Simulating the Effect of Automated Accounts in a Social Media Environment

Aaron Johnson
University of North Dakota
Grand Forks, ND, United States
aaron.r.johnson@und.edu

Derek Popowski
University of North Dakota
Grand Forks, ND, United States
Derek.a.popowski@und.edu

***Abstract*—With the rise of social media platforms such as Twitter and Facebook no longer being used only for communications between friends and relatives, and increasingly as a primary source of information on news and major events. There is a corresponding increase in the need for ways in which to verify the credibility of the information. This paper attempts to provide some insights into some of the ways in which one could potentially use an automated 'bot' account to artificially inflate traffic to a particular source of information which may make it seem more credible than it may be in reality. We created and built a simulation software written in Java that attempts to show how this process might work, the feasibility and scale that would be required to have an impact on the users of the platform. We did however find that real world data on this topic was rather limited as so a number of assumptions were made to model this system, and future work would need to do more data gathering on the real platform to ensure that the model follows the real distributions of the data. However, we did encounter some rather interesting results that would seem realistic considering the way in which these platforms operate and that these types of automated accounts have been used with success in recent news.**

***Index Terms*—binomial distribution, Bot, exponential distribution, normal distribution, simulation, social media, Twitter, uniform distribution.**

I. INTRODUCTION

THE main goal of this project was to design and implement a system to model and simulate the use of an automated bot account on a social media platform(namely twitter) as a way to inflate the traffic to a specific source of information(another account). We assumed this is accomplished only through the use of built in functions of the platform (retweets and following of a given account). We started by designing a way to model the accounts that we we concerned with, that is the ones that we wanted to increase the influence of. To accomplish this we used an object oriented approach(implemented in Java), and created an object that would track the account that the measurable metrics that we were focusing on, namely the number of followers and the number of bot accounts that were associated with the account, also the number of views(termed Impressions) that the accounts has generated. So the basic idea was to have the simulation generate a message for each of these accounts and draw from a random stream a certain number of views for the message as a function of some of the other metric that we were tracking. This is where some of the major roadblocks were encountered, as the distributions were not clear from any previous work that were are aware of. The data that has been gathered as far as we have found, has not been used in the way in which we were proposing. This must be considered when drawing conclusions from our results as we cannot say that this simulation model exactly replicates how these bots would affect the traffic of an account on the real social network. However, we hope to show in this paper that there is real conclusions to be drawn, and that our model shows that if bots are successful in inflating traffic to a certain user it can have a cascading effect on the increase of the account's influence to the environment. We built our model around three main ideas, that the popularity of an account would vary over the course of the simulation. This is increased as the number of views an account gains for a certain increases this is done to show that as an account becomes more popular in the network it would tend to generate more random views from users who are not followers of the account. This helps to account for increase in views from searches for a topic that the account is sending messages about, or from things such as 'trending' topics that are featured on Twitter. Second that there should be some randomness induced in the messages that the accounts were sending out this is a way to simulate that each account has a random chance to send a message that is relevant to the users on the network. This would then be used to either increase , or decrease the number of unique impressions that the certain message would generate. The last main component to our model of the accounts we wanted to track was the associated bots to each main or 'head' account that was considered to be in control of a certain number of other accounts. In this project there bots were only tracked in terms of a certain number of them and were not fully realized accounts as the 'head' accounts were. That is the follower and view counts for these bot accounts were not individually tracked. These accounts were mainly only used as a guaranteed number of views and retweets, of a certain message that the head account would send. The popularity multiplier element of the model was also implemented partially due to this limitation as it's increase over the time of the simulation helps to show that there would also be an associated increase in the traffic generated by the bot accounts as they grow in followers. This is one of the main limitations of the simulation and must be addressed if more work on the project is to continue. This simplified version of the model was used so that we may obtain useable results that we found reasonable by end of the time we had available to complete the project.

II. RELATED WORK

The main body of works related to this project where research that gathered and analyzed data from APIs of social media platforms(mainly twitter). Other works included research into the implementation of 'bot' accounts and how they might be detected once active on the social network. We drew quite heavily from ideas outlined in [2] Spread of information in a Social Network Using Influential Nodes, and [3] Measuring User influence in Twitter, to build our model and provide useful results. Specifically we incorporated the idea of the 'influence' of a specific user in our simulation. This means quantifying this influence to see the effect that using a bot account to boost the traffic to this account would affect the influence score that the account would obtain at the ending of the simulation. We built around some of the results and conclusions included in [3], especially, that the influence of a given user can not be determined only by the numbers of followers that the user has at any one time. As Cha states "The first finding in particular indicates that indegree alone reveals little about the influence of a user."[3] To that end we built our model around the idea that each account starts at a certain low influence level that grows over the course of the simulation based upon all the relevant metrics that are tracked, the number of views each message generates, the numbers of retweets, and the follower numbers. Next we looked at other work that had been done to model how a message propagated through a social network. The main works drawn upon here are [1] Model of information spread in social networks, and [2] Spread of Information in a Social Network Using Influential Nodes. These works helped to show how one might model a social network and analyze how the inflation of traffic might affect the spread of the information across the network. [1] uses an agent based model of simulation to create a user of the system then with 'likes/dislikes' and 'reposts' the agent gains 'energy'. New agents are created when a repost happens and at random intervals. Agents are destroyed when its energy becomes 0, as disliked reduce the energy. At the end of the simulation the number of agents and energy of a certain agent is used to show how far the given information was spread across the network. This system of energy is what was adapted in our simulation as our 'popularity' that was associated with each tracked account to help better represent the spread of the information that each account is tweeting about. [2] was used to help give us a better understanding about how to mathematically represent the network as a whole. As they showed in the paper it can be accurately represented by the use of a directed graph of nodes. With paths that join the nodes from the origin of the message to each follower and subsequent retweets. This does also help to show some of the limitations of our implementation of the simulation of the network. Since we knew we wanted to implement a way that the accounts were a system of connected nodes, we designed the model so that 'head' account would begin the generation of additional nodes of followers and retweeters of a given message. However given the time constraints of the project we had to limit the generation of followers and retweeter accounts to only a 'single layer' of the tree of nodes. Meaning that the cascade of retweets that may possibly be generated after initial retweets of the message are not modeled in our simulation. So to obtain more realistic results in possible future expansion of this project would have to include the full representation of all 'layers' of the graph. Lastly we looked at works that involved the implementation and possible detection of bot accounts that are used on a social media platform. These sources were used to help shape how our simulated bots would be used to help drive the inflation of the traffic the accounts that we were tracking. Specifically, [7] and [9] were used to determine that our assumption of bot followers generating guaranteed impressions and retweets was acceptable. In addition, [6] and [8] were used determine, given our discrete event approach to the simulation, that our assumption of the impressions and retweets of bot followers having equal weight as those of human followers was acceptable.

III. APPROACH

From our research, we determined a couple of key simplifications that could be made to approaching this otherwise complex problem. We modeled Twitter accounts on the basis of some research, although some quantities were assumed due to the lack of information about how these quantities could be represented by certain distributions. We also determined that using a discrete event simulation model was most appropriate for gathering the information we were looking for.

The first simplification we made was to reduce the structure of the real Twitter environment from a directed graph of many accounts down to a list of ten accounts. This simplification was made not only because of the time constraint, but also because we were only interested in the way bot accounts affected the influence immediately surrounding specifically monitored accounts, not

Quantity of Interest	Represented By
Human Follower Count	Integer
Bot Follower Count	Integer
Tweet Count	Integer
Total Impressions	Integer
Total Engagements	Integer

Fig. 1. Quantities of interest that help determine overall influence of a simulated human controlled account in a Twitter environment.

many layers deep like a true directed graph. We then used these ten accounts as the head accounts upon which our statistics were gathered.

The second simplification we made was to end the simulation when 5,000 total Tweets were generated among all ten head accounts. In reality, with any social media environment, it is possible to calculate statistics about the platform on the basis of some time interval, regardless of the number of messages created on it. In our case, the data we were interested in was oriented around the influence of bot accounts without regard for the passage of time, so we eliminated the time component entirely and decided to end the simulation on a condition more relevant to our study, that being the total number of Tweets generated by all ten head accounts in the simulated environment.

To model a human controlled Twitter account, we analyzed the Twitter API to determine the structure of our simulated accounts. This analysis lead us to use the key components in Fig. 1 to help quantify the influence of our simulated accounts.

In a real Twitter account, the Human Follower Count and Bot Follower Count are indistinguishable. However, to study if automated accounts have an influence in a social media environment, we further split the follower count into human followers and bot followers so we had

Parameter	Distribution
Tweet Quality	Normal(0.5, 0.1)
Tweet Probability	Normal(0.5, 0.1)
Initial Human Follower Count	Normal(100, 10)
Bot Follower Count	Uniform [0, 1000]
Engagement Rate	Exponential(0.05)
Popularity Multiplier	Initially Constant (0.01)

Fig. 2. Breakdown of distributions used to represent parameters of simulated human controlled accounts.

the ability to manipulate both, which allowed us to determine specifically whether the number of bot followers had any effect in the data generated by our simulation.

In order to simulate human activity for the head accounts during the simulation, we conducted research into how the rate at which human accounts Tweet and engage on Twitter, the quality of each Tweet, and the likelihood of human accounts to create and send a Tweet were best represented by a particular distribution, but were unsuccessful in finding specific information regarding the best distributions to use for these parameters. Instead, we assumed their distributions based on our most reasonable expectations of reality. Refer to Fig. 2 for a breakdown in how each of these parameters were distributed for each head account in our simulation.

We determined that a human controlled account’s Tweet quality was best represented by a Normal distribution based on our assumption that most Tweets aren’t spectacular or horrible in terms of quality, instead most hover around the average. For Tweet probability, we determined that our human controlled account’s likelihood of Tweeting was best represented by a Normal distribution, where only few accounts are extremely likely or very unlikely to Tweet, instead most hover around the average. We determined that each human controlled account should start with around 100 human followers based on a Normal distribution, and a random number of bot followers within the range 0 to 1,000 based on a Uniform distribution. Additionally, we had to incorporate a mechanism for including the chance that a Tweet may generate impressions if it is trending, searched by other accounts, or any other means of it being seen indirectly. To do this, we introduced a Popularity Multiplier for each head account which slightly increases the tendency of each account's Tweets to generate additional Impressions. This mechanism was intended to simulate human accounts on Twitter that grow in popularity over time.

We chose to start each head account's human follower count similarly so we could isolate the effects that the number of bot followers had on each account's key quantities of interest in Fig 1. throughout the simulation. It is important to note that these parameters may be altered when running the simulation for future work barring any significant research into the simulation of human accounts on social media that may eliminate some of the uncertainty of these assumptions.

To tie all of these elements together, we had to study the Twitter API to determine the process through which entities like Accounts, Followers, Tweets, Retweets, Impressions, and Engagements undergo in reality and how they affect overall influence of an account in order to accurately simulate the flow of the environment. We determined that the human controlled head accounts should be generated first, recalling that the quantities of interest for each account are seen in Fig. 1 and the parameters for the representation of each account are seen in Fig. 2. Next, we simply iterate through the list of accounts, generating a uniformly distributed random number in the range [0.0, 1.0], which we called the Tweet Probability Threshold, and compared it against each account's Tweet Probability. An account creates and sends out a Tweet if that account's Tweet Probability is less than or equal to the Tweet Probability Threshold. This Tweet has a quality value equal to the account's Tweet Quality parameter. Next, each Tweet generates a number of Impressions. The number of Impressions an account's Tweet generates is determined by taking the following calculation:

$$I = Exp(\ \lambda\ ) + F_B + R, \text{ where} \\ \lambda = F_H * (1 + M) * (1 + Q) \tag{1}$$

where $Exp(\ \lambda\ )$ is an exponential distribution with mean λ, F_H is the account's Human Follower Count, M is the account's Popularity Multiplier, Q is the account's Tweet Quality, F_B is the account's Bot Follower Count, and R is the number of Retweets received, which is given by:

$$R = Binom(\ I, P_R\ ) + F_B \tag{2}$$

where *Binom(n, p)* is a binomial distribution with n trials and probability p, I is the number of generated Impressions before adding R, and P_R is the probability of the account's Tweet getting a Retweet, which we assumed to be 0.01 or 1%, and F_B is again the account's Bot Follower Count. Next, the Total Impressions and Tweet Count are incremented appropriately for each account.

Importantly, in (1) and (2), F_B is generating guaranteed Impressions and Retweets respectively, whereas F_H is only generating Impressions based on the combined popularity of the account and the account's Tweet Quality in (1). Also of interest, (1) shows how we represented the tendency of a Tweet to go viral, basing the number of Impressions it generated on an exponential distribution of λ. This determination was based on the assumption that only a (3) very few number of Tweets will go viral and most stick around or below the average.

Next, we had to quantify the number of Engagements each Tweet generated, which is given by:

$$E = I * E_R \tag{3}$$

where I is the number of Impressions generated by the Tweet after Retweets are added and E_R is the account's Engagement Rate. Finally, we have to quantify how influential each account is in its environment. To do this, we calculate an Influence Score, which is given by:

$$A_I = \frac{E}{\sum_{i=1}^{n} E_i}, n = 10 \tag{4}$$

where A_I is each account's Influence Score, E is each account's Total Engagements, and n is the number of head accounts in the environment. This simply calculates the ratio of a specific head account's Total Engagements to the overall total Engagements generated in the environment. Next, each account's new human followers were calculated by the following:

$$F_N = Binom(\, I, P_F\,) \tag{5}$$

where F_N is the number of followers the account gained, I is the total number of Impressions after adding Retweets and P_F is the probability of gaining a new follower, which we assumed to be 0.005, or 0.5%. Lastly, we updated the Popularity Multiplier of each account, given by:

$$M = Binom(\, F_N, R_P\,) \tag{6}$$

where M is the new Popularity Multiplier for the account, F_N is the amount of new followers the account gained and R_P is the population increase rate, which we assumed to be 0.01, or 1%.

Combined, (6) all of the formulae show how we structured the relationship between all elements incorporated into our simulation of automated accounts in a Twitter environment to determine if bot followers do have an effect on how influential

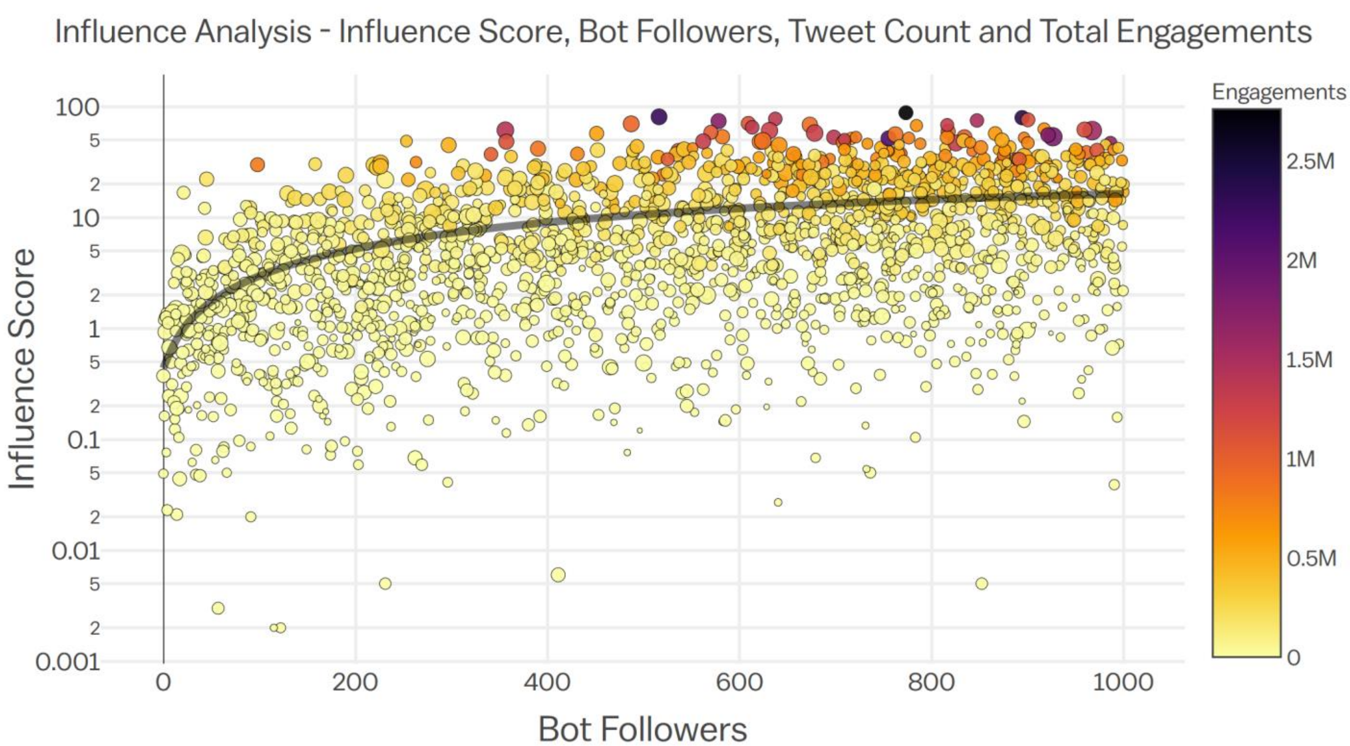

Fig. 3. A multivariate scatter plot showing each simulated head account as a point where Bot Follower Count is on the X-axis, Influence Score is on the Y-axis, Total Engagements is the color of each point and Tweet Count is the size of each point.

an account may become.

Finally, to gather enough data to analyze, we ran the simulation 200 times. Each time, 10 accounts were generated, as seen in (4), using 5,000 total Tweets as the ending condition. Ultimately, our data consists of 2,000 unique accounts and a combined 10,000,000 Tweets.

IV. RESULTS

The results of our simulation proved to be interesting in a few key areas. They demonstrate in multiple ways, given the methods in our approach to this problem, that the number of bot followers a human controlled account has does affect the influence of said account in its environment.

Firstly, we'll discuss Fig. 3, a multivariate scatter plot showing relationships between a few variables of interest, including the Influence Score, Bot Follower Count, Tweet Count, and Total Engagements. In Fig. 3, each one of the 2,000 simulated accounts is represented by a point in the scatter plot. The X-axis is representing the Bot Follower Count for each account, and the Y-axis is representing the Influence Score, A_I in (4), logarithmically. The color of each point is representing the Total Engagements for each account, and the size of each point is representing the Tweet Count for each account. Finally, the black line represents an exponential best fit line to the relationship between Bot Follower Count and Influence Score. A few notable relationships should stand out immediately.

Clearly, there is a darkening of color in the upper right corner of the points in the data set. This is showing that human controlled accounts, at least according to our approach, have a tendency to generate more Engagements on their Tweets if their Bot Follower Count is higher. This means that the guaranteed Impressions (1) and Retweets (2) that bot followers generate absolutely do have a positive effect in generating Engagement among the human controlled account's Tweets.

Secondly, a little less obvious is the difference

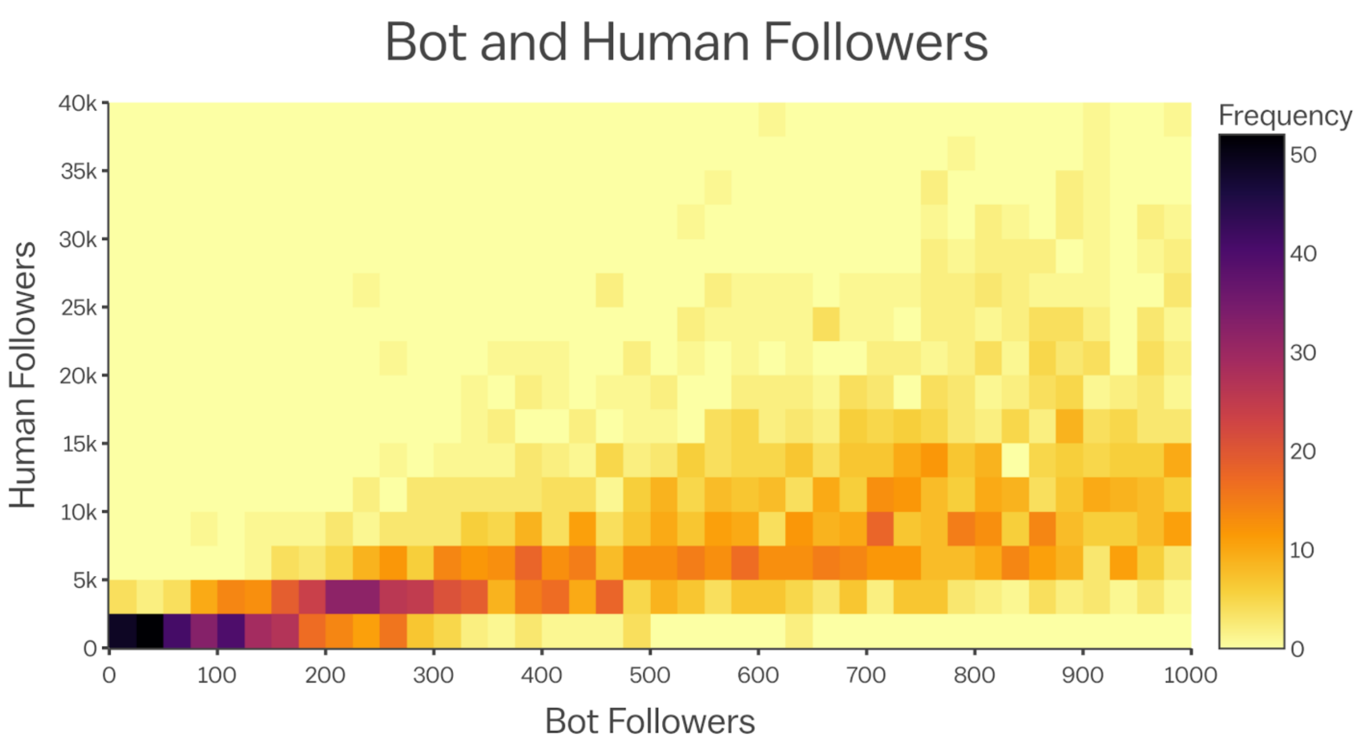

Fig. 4. A two-dimensional histogram showing the frequencies of simulated head accounts falling into bins represented by Bot Follower Count on the X-axis and Human Follower Count on the Y-axis. Bins are sized 25 bot followers by 2,500 human followers.

in the size of each point with respect to the Influence Score. It appears that the size of the points increases as Influence Score increases. This indicates that the human controlled account's tend to earn higher influence in their environment if they create and send out more Tweets than others in the same environment. This is intuitive because if the account has a higher Tweet Probability, the likelihood of said account gathering Engagements increases, thus increasing its Influence Score.

Thirdly, the exponential fit line shows a clear increase in Influence Score as the Bot Follower Count increases. Interestingly, the best fit line shows a dramatic increase in Influence Score from 0 to 200 bot followers, gradually becoming less effective with the successive addition of more bot followers. This means that adding any number of bot followers to a human controlled account with no bot followers to begin with would show a dramatic increase in influence in the environment initially, which seems to make intuitive sense based on our approach given that bot followers guarantee Impressions (1) and Retweets (2).

Finally, we have Fig. 4, a two-dimensional histogram showing collective bins of human controlled accounts with respect to Bot Follower Count and Human Follower Count.

In Fig. 4, the X-axis is representing the Bot Follower Count, the Y-axis is representing the Human Follower Count, and the color is representing the frequency of accounts that fall into certain bins, where the size of each bin in the X direction is 25 bot followers, and the size of each bin in the Y direction is 2,500 human followers.

This two-dimensional histogram shows at least three interesting observations in our data set. Firstly, the color of the histogram is very dark around the bottom left corner. This indicates that regardless of all the variability in the simulation, accounts starting with anywhere between 0 and 100 bot followers never made it past 10,000 human followers.

Secondly, there is clearly a positive relationship between Bot Follower Count and Human Follower Count, although they are not directly related in our approach. This seems to suggest that more human followers accumulate because of the increased number of Impressions and Retweets that bot followers are known to generate in (1) and (2) respectively.

Lastly, from 650 bot followers and up, there are 0 occurrences of accounts with less than 2,500 human followers, shown by the contiguous strip of light yellow on the bottom right corner of the histogram. This, without question, indicates that bot followers have an effect on how many human followers an account may reach when considering other variability factors mentioned previously in our approach. Thus, they have a higher tendency to generate Impressions, Engagements, and ultimately earn a higher Influence Score.

The visualization of our data set in Fig. 3 and Fig. 4 both show interesting results. They indicate that there is absolutely a positive trend between the influence of an account and the number of bot followers it has.

V. CONCLUSION

To conclude our study, we find that by simulating a social media environment as accurately as possible using assumption based distributions given the little amount of existing research in this area that automated accounts do have a tendency to inflate the influence of simulated human accounts. They inflate the influence of human accounts by following or interacting with information posted to the platform by the human accounts. Ultimately, we believe that because of this, it is possible for a number of automated accounts to increase the ability for information from a human controlled account to propagate further into the environment and reach more users of the platform. It could be the case that in reality, some human controlled accounts exist solely to work toward and achieve an agenda of spreading information or misinformation through a social media platform through the use of automated accounts. This simulation was created to conduct research into this idea, and we see that, given the assumptions made in our approach to solve this problem, that an increased number of automated followers for a human controlled account in a Twitter environment does lead the human controlled account to earn a higher influence in its environment.

We encourage future research into the simulation of a social media platform given our initial results because it may become necessary in the future to detect and eliminate automated accounts from social media platforms that are associated with human account a with malicious agenda. These findings are a good start and quite interesting, but more work is needed to fully simulate a social media environment using data structures that more accurately represent the environment, along with using research based distributions for all the parameters described in our approach.

REFERENCES

[1] Lande, D & A.M., Hraivoronska & B.O, Berezin. (2016). Model of information spread in social networks. European Journal Of Natural History. 41-44. Retrieved Nov 2018 from https://www.researchgate.net/publication/307598597_Model_of_information_spread_in_social_networks

[2] Chaudhury A., Basuchowdhuri P., Majumder S. (2012) Spread of Information in a Social Network Using Influential Nodes. In: Tan PN., Chawla S., Ho C.K., Bailey J. (eds) Advances in Knowledge Discovery and Data Mining. PAKDD 2012. Lecture Notes in Computer Science, vol 7302. Springer, Berlin, Heidelberg. Retrieved Nov 2018 from https://link.springer.com/chapter/10.1007/978-3-642-30220-6_11

[3] CHA, M.; HADDADI, H.; BENEVENUTO, F.; GUMMADI, K.. Measuring User Influence in Twitter: The Million Follower Fallacy. International AAAI Conference on Web and Social Media, North America, may. 2010. Available at: https://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/view/1538/1826 . Date accessed: 15 Nov. 2018.

[4] VS Subrahmanian, Amos Azaria, Skylar Durst, Vadim Kagan, Aram Galstyan, Kristina Lerman, Linhong Zhu, Emilio Ferrara, Alessandro Flammini, Filippo Menczer, Andrew Stevens, Alexander Dekhtyar, Shuyang Gao, Tad Hogg, Yan Liu, Onur Varol, Prashant Shiralkar, Vinod Vydiswaran, Qiaozhu Mei, Tim Hwang. The DARPA Twitter Bot Challenge, IEEE Computer 49(6), June 2016. Retrieved Nov 2018 from https://arxiv.org/ftp/arxiv/papers/1601/1601.05140.pdf

[5] Clayton A. Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, Filippo Menczer. BotOrNot: A System to Evaluate Social Bots, 2 Feb 2016. Retrieved Nov 2018 from https://arxiv.org/pdf/1602.00975.pdf

[6] Clayton A. Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, Filippo Menczer. Online Human-Bot Interactions; Detection, Estimation, and Characterization, ICWSM 2017. Retrieved Nov 2018 from https://aaai.org/ocs/index.php/ICWSM/ICWSM17/paper/view/15587/14817

[7] Nikan Chavoshi, Hossein Hamooni and Abdullah Mueen, On-Demand Bot Detection And Archival System, to appear In the Proceedings of World Wide Web Conference (Companion Volume), pp. xxx-xxx, WWW 2017. Retrieved Nov 2018 from https://www.researchgate.net/publication/314179650_On-Demand_Bot_Detection_and_Archival_System

[8] Nikan Chavoshi, Hossein Hamooni and Abdullah Mueen, Temporal Patterns In Bot Activities, to appear In the Proceedings of World Wide Web Conference (Companion Volume), pp. xxx-xxx, WWW 2017. Retrieved Nov 2018 from https://www.researchgate.net/publication/314179583_Temporal_Patterns_in_Bot_Activities

[9] Nikan Chavoshi, Hossein Hamooni and Abdullah Mueen, DeBot: Twitter Bot Detection via Warped Correlation, to appear In the Proceedings of the 8th International Conference on 6th IEEE International Conference on Data Mining, pp. xxx-xxx, ICDM 2016. Retrieved Nov 2018 from https://www.researchgate.net/publication/308021270_DeBot_Twitter_Bot_Detection_via_Warped_Correlation

APPENDIX
ARTIFACT DESCRIPTION APPENDIX: **Simulating the Effect of Automated Accounts in a Social Media Environment**

A. Abstract

This artifact description contains information on the environment that was used to develop, and test the simulation, both hardware and software, as well as direction to find the publicly available code.

B. Description

1) Check-list (artifact meta information):
- Program: JAVA code
- Compilation: compilation was done on the standard javac compiler JDK 11.0.1
- Binary: java .class file
- Run-time environment: Results obtained running on Linux mint 19 Tara, 64 bit
- Hardware: any hardware able to run the current version of the java compiler (11.0.1)and runtime environment
- Experimental Workflow: clone from git, compile java code, run with java command
- Output: tracked stats from each account at the end of the simulation
- Publicly available?: yes

2) How software can be obtained (if available): Obligatory if the paper contains computational results. https://github.com/arj0819/CSci445 Repository containing source code
3) Hardware dependencies: hardware that is capable to run JDK 11.0.1 and JSE 11
4) Software dependencies: JDK 11.0.1 for code compilation and JSE 11 for execution

C. Installation : follow instructions for installation of JDK 11 at https://jdk.java.net/11/ git clone from https://github.com/arj0819/CSci445 compile with JDK using javac.