. Then in the last step we stem each word based on the **Porter Stemmer** algorithm. For example, "running" becomes "run", "matches" becomes "match", etc. So our final bag of words is ["first", "comment"].

Reducing each comment to a bag of words has several advantages. It reduces to size of the corpus greatly, removes irrelevant terms, and normalizes words that have the same or similar meaning.

Once we have converted raw text into a bag of words, we can represent each comment as a vector. Below you can observe how three different comments are transformed from raw text into a vector.

Doc1 = "John loves to watch movies."
Doc2 = "Mary likes to watch movies with John."
Doc3 = "John loves to eat pizza."

Table 1: Raw comments

	John	loves	to	watch	movies	with	Mary	likes	eat	pizza
Doc1	1	1	1	1	1	0	0	0	0	0
Doc2	1	0	1	1	1	1	1	0	0	0
Doc3	1	1	1	0	0	0	0	0	1	1

Table 2: Feature vectors

Doc1 = [1, 1, 1, 1, 1, 0, 0, 0, 0, 0]
Doc2 = [1, 0, 1, 1, 1, 1, 1, 0, 0, 0]
Doc3 = [1, 1, 1, 0, 0, 0, 0, 0, 1, 1]

Table 3: Vectors

After the tokenization steps are done, each document can be represented as a vector. However, there is one more important step to do before we can run the classification algorithms. This step is converting the vectors into a term frequency–inverse document frequency model or tfidf model. This is an approach is an attempt to reflect how important a word is to a document in a corpus.

$$tf(t,d) = f_{t,d} \tag{1}$$

Where *tf* is the term frequency of term t in document *d*.

$$\mathit{idf}(t,D) = log\frac{N}{|\{d \in D : t \in d\}|} \tag{2}$$

Where *idf* is the inverse document frequency of term t in all documents D and N is the number of documents |D|.

$$\mathrm{tfidf}(t,d,D) = \mathrm{tf}(t,d) \cdot \mathrm{idf}(t,D) \tag{3}$$

With this tfidf equation I was able to calculate the tfidf values for each term in each document. This gives a more accurate weight for each term rather than simply using the raw count for each term.

2.3 Classification

Once I obtained a vector representation of each text document classification was possible. The classification algorithms I used were found in the **sklearn** library for Python. Each classification algorithm can be safely viewed as a black box. The results of each will be compared in the results section.

2.4 Evaluation metrics

In showing my classification results I will primarily be using the metrics of **support**, **accuracy**, **precision**, **recall** and **f1-score**. The table below is a **confusion matrix** which displays the metrics of true positive, false positive, false negative and true negative.

$$Support = Number\ of\ Documents\ from\ a\ certain\ class \tag{4}$$

	Actual: True	Actual: False
Predicted: True	True Positive (TP)	False Positive (FP)
Predicted: False	False Negative (FN)	True Negative (TN)

Table 4: Confusion Matrix

Based on the confusion matrix we can define the following terms.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{5}$$

Accuracy is simply the percentage of documents that were label correctly. This measure can be somewhat misleading, a high accuracy does not always mean that the classifier did a good job. For example, if we have 5 bots and 95 normal users in our data and our classifier classifies every user as a normal user our classifier will achieve an accuracy of 0.95 which seems good if you had no knowledge of the data. Because of this problem we cannot rely on accuracy alone.

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

Recall identifies the proportion of actual positives that were predicted correctly. For example, if we have 5 bots and 95 normal users in our data and we label 3 of the bots as bots and 2 of the normal users as bots then our recall is 0.60 because 3 / (3 + 2) = 0.60.

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

Precision identifies the proportion of positive predictions that were correctly labeled. For example, if we have 5 bots and 95 normal users in our data and we label all 5 bots as bots and 5 normal users as bots our precision would be 0.50 because 5 / (5 + 5) = 0.50. It measures the percentage of actual bots of all the accounts that were labeled as a bot.

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{8}$$

F1-Score combines precision and recall. Ideally, we want high precision and recall but the two metrics often have a tug of war relationship, i.e. if recall increases then precision decreases and vice versa.

3 Mechanics

In order to build a bot classifier data was first extracted then transformed into vector form. In the following sections I will describe how the data was extracted, stored, processed and finally classified. In all I scraped 937 bot accounts and 406 normal user accounts. A few of the bot accounts were discarded due to having no data.

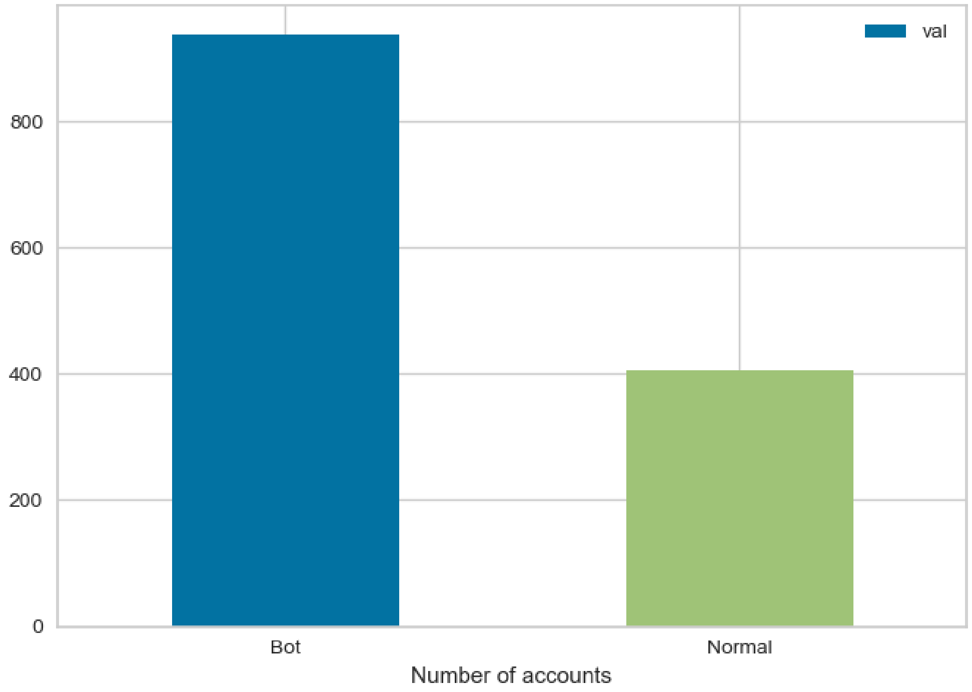


Figure 3: Number of accounts used in classification

3.1 Data Extraction

Reddit has a popular API called praw that is very compatible with Python. I decided to use this API for user account data only and not use it to extract a user's post and comment history. This is because praw limits API calls to less than 1,000 comments and posts and I needed to be able to extract every comment and post for the accounts I was analyzing. To get all of a user's post and comment history I used PushShift Reddit API, an API created by a 3rd part used to extract Reddit data.

Most of the bots were created in April 2015 and all the accounts were banned by April 10th, 2018. Because the nature and tactics of bots change over time, I only extracted normal user activity between April 2015 and April 2018. This was the ensure that I was comparing the activity of each bot with the activity of normal users within the same time frame. In my research I am assuming that none of the normal accounts that I am extracting are bots. If this assumption is wrong it could slightly taint me research.

I extracted normal users by generating all comments created on Reddit after April 10th, 2015 12:00:00AM UTC and then created a list of accounts from the authors of these comments. With this list of accounts, I extracted their account data along with all their posts and comments between April 2015 and April 2018, the same time frame in which the bot accounts were active. This method of normal user extraction ensured that there was not systematic bias in the subreddits that the normal users were regular contributors in.

Here are a few examples of the comments and posts between normal users and bot users. I found that most of the bot comments and posts were either very politically charged, clearly favoring one side or very simple and low effort. The politically charged content is likely to purpose of why the accounts were used and the low effort comments and posts to things to funny cat pictures and cute dogs were likely meant to distract and add some filler content among the propaganda.

Comment	subreddit
CNN = Clinton News Network. Even her daughter Chelsea used to work there for some time.	`politics`
I do not have such information. May be. The only thing i know is that CNN is in the top ten liberal new networks along with CBS, NBC, ABC and others.	`politics`
I agree, If we take the latest news about police fatal shootings into account, this guy had one chance in a hundred.	`Bad_Cop_No_Donut`
The only law Hillary Clinton knows is the law of Wall Street. If you take Something in return, soon you 'll have to repay a debt;) That's why ex-first lady tries her best during debates and primaries.	`ClintonforPrison2016`
this is from 'Super Troopers' movie.	`gifs`
That would be an awesome prank!	`funny`

Table 5: Bot User Comments

Comment	subreddit
You could work in conservation, research, education, rescue, zoos, veterinary care, etc. Where I live there are a lot of opportunities for various fields (Seattle, WA), but depending on where you are, you might have to move to find something that suits you. Think about what you have a passion for. If it isn't breeding, do you want to educate? Or just work with the animals? In addition to the schooling, find places beyond school that you can get experience. Volunteer with a local rescue or vet, work a few hours at a reptile store, etc. This could help you find the direction you want to go in. .	`snakes`
You don't have to explain why but saying a champion is OP doesn't mean shit to me unless you can back it up. I can run around talking shit how good jungle Teemo and Lux are. Basically what this video teaches people is sit in your jungle and wait until the enemy is at your turret and then gank. Kind of hard to carry games when that doesn't happen.	`leagueoflegends`
It's my money and my decision. If I end up liking how it looks then I'll preorder it. If I don't, I won't. I don't need you telling me how to handle my cash.	`CallOfDuty`
Isn't having all three akadora an optional yaku? I remember seeing it once in a video on Youtube.	`Mahjong`

Table 6: Normal User Comments

Post	subreddit
This could be my baby's first gun	`gifs`
Bubble Breathing Dragon	`aww`
Do you think Trump is racist? Or his followers have created a bad image for him? How about KKK group in support of Hilary Clinton because she stands for what they believe in? Does that make her racist too?	`AskReddit`
The number of people killed by police	`ProtectAndServe`
South Korea Is Not Banning Bitcoin Trade, Financial Regulators Clarify	`Bitcoin`

Table 7: Bot User Posts

Post	subreddit
Brothers, maidens of swole. Lend me thine ear muscles. My heart weighs heavy tonight after a night of unholy swolesting by vile brokain agents of the night.	`swoleacceptance`
Chelsea and Man City both eye Danny Rose - sources	`chelseafc`
I accidentally became my dad the other night.	`dadjokes`
Trying to find an App on Shopify. Please help.	`ecommerce`
Free RVCA Stickers - (xpost r/freebies)	`freestickers`

Table 8: Normal User Posts

3.2 Data Storage

For each user (bot or normal) I had access to the following attributes.

- `comment_karma`
- `comments`
- `created_utc`
- `has_verified_email`
- `icon_img`
- `id`
- `is_employee`
- `is_friend`
- `is_gold`
- `link_karma`
- `name`
- `submissions`
- `subreddit`
- `subreddit['banner_img']`
- `subreddit['name']`
- `subreddit['over_18']`
- `subreddit['public_description']`
- `subreddit['subscribers']`
- `subreddit['title']`

Many of these attributes were useless or were not used in my classification so I only stored the following in my database:

- `comment_karma`
- `comments`
- `created_utc`
- `link_karma`
- `name`
- `submissions`

For each post there were many possible attributes to scrape from the pushshift reddit api. The attribute over_18 is a boolean attribute that a user can put on their post if it is inappropriate content. The selftext attribute is an optional description a user can put on their post. I extracted only the following:

- `created_utc`
- `num_comments`
- `over_18`
- `score`
- `subreddit`
- `title`
- `selftext`

For each comment there were many possible attributes to scrape from the pushshift reddit api but I extracted only the following:

- `body`
- `created_utc`
- `score`
- `subreddit`

I used mongodb to store the Reddit user data. Mongodb is very compatible with python. Each entry in the database is a user with the attributes I listed earlier including an array of comments and an array of posts.

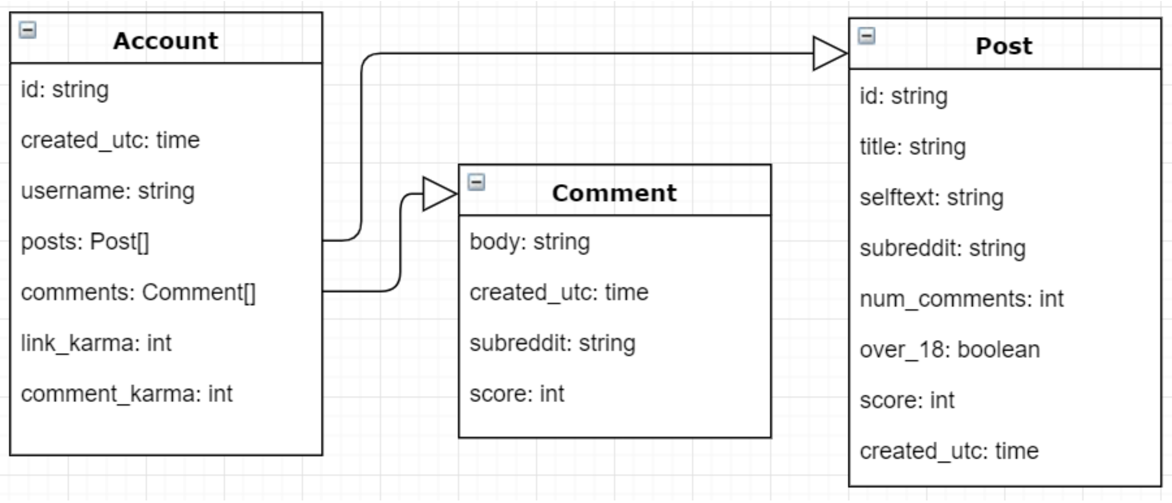


Figure 4: Database design

In total I scraped 937 bots and 406 normal users. The amount of comments and posts and the ratio of number of comments to posts for bots and normal accounts is markedly different. The ratio of bot posts to comments is approximately 1:2 while the ratio of normal user posts to comments is 1:40. This means that looking at the ratio of comments to posts alone can tell a lot about whether an account is a bot or not based on the data I was using.

	post	comment
bot	13,388	6,519
normal	35,209	1,409,256
total	48,597	1,415,775

Table 9: Number of Comments and Posts For Bot and Normal Users

3.3 Pipeline

Below is a diagram of the pipeline of the entire process of classification from when the data was first extracted from Reddit, stored in mongodb, transformed into a tfidf vector, classified and finally outputted as readable results.

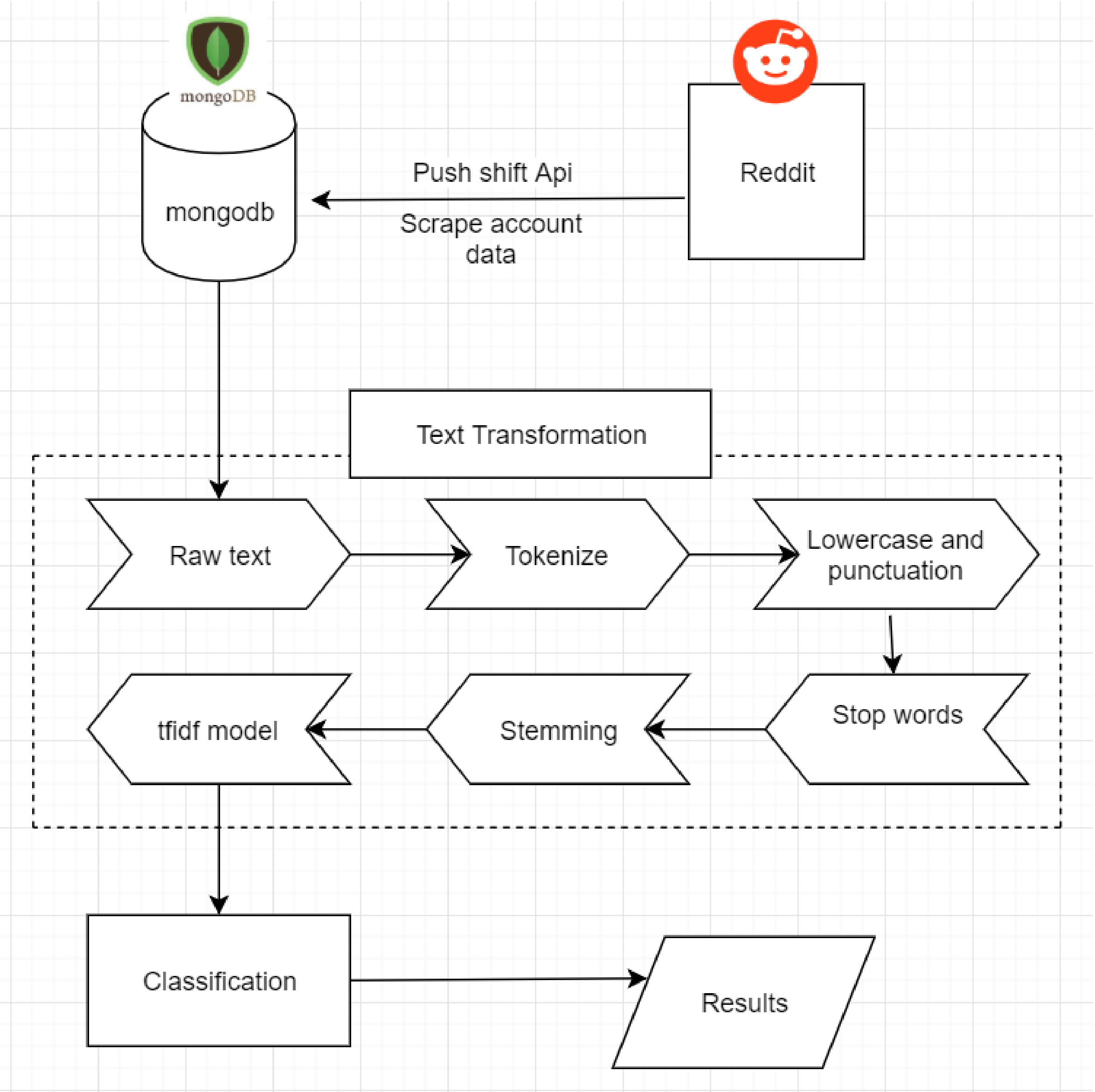


Figure 5: Pipeline

I performed classification on 4 different aspects of an account. The account's posts, comments, subreddit of post and subreddit of comment. For each classification I tested several algorithms and found that the **Extra Trees Classifier** consistently outperformed the other classification algorithms so I decided to use that in my final analysis. Below you will find the results of each classification method along with some explanation of the results.

4 Post Title

4.1 Data

For post title classification I viewed the title of each post as a document labeled either bot or normal. The text was transformed into the bag of words model and then tfidf vector as explained in section 2.3. After this transformation the data was split into 0.80 training data and 0.20 test data and each document was classified as bot or normal. My results are below.

	predicted: bot	predicted: normal
actual: bot	1885	479
actual: normal	718	6404

Table 10: Post Title Classification Confusion Matrix

	precision	recall	f1-score	support
bot	0.72	0.80	0.76	2364
normal	0.93	0.90	0.91	7122
micro avg	0.87	0.87	0.87	9486
macro avg	0.83	0.85	0.84	9486
weighted avg	0.88	0.87	0.88	9486

Table 11: Post Title Classification Metrics

$$\text{Accuracy} = 0.8738$$

4.2 Analysis

Classifying an account as a bot or normal user using only the text in the titles of their posts was very effective. The accuracy was very high as well as recall and precision, which combined to create a high f1-score. The number of bot posts and normal user posts was imbalanced but not to an extreme degree. Approximately 25% of the posts were bots and 75% of the posts were posted by normal users. Sometimes this imbalance in data can lead to a high accuracy but also cause other metrics to be poor. This was not the case in this classification.

My classification of bots achieved a recall of 0.80 which means that 80% of actual bots were correctly predicted to be bots. A precision of 0.72 means that 72% of posts were correctly predicted to be a bot was a bot. When dealing with an imbalanced dataset, the most important metric is precision. Since we have 75% negative (normal) documents and 25% positive (bot) documents, identifying more accounts as normal would result in a higher accuracy. So, in this classification when we label a document as a bot, 72% of the time it was correct.

	Top Normal	**Top Bot**
1	season	cop
2	thread	cops
3	event	clinton
4	goal	officer
5	recipe	hillary
6	diplomacy	america
7	comments	police
8	game	obama
9	scores	officers
10	r	american

Table 12: Top Words For Post Titles

	Word
1	trump
2	obama
3	hilary
4	diplomacy
5	reddit
6	cops
7	cop
8	facebook
9	andes
10	spoilers

Table 13: Most Characteristic Words For The Post Title Corpus

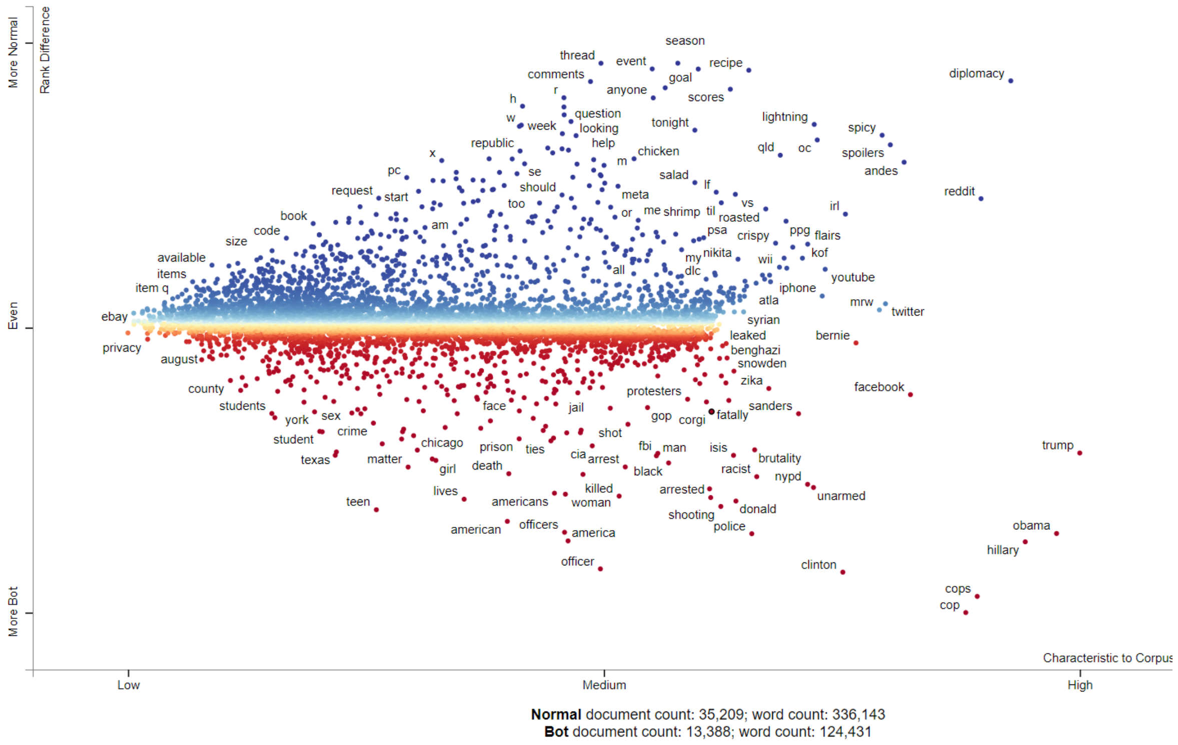


Figure 6: Post Title Word Visualization

Above is a visualization of the words within the post title classification corpus. Words represented by red dots are indicative of bots and the blue dots are for normal users. This chart is interactive, has a search function and provides statistics such as the frequency and word count for each word. It will also retrieve a list of every occurrence of a word.

Like the figure says, there were 35,209 normal user posts containing 336,143 words and 13,388 bot posts with 124,431 words.

5 Comment Body

5.1 Data

For comment classification I viewed the text body of each comment as a document with a label of either bot or normal. Just like post title, the text of each comment was transformed into a bag of words model and then a tfidf vector as explained in section 2.3. After this transformation the data was split into 0.80 training data and 0.20 test data and each document was classified as a bot or normal. My results are below.

	predicted: bot	predicted: normal
actual: bot	232	108
actual: normal	1094	273602

Table 14: Comment Body Classification Confusion Matrix

	precision	recall	f1-score	support
bot	0.17	0.68	0.28	340
normal	1.00	1.00	1.00	274696
micro avg	1.00	1.00	1.00	275036
macro avg	0.59	0.84	0.64	275036
weighted avg	1.00	1.00	1.00	275036

Table 15: Comment Body Classification Metrics

$$\text{Accuracy} = 0.9968$$

5.2 Analysis

Do not let the 99.68% accuracy fool you, this data was very imbalanced, a common problem in classification. Taken to an extreme degree, if 99% of accounts were normal and 1% were bots, then labeling all accounts as normal would result in a 99% accuracy. In my case 99.87% of the accounts were normal leaving 0.13% of the accounts to be bots. Therefore, it is not very helpful to look at the accuracy metric to determine whether this classification was successful. This means that if I were to blindly label every comment as a bot, I would achieve 99.87% accuracy. Instead, the metrics precision and recall must be examined.

To determine the effectiveness of the classifier we are most interested in detecting positives, which in this case is the bot label. Of the 1,326 accounts that were labeled as a bot, 17% were bots. Likewise, of the 340 bots the classifier was able to correctly predict 68% of them as bots. These numbers may seem low, but when you consider that we are analyzing 275,036 comments those numbers are that of an effective classifier.

It is important to keep in mind that these results are from simply viewing each comment as a bag of words. For a human, identifying whether comments come from a bot can be very challenging, especially when that human is dealing with the problem of imbalanced data online, seeing a small number of bot created comments among a sea of legitimate content. Once the complexity of this problem is appreciated, the precision and recall numbers of this classification become for impressive.

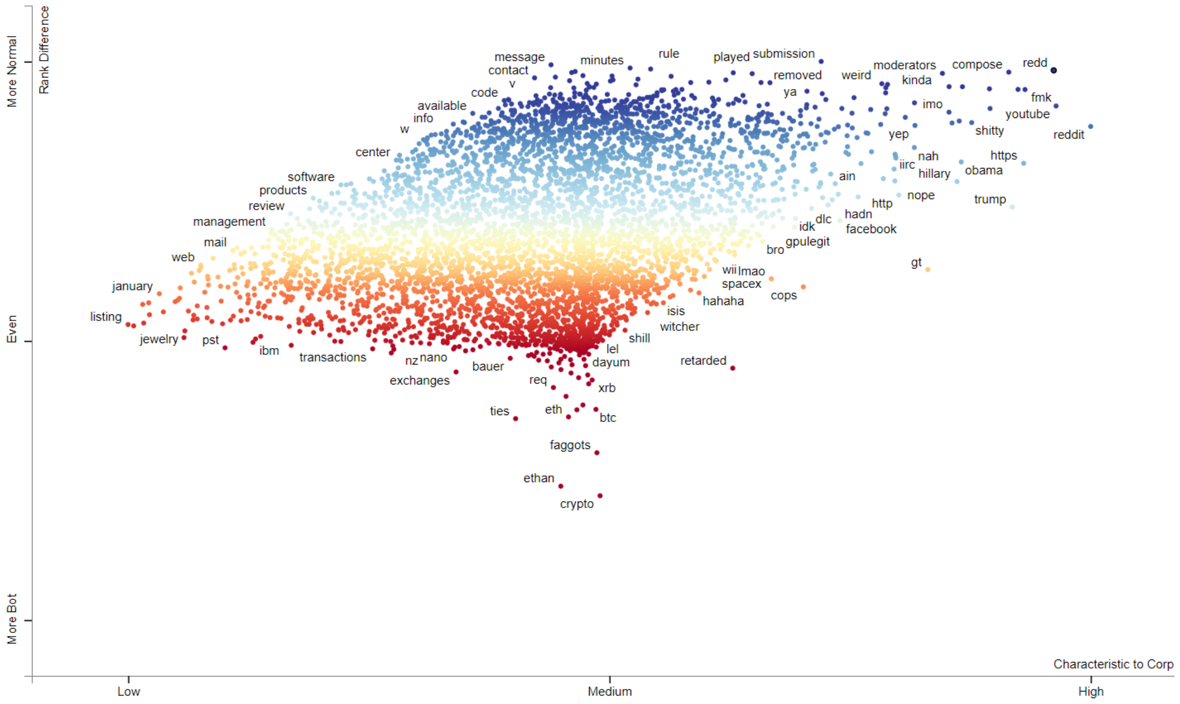
	Top Normal	**Top Bot**
1	submission	crypto
2	message	ethan
3	minutes	faggots
4	season	ties
4	redd	eth
5	image	tie
6	compose	btc
7	three	iota
8	automatically	tokens
9	moderators	req
10	performed	xrp

Table 16: Top Words For Comment Bodies

	Word
1	reddit
2	youtube
3	redd
4	fmk
5	tallies
6	trump
7	compose
8	bernie
9	tally
10	shitty

Table 17: Most Characteristic Words For Comment Corpus

The words reddit and youtube are likely the most common words because they are part of links.



Normal document count: 1,409,248; word count: 43,244,619
Bot document count: 6,519; word count: 113,350

Figure 7: Comment Body Word Visualization

6 Post Subreddit

6.1 Data

For the classification of the subreddit each post of a user I concatenated the subreddit of each post of a user together into one string. For example, if a user has three posts posted in the subreddits baseball, space and politics I concatenated the three together into one string, i.e. "baseball space politics". This string is then transformed into a bag of words model and then a tfidf vector as explained in section 2.3. Based on this concatenated string of subreddits we classify a user as a bot or normal. Below are my results.

	predicted: bot	predicted: normal
actual: bot	66	2
actual: normal	4	71

Table 18: Post Subreddit Classification Confusion Matrix

	precision	recall	f1-score	support
bot	0.94	0.97	0.96	68
normal	0.97	0.95	0.96	75
micro avg	0.96	0.96	0.96	143
macro avg	0.96	0.96	0.96	143
weighted avg	0.96	0.96	0.96	143

Table 19: Post Subreddit Classification Metrics

$$\text{Accuracy} = 0.9580$$

6.2 Analysis

This method of classification was very successful. The accuracy of 95.8% is backed up by a precision of 0.96 and a recall of 0.96. Of the 143 users examined only 6 were labeled incorrectly. This means that based on which subreddit the user is posting in alone I was able to predict whether that account was a bot or not correctly 95.8% of the time.

To give more insight into which subreddit the bots and normal users are posting in I will provide some statistics below.

	Top Normal	**Top Bot**
1	newsonreddit	bad_cop_no_cop
2	mylittlepony	uncen
3	onetruebiribiri	racism
4	spam	copwatch
5	postworldpowers	uspolitics
6	westernbulldogs	police
7	wastelandpowers	blackpower
8	libertarian	blackfellas
9	canucks	hillaryforprison
10	fivenightsatfreddys	police_v_video

Table 20: Top Subreddits for Posts

These subreddits are the most useful when determining if the account's post history is a bot or a normal user. For example bad_cop_no_cop is very characteristic of a bot account and newsonreddit is very characteristic of a normal account. An important note is that since I scraped the data of random accounts for normal users this list of subreddits is not necessarily typical of the "average" Reddit user. If this was the goal I would have had to scrape many accounts because there are so many niches within Reddit.

	Word
1	worldpowers
2	uncen
3	tampabaylightning
4	askreddit
5	politicalhumor
6	streetfightercj
7	foodporn
8	fireteams
9	fivenightsatfreddys
10	fireemblemheroes

Table 21: Most Characteristic Subreddits for Posts

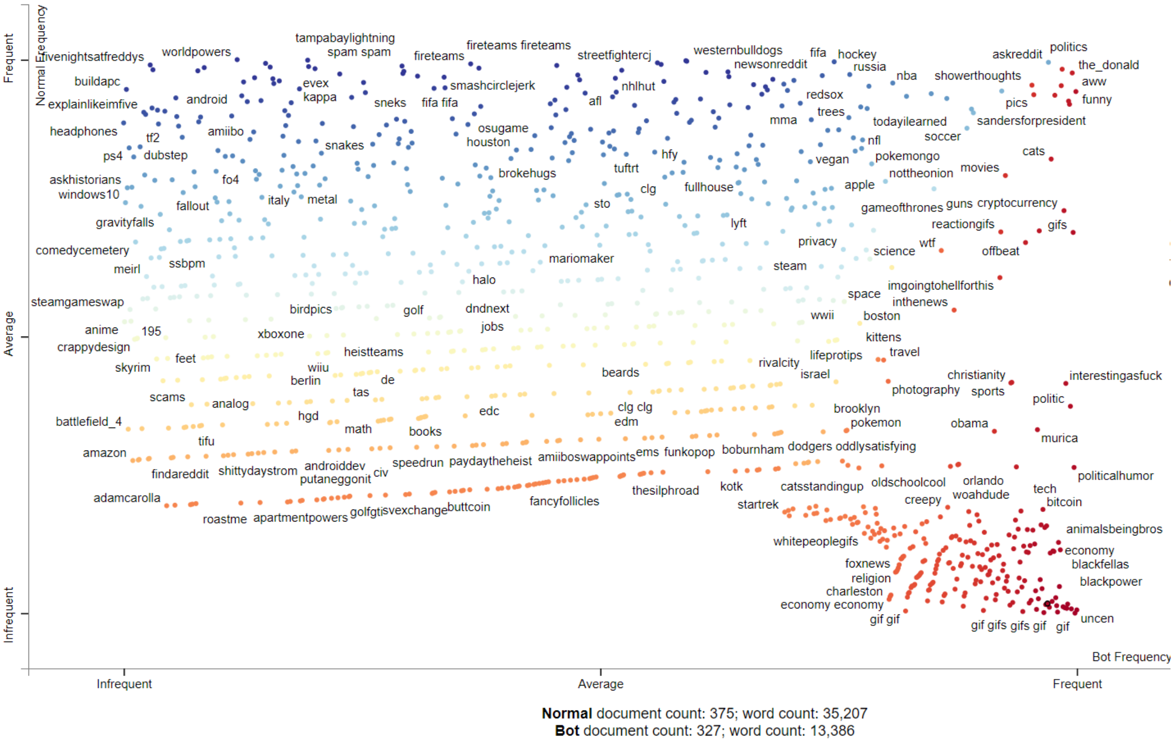


Figure 8: Post Subreddit Visualization

7 Comment Subreddit

7.1 Data

For classification of the subreddit each comment of a user I concatenated the subreddit of each comment of a user together into one string. For example, if a user has three comments posted in the subreddits baseball, space and politics I concatenated the three together into one string, i.e. "baseball space politics". This string is then transformed into a bag of words model and then a tfidf vector as explained in section 2.3. Based on this concatenated string of subreddits we classify a user as a bot or normal. Below are my results.

	predicted: bot	predicted: normal
actual: bot	70	14
actual: normal	0	59

Table 22: Comment Subreddit Classification Confusion Matrix

	precision	recall	f1-score	support
bot	1.00	0.83	0.91	84
normal	0.81	1.00	0.89	59
micro avg	0.90	0.90	0.90	143
macro avg	0.90	0.92	0.90	143
weighted avg	0.92	0.90	0.90	143

Table 23: Comment Subreddit Classification Metrics

$$\text{Accuracy} = 0.9021$$

7.2 Analysis

Classification of a user based on the subreddit's of their comments was very effective. In my study, classifying a user based on the subreddit of their posts and comments had better results than examining the text of their posts and comments alone. There is a very strong pattern in subreddits that the bots post in. Below is a list of the most common subreddits that the bots commented in.

	Top Normal	**Top Bot**
1	adviceanimals	cryptocurrencies
2	spacex	cryptocurrency
3	anime	altcoin
4	canucks	blockchain
5	opieandanthony	femboys
6	newmarvelrp	ggcrypto
7	rupaulsdragrace	sissies
8	comicbooks	bitcoinall
9	streetfightercj	cryptomarkets
10	paydaytheheist	icocrypto

Table 24: Top Subreddits for Comments

	Word
1	askreddit
2	monarchyofequestria
3	redsox
4	maddenultimateteam
5	destinythegame
6	pcmasterrace
7	theoddadventures
8	todayilearned
9	cfbofftopic
10	worldnews

Table 25: Most Characteristic Subreddits For Comments

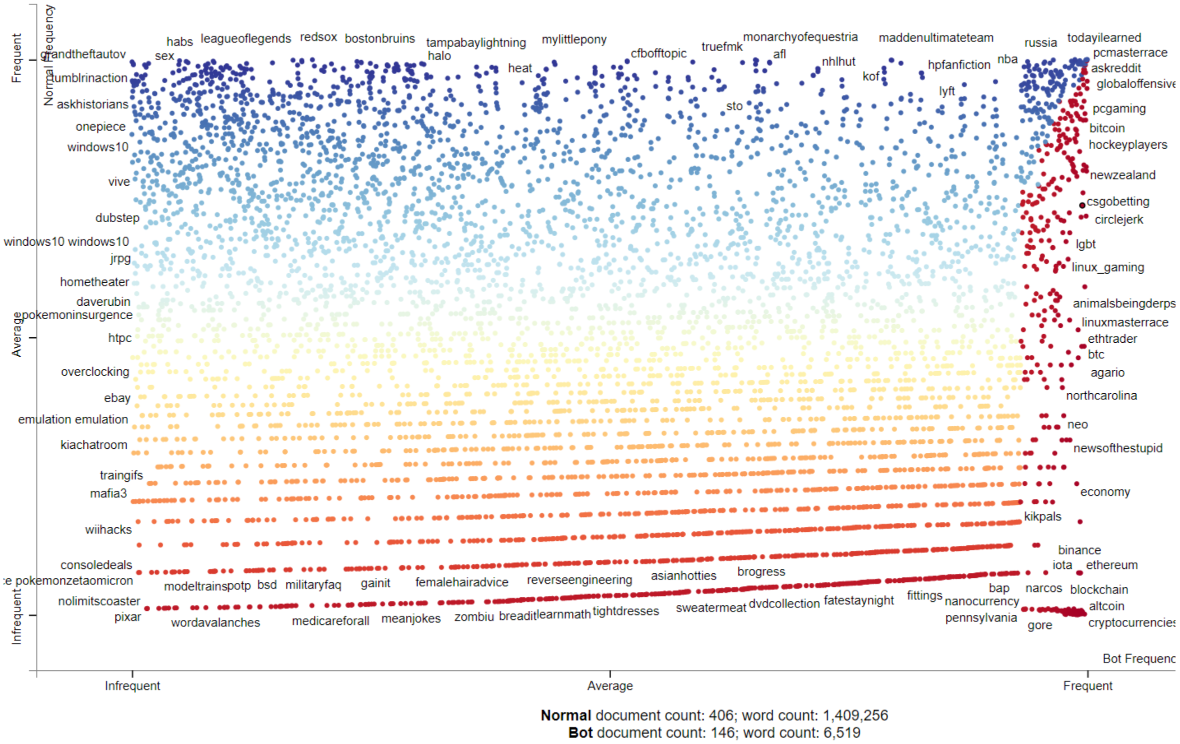


Figure 9: Comment Subreddit Visualization

8 Account Characteristics

8.1 Data

This section contains many findings related to the differences in account characteristics between bot and normal users. None of this data was used in classification. Using the account data of the date the account was created, number of comments, number of posts, date and time of comments and date and time of posts I was able to find some interesting patterns. In the next few sections I will display some of my findings.

First we notice the dates in which the accounts were created. Below is a histogram of the frequency of accounts created per year and month.

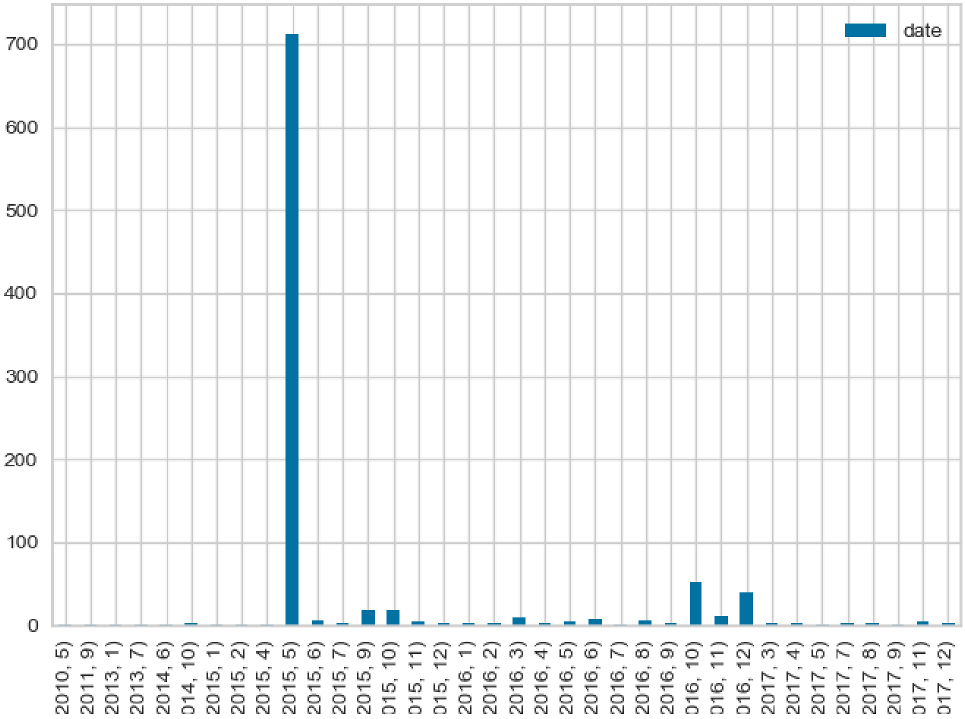


Figure 10: Date of account creation of bots

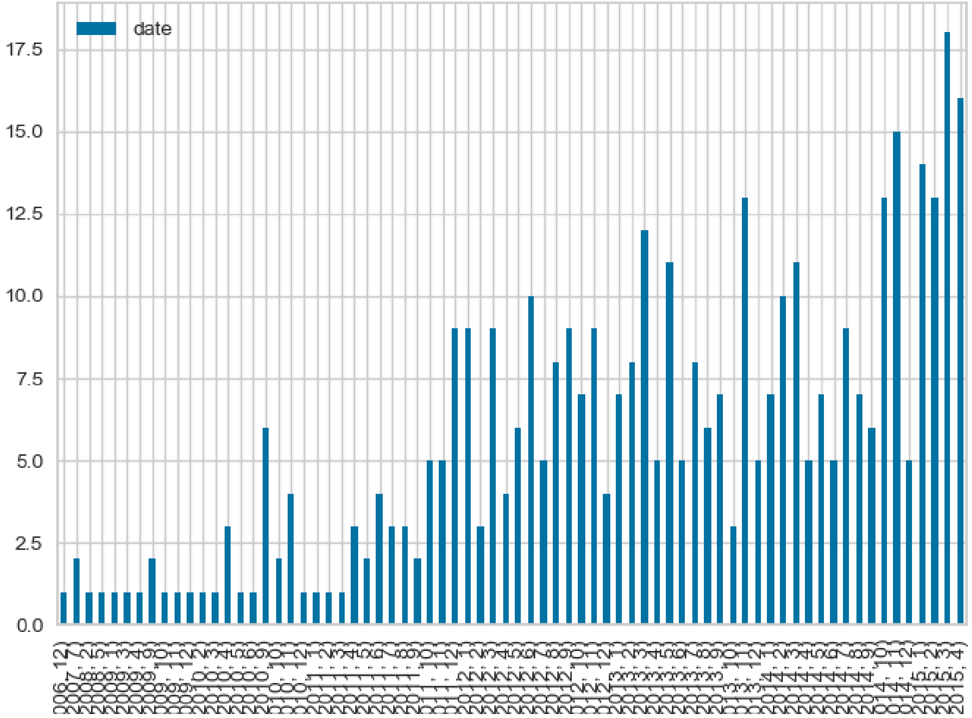


Figure 11: Date of account creation of normal users

From these two histograms it appears that most of the bot accounts were made in one batch. Identifying this is an important indicator that many of these accounts came from a single source. The histogram for normal user accounts follows a pattern that seems to be likely for regular Reddit users, a higher number of the accounts created recently and a small number of older accounts. I do not have information from Reddit to back up this claim.

Next I have two histograms of the frequency of comments and posts between normal and bot accounts. I believe that these two graphs are evidence that the two account groups are from different time zones based on the time of their activity.

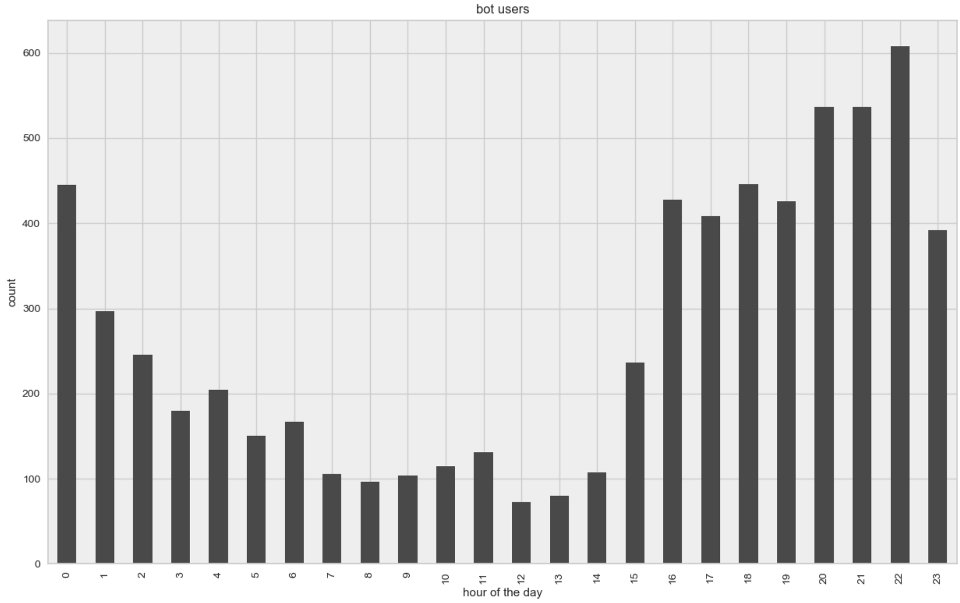


Figure 12: Frequency of Bot Comments By Hour

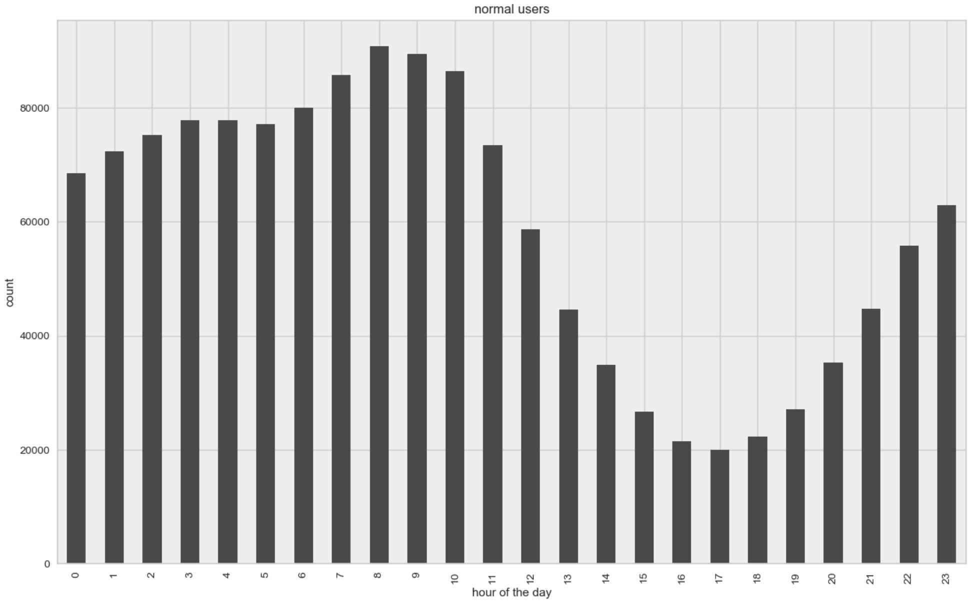


Figure 13: Frequency of Normal User Comments By Hour

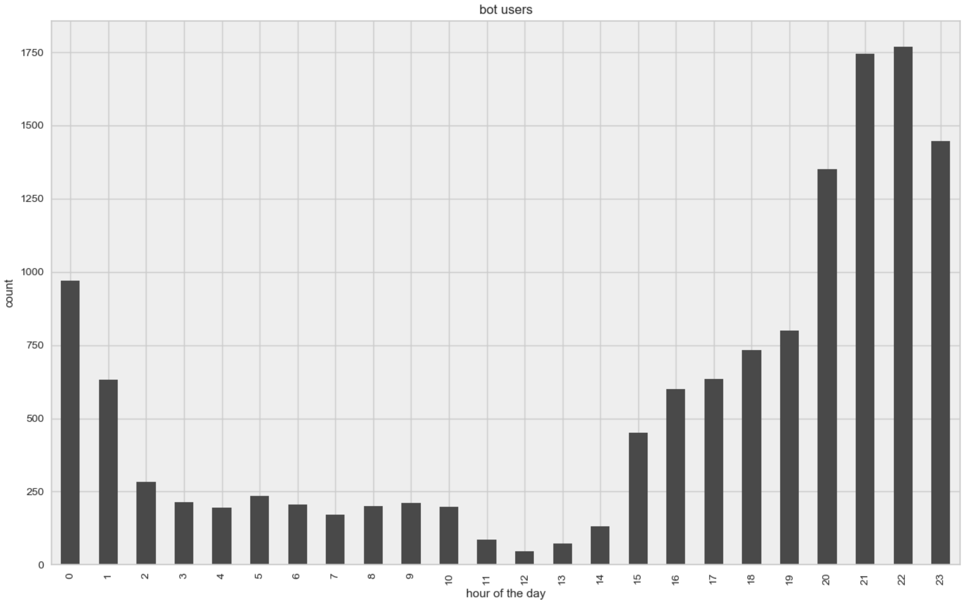


Figure 14: Frequency of Bot Posts By Hour

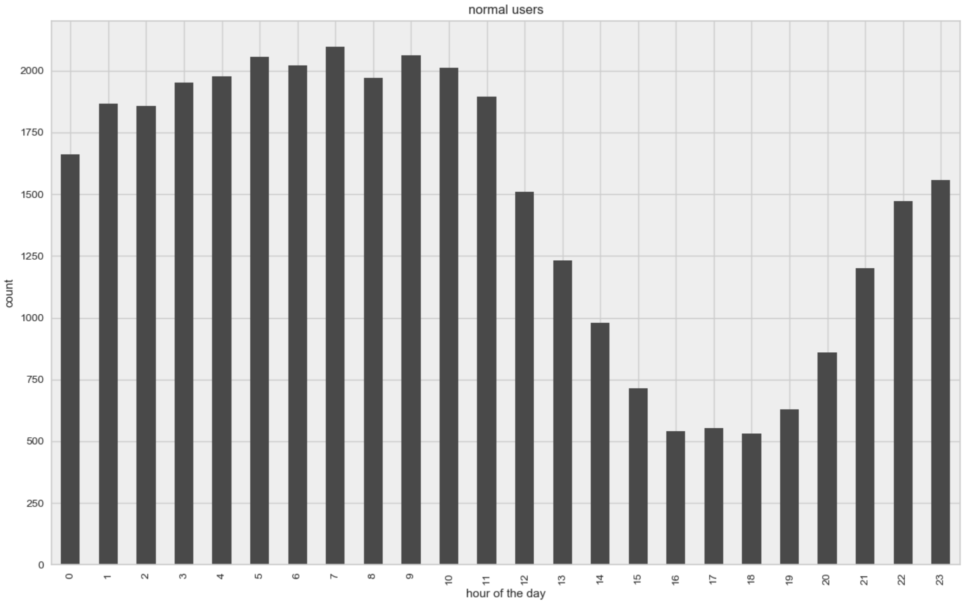


Figure 15: Frequency of Normal User Posts By Hour

From these four figures it seems likely that the bots are from a different time zone than the average user of Reddit, which is from the United States [2] as seen below.

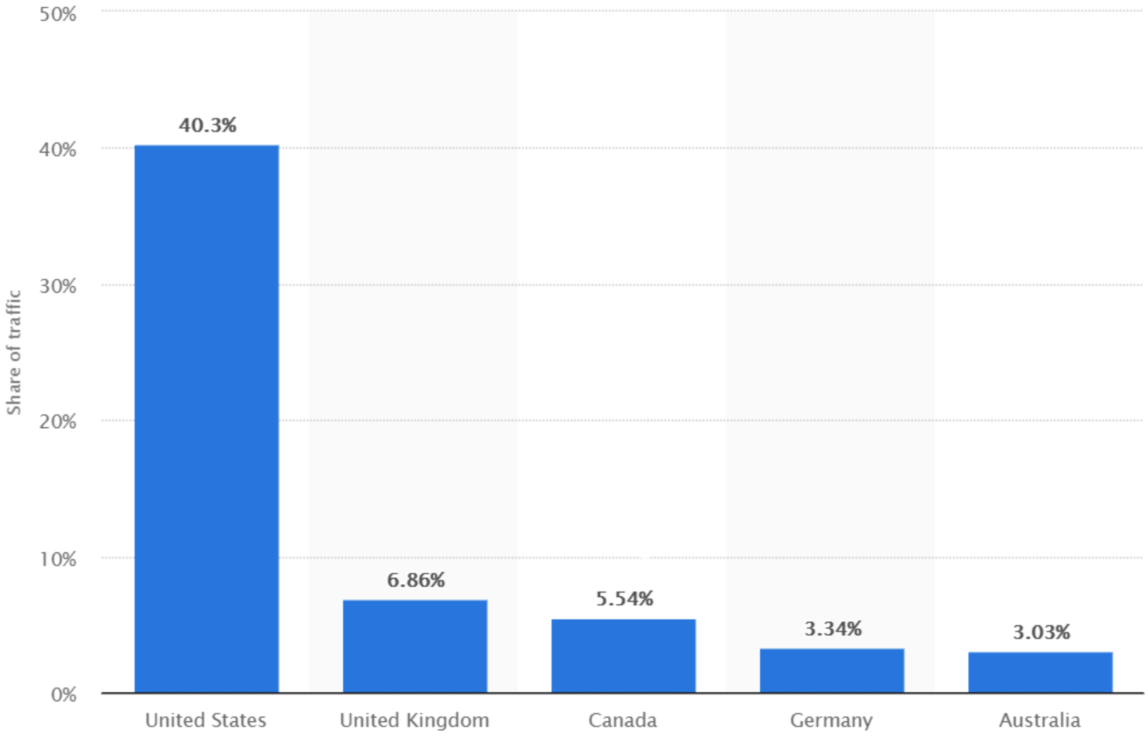


Figure 16: Distribution of Reddit Users By Country

Below we observe the amount of comments per account. We can see that the amount of comments for the bot accounts, on average, is much lower than the normal accounts. On Reddit, posts typically reach a larger audience, therefore having a larger impact if an account was maliciously spreading propaganda. Attempting to influence users through comments would work, it would just require a lot more effort. **On the next two figures notice the difference in the X-axis scale.**

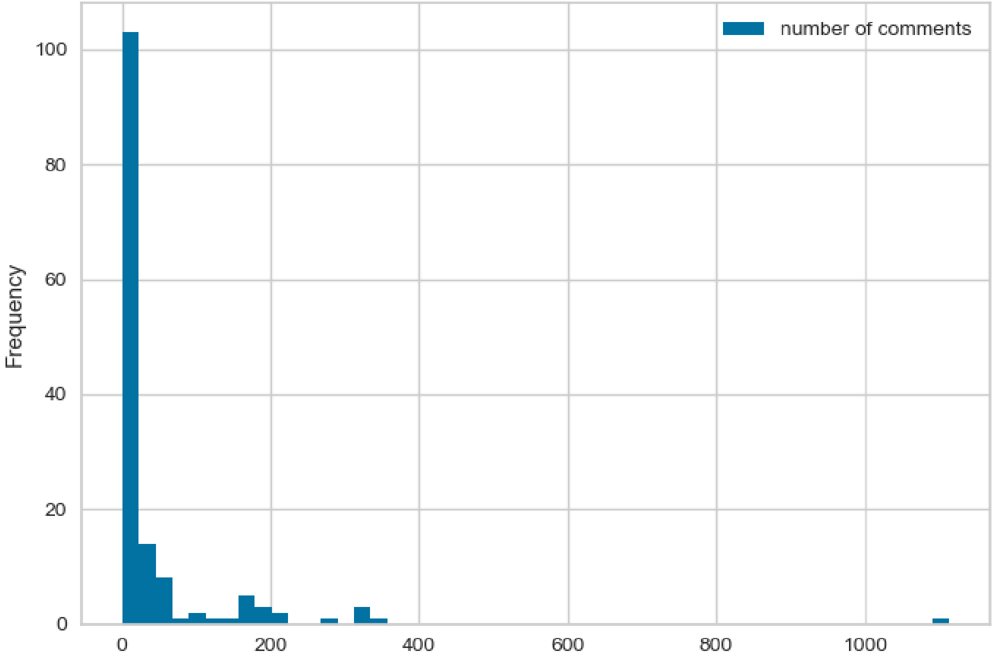


Figure 17: Bot account number of comments

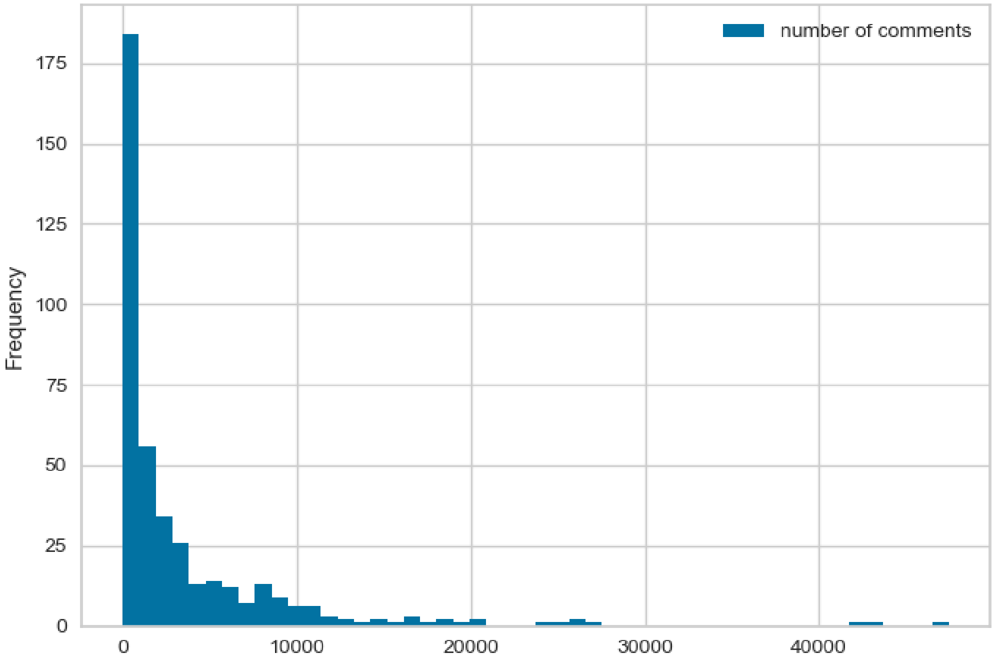


Figure 18: Normal account number of comments

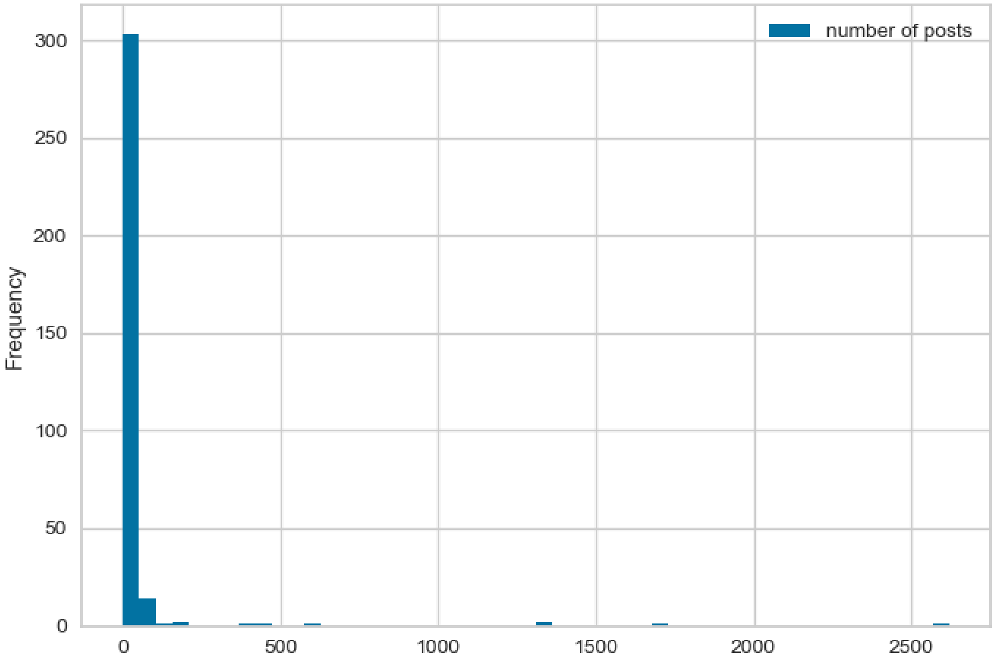


Figure 19: Bot account number of posts

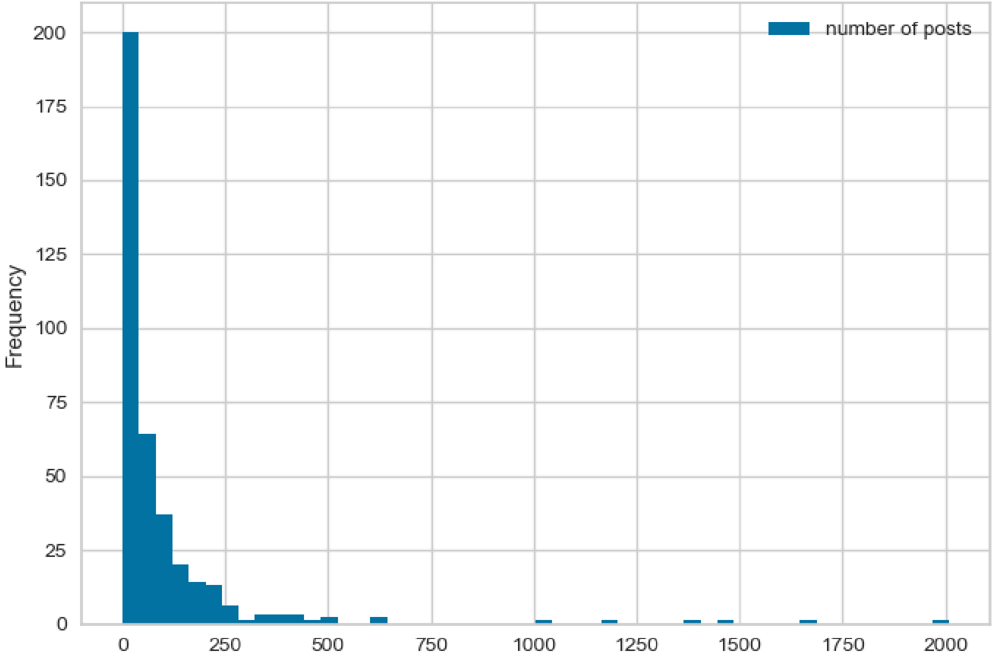


Figure 20: Normal account number of posts

Both bot and normal user accounts have a large number of accounts that have little to no post and comments history. The normal user accounts have a realistic looking drop off in activity, the bots have almost no accounts that just have some infrequent activity. Both groups of accounts have a few, sporadic users who create a lot of posts and comments.

Lastly we observe the time of day that the accounts were created. This metric, similar to the time of the day that the accounts commented and posted, is telling of the time zone that the accounts are in. Additionally, if a large amount of accounts were created by a script and not a human then we would observe a large amount of bot accounts created in a time span shorter than that a human could do.

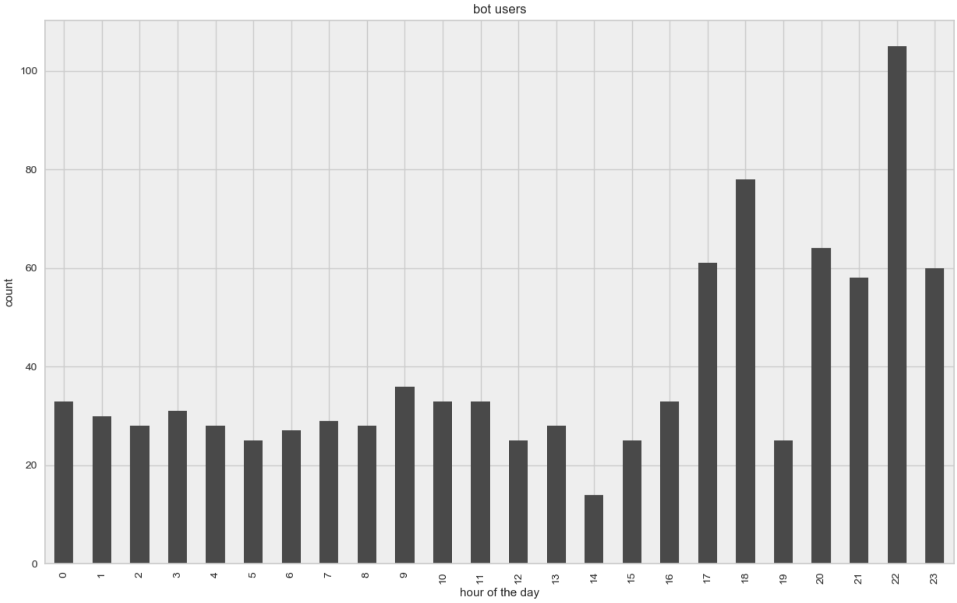


Figure 21: Bot Hour of the Day Account was Created

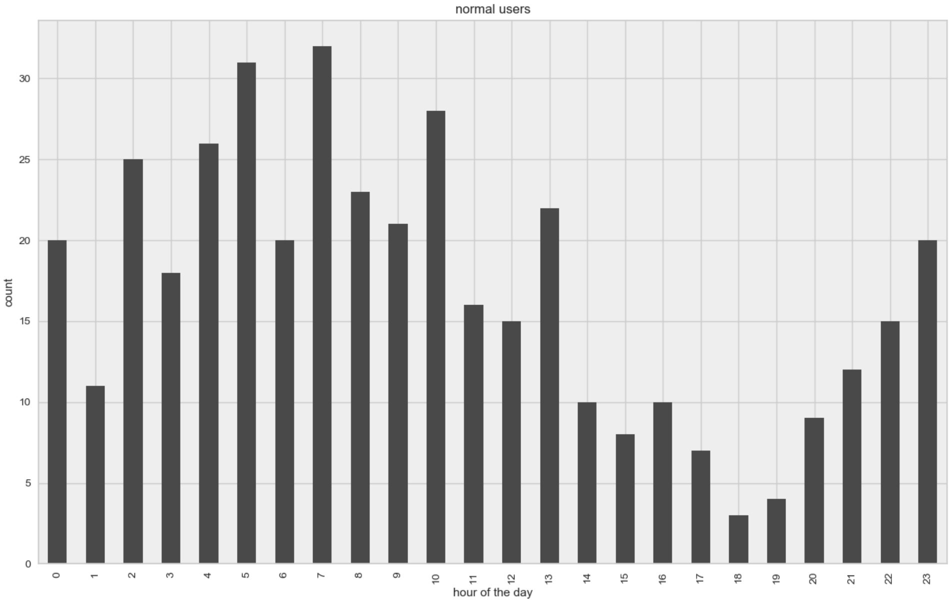


Figure 22: Normal User Hour of the Day Account was Created

9 Conclusion

The classification of Reddit user accounts was very effective. Simply treating each comment or post of a user as a bag of words resulted in significant success. What results in the most success was classification based on the subreddit that an account was posting and commenting in. Based only on the list of subreddits that an account posted and commented in the f1-score was 0.96 and 0.9 respectively. Other the classification done there were other account metrics that was very suspicious such as many accounts being created on the same date, bot accounts posting and commenting during different times than the typical Reddit user, and also the number of comments and posts per account.

In the future I would like to create a classifier that took into account the posts, comments, subreddits of posts and comments, and included account characteristics as well. To do this the weights of each classifier would have to be tweaked correctly but based on my project I believe that if such a classifier were to be made it would be very successful in identifying bots. It is my hope that going forward bot detection incorporates natural language processing and an account's information to correctly identify bots and malicious users. Going forward the problem of bots and fake information online is only becoming more common and sophisticated, making it necessary for social media platforms and online forums every to be able to have tools in place to detect such users and deal with them accordingly.

You can see all of the code I used for this project on Github , Reddit's 2017 Transparency - my inspiration for this project and another short write up of my project on my personal website.

References

[1] "Reddit in 2015", [Online] Available: https://redditblog.com/2015/12/31/reddit-in-2015/

[2] "Regional distribution of desktop traffic to Reddit.com as of October 2018, by country", [Online] Available: https://www.statista.com/statistics/325144/reddit-global-active-user-distribution/

[3] "Reddit's 2017 transparency report and suspect account findings", [Online] Available: https://www.reddit.com/r/announcements/comments/8bb85p/reddits_2017_transparency_report_and_suspect/

[4] "sklearn Extra Trees Classifier Documentation", [Online] Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html