Cascading models using Twitter Sentiment Analysis

Christopher Pereyda and Chance DeSmet

School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA, USA

Abstract - With Twitter becoming an increasingly popular method for politicians to interact with their audience, the number of users "following" or keeping track of the different "Tweets" that these politicians have been making has increased dramatically. The relationship of users following politicians and other users can be represented by a graph, where the nodes are the individual Twitter accounts and a directed edge connects a node to a node that follows it. Once this graph has been created, it then becomes possible to label the nodes for certain politicians as either right or left leaning. Then, using different cascading principles, it is possible to simulate the flow of information from the principal political nodes to the nodes that represent the followers of those accounts, and continue cascading based upon these principles until the cascade has stabilized.

I. Introduction

Twitter allows for the tracking of discussion and the many connections made by different users, the movement of political discourse into this sphere has enabled the tracking of this discussion as well as following the flow of information. In the political spectrum, the information flow is expected to be from the originating politician and news accounts to the general members of the population. We are modeling this flow by expecting that the information and beliefs will be diffused through the regular posts that users make, and thus the followers of these accounts will be most likely to be the recipients of the information dissemination.

The motivation behind this problem is to attempt, through the use of machine learning and network theory, to model the flow of information across the twitter social network as it relates to online politics. A literature review finds relatively few scholarly inquiries into this specific topic, so we are implementing a novel method in which to view on-line information flow. Additionally, this method is easily transferable into several different non-political fields, such as research, entertainment, news, and other types of shared information. A novel working model for these subjects would be very useful in providing a new perspective on how information flow occurs.

From this set up, we were able to identify generate 5 different styles of cascading, as well as identify a trend of both Democrat and Republican Twitter accounts to follow a significant amount of Twitter accounts that do not share their beliefs, with Democratic accounts being more likely to follow a wide variety of account types.

II. Problem Definition

The basic problem is comprised of several different problems, gathering the Twitter data, labelling the initial "political" nodes, creating a graphical representation of the Twitter social network, simulating several different methods of cascade algorithms, and then generating graphs and statistics from the resulting graphs. One of the most difficult parts of this research is running computations on the extremely large dataset that is generated by collecting Twitter data. Our graph representation of the twitter network has over 3 million unique nodes which makes complex computations extremely time consuming. To mitigate this, we have performed analysis on representative sub-graph samples of our network and left many full network computations to super computers.

The problem that we wish to solve is to create a method to simulate diffusion of political thought and ideas by using network science principles of cascading and thresholds. In order to provide more in depth and generalizable results, we will be testing several different types of cascades. This is a relevant problem because applying cascading methodology to Twitter networks has experienced very little inquiry, and upon investigation our method could become a useful way to analyze data flow on social networks for other researchers.

Since we are interested in examining how information flows through the network (and thusly bias flow), we wanted to examine the friendship relation between users. Twitter defines the friendship relation as the people that the user follows. That is, if person A follows person B, A is a friend of B. We did not examine the follower relation because of the difference in size between followers and friends. A popular user can have upwards of 4,000 friends, and beyond 1 million followers. It would be too impractical to try and navigation the network via the follower relationship.

III. Models

The goal of our research is to simulate cascading effects on our generated network of political bias across Twitter. As this is a novel problem, the algorithms and methods used were designed by our group. This first step in our problem was to gather Twitter data, as there was no existing dataset that focused on Twitter data regarding politics and followers. Twitter restricts the amount of data requested to fifteen nodes every fifteen minutes, so to get a sufficient amount of data to generate meaningful graphs and conclusions, it was necessary to gather data and keep the data collection process ongoing for the majority of our research.

Data was collected from twitter using their open API. We created an application under the terms and conditions applied by twitter and used this application to query the twitter database. We ran our network structure collection system for a period of 1 month. This was done by performing Fair-BFS on our initial 4 seeders. We gathered the friendship relation of these users to further expand the network topology. This concluded in the scanning 106,000 unique twitters users which resulted in approximately 1 million known twitter users. From these users, a random scan was performed to gather their most recent 100 tweets. This scan was left running for 1 week and gathered 50,000 users tweet data.

To gather our friendship relation data, we implemented a modified Breadth First Search (BFS). This BFS was altered to make it fair. Fairness in that, we do not want to prioritize one user over another and make sure to examine all users a certain depth away from the initial nodes before exploring further. We begin by setting our first 4 nodes for Fair-BFS (seeders). For each of our 4 nodes, we gather the friends of each of these nodes and place them in separate sets. For each of our seeders, we rotate through gathering the friends from each node in the set. An example of this process can be seen in Figure 1. This process is repeated until all nodes in each set are explored and those explored nodes become the new set we rotate through. We implemented this Fair-BFS algorithm because we were uncertain to what depth we could feasibly reach given a time-limited number of requests. We also did this because

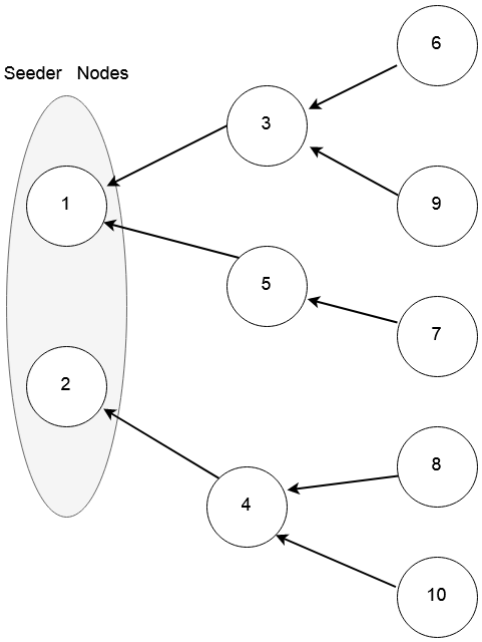

Figure 1: A demonstration of our Fair-BFS. We begin by adding two nodes to our initial search set. The numbers inside the nodes correspond to the order of searching.

we did not want our network structure to be biased towards one person compared to another if they had a larger number of friends.

To determine the political bias of the twitter users, we implemented a well-known ML approach to classification. We began by collecting a set of users with known political biases, who were: Donald Trump, Ted Cruz, Hillary Clinton, and Bernie Sanders. These users were classified as Republicans and Democrats. We collected the last 3200 tweets of these individuals (including retweets) and gave the corresponding political leaning to each tweet. We proceeded to train our Neural Network with this set of data, leaving out a 10% random sample for validation. The results for this training process can be seen Table 1. From these results, we know that our model has 90% accuracy when classifying our sample set of people. We do not know how this result can be further extrapolated to entities with different speech characteristics from our sample (e.g. Businesses, Advertisements, Blogs, …).

		Predicted Bias R	Predicted Bias D	total
Actual Bias	R′	589	78	667
	D′	51	562	613
	total	640	640	

Figure 3: Confusion matrix of our classification method. Our model is slightly better at classifying Republicans vs. Democrats but still has an overall accuracy of about 90%.

The machine learning model was a neural network with an embedding layer, an LSTM layer, and a dense layer. We began by first filtering the raw tweet data to exclude links, special characters, and Twitter formatting. The tweets were then tokenized to remove the necessity to process natural languages. Tokenizing is the act of assigning an integer value to each word. These tokens were then fed into an embedding layer. This effectively creates a vector space representation of our words. It is very beneficial to do this because some words are more

important than others and words linked to other words will become important in terms of classification. The vector space is then fed into a Long Short Term Memory (LSTM) layer. LSTM is subset of the Recurrent Neural Network layer which allows for past data to be seen by future nodes in the layer. RNNs and its subtypes are effective for structure prediction and recognition. An LSTM is used instead of a more general RNN because LSTM allows the data flow almost linearly from previous nodes to future nodes. This makes it considerably better at recognizing larger structures than an RNN. It is necessary to recognize longer structures because sentences and ideas used in tweets can be very long. The result of this layer is then passed into a fully connected layer which then outputs a binary classification of either Republican or Democrat.

The only methods exterior to this project is the "igraph" and "graph-tool" libraries developed for Python. These libraries allow for the easy use of graph functions, and plotting very large graphs, giving the project a method to perform and visualize cascading in our networks. Our group used these methods to develop and algorithm to create different sized graphs, perform cascade operations on these graphs, and analyze the results of the different cascades.

IV. Implementation and Analysis

To generate a greater amount of data to analyze, three different initial graphs were generated from the Twitter data: a "large" graph, with only single edge nodes removed, containing approximately 40,000 nodes; a "medium" graph, where every node had a connectivity of at least 4, with the graph containing approximately 14,000 nodes; and finally, a "small" graph, with all nodes connecting to at least 10 other nodes and containing approximately 3,000 nodes.

With these three graphs, we will be able to determine how the differences in connectivity and sample size affect the cascading and resultant distribution.

We ran our classification method on 50,000 twitter users. We classify each tweet as either a republican or democrat using our neural network. A user's bias is then the scaled sum of their tweet's classifications. To match the real world more, we scaled our twitters use to have a normal distribution about zero (where democrat is -1 and republican is 1). We found this was necessary as we were over classifying users as republican. To perform a baseline measurement of the real-world accuracy of our system, we ran several political people whose bias is well known through our classification system. The results of this can be seen in Table 1.

Name	Political-Bias	Classification-Score
Maxine Waters	alt-left	0.215
Bernie Sanders	strong-left	0.440
Hillary Clinton	left	0.507
Barack Obama	left	0.526
Donald Trump	right	0.532
Tucker Carlson	strong-right	0.547
Richard Spencer	alt-right	0.556

Table 1: Well-known political figures with known political bias and classification score. The score is not normalized to have a normal distribution of people about 0.

Once we had generated different the different sized graphs, labels were assigned to each of the nodes. "R" if the node was republican, "D" if the person was a democrat, and "Neutral" if the node hadn't been labelled. A person can be not labelled if they did not have enough tweets, their bias was not strong enough to one party, or their data was not collected. Then, this graph was put into a function that analyzed the connections between nodes, simulating a cascading effect. We implemented five different cascading metrics to provide information on how the threshold for cascading influenced the final network. The first method used, labelled "MAJ", only required a majority of either Republican or Democrat accounts followed by a node to label that node either Republican or Democrat. The next, similar methods used were "MEDMAJ" and

"SMAJ", representing "medium majority" and "super majority". These methods required the accounts that a node is following to be over 55% and 66% more of one ideology than another to become labelled. For example, if the node follows 10 Republican accounts, then it must follow 13 Democrat accounts to be labelled Democratic for the "MEDMAJ" cascade and 15 for the "SMAJ" cascade. With these three metrics, the cascading effect will happen in differing fashions as increasing thresholds have to be met. In this way it can become visible by comparing the resulting graph if nodes tend to follow only a certain side (all three cascades produce a similar split between political ideologies), or if nodes tend to follow accounts from both sides of the political spectrum (three cascades produce different splits, as well as an increased number of neutral nodes). The other two types of cascade metrics created were "NMAJ" and "FMAJ", representing "neutral majority" and "full majority". The criteria for "NMAJ" was that the amount of a political ideology had to be greater than both the other side, as well as the number of neutral nodes it followed. Similarly, "FMAJ" required that the number of accounts a node follows must be greater than that of the opposing viewpoint PLUS the amount of neutral accounts that the node follows in order to assign it that belief.

With the cascading methodologies in place, we performed the cascading operation on the three graphs. To have sufficient benchmarks for comparison, we saved the distribution of the network after 0, 5, 10, 50, and 100 cascades had been performed. This is to allow for easy visualization as to how the cascading process took place, and how quickly individual metrics allowed the cascade to occur. Finally, we plotted the resulting cascades, providing a visual of how the networks looked once they were fully cascaded.

V. Results

After computing the graphs after formation and the cascading criteria, several trends were apparent, not only in the individual "small", "medium", and "large" graphs, but between the three graphs as a whole collection, in terms of clustering, cascading, and connectivity.

The global clustering coefficient of the "small" was ~.191, the global clustering coefficient of the "medium" graph was ~.08, and the global clustering coefficient of the "large" graph was .05. These coefficients indicate that as the size of the graph increases, the overall connectivity between nodes decreases and the graph gets less and less centrally clustered. This is easily understandable upon reflection of how the size of the graphs were generated; in order to create smaller, less tree-like graphs, the nodes selected to be contained in the smaller graphs possessed increasingly higher connectivity, resulting in connectivity, and the resulting global clustering coefficients, to be inversely proportional to the size of the graphs.

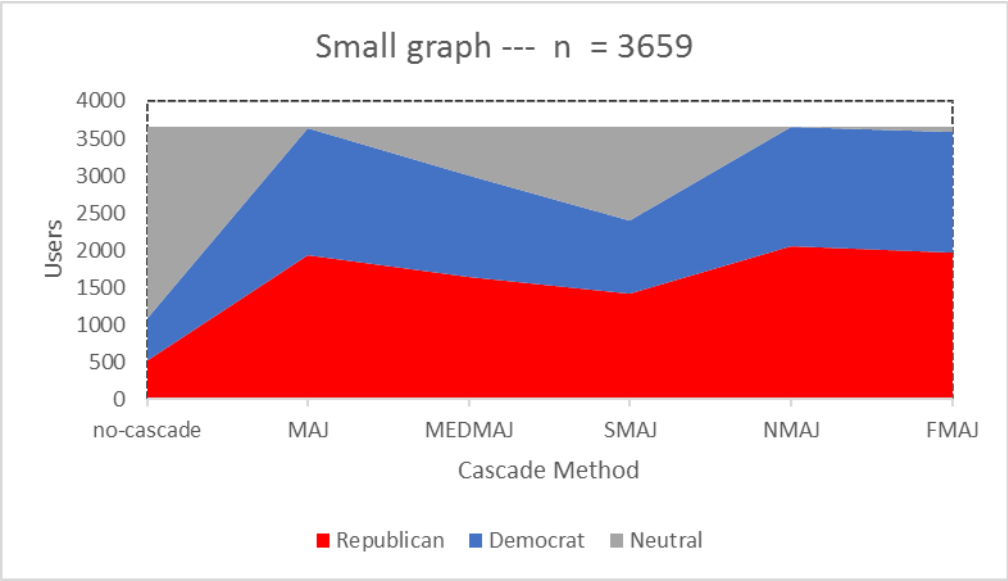

Figure 4: The largest number of Republican nodes is created by the "NMAJ" cascade and is 2059, the largest number of Democratic nodes is observed in the "MAJ" cascade category and is 1691, and the greatest number of Neutral nodes is generated by the "SMAJ" cascade, and is 1261[1]

[1] Appendix A

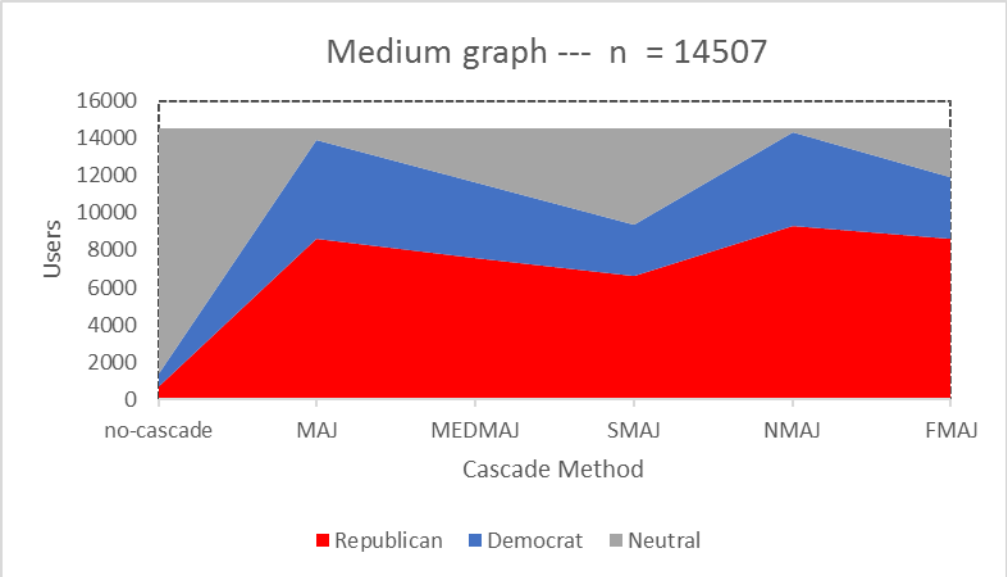

Figure 5: This graph has the greatest number of Republican nodes created in NMAJ, with 9286; the Greatest amount of Democrat nodes is created by "MAJ", with 5295; and the greatest number of Neutral nodes is created by "SMAJ", with 5145[1].

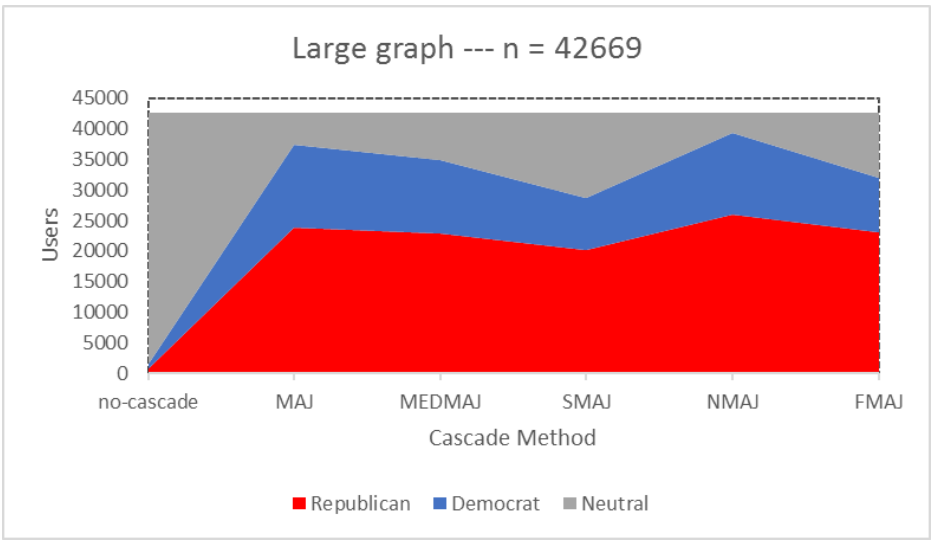

Figure 6: The maximum value categories the same as "medium" and "small", with the largest generator of Republican nodes being "NMAG" with 25953, the largest generator of Democrat nodes being "MAJ" with 13561, and the largest generator of Neutral nodes being "SMAJ" with 14029[1].

For each graph, the initial values of the labelled nodes contained slightly more Democratic nodes, however, once cascading had occurred, a significant majority of nodes were Republican in every category for every graph size. This would suggest that while there exist a relatively even split between Republican and Democratic Twitter accounts, Republicans as a

group tend to follow proportionally more Republican Twitter accounts, leading to the cascading models in general choosing to label a given node Republican.

As the largest element in each class of political affiliation is present in the same category for each graph size, concussions can be drawn as to why the maximal value exists in a certain category. For the Republican nodes, the greatest amount existing in the "NMAJ" category is consistent with previous observations. Recall that the "NMAJ" classifier labels a node a specific belief if its majority belief is greater than the other belief as well as Neutral nodes. If the Republican nodes are less likely to follow nodes that do not share their beliefs, they will be following relatively fewer Neutral nodes, and then will be more easily labelled Republican by the "NMAJ" classifier.

The classifier that the Democratic nodes were generated the most on was the "MAJ" classifier. This classifier labelled a given node whichever was greater of the nodes that they followed, the number of Republican nodes or the number of Democrat nodes. As the Democratic nodes are hypothesized to follow a wider variety of political views, then the cascade category that generates the highest number of Democrat labelled nodes would be "MAJ" as all it checks is the highest number of either class of belief that a node is following, and labels it the same as that of the most common class.

The Neutral nodes, or the nodes that remained unlabeled, occurred most frequently when a cascade was generated using the "SMAJ" classifier. This is not surprising, as the "SMAJ" category required a node to follow 150% more nodes of a certain belief than others, and as all graphs are relatively well connected, it is increasingly rare that a node exclusively follows a certain political class of node. Interestingly, the categories "SMAJ" and "FMAJ" also exhibits the highest percentage of Republican to Democrat nodes, and as both are biased toward assigning a

view to nodes that primarily follow a single political class, this further strengthens the idea that Republican nodes are more likely to follow other Republican nodes than Democratic or Neutral nodes.

In the "small", "medium", and "large" graphs, two trends exist: among the criteria that cascades without respect to the number of neutral nodes, and among the criteria that compares against the neutral nodes before it cascades. The first of these groups contains "MAJ", "MEDMAJ", and "SMAJ". As can be seen, as the criteria gets more and more selective, the amount of "Neutral", unlabeled nodes greatly increases. This indicates that many Twitter accounts follow members from both political parties, as the stricter the classifier is about the majority class' dominance in order to label the node, the fewer labelled nodes exist. Another trend that is noticeable is present in the neutral tracking classifiers, "NMAG" and "FMAG". In this instance, as the more Neutral nodes a node follows decreases the likelihood that a node will be labelled as certain political class, this trend is also indicative that in general, Twitter users tend to follow multiple political classes of accounts, not just those the same class as their belief.

Finally, the fact that all three graphs present similar data is important, because it suggests that when analyzing political Twitter data, it might not be necessary to construct an extremely large graph with millions of nodes and edges. Choosing a representative sample that is significantly smaller than the whole dataset could provide similar results and conclusion, while being much less computationally intensive to analyze.

Overall, our approach was successful in our original goal. We were able to collect Twitter data, separate it into different sizes of graphs, run the graphs through several cascading algorithms, and analyze the resultant data. We were able to get consistent results across graph sizes and categories, which indicates that our methods were effective in categorizing the data in

the graph. There were some weaknesses in our approach however, namely the collection period, cascading intervals, and attempts at plotting. The collection period was a weakness because in order to get what we thought was enough data, most of the project could not be started until the final weeks of class. As we discovered, a smaller sample of the graph can yield the same results, thus it might have been unnecessary to spend so much time on data collection. The cascading periods chosen were also a weakness as a significant computation time was spent cascading each generated graph up to 100 times, which, as it turned out was unnecessary, as no graph needed more than 10 iterations to finish the cascading process. Finally, the time spent on attempting to generate meaningful plots of the graphs was not fruitful, as with graphs this large and interconnected, we were unable to create plots that revealed any additional information on the properties of the graphs that we had created.

VI. Related Work

One foundational paper explored the structure of the Twitter social network [4]. They scanned approximately 42 million users which resulted in about 1.5 billion social relationships. In their work they examined tweet diffusion across the network. They found that 75% of a tweets effective life (amount of people retweeting) is about 1 day and 50% is in one hour. From this we can conclude that time of flight for tweets is very small and thus propagation through the network happens very fast. They did not examine how the information diffuses to users groups (clusters) or how tweets are retweeted by certain groups. They found the average path length of twitter to be approximately 4.12. This is significantly smaller from the theorized small world phenomenon of 6, they theorize that this is because users can connect with many more people and may not even know to whom they are connecting. In our work, we found that the average

path length is approximately 5. This number is larger most likely because we have 4 linked clusters and not a large random sample of the graph.

Another well-known paper is the work done by Baksy in, “Everyone’s an influencer” [5]. In their work, they examined information diffusion not to target user groups but diffusion based on topics. They found that the success of propagation of a topic depends largely on what the topic is not who it is being sent to. They found that past local performance (in regards to influencers) and number of followers can be highly suggestive of performance in a tweets success. This means that people who regularly get retweeted are more likely to have their tweets retweet regardless of what is actually being sent. Using this information, we could effectively target these normal influencers for things like adds to increase the likelihood of propagation. Similarly, in our work we could target users who have strong retweet pasts to increase the chance of propagation through the network. It would be cool to examine how confirmation bias can play a role in retweets propagating between different political groups and see if this is a stronger indicator of tweet propagation than just retweet past.

VII. Conclusion

In conclusion, our group was able to successfully extract user data, assign sentiment to a few nodes, perform cascading based upon several created algorithms, and analyze the resulting data for notable trends. Our methods were consistent regardless of graph size as our cascading categories produced similar results in all three runs. From the resulting data, we were able to see that both Democratic and Republican users tend to follow users outside of their political views, with Democrats being slightly more likely to do so. We also introduced our suit of cascading classes used to label political views, based upon likely trends in Twitter usage.

Further work in this area could involve a non-binary approach to labelling and increased cascade classifiers. Using a non-binary approach to labelling different political views could convey a spectrum of beliefs, offering a greater dataset with more possibilities of in depth analysis. Similarly, creating more complex cascade criteria could split the data in more unique ways, offering an even greater amount of information for potential analysis.

References

[1] Csardi G, Nepusz T: "The igraph software package for complex network research", *InterJournal*, Complex Systems 1695. 2006. http://igraph.org

[2] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780.

[3] Tiago P. Peixoto, "The graph-tool python library", figshare. (2014) DOI: 10.6084/m9.figshare.1164194 [sci-hub, @tor]

[4] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. 2010. What is Twitter, a social network or a news media?. In *Proceedings of the 19th international conference on World wide web* (WWW '10). ACM, New York, NY, USA, 591-600. DOI=http://dx.doi.org/10.1145/1772690.1772751

[5] Eytan Bakshy, Jake M. Hofman, Winter A. Mason, and Duncan J. Watts. 2011. Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining* (WSDM '11). ACM, New York, NY, USA, 65-74. DOI: https://doi.org/10.1145/1935826.1935845

Appendix A – Graph Data

"Small" Graph

Cascade Criteria:	MAJ	MEDMAJ	SMAJ	NMAJ	FMAJ
0 Cascade cycles	526 Republican 571 Democrat 2562 Neutral	526 Republican 571 Democrat 2562 Neutral	526 Republican 571 Democrat 2562 Neutral	526 Republican 571 Democrat 2562 Neutral	526 Republican 571 Democrat 2562 Neutral
5 Cascade cycles	1943 Republican 1691 Democrat 25 Neutral	1636 Republican 1362 Democrat 661 Neutral	1417 Republican 981 Democrat 1261 Neutral	2059 Republican 1593 Democrat 7 Neutral	1227 Republican 571 Democrat 2161 Neutral
10 Cascade cycles	1943 Republican 1691 Democrat 25 Neutral	1636 Republican 1362 Democrat 661 Neutral	1417 Republican 981 Democrat 1261 Neutral	2059 Republican 1593 Democrat 7 Neutral	1974 Republican 1589 Democrat 96 Neutral
50 Cascade cycles	1943 Republican 1691 Democrat 25 Neutral	1636 Republican 1362 Democrat 661 Neutral	1417 Republican 981 Democrat 1261 Neutral	2059 Republican 1593 Democrat 7 Neutral	1974 Republican 1611 Democrat 74 Neutral
100 Cascade cycles	1943 Republican 1691 Democrat 25 Neutral	1636 Republican 1362 Democrat 661 Neutral	1417 Republican 981 Democrat 1261 Neutral	2059 Republican 1593 Democrat 7 Neutral	1974 Republican 1611 Democrat 74 Neutral

"Medium" Graph

Cascade Criteria:	MAJ	MEDMAJ	SMAJ	NMAJ	FMAJ
0 Cascade cycles	682 Republican 731 Democrat 13094 Neutral	682 Republican 731 Democrat 13094 Neutral	682 Republican 731 Democrat 13094 Neutral	682 Republican 731 Democrat 13094 Neutral	682 Republican 731 Democrat 13094 Neutral
5 Cascade cycles	8616 Republican 5295 Democrat 596 Neutral	7584 Republican 4044 Democrat 2879 Neutral	6627 Republican 2735 Democrat 5145 Neutral	9286 Republican 5049 Democrat 172 Neutral	8586 Republican 3341 Democrat 2601 Neutral
10 Cascade cycles	8616 Republican 5295 Democrat 596 Neutral	7584 Republican 4044 Democrat 2879 Neutral	6627 Republican 2735 Democrat 5145 Neutral	9286 Republican 5049 Democrat 172 Neutral	8586 Republican 3341 Democrat 2601 Neutral
50 Cascade cycles	8616 Republican 5295 Democrat 596 Neutral	7584 Republican 4044 Democrat 2879 Neutral	6627 Republican 2735 Democrat 5145 Neutral	9286 Republican 5049 Democrat 172 Neutral	8586 Republican 3341 Democrat 2601 Neutral
100 Cascade cycles	8616 Republican 5295 Democrat 596 Neutral	7584 Republican 4044 Democrat 2879 Neutral	6627 Republican 2735 Democrat 5145 Neutral	9286 Republican 5049 Democrat 172 Neutral	8586 Republican 3341 Democrat 2601 Neutral

"Large" Graph

Cascade Criteria:	MAJ	MEDMAJ	SMAJ	NMAJ	FMAJ
0 Cascade cycles	708 Republican 789 Democrat 41172 Neutral	708 Republican 789 Democrat 41172 Neutral	708 Republican 789 Democrat 41172 Neutral	708 Republican 789 Democrat 41172 Neutral	708 Republican 789 Democrat 41172 Neutral
5 Cascade cycles	23870 Republican 13561 Democrat 5238 Neutral	22780 Republican 12055 Democrat 7834 Neutral	20101 Republican 8539 Democrat 14029 Neutral	25953 Republican 13411 Democrat 3305 Neutral	23099 Republican 8901 Democrat 10669 Neutral
10 Cascade cycles	23870 Republican 13561 Democrat 5238 Neutral	22780 Republican 12055 Democrat 7834 Neutral	20101 Republican 8539 Democrat 14029 Neutral	25953 Republican 13411 Democrat 3305 Neutral	23099 Republican 8901 Democrat 10669 Neutral
50 Cascade cycles	23870 Republican 13561 Democrat 5238 Neutral	22780 Republican 12055 Democrat 7834 Neutral	20101 Republican 8539 Democrat 14029 Neutral	25953 Republican 13411 Democrat 3305 Neutral	23099 Republican 8901 Democrat 10669 Neutral
100 Cascade cycles	23870 Republican 13561 Democrat 5238 Neutral	22780 Republican 12055 Democrat 7834 Neutral	20101 Republican 8539 Democrat 14029 Neutral	25953 Republican 13411 Democrat 3305 Neutral	23099 Republican 8901 Democrat 10669 Neutral