Twitter Bots Multiclass Classification Using Bot-Like Behavior Features.

by Lulwah Ahmad AlKulaib

B.S. in Computer Science, June 2011, Gulf University for Science and Technology.

A Thesis submitted to

The Faculty of
The School of Engineering and Applied Science
of The George Washington University
in partial fulfillment of the requirements
for the degree of Master of Science

May 20, 2018

Thesis directed by

David Broniatowski
Assistant Professor of Engineering Management and Systems Engineering

Robert Pless
Patrick & Donna Martin Professor of Computer Science

Dedication

This thesis is dedicated to my father and mother, who are my great sources of inspiration and whose good examples have taught me to work hard for the things that I aspire to achieve.

Acknowledgement

"He has not thanked Allah who has not thanked people."

Prophet Muhamad, peace and blessings be upon him.

I would like to express my sincere gratitude and appreciation to my advisor, Prof. David Broniatowski, for his continuous support of my research as well as his patience, motivation, enthusiasm, and immense knowledge. I would also like to thank Prof. Robert Pless for his insight and guidance throughout this process.

I would like to thank the rest of my thesis committee: Prof. Abdou Youssef , Prof. Robert Pless, and Prof. David Broniatowski, for their encouragement, comments, and questions.

I appreciate Sihua Qi's efforts and contributions. Sihua is a fellow GWU MS student, who worked on the Bot-like behavior project that served as the basis for multi-class classification. I am grateful for our extensive discussions, the brainstorming sessions, and his patience as he explained the intricacies of statistical analysis.

Special thanks to Adrian Benton, who is a Computer Science PhD candidate at Johns Hopkins University, for his help with data collection that was essential to this project's completion.

I am very grateful to my mother, Najeebah AlMudhaf, my father, Ahmad AlKulaib, my brothers, sister, and sisters in-law for their tremendous support and encouragement. I would like to extend my genuine thanks to all my friends, colleagues, and extended family for their assistance during these past few months.

Lastly, I thank the Kuwait Institute for Scientific Research (KISR) for sponsoring and funding my journey.

Abstract of Thesis

Twitter Bots Multiclass Classification Using Bot-Like Behavior Features

Bots on Twitter are accounts that are controlled by computer programs, automatically producing content, and interacting with other accounts. These programs are turned on and off without following a pattern, making them hard to identify. Using previous work that identifies bot accounts bot-like behavior features, we identified features that are more relevant to some bot types than others. In this thesis, we propose a novel bot type classification method by using bot-like behavior features.

Multiclass classification is our proposed idea for this project. We use the output data from the bot-like behavior to train a MaxEnt classifier to identify 6 different classes (5 bots, 1 human). We collect our test dataset to match the structure of our training set except for labels, then use our classifier to test it. Moreover, we analyze an additional holdout set to test the polarity of bot classes in vaccine topics on Twitter.

Table of Contents

List of Figures

List of Tables

Glossary of Terms

Google's Machine Learning glossary of terms [5] defines the following terms as follows:

Accuracy	The fraction of predictions that a classification model got right. In multi-class classification, accuracy is defined as follows: $$Accuracy = \frac{Correct\ predictions}{Total\ Number\ of\ Examples}$$
Bot	An automated account, that is controlled by a program, automatically producing content, and interacting with other accounts.
Class	One of a set of enumerated target values for a label. For example, in a binary classification model that detects bots, the two classes are bot or not bot. In a multi-class classification model that identifies bot types, the classes would be social spam bot, content polluter bot, fake follower, and so on.
Confusion Matrix	An NxN table that summarizes how successful a classification model's predictions were; that is, the correlation between the label and the model's classification. One axis of a confusion matrix is the label that the model predicted, and the other axis is the actual label. N represents the number of classes.
Crawl	Collect the most recent N posts on a user's timeline.
Cyborg	A bot acting like a human or a human acting like a bot.
Dataset	A collection of rows with one or more features and possibly a label.
Deduplicate	A specialized data compression technique for eliminating duplicate copies of repeating data.
Holdout data	A dataset intentionally not used ("held out") during training. The validation dataset and test dataset are examples of holdout data. Holdout data helps evaluate a model's ability to generalize to data other than the data it was trained on.
Label	In supervised learning, it is the "answer" or "result" portion of a row.
Logistic regression	A model that generates a probability for each possible discrete label value in classification problems by applying a sigmoid function to a linear prediction. Although logistic regression is often used in binary classification problems, it can also be used in multi-class classification problems (where it becomes called multi-class logistic regression or multinomial regression).
Machine learning	A program or system that builds (trains) a predictive model from input data. The system uses the learned model to make useful predictions from new (never-before-seen) data drawn from the same distribution as the one used to train the model.

Model	The representation of what an ML system has learned from the training data.
Model training	The process of determining the best model.
Multiclass classification	Classification problems that distinguish among more than two classes.
Multinomial classification	Synonym for multiclass classification.
One-vs-all	Given a classification problem with N possible solutions, a one-vs.-all solution consists of N separate binary classifiers—one binary classifier for each possible outcome.
Precision	A metric for classification models. Precision identifies the frequency with which a model was correct when predicting the positive class. $Precision = \frac{True\ positives}{True\ positives + false\ positives}$
Recall	A metric for classification models that answers the following question: Out of all the possible positive labels, how many did the model correctly identify? $Recall = \frac{True\ positives}{True\ positives + false\ negatives}$
Scikit-learn	A popular open-source ML platform. See www.scikit-learn.org.
Semi-supervised learning	Training a model on data where some of the training examples have labels but others don't. One technique for semi-supervised learning is to infer labels for the unlabeled examples, and then to train on the inferred labels to create a new model. Semi-supervised learning can be useful if labels are expensive to obtain but unlabeled examples are plentiful.
Supervised learning	Training a model from input data and its corresponding labels. Supervised machine learning is analogous to a student learning a subject by studying a set of questions and their corresponding answers. After mastering the mapping between questions and answers, the student can then provide answers to new (never-before-seen) questions on the same topic.
Test set	The subset of the data set that is used to test the model after the model has gone through initial vetting by the validation set.
Training set	The subset of the data set used to train a model.

Chapter I: Introduction

Twitter is a social networking service, where registered users post 280 character messages that are known as tweets and they are broadcasted to users in their respective networks. Twitter is an open platform that allows anyone with a valid email to create an account [13]. The relaxed rules of profile creation on this platform allows account automation. Twitter permits automated accounts as long as the owner states it clearly in the "bio" section. An automated account is an account that is controlled by a computer program, which we will refer to as a "bot" going forward. Bots post content that ranges from helpful tips to malicious misinformation and it depends on the goal of its creator. Some Twitter bots do not state that they are automated in their bio, which violates Twitter's terms of service and raises questions about the account's intended purpose. Many bots are created to disseminate news, advertising marketing services, increase the popularity of other accounts [16], enhance message outreach, and influence followers.

Bots can be easy to detect even if they don't mention it in their bio because of their behavior on the social media platform. Periodic and regular timing of tweets, minimal original content tweets, and high rates of interaction with a tweet even with a small number of followees are common indicators of automation. On the other hand, some bots try to imitate human behavior and this programming makes it harder to detect. These accounts are turned on and off, which creates a more authentic gap as well as a difference in pattern and behavior. [14] Recently, such bots have become more sophisticated as they are able to search the internet for information to post on their profiles. These attempts at human emulation open the door for greater engagement with legitimate users, which helps to mask their presence. [12]

Since it is getting harder to distinguish between bots and humans accounts, it makes more sense to try to study the behavior of bots in order to determine better ways to identity bot-like behavior.

In our work, we propose a new bot-like behavior detection method. We utilized multiple datasets that contained four known bot types [4] and then crawled the last 200 posts per user id, which resulted in a bot-like tweet set. Additionally, we leveraged a legitimate users' dataset [15] and collected the most recent 200 posts per user id from that group. This process resulted in a pool of 15,000 user profiles.

We created scripts to detect 19 features of bot-like behavior. Twelve of these features were inspired by Nimmo's article [14], while the other 7 were patterns that we observed on our own. This program script ran on the crawled 15k user dataset and generated reports based on our key criteria. We then applied a logistic regression model on the results and used Akaike Information Criteria values to determine how many of the 19 features were relevant for the purposes of detecting bot-like behavior for each bot type.

Noticing that some bot types have features more relevant to them than others, we thought of building a multiclass classifier that is able to distinguish between different bot types. Using results from the bot like behavior project, we train a Max Entropy classifier to label a given Twitter's account recent activity as one of 5 labels: human, social spam bot, traditional spam bot, content polluter, or fake follower. We tested the classifier with a dataset that followed the same collection procedure of the training data. The test generated a report that had each user's data with a bot label. The classifier is then used to identify types of bots in a holdout dataset and produce a report further discussed in chapter IV.

The remainder of this thesis is structured as follows: In chapter II, we review the previous work in bot detection and bot type identification. In chapter III, we explain our bot-like behavior project in detail. In chapter IV, we thoroughly explain the details of the multiclass classifier that differentiates between bot types and address our findings. In chapter V, we summarize our work and give out the conclusion.

Chapter II: Review of the Literature

This chapter reviews the existing works on bot detection and bot type identification on Twitter. Section 1 reviews two papers that addressed bot detection techniques and the methods used in those different systems. Section 2 reviews two papers that identified bot types using machine learning techniques.

1 Twitter Bot Detection

1.1 DARPA

DARPA held a four-week competition between Feb-Mar 2015, where six participating teams competed to identify a set of influence bots on Twitter with the support of DARPA's Social Media in Strategic Communications program. [19]

1.1.1 Challenge

DARPA asked the participants to identify influence bots that supported a pro-vaccination discussion on Twitter. Dealing with this challenge meant that teams had to consider: (1) separating influence bots from other types of bots; (2) separating influence bots about that topic from those about other topics; and (3) separate influence bots about that topic that sought to spread pro-sentiment from influence bots that were either neutral or anti-vaccine in sentiment.

1.1.2 Dataset

DARPA provided a dataset from the Pacific Social Architecting Corporation, which is a research group that studies how bots and technology shape social behavior on Twitter. They focused on the usage of bots in combating misinformation online with specific attention to the anti-vaccine communities on Twitter. The data provided consisted of:

- 7,038 user accounts;
- redacted user profiles with Twitter-like format: user image, website, number of friends and followers, and user bio;
- tweets with timestamp data for each user (4,095,083 tweets in total); and
- A weekly network snapshot of (from_user, to_user, timestamp, weight) tuples. Weight was 0 if "from_user" was not following "to_user", and was 1 otherwise.

1.1.3 Bot detection approaches

The top three teams in the challenge all agreed that machine learning techniques alone were insufficient because of the lack of training data. However, a semi-automated process that included machine learning proved useful. Regardless of method specifications, the features listed below were of interest to all teams.

1.1.3.1 Features used to identify influence bots

1. Tweet syntax:

   In this category, competing teams considered:

   - If user's tweets were similar to the natural language generation program ELIZA and auto generated language.

   - Average number of hashtags, user mentions, links, special characters in tweets.

   - Moreover, teams studied the average number of retweets by the user.

   - Whether tweets are geo-labeled.

   - Percentage of tweets ending with punctuation, hashtag, or link as such tweets have a higher probability of being auto generated.

2. Tweet semantics:

In this category, competing teams considered:

- Number of posts related to vaccination.

- User's average sentiment score in vaccine-related tweets.

- Measures of contradiction in posts on vaccination-related tweets using functions such as contradiction rank.

- Positive or negative sentiment strength.

- Most frequent topics tweeted about by the user.

- Number of languages in which tweets were generated.

-Sentiment inconsistency.

3. Temporal behavior features:

In this category, competing teams considered:

- Variance in tweet sentiment over time.

- Entropy of inner-tweet time distribution.

- Predictability of tweet timing based on a transfer entropy approach.

- The duration of the longest session by a user without any short (5-10 minute) breaks.

- Average number of tweets per day.

- Percentage of dropped followers.

4. User profile features:

In this category, competing teams considered:

- If the user's profile has a photo, is it from a stock database?

- If the user's profile has an associated website, does it have a clone elsewhere?

- Is the username auto-generated?

- Number of posts, retweets, replies, mentions.

- Number of followers/followees.

- Number of devices used.

- Similarity of the user profile to known bots.

5. Network features:

In this category, competing teams considered:

- Average deviation of user sentiment scores from followers and followees.

- Degree of centrality.

- Average clustering coefficient of retweet and mention network associated with each user.

- Number of known bots followed by a user.

- Number/percentage of bots in the cluster that a user belonged to.

Some teams added more features once the challenge started and some bots had been discovered. Teams used insights from previous work in this area [2,3] to identify a small number of suspicious accounts that were then manually confirmed as bots. However, not all bots could be found using these past insights.

1.1.4 Bot Analysis Algorithms used:

The top three teams used several bot analysis algorithms as described below.

1. Hashtag co-occurrence network

Nodes represent unique hashtags, and edges between two nodes were weighted by the number of times those hashtags co-occurred. These were then used to separate users into pro- and anti-vaccine categories. Also, the proportion of tweets containing any of these hashtags resulted in a predictive feature.

2. Distance Measure

   Identifying bots by computing the cosine similarity between users and known bots.

3. Outlier detection

   Identifying bots by applying orthogonal non-negative matrix factorization (NMF) to the data features, find low-dimensional vector representation of each user. Then use a clustering based outlier detection to fine outliers in this low-dimension latent space. Next, perform micro-level clustering by using the same feature representation and reapplying NMF again to cluster outliers.

1.1.5 Conclusion

The takeaway from the DARPA challenge is that a bot-detection system needs to be semi-supervised. Utilizing the available data and machine learning techniques could help ease the automation process of bot detection as bots become more sophisticated.

1.2 BOT OR NOT

Bot or Not is a publicly available service since 2014 that uses more than one thousand features to evaluate the extent to which a Twitter account exhibits similarities to the known characteristics of social bots. [21] BotOrNot takes a Twitter screen name, retrieves the account's recent activity, then computes and returns a bot-likelihood score.

1.2.1 BotOrNot Service

The service can be used in one of two ways. Either the user checks an account's bot-likelihood or generates a report about one's following and followees. In both cases, a user must have a Twitter account in order for BotOrNot to make inquires to Twitter's REST API on their behalf. The server then computes the score using the classification algorithm. While BotOrNot does not collect data about users submitting the requests, they store computed classification results for future use.

1.2.2 Classification System

BotOrNot's classification system generates over a thousand features using available metadata that is extracted from interaction patterns and content. Those features are represented in the following classes:

Network features

Building networks based on retweets, mentions, and hashtag co-occurrence then extract their statistical features. Allowing them to capture various dimensions of information diffusion patterns.

User features

These features are based on Twitter account metadata such as language, geographic location, and account creation time.

Friends features

Such features include descriptive statistics relative to an account's social contacts, such as the median, moments, and entropy of the distributions of their number of followers, followees, posts, ... etc.

Temporal features

Capturing timing patterns of content generation and consumption such as tweet rate and inner-tweet time distribution.

Content features

These features are based on linguistic cues computed through natural language processing, especially part-of-speech tagging.

Sentiment features

Using general purpose and Twitter specific sentiment analysis algorithms like happiness, arousal-dominance-valence, and emotion scores.

1.2.3 How classification works

The model is trained with instances of social bots and human classes. BotOrNot used the Twitter Search API to collect 200 of their most recent tweets and 100 of the most recent tweets mentioning them. This data collection method yielded 15k manually verified social bots and 16k human accounts. The dataset consisting of 5.6 M tweets was then used to train models and benchmark classification performance.

BotOrNot's classifier used a Random Forest algorithm, which is an ensemble supervised learning method. Extracted features were used to train seven classifiers, one for each of the subclass features and one for the overall score.

1.2.4 Conclusion

BotOrNot's authors offer a free service that takes a Twitter screen name, collects the account's recent activity, then returns a bot-likelihood score as a response. Their supervised learning method uses a random forest algorithm that leverages extracted features to train a classifier for each feature subclass and one for the overall score.

2 Identifying Bot Types

2.1 Who is Tweeting on Twitter: Human, Bot, or Cyborg?

Chu Et al. [18] talk about how the growing user base and the openness of Twitter made it an ideal target for automated accounts. Legitimate bots are bots that follow Twitter's rules and are usually used to deliver news, while malicious bots spread spam and malicious content. They also mention a new type of account that exists between bots and human, which the authors refer to as cyborg. A cyborg is either a bot-assisted human or a human-assisted bot account. They collected data and characterized the differences between human, bots, and cyborgs in terms of tweeting behavior, tweet content, and account properties. The data was then used in classification described in further detail below.

2.1.1 Data collection

They used two methods of data collection to cover 512,407 users. The first method is Depth-First Search (DFS) based crawling, which is fast and uniformed for traversing a network and is reliable for network locality and clustering. The second method was using the public timeline (Twitter API) to collect information about active users which diversifies the user pool set.

To be able to classify automatically, the authors had to create a manually labeled dataset for training and testing. The training set contained 1000 users per class of human, bot, and cyborg, thus in total 3000 labeled to serve as ground truth and a test set of 3000 users created in the same way.

2.1.2 Classification

Their classification system consists of several components:

- Entropy Component:

  Uses corrected conditional entropy to detect periodic or regular timing of messages posted by a Twitter user.

- Machine Learning Component:

  Uses a variant of Bayesian classification to detect text patterns of known spam on Twitter.

- Account Properties Component

  Uses account related properties to detect bot deviation from normal human distribution.

- Decision Maker Component

Uses Linear Discriminant Analysis (LDA) to analyze the features identified by the other components and produces a decision class: human, cyborg, or bot.

2.1.3 Evaluation

		Classified			**Total**	**True Pos. %**
		Human	Cyborg	Bot		
Actual	Human	949	51	0	1000	94.9%
	Cyborg	98	828	74	1000	82.8%
	Bot	0	63	937	1000	93.7%

Table 1: Confusion Matrix

According to the results displayed in the confusion matrix, BotOrNot's classification system can accurately differentiate human accounts from bot accounts. However, it is more difficult to distinguish between cyborg from human or bot.

2.1.4 Conclusion

The authors of this paper created a system that identifies accounts as the following types: human, cyborg, bot. They collected one month's worth of data with over 500,000 users and more than 40M tweets. Based on the data, they identified features to differentiate the three types of accounts, designed an automated classification system with 4 components, and then evaluated the effectiveness of the classification through their test dataset.

2.2 Seven Months with the Devils: A Long-Term Study of Content Polluters on Twitter

Lee Et al. [16] deployed 60 honeypots on Twitter, which resulted in a harvest of 36,000 candidate content polluters within a 7-month period. They examined the harvested users' behavior over time, followers/following network dynamics, and evaluated a wide range of features to investigate the effectiveness of their automatic content polluter identification system.

2.2.1 Social Honeypots

The authors designed and deployed 60 social honeypot accounts on Twitter, whose purpose was to pose as Twitter users, and reported back what accounts follow or otherwise interact with them. The honeypots only interacted with each other until an external user initiated contact. Afterwards, the social honeypot passed the external user's information to the observation system that keeps track of all users discovered by the system.

2.2.2 Harvested Users

The overall goal of their research was to automatically detect content polluters via social honeypots. So, to understand those users, they manually investigated them using Expectation Maximization (EM) cluster analysis algorithm. EM allowed for the grouping of harvested users with similar appearances and/or behavior by examining the following features: followers and following, tweeting activity, and behavior over time.

2.2.3 Classification

The authors used the Weka machine learning toolkit to test 30 classification algorithms. These included, but were not limited, to: Naïve Bayes, logistic regression, support vector machine, and tree based algorithms. They used a dataset of content polluters extracted by the honeypots and legitimate users sampled from Twitter. Using 15 features to differentiate between content polluters and legitimate users, they found that the Random Forrest classification algorithm produced highest accuracy (98.42%).

2.2.4 Conclusion

The authors designed a system for automatically detecting and profiling content polluters on Twitter, and subsequently evaluated its merits. They were able to study content polluters and isolate the distinguishing features in their behavior, which lead to developing their classifier.

3 Summary

In this chapter, we reviewed 4 publications that relate to bot detection, bot behavior, and bot types on Twitter. DARPA's report summarizes the features of influential bots and explains their reasoning through a suggestion of semi-supervised machine learning methods to automate bot detection. Davis et al. [21] provides a service to evaluate whether a Twitter account is controlled by a bot using information extracted from interaction patterns and content from users. These two papers gave us a comprehensive overview of the features as well as the structure of bot accounts, which lead us to a better understanding of bots.

Meanwhile, Chu et al. [18] and Lee et al. [16] use extracted features to identify certain types of bots and classifying them into groups based on similar behavior.

Previous work has detected bots with high accuracy, but they do have limitations. Bots are now more sophisticated and classifying an account to a bot or not label is no longer acceptable because of the rise of bot accounts that are routinely turned on and off.

A solution to that is to investigate bot-like behavior and to examine features to show the likelihood of an account to be a bot. Also, if an account exhibits bot-like behavior, then we should be able to classify it to a bot type. There are many types of bots on Twitter and knowing the features that an account exhibits would allow us to group it with similar bot types.

Chapter III: Bot-Like Behavior

In this chapter, we define 19 features used to detect bot-like behavior. Then we explain our experiment in detecting bot-like behavior in a dataset that we've combined from multiple sources. We also performed a logistic regression to calculate which features matter most to each bot type. In section 1 we give an overview of Twitter, bots, and previous work. In section 2, we explain our dataset collection methods. In section 3, describe our methods. In section 4, we illustrate our analysis and conclusion.

1 Introduction

Social networks are online platforms used to connect users with others that share their interests, or to create and maintain interpersonal relationships. [9] Twitter is a social network that allows users to broadcast 280-characters long posts to users that they're connected to on the platform. [8] Statista reports that Twitter has 330 M active users as of April 2018 which puts it in the top 15 most popular social networks worldwide. [10] Unfortunately, a lot of accounts on Twitter are automated. Having an account that is controlled by a software, automatically producing content, and interacting with other accounts is called a 'bot'. Although Twitter allows identified bots to run accounts that clearly state that they are automated [13], the platform houses a huge amount of unidentified bot accounts. These accounts are designed to mimic human behavior online to pursue certain goals such as: increasing accounts' popularity by having fake followers, [17] spreading information, and influencing targets.

The DARPA Twitter bot challenge in 2016 shed some light on the malicious activities bots are involved in on the platform. [19] There was a need to identify "influence bots" which are automated accounts that illicitly shape discussions before they get too influential. Participant teams in the challenge created bot detection systems that were all semi-supervised and used human judgement to augment automated bot identification processes. Davis Et al. [21], created a platform that evaluates a given account's likelihood to be a bot by comparing similarities to known characteristics of social bots then returning a percentage score per username. Dickerson Et al. [11], explains that using sentiment analysis and linguistic features studies to distinguish between humans and bots on Twitter.

Previous work on this topic focuses on bot detection, where it is important to note that bot accounts' behavior has "evolved" over time. With the growth in programming techniques, these users try to hide that their accounts are automated by turning the bot on and off. When the bot is turned off, a human would post using the same account which makes it harder to detect. In this chapter, I refer to a previous collaboration [4] where a colleague and I defined bot-like behavior criteria. We used features from an article published by Nimmo [14], and some others that we added as the study developed. Unlike other approaches that try to predict whether an account is a bot or not based on holdout data [20], we use a statistical approach that aims to provide explanatory insight into why our assignment is made.

2 Data collection

Twitter provides developers and researchers with API functions that support public user accounts information collection. We used a known bots list [15] as our source, then crawled the most recent 200 posts from users on that list. The dataset consists of 4 types of Twitter bots: Fake Followers, Traditional Spam Bots, Social Spam Bots and Content

Polluters. We then collected the most recent 200 posts from a list of legitimate users from the same source.

Fake Followers Bots	Accounts that inflate number of followers of another account. [15]
Traditional Spam Bots	Accounts that were spamming job offers. [15]
Social Spam Bots	Accounts that spam products on sale at Amazon or spammers of paid apps for mobile devices. [15]
Content Polluters	Accounts designed to generate spam while masquerading as humans. [16]

Table 2. Description of different bot data sets.

3 Methods

Feature name	**Explanation**
digit_screen_name	screen_name consists of digits only
scramble_name	screen_name consists of alpha numeric scrambles
default_profile_image	using default profile image
default_background_image	using default background image
url_shortner	using url shorteners in tweet content
low_post_high_result	retweet count or like count is more than number of followers for given account
multi_language	more than 2 languages appeared in tweets crawled

tweet_frequency	average daily tweet number
time_range	average days between two consecutive tweets
rt_number	number of retweets/ total tweets crawled
#of_mentions	average number of mentions in original tweets crawled for this account
#of_hyperlinks	average number of hyperlinks in original tweets crawled for this account
#of_friends	number of friends
#of_followers	number of followers
status_num	number of tweets
#of_favorites	number of favorited tweets
most_recent_time	most recent tweet timestamp
tweet_avg_word_number	average number of words in each original tweet
tweet_lexical_diversity	number of unique words used in all crawled original tweets

Table 3. Features used for bot-like-behavior detection.

In this project, we designed a study to describe a list of user behaviors found in Twitter accounts. We created a program that detects 19 features indicating bot-like behavior [Table 3]. When running the script on our dataset, it generates results for each user against the criteria. Applying a stepwise logistic regression model based on Akaike

Information Criteria (AIC) values to those results determined which of the 19 features were relevant when detecting bot-like behavior to each bot type [figure 1].

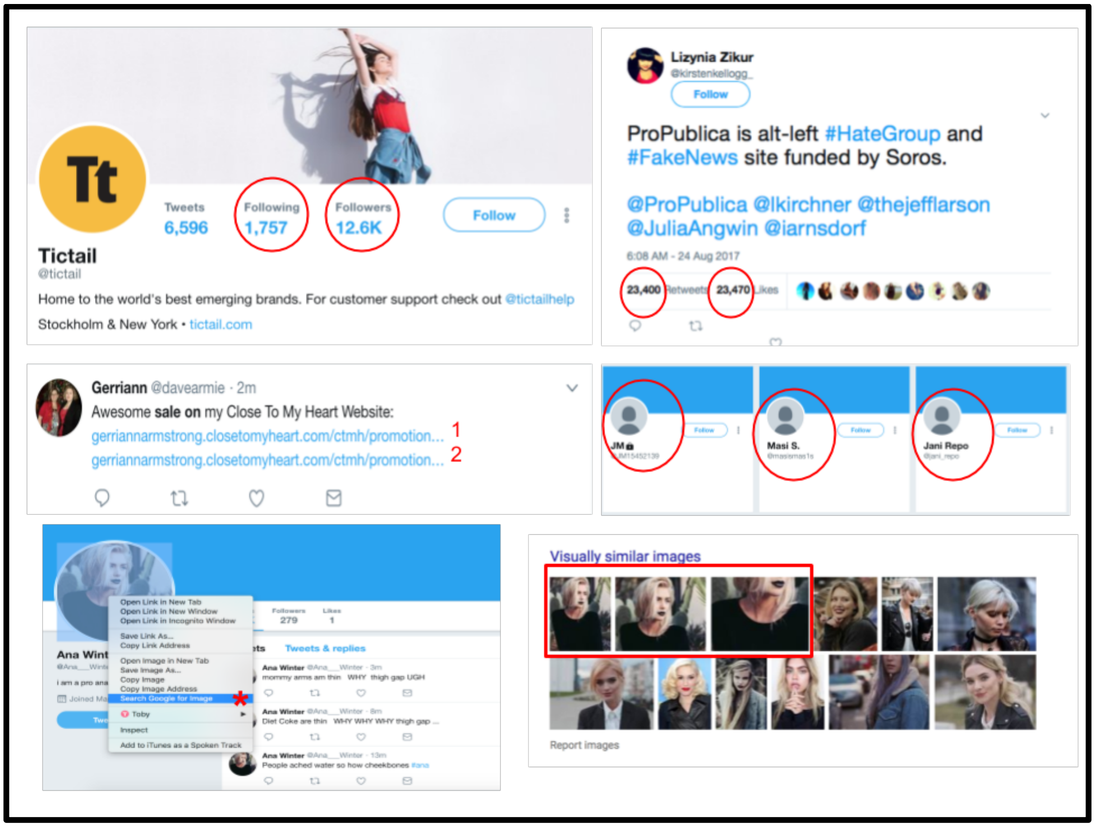


Figure 1: Visual captures of some bot-like behavior features.

4 Analysis and Conclusion

Testing our script on the dataset from the four bot types previously collected indicated that there are features that matter more to one type more than others. We used the cut off value $|z| = 2$ as a threshold to extract features which are more relevant to the model. The chosen cut off value means that we have a two-sided hypothesis test with significance level of 0.05. Having a big magnitude of z-score indicates that the true regression coefficient is not 0, therefore that feature matters, thus we were able to describe a series of bot-like behaviors.

We found two common features among all bot types which were “most_recent_time” and “status_num” with negative and positive z values respectively. It means that the user is most-likely not a bot if the account was active recently, and a user with a high number of tweets is more likely to be a bot.

Feature name	**Fake followers**		**Content Polluters**		**Traditional spam**		**Social spam**	
	z	P>|z|	z	P>|z|	z	P>|z|	z	P>|z|
status_num	4.64	0	13.555	0	3.367	0.001	2.71	0.007
tweet_frequency	4.541	0	13.598	0	-4.378	0	-4.986	0
#of_friends	3.409	0.001	5.242	0	4.239	0	-	-
avg_word_number	-2.239	0.025	-	-	-	-	-	-
multi_language	-2.732	0.006	-	-	-	-	-	-
most_recent_time	-3.177	0.001	-5.005	0	-5.316	0	-8.441	0
scramble_name	-	-	-2.994	0.003	4.03	0	-3.55	0
rt_number	-	-	-4.51	0	-3.74	0	-	-
#of_favorites	-	-	3.648	0	-	-	-	-

url_shortner	-	-	-	-	-2.379	0.017	-	-
avg_time_btw_status	-	-	-	-	-3.263	0.001	-	-
#of_hyperlinks	-	-	-	-	-4.098	0	2.93	0.003
#of_followers	-	-	-	-	-	-	4.383	0
default_background_image	-	-	-	-	-	-	-2.147	0.032
low_post_high_result	-	-	-	-	-	-	-2.579	0.01
#of_mentions	-	-	-	-	-	-	-2.797	0.005

Table 4. Features relevant to bot types.

Based on Table 4, we can associate each bot type with features more significant to it. Our results showed that fake follower bots do not tweet frequently but they have a significant number of friends which aligns with their purpose, increasing accounts' popularity. On the other hand, content polluters have a high average number of tweets per day, and a significant amount of friends which is consistent with spam accounts behavior: increasing outreach. Additionaly, the analysis of traditional spam bots' behavior shows that they rarely post retweeted content and that the average time between two posts is short. This behavior is true to the type of bots and reflects on the data set used. Finally, we noticed that social spam bots do not engage in conversations with other users and post multiple hyperlinks in the same tweet. These observations are consistent with the content provided by the source.

In conclusion, the results presented by this study demonstrate that bot-like behavior differs significantly with bot design. Particularly, one can infer the purpose of the bot creation by exploring the features presented in that account's history.

Chapter IV: Multiclass classification

In this chapter, we address our classification problem in detail. In Section 1 we introduce our idea. In section 2, we define multiclass classification and how a one-vs-all classifier works. In section 3, we introduce the MaxEnt classifier. In section 4, we describe out data collection method. In section 5, we explain how our methods. In section 6, we analyze the results. Finally, in section 7, we summarize the method and give our conclusion.

1 Introduction

Following the completion of the bot-like-behavior project and noticing that bot types in our dataset were associated with certain behavior, we discussed training a multiclass classifier to detect different bot types. Since the existing dataset is labeled, and we had the logistic regression showing the related features, we decided to use a Maximum Entropy classifier for this task.

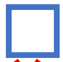
2 Multiclass classifier

Multiclass classification is constructing a function which, when given a new data-point, would correctly predict the class to which the new point belongs, conditioned that the number of classes available is more than two. A Multiclass classifier is trained using labeled data-points where each one belongs to one of *N* different classes that enables correct prediction afterwards. [7]

2.1 One-versus-all

Supposing that we have a classifier sorting input data into 3 categories [6]:

- Class 1: 

- Class 2: 
- Class 3: 

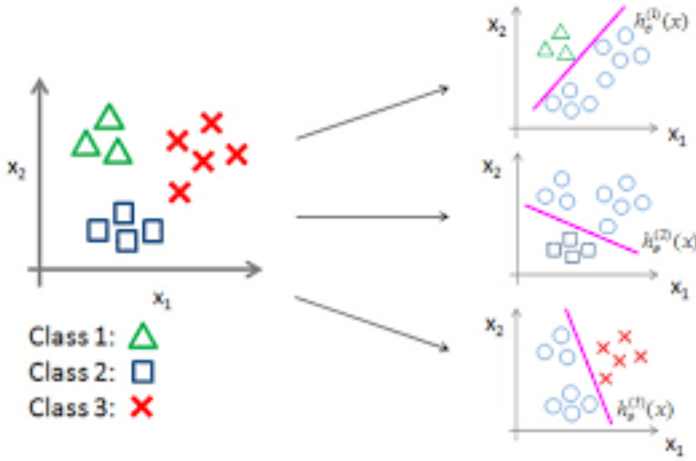


Figure 2: Multiclass classification to binary classification. [6]

We can easily turn this into a binary classification problem where we only predict if class A ∈ {0,1} by taking values of one class and turning them into positive examples and the remaining classes into negative examples and we run the classifier 3 times calculating $h_\theta(x)$ for each class which results in 3 fit classifiers:

$$h_\theta^{(i)} = P(y = i|x; \theta), i = 1,2,3$$

Since we now have calculated the vector: $h_\theta\ (x) = [h_\theta^{(1)}(x), h_\theta^{(2)}(x), h_\theta^{(3)}(x)]$, all that is left to complete the prediction is calculate the maximal value which will give the predicted class: $\max_i h_\theta^{(i)}(x)$.

3 Max Entropy Classifier

The Max Entropy classifier is a probabilistic classifier for multinomial cases. It is also considered a generalization of the logistic regression for multiclass problems. In both models, we want a conditional probability where $p(y|x)$ in which y is the target class and x is a vector of features.

4 Data Collection

Since this project is based on the bot-like behavior project, we used the same dataset for training to keep it unified. We used the output of the bot-like behavior project where we had results for each user against the 19-bot-like behavior criteria as our training set. The dataset consists of 5 types of Twitter bots and 1 human, types were labeled as follows: Fake Followers, Traditional Spam Bots, Social Spam Bots, Content Polluters, Varol Spam Bots, and Human.

Our testing dataset comprised of tweets collected using the Twitter API in late 2017. Those tweets included the words "vaxx|vacc" in their text. In order to have our training and testing datasets aligned and similar, we checked for ids that are still active in our test dataset and crawled their most recent 200 tweets. We then ran the script from the previous project on the test dataset which generated the results for each user against the 19-bot-like behavior criteria. The results were used as our test data.

5 Methods

In this section, we describe how we created a MaxEnt multiclass classifier by explaining data preprocessing, classifier training, and classifier testing.

5.1 Data Preprocessing

As mentioned above, datasets used in training and testing had similar structure. The files contained rows each one with: user id, tweet id, tweet text, other information from that user's profile and how that user evaluates against the 19 bot-like-behavior criteria. Aggregating all bot types rows into one file for the training set resulted in having one column missing for it to be complete, which is the bot type label for each row. To keep track, we used a list that had each user id and associated label. In order to use that file, we needed to deduplicate some of the ids, by eliminating user ids that were repeated. If the user id had different labels, we would give it the label of the majority class. We were then able to use the file with user id labels to fill the missing column and our training dataset was complete.

5.2 Classifier Training

After preprocessing the training dataset, we used sklearn [1] library to train a MaxEnt classifier. The maximum entropy classifier is also known as a logistic regression classifier with multi-classes which using supervised learning, converts labeled feature sets to vectors using encoding. The encoded vector is then used to calculate weights for each feature that can be combined to determine the most likely label for a feature set. Our training dataset statistics were as follows:

	Number of Rows	**Percentage**
Total records	1,580,999	
content_polluters	491,290	31.07%
fake_followers	41,204	2.61%
traditional_spambots	165,058	10.44%
social_spambots	456,478	28.87%
varol_spambots	139,967	8.85%
human	287,002	18.15%

Table 5: training dataset statistics

5.3 Classifier Testing

To test the classifier, we used a dataset with similar structure as the training set but the file did not include the label's column. When testing, we perform the following steps:

- Loading test data.
- Loading vectorizer.
- Vectorize the text.
- Load the classifier.
- Check the accuracy on test data.
- Calculate precision, recall, and f-measure.
- Output confusion matrix.

The test dataset statistics were as follows:

	Number of Rows	**Percentage**
Total records	102,230	
content_polluters	66,586	65.13%
fake_followers	305	0.30%
traditional_spambots	3,591	3.51%
social_spambots	18,806	18.40%
varol_spambots	12,942	12.66%
human	0	0.00%

Table 6: test dataset statistics

6 Analysis

After performing multiple tests on the maximum entropy (MaxEnt) classifier, it has shown that the best performance occurred when running the default settings. Meaning, that classes were not balanced, and performing one-vs-all classification.

Accuracy on test set	**45.16%**		
Label	Precision	Recall	f-measure

content_polluters	33.76%	35.66%	34.68%
fake_followers	25.65%	45.57%	32.82%
traditional_spambots	65.44%	38.18%	48.22%
social_spambots	54.19%	76.10%	63.30%
varol_spambots	60.35%	11.91%	19.90%
human	0.00%	0.00%	0.00%

Table 7: Testing MaxEnt one-vs-all classifier

The highest accuracy from 5 different Max Entropy models achieved was 45.16%. It shows that the classifier learned most about social spam bots as that type is best detected. The confusion matrix #1 in Appendix A, shows that 14311 spam bots from the test dataset were correctly identified.

7 Holdout Dataset

While planning this project we considered having a holdout dataset to analyze using the multiclass classifier. The dataset consisted of 10,000 tweets with the keywords “vacc” or “vaxx” in their text, and collected between Nov 11, 2014 and Oct 11, 2017. The dataset was previously annotated as pro-vaccine, anti-vaccine, and neutral. Before we could run the dataset to be analyzed, we collected the most recent 200 tweets from ids that were in the dataset to keep the study structure unified with the classifier input. We then provided the classifier with the input required to classify the 10,000 user ids into bot-types. The results are presented in table x.

	Anti-Vaccine	**Pro-Vaccine**	**Neutral**
content_polluters	21.41%	15.55%	63.04%
fake_followers	28.26%	19.57%	52.17%

traditional_spambots	22.99%	12.86%	64.16%
social_spambots	16.61%	17.08%	66.30%
varol_spambots	22.41%	13.55%	64.03%
human	27.44%	16.06%	56.50%

Table 8: Classifying bots polarity in vaccine topics.

The results show that fake followers bots tweet more than humans in anti-vaccine topics. It is also shown that all bot types excluding social spam bots post more anti-vaccine content than pro-vaccine.

8 Future Work

Using a supervised learning algorithm as MaxEnt for Twitter account type classification has demonstrated reasonable performance considering having 6 classes. Some areas of this thesis remain for future work. The results indicate that maximum entropy may be sensitive to limited feature selection. Increasing features and data points could enhance learning.

Chapter V: Conclusion

In this thesis, we study the problem of classifying the different types of bots on Twitter. First, we reviewed related work. We selected the following publications: The DARPA Challenge, [19] and Davis et. al., [21] which gave us a comprehensive overview of bot account features. Meanwhile, we reviewed the publications by Chu et. al., [18] and Lee et. al. [16] that use techniques to identify specific types of bot accounts. Although the existing methods these papers presented high accuracy in detection, they all study the problem in bot identification.

In our research, we target studying malicious accounts' behavior as means to discover bot-like behavior. We focus on finding features and behavior in an account's recent activity. By running our script to detect the handpicked criteria and generating the report, we are able to identify bot-like behavior accounts and can identify what type of bots too.

In chapter IV, we train an sklearn MaxEnt classifier using the features of each bot type that we have learned previously. We are able to create a multiclass classifier that performs with 45% accuracy when classifying into one of 6 classes. By analyzing the confusion matrix, we are effectively detecting social spam bots. The summary of our findings is referred to in the confusion matrix #1 in Appendix A.

Bibliography

[1] F. Pedregosa *et al*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research,* vol. 12, *(Oct),* pp. 2825-2830, 2011.

[2] S. Lehmann and P. Sapieżyński, "You're here Because of a Robot," *Blog,* 2013.

[3] P. R. Gregory, "Inside putin's campaign of social media trolling and faked ukrainian crimes," *Forbes, May,* vol. 11, 2014.

[4] S. Qi, L. AlKulaib and D. Broniatowski, "Detecting and characterizing bot-like behavior on twitter," in *2018 International Conference on Social Computing, Behavioral-Cultural Modeling, & Prediction and Behavior Representation in Modeling and Simulation* July, 2018, .

[5] *ML Glossary of Terms*. Available: https://developers.google.com/machine-learning/glossary/#training_set.

[6] A. Ng, "Multiclass classification: one-vs-all," .

[7] R. Rifkin, "Multiclass Classification," "February 25th, ", 2008.

[8] (*Most popular social networks worldwide as of April 2018, ranked by number of active users (in millions)*. Available: https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/.

[9] *What Is Twitter?*. Available: https://whatis.techtarget.com/definition/Twitter.

[10] *Social Network Definition*. Available: https://www.merriam-webster.com/dictionary/social%20network.

[11] J. P. Dickerson, V. Kagan and V. S. Subrahmanian, "Using sentiment to detect bots on twitter: Are humans more opinionated than bots?" in 2014, Available: https://ieeexplore.ieee.org/document/6921650. DOI: 10.1109/ASONAM.2014.6921650.

[12] E. Ferrara *et al*, "The rise of social bots," *Communications of the ACM,* pp. 96-104, 2016. Available: http://dl.acm.org/citation.cfm?id=2818717. DOI: 10.1145/2818717.

[13] *Twitter Rules*. Available: https://help.twitter.com/en/rules-and-policies/twitter-rules.

[14] B. Nimmo, "#BotSpot: Twelve Ways to Spot a Bot," Aug, 28,. 2017.

[15] *Bot Repository*. Available: https://botometer.iuni.iu.edu/bot-repository/datasets.html.

[16] K. Lee, B. Eoff and J. Caverlee, "Seven Months with the Devils: A Long-Term Study of Content Polluters on Twitter," .

[17] N. CONFESSORE *et al*, "The Follower Factory," *New York Times,* 2018. Available: https://www.nytimes.com/interactive/2018/01/27/technology/social-media-bots.html.

[18] Z. Chu *et al*, "Who is tweeting on twitter," in Dec 6, 2010, Available: http://dl.acm.org/citation.cfm?id=1920265. DOI: 10.1145/1920261.1920265.

[19] V. S. Subrahmanian *et al*, "The DARPA Twitter Bot Challenge," *Mc,* vol. 49, *(6),* pp. 38-46, 2016. Available: http://ieeexplore.ieee.org/document/7490315. DOI: 10.1109/MC.2016.183.

[20] O. Varol *et al*, "Online Human-Bot Interactions: Detection, Estimation, and Characterization," 2017. Available: http://arxiv.org/abs/1703.03107.

[21] C. A. Davis *et al*, "BotOrNot: A System to Evaluate Social Bots," 2016.

Available: http://arxiv.org/abs/1602.00975. DOI: 10.1145/2872518.2889302.

Appendix A – Confusion Matrices of MaxEnt models

Table A.1 Confusion Matrix of MaxEnt: liblinear.

Using the liblinear library from sklearn which uses a coordinate descent algorithm (CD). The CD algorithm implemented in liblinear cannot learn a true multinomial model; instead the optimization problem is decomposed in a "one-vs-all" fashion.

	content_polluters	fake_followers	human	social_spambots	traditional_spambots	varol_spambots
content_polluters	4749	74	1277	6473	239	505
fake_followers	105	139	7	13	28	13
human	0	0	0	0	0	0
social_spambots	3569	6	158	14311	325	437
traditional_spambots	1779	45	40	298	1371	58
varol_spambots	3866	278	1808	5316	132	1542

Table A.2 Confusion Matrix of MaxEnt: liblinear, scaled.

Using the liblinear library from sklearn which uses a coordinate descent algorithm (CD). The CD algorithm implemented in liblinear cannot learn a true multinomial model; instead the optimization problem is decomposed in a “one-vs-all” fashion.

Scaled is using sklearn preprocessing library which scales each feature by its maximum absolute value.

	content_polluters	fake_followers	human	social_spambots	traditional_spambots	varol_spambots
content_polluters	11017	124	864	159	660	493
fake_followers	87	189	5	1	20	3
human	0	0	0	0	0	0
social_spambots	10839	393	149	4950	332	2143
traditional_spambots	1400	39	25	4	2097	26
varol_spambots	7970	96	1432	28	377	3039

Table A.3 Confusion Matrix of MaxEnt: liblinear, scaled, and balanced.

Using the liblinear library from sklearn which uses a coordinate descent algorithm (CD). The CD algorithm implemented in liblinear cannot learn a true multinomial model; instead the optimization problem is decomposed in a "one-vs-all" fashion.

Scaled means using sklearn preprocessing library which scales each feature by its maximum absolute value.

Balanced mode uses the values of y to automatically adjust weights inversely proportional to class frequencies in the input data as:

$$n\_samples \ / \ (n\_classes \ * \ np.bincount(y)).$$

	content_polluters	fake_followers	human	social_spambots	traditional_spambots	varol_spambots
content_polluters	9989	513	914	109	932	860
fake_followers	36	243	4	1	18	3
human	0	0	0	0	0	0
social_spambots	9047	1648	177	4132	887	2915
traditional_spambots	1145	69	15	4	2296	62
varol_spambots	6880	216	1489	19	579	3759

Table A.4 Confusion Matrix of MaxEnt: sag, scaled.

Using the “sag” solver uses a Stochastic Average Gradient descent. It is faster than other solvers for large datasets, when both the number of samples and the number of features are large.

Scaled means using sklearn preprocessing library which scales each feature by its maximum absolute value.

	content_polluters	fake_followers	human	social_spambots	traditional_spambots	varol_spambots
content_polluters	11017	124	863	160	660	493
fake_followers	87	189	5	1	20	3
human	0	0	0	0	0	0
social_spambots	10837	393	149	4951	332	2144
traditional_spambots	1400	39	25	4	2097	26
varol_spambots	7970	96	1431	28	377	3040

Table A.5 Confusion Matrix of MaxEnt: sag, scaled, and balanced.

Using the “sag” solver uses a Stochastic Average Gradient descent. It is faster than other solvers for large datasets, when both the number of samples and the number of features are large.

Scaled means using sklearn preprocessing library which scales each feature by its maximum absolute value.

Balanced mode uses the values of y to automatically adjust weights inversely proportional to class frequencies in the input data as:

$$n\_samples \;/\; (n\_classes \; * \; np.bincount(y)).$$

	content_polluters	fake_followers	human	social_spambots	traditional_spambots	varol_spambots
content_polluters	9752	657	1039	107	1008	754
fake_followers	40	235	5	1	22	2
human	0	0	0	0	0	0
social_spambots	8403	2454	188	4318	1103	2340
traditional_spambots	1131	66	22	2	2327	43
varol_spambots	6621	238	1588	24	707	3764

Appendix B – Holdout Dataset Results

Table B.1 Raw counts for 10K dataset.

Table B.2 % results for 8K dataset.

Table B.3 Raw counts for 8K dataset.

Table B.4 STD deviation of botscores for 8K dataset.

Table B.1 Raw counts for 10K dataset.

	Anti-Vaccine	Pro-Vaccine	Neutral
content_polluters	1273	925	3749
fake_followers	13	9	24
traditional_spambots	177	99	494
social_spambots	106	109	423
varol_spambots	210	127	600
human	456	267	939

Table B.2 % results for 8K dataset.

Some users in the 10K dataset were no longer active on twitter, so we had to reduce the dataset to include only active users ~8K.

	Anti-Vaccine	Pro-Vaccine	Neutral
content_polluters	21.22%	15.74%	63.04%
fake_followers	31.25%	25.00%	43.75%
traditional_spambots	16.33%	10.20%	73.47%
social_spambots	16.72%	17.19%	66.09%
varol_spambots	25.17%	11.92%	62.91%
human	27.30%	15.90%	56.81%

Table B.3 Raw counts for 8K dataset.

	Anti-Vaccine	Pro-Vaccine	Neutral
content_polluters	1033	766	3068
fake_followers	10	8	14
traditional_spambots	8	5	36
social_spambots	106	109	419
varol_spambots	114	54	285
human	419	244	872

Table B.4 STD deviation of bot scores for 8K dataset.

	universal		english	
	mean	std dev	mean	std dev
content_polluters	0.417	0.183	0.474	0.168
fake_followers	0.337	0.093	0.363	0.091
traditional_spambots	0.500	0.166	0.536	0.178
social_spambots	0.258	0.182	0.304	0.174
varol_spambots	0.460	0.196	0.504	0.161
human	0.308	0.140	0.354	0.128

Appendix C – Authorization to use previous work

Attachment C.1: License Number 4345451408096

Attachment C.2: SBP Author's rights

Attachment C.3: Authorization from Prof. Andrew Ng to use ML Course content

Attachment C.1: License Number 4345451408096

ASSOCIATION FOR COMPUTING MACHINERY, INC. LICENSE TERMS AND CONDITIONS

May 10, 2018

This Agreement between Ms. Lulwah AlKulaib ("You") and Association for Computing Machinery, Inc. ("Association for Computing Machinery, Inc.") consists of your license details and the terms and conditions provided by Association for Computing Machinery, Inc. and Copyright Clearance Center.

All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.

License Number	4345451408096
License date	May 10, 2018
Licensed Content Publisher	Association for Computing Machinery, Inc.
Licensed Content Publication	Proceedings
Licensed Content Title	Who is tweeting on Twitter: human, bot, or cyborg?
Licensed Content Author	Zi Chu, et al
Licensed Content Date	Dec 6, 2010
Type of Use	Thesis/Dissertation
Requestor type	Academic
Format	Print and electronic
Portion	figure/table
Number of figures/tables	1
Will you be translating?	No
Order reference number	
Title of your thesis/dissertation	Twitter Bots Multiclass Classification Using Bot-Like Behavior Features
Expected completion date	May 2018
Estimated size (pages)	60
Requestor Location	Ms. Lulwah AlKulaib [redacted]
Billing Type	
Credit card info	
Credit card expiration	
Total	
Terms and Conditions	

Rightslink Terms and Conditions for ACM Material

1. The publisher of this copyrighted material is Association for Computing Machinery, Inc. (ACM). By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the Billing and Payment terms and conditions established by Copyright Clearance Center, Inc. ("CCC"), at the time that you opened your Rightslink account and that are available at any time at).

2. ACM reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.
3. ACM hereby grants to licensee a non-exclusive license to use or republish this ACM-copyrighted material* in secondary works (especially for commercial distribution) with the stipulation that consent of the lead author has been obtained independently. Unless otherwise stipulated in a license, grants are for one-time use in a single edition of the work, only with a maximum distribution equal to the number that you identified in the licensing process. Any additional form of republication must be specified according to the terms included at the time of licensing.
*Please note that ACM cannot grant republication or distribution licenses for embedded third-party material. You must confirm the ownership of figures, drawings and artwork prior to use.
4. Any form of republication or redistribution must be used within 180 days from the date stated on the license and any electronic posting is limited to a period of six months unless an extended term is selected during the licensing process. Separate subsidiary and subsequent republication licenses must be purchased to redistribute copyrighted material on an extranet. These licenses may be exercised anywhere in the world.
5. Licensee may not alter or modify the material in any manner (except that you may use, within the scope of the license granted, one or more excerpts from the copyrighted material, provided that the process of excerpting does not alter the meaning of the material or in any way reflect negatively on the publisher or any writer of the material).
6. Licensee must include the following copyright and permission notice in connection with any reproduction of the licensed material: "[Citation] © YEAR Association for Computing Machinery, Inc. Reprinted by permission." Include the article DOI as a link to the definitive version in the ACM Digital Library. Example: Charles, L. "How to Improve Digital Rights Management," Communications of the ACM, Vol. 51:12, © 2008 ACM, Inc. http://doi.acm.org/10.1145/nnnnnn.nnnnnn (where nnnnnn.nnnnnn is replaced by the actual number).
7. Translation of the material in any language requires an explicit license identified during the licensing process. Due to the error-prone nature of language translations, Licensee must include the following copyright and permission notice and disclaimer in connection with any reproduction of the licensed material in translation: "This translation is a derivative of ACM-copyrighted material. ACM did not prepare this translation and does not guarantee that it is an accurate copy of the originally published work. The original intellectual property contained in this work remains the property of ACM."
8. You may exercise the rights licensed immediately upon issuance of the license at the end of the licensing transaction, provided that you have disclosed complete and accurate details of your proposed use. No license is finally effective unless and until full payment is received from you (either by CCC or ACM) as provided in CCC's Billing and Payment terms and conditions.
9. If full payment is not received within 90 days from the grant of license transaction, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment terms and conditions, the license is automatically revoked and shall be void as if never granted.
10. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and publisher reserves the right to take any and all action to protect its copyright in the materials.
11. ACM makes no representations or warranties with respect to the licensed material and adopts on its own behalf the limitations and disclaimers established by CCC on its behalf in its Billing and Payment terms and conditions for this licensing transaction.
12. You hereby indemnify and agree to hold harmless ACM and CCC, and their respective officers, directors, employees and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.
13. This license is personal to the requestor and may not be sublicensed, assigned, or transferred by you to any other person without publisher's written permission.

14. This license may not be amended except in a writing signed by both parties (or, in the case of ACM, by CCC on its behalf).
15. ACM hereby objects to any terms contained in any purchase order, acknowledgment, check endorsement or other writing prepared by you, which terms are inconsistent with these terms and conditions or CCC's Billing and Payment terms and conditions. These terms and conditions, together with CCC's Billing and Payment terms and conditions (which are incorporated herein), comprise the entire agreement between you and ACM (and CCC) concerning this licensing transaction. In the event of any conflict between your obligations established by these terms and conditions and those established by CCC's Billing and Payment terms and conditions, these terms and conditions shall control.
16. This license transaction shall be governed by and construed in accordance with the laws of New York State. You hereby agree to submit to the jurisdiction of the federal and state courts located in New York for purposes of resolving any disputes that may arise in connection with this licensing transaction.
17. There are additional terms and conditions, established by Copyright Clearance Center, Inc. ("CCC") as the administrator of this licensing service that relate to billing and payment for licenses provided through this service. Those terms and conditions apply to each transaction as if they were restated here. As a user of this service, you agreed to those terms and conditions at the time that you established your account, and you may see them again at any time at http://myaccount.copyright.com
18. Thesis/Dissertation: This type of use requires only the minimum administrative fee. It is not a fee for permission. Further reuse of ACM content, by ProQuest/UMI or other document delivery providers, or in republication requires a separate permission license and fee. Commercial resellers of your dissertation containing this article must acquire a separate license.
Special Terms:

Questions? customercare@copyright.com or +1-855-239-3415 (toll free in the US) or +1-978-646-2777.

Attachment C.2: SBP–BRIMS rights retained by Author

Consent to Publish

Lecture Notes in Computer Science

Title of the Book or Conference Name: 2018 International Conference on Social Computing, Behavioral-Cultural Modeling, & Prediction and Behavior Representation in Modeling and Simulation

Volume Editor(s) Name(s):

Title of the Contribution:

Author(s) Full Name(s): SiHua Qi . Lulwah AlKulaib, David A. Broniatowski

Corresponding Author's Name, Affiliation Address, and Email:

SiHua Qi, 2121 I St NW, Washington, DC 20052
qisihu@gwu.edu

When Author is more than one person the expression "Author" as used in this agreement will apply collectively unless otherwise indicated.

The Publisher intends to publish the Work under the imprint **Springer**. The Work may be published in the book series **Lecture Notes in Computer Science (LNCS, LNAI or LNBI)**.

§ 1 Rights Granted

Author hereby grants and assigns to **Springer International Publishing AG, Gewerbestrasse 11, 6330 Cham, Switzerland** (hereinafter called **Publisher**) the exclusive, sole, permanent, world-wide, transferable, sub-licensable and unlimited right to reproduce, publish, distribute, transmit, make available or otherwise communicate to the public, translate, publicly perform, archive, store, lease or lend and sell the Contribution or parts thereof individually or together with other works in any language, in all revisions and versions (including soft cover, book club and collected editions, anthologies, advance printing, reprints or print to order, microfilm editions, audiograms and videograms), in all forms and media of expression including in electronic form (including offline and online use, push or pull technologies, use in databases and data networks (e.g. the Internet) for display, print and storing on any and all stationary or portable end-user devices, e.g. text readers, audio, video or interactive devices, and for use in multimedia or interactive versions as well as for the display or transmission of the Contribution or parts thereof in data networks or search engines, and posting the Contribution on social media accounts closely related to the Work), in whole, in part or in abridged form, in each case as now known or developed in the future, including the right to grant further time-limited or permanent rights. Publisher especially has the right to permit others to use individual illustrations, tables or text quotations and may use the Contribution for advertising purposes. For the purposes of use in electronic forms, Publisher may adjust the Contribution to the respective form of use and include links (e.g. frames or inline-links) or otherwise combine it with other works and/or remove links or combinations with other works provided in the Contribution. For the avoidance of doubt, all provisions of this contract apply regardless of whether the Contribution and/or the Work itself constitutes a database under applicable copyright laws or not.

The copyright in the Contribution shall be vested in the name of Publisher. Author has asserted his/her right(s) to be identified as the originator of this Contribution in all editions and versions of the Work and parts thereof, published in all forms and media. Publisher may take, either in its own name or in that of Author, any necessary steps to protect the rights granted under this Agreement against infringement by third parties. It will have a copyright notice inserted into all editions of the Work according to the provisions of the Universal Copyright Convention (UCC).

The parties acknowledge that there may be no basis for claim of copyright in the United States to a Contribution prepared by an officer or employee of the United States government as part of that person's official duties. If the Contribution was performed under a United States government contract, but Author is not a United States government employee, Publisher grants the United States government royalty-free permission to reproduce all or part of the Contribution and to authorise others to do so for United States government purposes. If the Contribution was prepared or published by or under the direction or control of the Crown (i.e., the constitutional monarch of the Commonwealth realm) or any Crown government department, the copyright in the Contribution shall, subject to any agreement with Author, belong to the Crown. If Author is an officer or employee of the United States government or of the Crown, reference will be made to this status on the signature page.

16.01.2018 10:38

§ 2 Rights Retained by Author

Author retains, in addition to uses permitted by law, the right to communicate the content of the Contribution to other research colleagues, to share the Contribution with them in manuscript form, to perform or present the Contribution or to use the content for non-commercial internal and educational purposes, provided the original source of publication is cited according to the current citation standards in any printed or electronic materials. Author retains the right to republish the Contribution in any collection consisting solely of Author's own works without charge, subject to ensuring that the publication of the Publisher is properly credited and that the relevant copyright notice is repeated verbatim. Author may self-archive an author-created version of his/her Contribution on his/her own website and/or the repository of Author's department or faculty. Author may also deposit this version on his/her funder's or funder's designated repository at the funder's request or as a result of a legal obligation. He/she may not use the Publisher's PDF version, which is posted on the Publisher's platforms, for the purpose of self-archiving or deposit. Furthermore, Author may only post his/her own version, provided acknowledgment is given to the original source of publication and a link is inserted to the published article on the Publisher's website. The link must be provided by inserting the DOI number of the article in the following sentence: "The final authenticated version is available online at https://doi.org/[insert DOI]." The DOI (Digital Object Identifier) can be found at the bottom of the first page of the published paper.
Prior versions of the Contribution published on non-commercial pre-print servers like ArXiv/CoRR and HAL can remain on these servers and/or can be updated with Author's accepted version. The final published version (in pdf or html/xml format) cannot be used for this purpose. Acknowledgment needs to be given to the final publication and a link must be inserted to the published Contribution on the Publisher's website, by inserting the DOI number of the article in the following sentence: "The final authenticated publication is available online at https://doi.org/[insert DOI]".
Author retains the right to use his/her Contribution for his/her further scientific career by including the final published paper in his/her dissertation or doctoral thesis provided acknowledgment is given to the original source of publication. Author also retains the right to use, without having to pay a fee and without having to inform the Publisher, parts of the Contribution (e.g. illustrations) for inclusion in future work. Authors may publish an extended version of their proceedings paper as a journal article provided the following principles are adhered to: a) the extended version includes at least 30% new material, b) the original publication is cited, and c) it includes an explicit statement about the increment (e.g., new results, better description of materials, etc.).

§ 3 Warranties

Author agrees, at the request of Publisher, to execute all documents and do all things reasonably required by Publisher in order to confer to Publisher all rights intended to be granted under this Agreement. Author warrants that the Contribution is original except for such excerpts from copyrighted works (including illustrations, tables, animations and text quotations) as may be included with the permission of the copyright holder thereof, in which case(s) Author is required to obtain written permission to the extent necessary and to indicate the precise sources of the excerpts in the manuscript. Author is also requested to store the signed permission forms and to make them available to Publisher if required.
Author warrants that Author is entitled to grant the rights in accordance with Clause 1 "Rights Granted", that Author has not assigned such rights to third parties, that the Contribution has not heretofore been published in whole or in part, that the Contribution contains no libellous or defamatory statements and does not infringe on any copyright, trademark, patent, statutory right or proprietary right of others, including rights obtained through licences; and that Author will indemnify Publisher against any costs, expenses or damages for which Publisher may become liable as a result of any claim which, if true, would constitute a breach by Author of any of Author's representations or warranties in this Agreement.
Author agrees to amend the Contribution to remove any potential obscenity, defamation, libel, malicious falsehood or otherwise unlawful part(s) identified at any time. Any such removal or alteration shall not affect the warranty and indemnity given by Author in this Agreement.

§ 4 Delivery of Contribution and Publication

Author agrees to deliver to the responsible Volume Editor (for conferences, usually one of the Program Chairs), on a date to be agreed upon, the manuscript created according to the Publisher's Instructions for Authors. Publisher will undertake the reproduction and distribution of the Contribution at its own expense and risk. After submission of the Consent to Publish form signed by the Corresponding Author, changes of authorship, or in the order of the authors listed, will not be accepted by the Publisher.

§ 5 Author's Discount for Books and Electronic Access

Author is entitled to purchase for his/her personal use (if ordered directly from Publisher) the Work or other books published by Publisher at a discount of 40% off the list price for as long as there is a contractual arrangement between Author and Publisher and subject to applicable book price regulation.
Resale of such copies is not permitted.

§ 6 Governing Law and Jurisdiction

If any difference shall arise between Author and Publisher concerning the meaning of this Agreement or the rights and liabilities of the parties, the parties shall engage in good faith discussions to attempt to seek a mutually satisfactory resolution of the dispute. This agreement shall be governed by, and shall be construed in accordance with, the laws of Switzerland. The courts of Zug, Switzerland shall have the exclusive jurisdiction.

Corresponding Author signs for and accepts responsibility for releasing this material on behalf of any and all Co-Authors.

Signature of Corresponding Author: **Date:**

SiHua Qi .... 04/18/2018 ....

☐ I'm an employee of the US Government and transfer the rights to the extent transferable (Title 17 §105 U.S.C. applies)

☐ I'm an employee of the Crown and copyright on the Contribution belongs to the Crown

For internal use only:
Legal Entity Number: 1128 Springer International Publishing AG
Springer-C-CTP-01/2018

16.01.2018 10:38

Attachment C.3: Authorization from Prof. Andrew Ng to use ML Course content

Referring to the image used from the online course by A. Ng, "Multiclass classification: one-vs-all,"

5/14/2018 The George Washington University Mail - Authorization to use content in thesis

THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

Lulwah Alkulaib <lalkulaib@gwmail.gwu.edu>

Authorization to use content in thesis

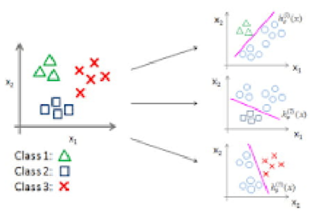
Lulwah Alkulaib <lalkulaib@gwmail.gwu.edu> Thu, May 3, 2018 at 10:50 AM
To: ang@cs.stanford.edu

Hello Prof. Andrew,

I am a MS student in Computer Science at GWU. I am in the process of publishing my thesis and have used a photo from your coursera course explaining mutliclass classifiers that I would like to have authorization to use. GW requires a written letter attached in my appendix in addition to the citation to be able to use it.

I used the attached image.

Best,
Lulwah AlKulaib



multiclassclassifier.png
9K

Lulwah Alkulaib <lalkulaib@gwmail.gwu.edu>

Authorization to use content in thesis

Andrew Y. Ng <ang@cs.stanford.edu> Mon, May 14, 2018 at 1:00 AM
To: Lulwah Alkulaib <lalkulaib@gwmail.gwu.edu>

Please feel free to do so. Best of luck!

Andrew

On Thu, May 3, 2018 at 7:50 AM, Lulwah Alkulaib <lalkulaib@gwmail.gwu.edu> wrote:

> Hello Prof. Andrew,
>
> I am a MS student in Computer Science at GWU. I am in the process of publishing my thesis and have used a photo from your coursera course explaining mutliclass classifiers that I would like to have authorization to use. GW requires a written letter attached in my appendix in addition to the citation to be able to use it.
>
> I used the attached image.
>
> Best,
> Lulwah AlKulaib