Universität Zürich[UZH]

Blockchain & Distributed Ledger Technologies

Influence Maximization in Twitter Network

Li Nan (22-736-169), Liu Yingying (22-736-201), Guzman Piedrahita, David (22-737-571), Harputluoglu Eylül (22-740-856)

Network Science (HS22)

Faculty of Business, Economics and Informatics

21.11.2022 Monday

ABSTRACT

We study the problem of maximizing the expected spread of influence within a social network. In this report, we used the Twitter accounts of "ProudBoys" members and of other users associated with them which form of snapshot a social network in the year 2020. We analyzed the influence maximization problem in several of the most widely studied models in social network analysis such as Independent Cascade, Decreasing Cascade, Weighted Cascade, TRIVALENCY, Linear Threshold, and Generalized Threshold models, along with 3 influence maximization algorithms: Naive Greedy, Cost Effective Lazy Forward (CELF) and Maximum Influence Arborescence (MIA). We compared the results from each algorithm paired with each model. As result, we detected accounts with the most influence on the network as it pertains to diffusion of information. Click here to access our GitHub.

1 INTRODUCTION

The widespread use of the internet has led to billions of people being connected through online social media platforms like TikTok, Twitter, and Instagram. These platforms generate a large amount of data, which has led to increased research on social networks. In addition to being a means of communication, social networks also serve as a platform for sharing information, providing public services, and marketing.

In recent years, with the popularity of social networks, the influence maximization problem has become a pressing issue in this field. The Influence Maximization problem identifies a small subset of the most important influencers in the network to tackle some real-world problems and activities(Singh et al. 2022). Numerous techniques to improve the performance of Influence Maximization have been proposed. In this project, first, we will explore how to model the diffusion process to propagate the information by adapting several well-accepted diffusion models. Second, we will compare and analyze the outcome of existing Influence Maximization algorithms deployed on our dataset. Our goal is to identify the top influential users in Twitter's network and determine which models and algorithms perform best on the dataset.

2 DATA DESCRIPTION

The dataset is a sample of the Twitter follower network in 2020. It's a directed, unweighted graph with nodes representing Twitter accounts of members and associates of the extremist group Proud Boys. The directed links represent the following relationship between users. In the dataset, the arrows point from the followers to the accounts they are following. This is the opposite of the direction defined in diffusion models and for this reason the direction of the edges was inverted.

Agents are users' accounts, each of which has two states determined by whether they have been presented with some piece of information. Thus we define the two states as follows:

1. An uninformed state describes those agents who have not been exposed to the information but are likely to see it if the information is re-tweeted by other agents that they are following.

2. An Informed state describes those who have seen the information and also are able to re-tweet it and spread it.

3 THEORY

This section focuses on the theory behind the result of this project. The first six concepts are diffusion models, which are used to study how information is spread through a complex network, and the last three are influenced maximization algorithms, which are used to identify the subset of nodes in a complex network that is most influential in terms of spreading information.

3.1 Independent Cascade Model

The Independent Cascade (IC) Model (Kempe et al. 2003) is a probabilistic model. The information spreads according to probability $P_{u,v}$, which is the probability of node u passing the information to node v. Under the IC model dynamics, at each time step t, the nodes' state will change according to the informed nodes at time step $t-1$ and the probability. Note that, if S_{t-1}^{new} is the set of newly activated nodes at time step $t-1$, only the neighbors of nodes in S_{t-1}^{new} can be informed/activated.

As the example shown in Figure 1, active nodes are in cyan which can activate other nodes. Yellow nodes are activated in this time step. Blue nodes are activated before but cannot activate other nodes. For the first time step, node A is activated. In the next time step, node B and node E can be activated by node A with probability 0.8 and 0.5. Node B has successfully been activated at time step 2, but node E has not. At time step 3, node C and node D are activated by node B. But node A cannot try to activate node E once again. In the last step, node D tries to activate node E with a probability of 0.5 but failed and no new node is activated in this step, so the diffusion process stops.

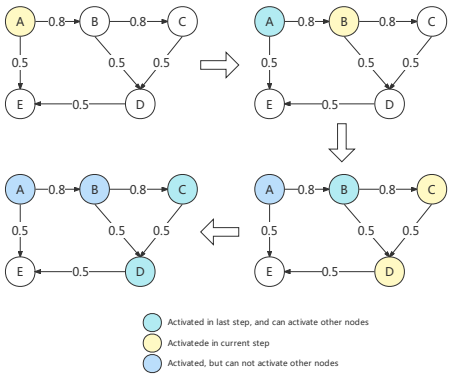


Fig. 1. Independent Cascade Model

Normally, The probability can be assigned according to geographic closeness, past infection traces, or the frequency of infection. For our Twitter data set, if one user follows n users, s/he has a probability proportional to $1/n$ to be informed by one of them. Specifically, the probability is equal to the reciprocal of the natural logarithm of the in-degree of the user.

3.2 Decreasing Cascade Model

For Decreasing Cascade Model(Kempe et al. 2005), it is similar to the IC model with exception that the probabilities change during the diffusion process. In fact, the "Decreasing" in the name of this model means the probability will decrease along with diffusion, so it is not "Independent". So for the probability $P_{u,v}$ of node u passing the information to node v, if more nodes try to inform node v, the $P_{u,v}$ for all possible u is reduced, which means it's 'Decreasing'. We define the probability as $P_{u,v}(u, S_u)$, where S_u is the subset of informed

neighbors of v. And DC model needs $P_{u,v}(u, S_u) \geq P_{u,v}(u, T_u), S \subseteq T$ to capture the property of diminishing return.

3.3 Weighted Cascade Model

The Weighted Cascade model is a special case of the Independence Cascade model. In the WC model, the probability $P_{u,v}$ is assigned as the in-degree $1/d_v$ of node v.

3.4 TRIVALENCY Model

The TRIVALENCY model (Chen et al. 2010a) is also similar to the Independent Cascade model, but for the TRIVALENCY model, the probability of each node is assigned to a randomly from 0.5, 0.3, 0.1, which represents high, medium, and low influence levels.

3.5 Linear Threshold Model

The Linear Threshold Model (Kermack and McKendrick 1927) is a mathematical model describing how information or influence spreads through a social network. It assumes that each directed edge $e(u, v) \in E$ has a non-negative weight $w(u, v)$ indicating the influence of the node to the node at the end of the edge, and each node selects a threshold $\theta_v \in [0, 1]$ determining how likely it is to be activated in the spreading process. Under the LT model dynamics, at each step t, if $S_t^{activated} S_t^{wait}$ are collections of activated nodes and non-activated nodes respectively, nodes are in two statuses, either activated or waiting, since $S_t^{activated} + S_t^{wait} = S$, where S is the set of all nodes. At the step t, where $S_{t-1}^{activated}$ is a set of informed nodes, and $u \in \eta^{in}(v)$ if $\exists e(u, v)$, nodes in S_{t-1}^{wait} will be informed if $\sum_{u \in S_{t-1}^{activated} \bigcap \eta^{in}(v)} w(u, v) \geq \theta_v$, where $\eta^{in} v$ is the set of node v's neighbors.

3.6 Generalized Threshold Model

The Generalized Threshold Model is a more flexible and general version of the Linear Threshold Model. Same as LT model, in GT model (Pathak et al. 2010) node v selects a threshold $\theta_v \in [0, 1]$. While the difference is that each node in the GT model has more than one threshold. For example, except for the threshold $\theta_v^{activated}$ determining whether it can be activated, another one θ_v^{spread} is defined to determine whether enable this node to spread information. Under the GT model dynamics, at each step t, nodes are in three statuses, wait S_t^{wait}, activated $S_t^{activated}$, or spread S_t^{spread}. In this case $S_t^{wait} + S_t^{activated} = S$ still holds, since $S_t^{spread} \subseteq S_t^{activated}$. Given node v, it will be informed at time step t if $\sum_{u \in S_{t-1}^{activated} \bigcap \eta^{in}(v)} w(u, v) \geq \theta_v^{activated}$. At the same time, it will be a spreading node if $\sum_{u \in S_{t-1}^{activated} \bigcap \eta^{in}(v)} w(u, v) \geq \theta_v^{spread}$.

In our project, the activation and spread threshold of each node is related to its degree centrality. According to the performance of different threshold values, we take $centrality/20$ as the activation threshold, and the spreading threshold doubles the activation, which is $centrality/10$.

3.7 Naive Greedy Algorithm

For Naive Greedy Algorithm, it starts with an empty node set S and returns a list of k nodes as the most influential nodes. In each step, it iterates every node outside S and selects the node which has the greatest influence to add in the next step adding into S. The meaning of having the greatest influence is the node v together with S can activate the most number of new nodes in the next time step of the diffusion model. The process is shown in Algorithm 1.

3.8 Cost-effective Lazy Forward

The CELF algorithm utilizes the sub-modular property of the influence maximization objective function to reduce the number of assessments on the influence spread of nodes, where CELF needs to repeatedly calculate the marginal influence spread of each candidate node in the node-selecting process using Monte Carlo simulations which will give accurate results, but also makin it time consuming. However, according to the paper [Leskovec et al. 2007], the efficiency of CELF algorithm is 700 times faster than the naive greedy algorithm. Therefore, it is still a valid algorithm to analyse the influence maximization.

The pseudocode 2 of CELF helps explain the main idea. It has a queue Q(k, mg) where k is a node and mg is the most recently calculated marginal gain of the node k. The queue is sorted by marginal gain of the nodes in the decreasing order. If the selected node in the loop has marginal influence computed in the current iteration,

which means that it has the maximal marginal influence of all the other nodes, even if their gains were computed in the previous iterations. The reason for this is that CELF is using sub-modularity and the marginal gains those nodes will have on the current set should not be larger than the gains on the smaller set from previous iteration. Consequently, it is possible to eliminate re-computation costs to calculate marginal gain for each node.

Algorithm 1 Naive Greedy Algorithm

Input: $k \in N^+$ and Network $G(V, E)$
Output: Seed set S for diffusion with k nodes inside

Initialization: $S \leftarrow \emptyset$;
for $i = 1$ to k **do**
  $min \leftarrow \infty$
  for v in $V - S$ **do**
    $num =$ diffusion one step with $S + v$ as seed
    if $num < min$ **then**
      $min = num$
      $node = v$
    end if
  end for
  $S \leftarrow S + node$
end for

Algorithm 2 CELF Algorithm

Require: $k \in N^+$ and Network $G(V, E)$
Ensure: Seed set S for diffusion with k nodes inside

Initialization: $S \leftarrow \emptyset$;
Initialization: $Q \leftarrow \emptyset$;
for v in V **do**
  $mg \leftarrow$ f({k})
  $Q \leftarrow$ (k, mg)
end for
for $i = 1$ to k **do**
  while $|S| < k$ **do**
    n = Q [0] [0]
    $mg = f(\text{S} \cup \{n\}) - f(\text{S})$
    Resort Q
    $S \leftarrow S \cup \arg\max_{v \in \mathcal{V}} f(S \cup \{n\}) - f(S)$
  end while
  $S = S \cup \{n\}$
  $Q = Q \cap \{n\}$
end for
return S

3.9 Maximum Influence Arborescence

MIA (Chen et al. 2010b) is an algorithm that aims to improve upon previous influence maximization techniques by only considering the local influence regions of nodes in a network. Previous attempts have mostly used approximation techniques, such as Monte-Carlo simulations. However, MIA proposes a more sophisticated approximation that requires shorter running times, allowing it to be applied to larger networks.

To achieve this, the algorithm starts by considering the different building blocks that allow for a localized study of the influence region of a node. The first building block is the propagation probabilities of each edge, which are determined by some criterion exogenous to the algorithm. These probabilities are used to define the propagation probability of a path as the joint probability of the node at the beginning of the path influencing the one at the end.

Armed with this concept, it is possible to consider any given node and study its likelihood of influencing any other node it can reach through an existing path. However, since the propagation probabilities of paths are calculated as the product of all the probabilities of edges within it, the result of this operation tends to be smaller and smaller for edges with low probabilities and/or paths composed of many edges. As a consequence, certain paths can have probabilities that may be negligible in the dataset's context, therefore, to account for this and to further optimize the runtime of the algorithm, a threshold θ was introduced: any paths under the threshold are not considered during its execution.

The next building block is the MIP (Maximum Influence Path), which is defined as the path with the highest probability between any two nodes u and v. The MIIA (Maximum Influence In Arborescence) and the MIOA (Maximum Influence Out Arborescence) are then defined as the union of the maximum influence paths ending at and starting from a particular node, respectively. In other words, MIIA(v, θ) represents all of the paths through which node v can be influenced by other nodes with a likelihood higher than the threshold, and MIOA(v, θ) represents all of the paths through which v can influence other nodes, again, with a likelihood higher than the threshold. The size of a node's local influence region can therefore be controlled by altering the value of the threshold, θ.

Using these concepts, the algorithm calculates the Activation Probability, or the probability that a node will be activated by a given set of seed nodes, through recursive calculations.

Given that this is an influence maximization algorithm, its goal is to maximize the Influence Spread of the seed it gives as an output. In this case, the Influence Spread of a seed is defined as the sum of the activation probabilities of all nodes, therefore, the bigger it is, the stronger the spread.

The seed calculated by MIA is determined iteratively. During each iteration, we evaluate the potential increase in Influence Spread that each node can contribute, called the node's Incremental Influence Spread. We then select the node with the largest Incremental Influence Spread to be added to the seed, repeating this process until we have reached the desired number of nodes in the seed.

However, these Incremental Influence Spread metrics need to be updated at each iteration: when a node u is added to the seed, it only reaches the nodes in MIOA(u, θ), consequently, the Incremental Influence Spread of any other node needs to be updated only if it is in MIIA(v, θ) for some v $\in$ MIOA(u, θ).

By only considering the local influence regions of the nodes, the algorithm is able to approximate the Independent Cascade model while still requiring shorter running times.

4 RESULTS

The different influence maximization algorithms produced seeds containing the subsets of users which are expected to maximize the influence spread in the network. However, since their internal mechanisms and the diffusion models on which they rely behave differently, results can vary.

An analysis of the seeds generated by the 18 different combinations of maximization algorithms and diffusion models led to the construction of table 1, which contains the top 5 most influential users in the network, for which all model-algorithm pairs are in agreement, except for a small difference in the Trivalency model.

To illustrate this, Figure 2 contains a visualization of the entire network where the color reflects k-core (orange: 4, cyan: 3, green:2, dark blue: 1) and the size reflects the amount of people reached using MIA on the WC model. The labeled nodes are the top 5 influencers.

IC, DC, WC, LT, GT	TR
principe_giovan	principe_giovan
Premises187	Premises187
MoralDK	MoralDK
proudboy_	proudboy_
enrique_tarrio	GavinM_ProudBoy

Table 1. Top 5 users with the highest influence for the different diffusion models.

5 DISCUSSION

5.1 Models

Models	IC	DC	WC	TR	LT	GT
Type	Prob	Prob	Prob	Prob	Math	Math
Avg Probability or Influence	2.06^{-5}	2.06^{-5}	9.78^{-6}	2.49^{-5}	9.77^{-6}	9.77^{-6}
Std Probability or Influence	3.4^{-3}	3.4^{-3}	3.0^{-3}	3.1^{-3}	3.0^{-3}	3.0^{-3}
Avg Threshold	-	-	-	-	0.001	activate 4.14^{-5} spread 8.29^{-5}

Table 2. Difference between models

Table 2 lists the attributes of the graph fitted by each model. These numbers are calculated before the diffusion begins.

We tested the 6 models with 500 iterations and calculated averages. In Figure 3, we compare the average running time the average number of diffusion steps, and the average number of informed nodes for 6 different models as a function of the number of initial seed nodes.

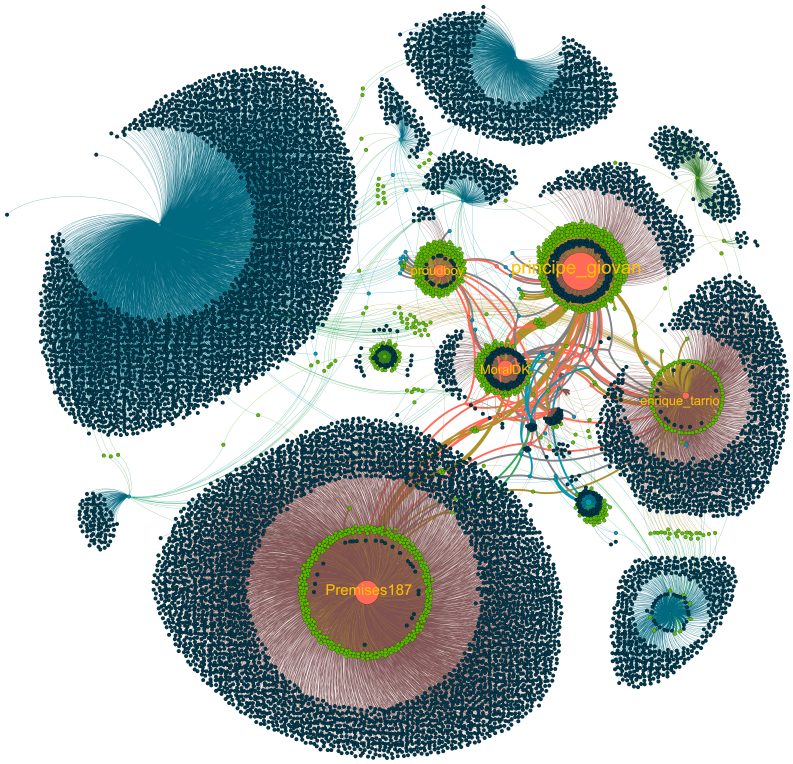


Fig. 2. Network visualization: color reflects k-core (orange: 4, cyan: 3, green: 2, dark blue: 1) and size reflects the amount of people reached using MIA on the WC model. The labeled nodes are the top 5 influencers.

It can be observed that all of the models show a similar increasing trend in Figure 3(a) and 3(c), the running times and a number of activated nodes increased with the number of seed nodes. This aligns with the common understanding that more seed nodes lead to a greater spread of information. The differences in the values on the y-axis can be attributed to the different activation probabilities or thresholds of the models.

Regarding specific models, The IC model and the DC model show similar trends in all three figures due to their similar activation probability. The main difference between these two models is that the probability of information spread in the DC model gradually fades over time, resulting in a small variance in the results.

The TR model has a larger average probability than other probability models, but it activated fewer nodes than IC and GC models. Because the TR model has random propagation probability for all nodes, those nodes with the greatest influence may not have the greatest probability to spread the information.

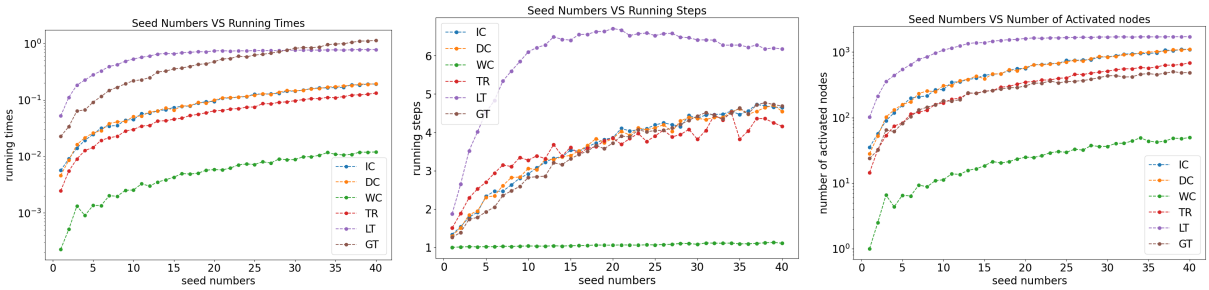
In comparison, the WC model has a much smaller probability of activation, leading to smaller numbers of activated nodes and diffusion steps compared to the other models.

For the LT model, Figure 3 shows that LT always gets the most running steps and activated nodes. Because nodes in LT will be influenced by all of the activated neighbors compared to those probabilistic models, in which neighbors' influence on the node is independent. Since GT has another threshold for spreading, which doubles the activation threshold, the running steps, and the final activated nodes are lower than LT.

Of particular note is the fact that the lines of LT in Figure 3(c) and 3(a) stabilize when there are more than 20 seeds, which means it can achieve the max activated node number with 20 nodes. At the same time, in Figure 3(b) we can see a slight drop in running steps. Because with the help of more nodes, it needs fewer steps to fully activate the network.

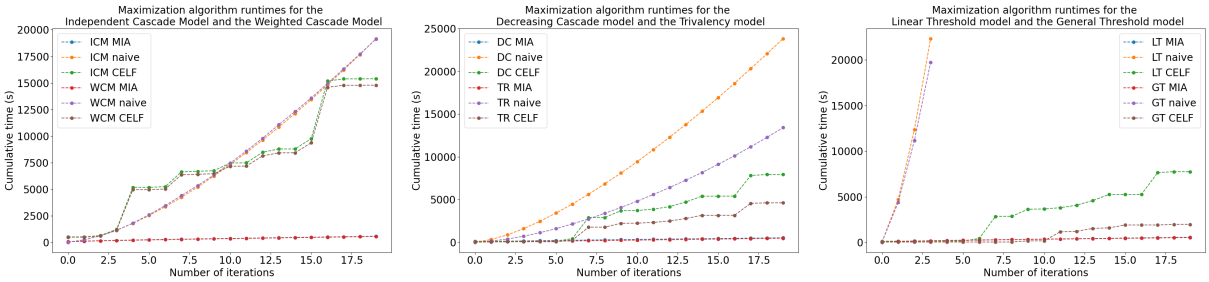
5.2 Models with Algorithms

Non-backtracking centralities and k-core centralities were used as a baseline. For the latter, nodes in the same shell were sorted by out-degree in decreasing order, because outgoing edges represent pathways that can be used to spread influence. This gives it an advantage in all models, except Trivalency, because their propagation probabilities are determined using degree values. Despite this advantage, all the Influence Maximization algorithms tend to outperform them.



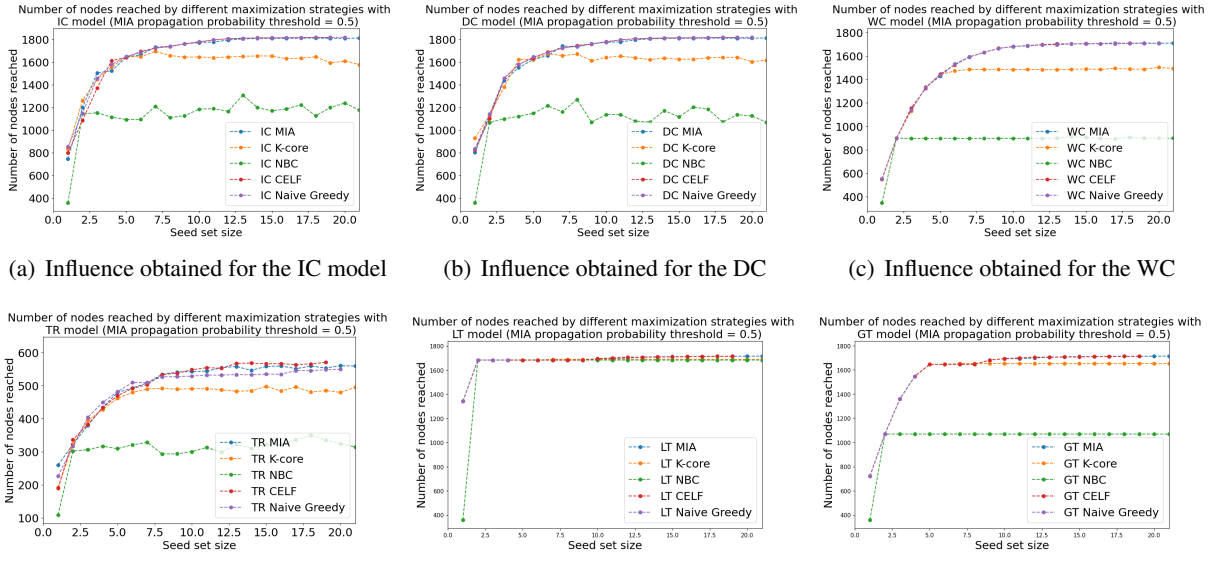
(a) Average Running Time Length (b) Average Number of Diffusion Step (c) Average Informed Nodes Number

Fig. 3. Compare Models By Random Selected Nodes



(a) run-times for IC and WC models (b) run-times for DC and TR models (c) run-times for LT and GT models

Fig. 4. Comparison of the algorithm run-times for different diffusion models



(a) Influence obtained for the IC model (b) Influence obtained for the DC (c) Influence obtained for the WC

(d) Influence obtained for the TR (e) Influence obtained for the LT (f) Influence obtained for the GT

Fig. 5. Comparison of the influence obtained by the different algorithms for different diffusion models

Figure 5 reflect the level of influence achieved by each influence maximization algorithm as a function of the size of the seed set (i.e. the set of individuals chosen as initial influencers) for different diffusion models, the run-time of the algorithms was limited to a maximum of 12 hours to limit the computational load, for this reason certain algorithms were able to calculate bigger seeds than others.

Greedy, CELF and MIA algorithms tend to perform the best at any given seed size. Since CELF doesn't compute spread for all nodes in each iteration, the computation time for CELF can be shorter when compared to Naive Greedy algorithm, while the influence is similar to each other, which can be seen in the plots.4 and 5,

Fig. 6. Correlation matrix showing the relationship between the individual influence of a user (measured as the amount of people they can reach individually) and different user properties.

but, notably, MIA takes only a fraction of the time to achieve results similar to the ones put forward by the Naive Greedy algorithm and CELF (Figure 4). This is reasonable given that MIA's goal is to be a heuristic alternative to Naive Greedy with similar or better performance and better scaling: the experiments presented confirm that it succeeds at its goal in this particular dataset.

5.3 Analysis of correlations between influence metrics and user attributes

After having determined the most influential nodes or individuals in the dataset, the correlation of their influence metrics with respect to other personal attributes can be evaluated. Three subsets of attributes were subject to consideration: Twitter metrics, such as the individual's follower count and the number of tweets; node centralities, such as degree, closeness, or betweenness centrality and node properties such as in-degree and out-degree.

In this case, the influence metrics are calculated according to the number of people each user can reach individually when their diffusion is simulated using diffusion models.

As far as Twitter metrics are concerned, Figure 6 shows how the influence metrics have little to no correlation to values like the number of followers, favorite tweets, statuses (tweets), or friends.

There is no correlation between follower count and influence, which seems counterintuitive given that common sense would suggest that users with more followers should have a greater influence. This could be due to two factors: first, not all followers are equally likely to share or redistribute information, as their own follower count and the influence of their followers can affect this. Secondly, the dataset only includes a small subset of Twitter users, many of whom are connected to the Proud Boys group, and is therefore not representative of the entire social network. The out-degree of a user in the dataset, on the other hand, may be a more accurate measure of their potential influence than the total number of followers they have across the entire platform.

In fact, Figure 6 shows the correlations of influence metrics to in-degree and particularly out-degree are very high despite having different absolute values. This is to be expected given that most models calculate probabilities based on degree values. The Trivalency model, on the other hand, assigns probabilities 0.1, 0.3, or 0.6 randomly, and therefore the correlation is not as strong, but still significant because outgoing edges represent pathways for influence in all models.

Finally, the correlations to other centrality metrics, show a similar story, with all models being moderately correlated to degree-centrality and more so to betweenness centrality. This is sensible given that the latter ascribes centrality based on whether a node is on many shortest paths between other nodes in the network, and this reflects a node's ability to route information, which is akin to the objective of influence maximization.

6 AUTHOR CONTRIBUTIONS

N.L and Y.L proposed the initial project idea and wrote the proposal.

During the execution of the project, D.G.P refined the proposal by deciding what research question to answer, which analyses and plots were needed for it, and in general decided how to structure the results and discussion part. D.G.P also proposed to add the WC and TR models which hadn't been considered initially.

E.G.H was responsible to develop the CELF algorithm, D.G.P implemented the MIA algorithm, Y.L was responsible for the Linear threshold as well as general threshold models, N.L implemented the Naive Greedy algorithm, and the IC, DC, WC, and TR models.

Y.L assisted N.L for the experiment of models comparison. E.G.H and Y.L wrote the abstract and introduction. Y.L and N.L prepared a data description which is used in the report, with some additional input from D.G.P. In the theory part, everyone filled the subsections of the parts that correspond to the models or algorithms they implemented. Results were written by D.G.P along with the creation of the visualization of the network. 'Models discussion' was written by Y.L and N.L, while 'models with algorithms' was written by D.G.P. Analysis of correlations between influence metrics and user attributes was written by D.G.P.

Plots in the discussion part were reviewed and accepted by all members.

Most of the presentation slides were prepared by Y.L with the help of N.L. And Y.L added the subtitle to the video with the help of N.L and D.G.P. The author's contribution was written by E.G.H with the help of D.G.P.

D.G.P completed a final read-through of the document to improve coherence with the help of N.L.

All authors revised and accepted the final version of this document.

REFERENCES

Chen, Wei, Chi Wang, and Yajun Wang (2010a) "Scalable Influence Maximization for Prevalent Viral Marketing in Large-Scale Social Networks". In: *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '10. Washington, DC, USA Association for Computing Machinery, pp. 1029–1038. ISBN: 9781450300551. DOI: 10.1145/1835804.1835934.

– (2010b) "Scalable influence maximization for prevalent viral marketing in large-scale social networks". In: *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1029–1038.

Kempe, David, Jon Kleinberg, and Éva Tardos (2003) "Maximizing the spread of influence through a social network". In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146.

– (2005) "Influential nodes in a diffusion model for social networks". In: *International Colloquium on Automata, Languages, and Programming*. Springer, pp. 1127–1138.

Kermack, William Ogilvy and Anderson G McKendrick (1927) "A contribution to the mathematical theory of epidemics". In: *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character* 115.772, pp. 700–721.

Leskovec, Jure, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance (2007) "Cost-effective outbreak detection in networks". In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 420–429.

Pathak, Nishith, Arindam Banerjee, and Jaideep Srivastava (2010) "A Generalized Linear Threshold Model for Multiple Cascades". In: *2010 IEEE International Conference on Data Mining*, pp. 965–970. DOI: 10.1109/ICDM.2010.153.

Singh, Shashank Sheshar, Divya Srivastva, Madhushi Verma, and Jagendra Singh (2022) "Influence maximization frameworks, performance, challenges and directions on social network: A theoretical study". In: *Journal of King Saud University - Computer and Information Sciences* 34.9, pp. 7570–7603. DOI: https://doi.org/10.1016/j.jksuci.2021.08.009.