Influence Maximization in Social Network with Ground-truth Community

Xin Guo and Nai-Chia Chen
xguo4@alumni.jh.edu, chen1945@umn.edu

1 Introduction and Motivation

Influence maximization is to find a small set of most influential nodes in a social network that maximizes the aggregated influence in the network. Such network influence process has many important applications and attracts interests from both industry and academia [Chen et al., 2010, Gionis et al., 2013, Kempe et al., 2003, Singer, 2012]. For example, a company may want to design a "word-of-mouth" marketing strategy to promote their products or service through social networks. The company needs to know who should be targeted by the promotion with limited budget. To successfully achieve their goals, it is crucial to study, model, and simulate the network influence process.

Nowadays, online social networking sites, e.g., Facebook, LiveJournal, and Linkedin, serve as an important platform to spread information, news, and new technology. Different from traditional social networks, users in online social network site form small communities with their common properties, e.g., users in one community may be from the same high-school, the same company, or the same club; users in another community may have the same hobbies or interests. One user may also be in several different communities simultaneously.

The community structure in online social networking plays a significant role on the dynamics of information spreading. Users in the same community may also be friends. When one person receives news or acquires new techniques, he/she may be more likely to spread the information through the online community. Other persons in the same community, even not a friend, may be influenced with high probability, whereas friends not in the same community may be influenced with low probability. Nevertheless, online social networks were only treated as simple undirected graphs in previous studies [Chen et al., 2010, Mao & Zhang, 2016, Singer, 2012, Song et al., 2016] and the effect of community is ignored completely.

On the other hand, online social networks usually are huge, e.g., Facebook has more than one billion users; LiveJournal has more than 10 millions unique monthly visitors. The simulation of network influence process on such huge networks requires dramatic computational resource. To reduce the simulation time, it is important to have smart and scalable algorithm to select initial influential nodes and simulate the network influential process.

In the present project, we are interested in investigating the algorithm to select initial influential nodes with ground-truth community information. The report is organized as follows. In section 2, we discuss the related works and their relation to the present project. The proposed algorithms to select initial influential nodes are described in section 3. We summarized our key findings and results in section 4. The conclusion and future work are in section 5. Section 6 lists the contribution.

2 Related Work

The studies on influence maximization mainly focus on two aspects, i.e., descriptive influence model, and efficient heuristics to choose initial nodes to influence. Kempe et al. [2003] first systematically studied the influence maximization problem as a discrete optimization problem and proposed two families of classic stochastic influence models, i.e., linear threshold model and independent cascade model. They showed that the influence maximization problem is NP-hard. Many previous works focused on the characteristic of the influence models and their approximation guarantees [Kempe et al., 2015, Singer, 2012, Song et al., 2016]. For the greedy hill-climbing algorithm, both of the two models yield $1 - e^{-1}$ approximation guarantee [Kempe et al., 2015, Mossel & Roch, 2010]. Besides the family of the two classic models initialized by Kempe et al. [2003], researchers also proposed other types of models, e.g., triggering model [Kempe et al., 2003, 2015], linear threshold model with a directed acyclic graph [Chen et al., 2010], voter model [Singer, 2012], and coverage model [Singer, 2012] to mimic the influence spreading process.

Researchers proposed heuristics to choose initial nodes to approximate network influence to be maximum and extensively compared the performance with different influence models. Many algorithms are proposed, e.g., degree-centrality algorithm [Chen et al., 2010, Gionis et al., 2013, Kempe et al., 2003], free-degree algorithm [Gionis et al., 2013], random walk with restart algorithm [Gionis et al., 2013], min-S and min-Z algorithms, distance-centrality algorithm [Kempe et al., 2003], and PageRank algorithm [Chen et al., 2010, Mao & Zhang, 2016]. The baseline of the study is to choose initial nodes randomly [Chen et al., 2010, Kempe et al., 2003, Leskovec et al., 2007, Singer, 2012]. The best result is always given by the greedy hill-climbing algorithm [Gionis et al., 2013, Kempe et al., 2003, Singer, 2012, Song et al., 2016]. However, the greedy hill-climbing algorithm is extremely slow. For some cases, e.g., the CAMPAIGN problem, the computational time of the greedy algorithm is super-quadratic and it is hard to scale to large data set [Gionis et al., 2013].

To improve the simulation time and model estimation accuracy, Leskovec et al. [2007] proposed a "lazy-forward" algorithm in selecting new seeds. They used the idea that most of outbreaks only affects a small area of the network to reduce the function evaluations. Although Leskovec et al. [2007] showed that their algorithm can be hundreds times faster than the greedy algorithm and only needs very small memory, the algorithm is still slower than the none-greedy algorithms and is not scalable when the data size is dramatically large. Nevertheless, if we consider the influence spreading through a community, the small area affected by a outbreak is most likely in the same or neighboring communities.

Chen et al. [2010] proposed the first scalable influence maximization algorithm based on the linear threshold model. In the algorithm, a local directed acyclic graph (LDAG) is constructed surrounding every node in the network and the influence to the nodes is restricted within the local DAG structure. They claimed that this algorithm can be done in time linear to the size of the graph and is much faster than previous algorithms. Nevertheless, they admitted in the paper that social networks in reality is typically not DAG. So it is hard to say that the information spreading in real-world social networks does follow the DAG as described by the algorithm. Meanwhile, the LDAG is still NP-hard and in the simulation, the LDAG has to be built around each node. As shown in the paper, the simulation time is still much longer than the non-greedy algorithms.

Yang & Leskovec [2012] studied the overlapping of network communities in real-world social networks. In real-world networks, the nodes in a community are densely connected and the edges highly concentrate around the community members. They found that the overlapping part of the communities are much denser than the nonoverlapping part. They showed that in ground-truth communities, the more communities two nodes share, the higher probability the two nodes are connected.

In summary, those properties of community in real-world social networks discussed in Yang & Leskovec [2012] inspire us to investigate if there is a good way to select initial seeds based on the information within ground-truth communities. Such idea is within the same domain as Leskovec et al. [2007] and Chen et al. [2010], in both of which the influence is assumed to only happen in a small area around a node, although the selection of the small area in their work did not consult community information or was arbitrary defined. Using the community information also follows the idea that community structure plays a key role in the influence spreading.

3 Model

We propose methods to utilize community information to select initial seeds. A property of community is that a community is densely-connected internally, and therefore, it is easier to infect a community than an arbitrary set of nodes with the same size.

3.1 Initial Seed Selection Heuristics with Ground-Truth Community

In method M1 and M2, instead of applying heuristics to the whole graph to select initial seeds, we first decompose the whole graph into sub-graphs, select the top n-largest sub-graphs, and then apply heuristics to the sub-graphs. With the ground-true community information, we propose the following two algorithms to build sub-graphs:

(M1) For each ground-true community, we construct a sub-graph, the nodes of which are members of the community, and the edges are those edges in the whole graph whose both ends belong to the community.

(M2) For each ground-truth community, we construct a sub-graph, i.e., the edges are those edges in the whole graph one of its ends belong to the community and the nodes are the members of the community and their first degree friends.

One benefit of the above two methods is that the running-time and memory are reduced significantly comparing to apply heuristics to the whole graph since each sub-graph is much smaller than the whole graph. Meanwhile, since each sub-graph can be processed independently, parallel computing can further accelerate the simulation.

The idea of M1 and M2 algorithms is to maximize influence by infecting the largest communities. On the other hand, another idea is to maximize influence by infecting as many communities as possible. We propose another method, M3, as described below:

(M3) We rank the nodes in the network according to the number of communities a node belongs to. The initial seeds are the top-n nodes.

3.2 Community Selection Heuristics

In the algorithms described in section 3.1, we select communities based on the size of communities. As evidenced in our experiments result in section 4.2, there are other factors that impact how influential a community is.

In this section, we study more sophisticated methods to select communities and initial seeds. We propose to first build up a "community" network with the following two algorithms:

(M4) We construct an undirected unweighted graph, where each node represents a community. For the nodes not belonging to any community in the whole graph, we consider each such node as a community. Two "community" nodes are connected by an edge if any pair of nodes from the two communities are connected in the original graph.

(M5) We construct an undirected weighted graph, where each node represents a community. Two "community" nodes are connected by an edge if any pair of nodes from the two communities are connected in the original graph, and the weight of the edge is the number of edges between the two communities in the whole graph.

With the community graph, we select the initial seeds by following the steps below:

1 We apply initial seed selection heuristics on the "community" graph and select the influential "community" nodes.

2 We apply the M1 or M2 algorithms described in section 3.1 to build up sub-graphs.

3 We apply the initial seed selection heuristics to the selected sub-graphs.

4 Results and Findings

In this section, we first introduce the properties and interesting findings of the experiment data in section 4.1. Then we shows the results of the M1, M2, and M3 algorithms in section 4.2. Finally, we study the performance of the algorithm M4 and M5 in section 4.3.

4.1 Research Data and Experiment Parameters

We used the LiveJournal social network data with the ground-truth community information [Yang & Leskovec, 2012]. The network has $3,997,962$ nodes and $34,681,189$ edges (snap.stanford.edu/data/com-LiveJournal.html). The average clustering coefficient and diameter are 0.2843 and 17, respectively. There are $664,414$ communities in the network. The largest community has $178,899$ members. Figure 1 shows the cumulative distribution function of the number of nodes in communities. The nodes are mainly in a flew communities, i.e., about 10 communities have more than 10000 nodes and about 10000 communities (i.e., top 1%) have more than 100 nodes.

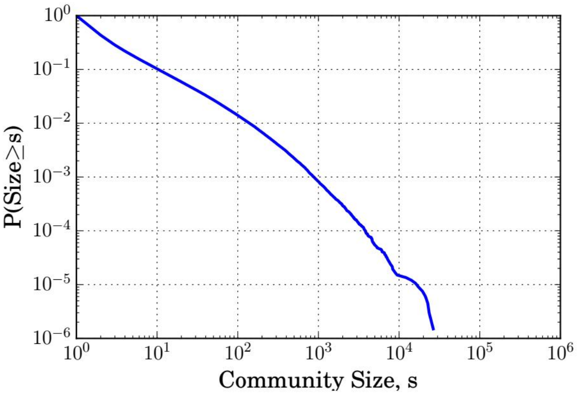


Figure 1: Complementary cumulative distribution function of community sizes.

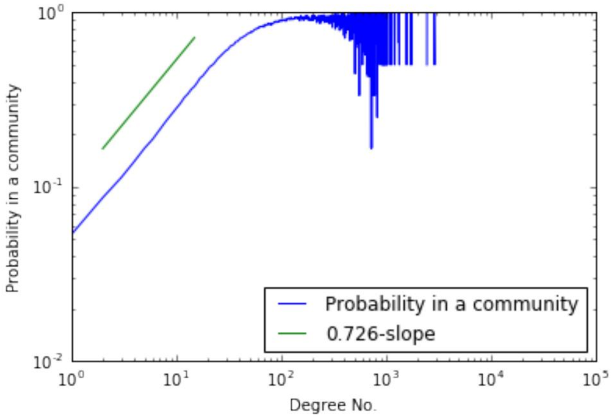


Figure 2: Probability of nodes belonging to a community VS node degree.

Figure 2 shows the distribution of the probability of nodes belonging to a community as a function of node degree. As the node degree increases, the probability increases. Therefore, a node with high degree is most likely in a community. Meanwhile, our statistics show that in most communities (specially the large ones), 95% edges have one node in one community and the other node in a different community. This result implies that targeting at the nodes in a community not only helps to spread the influence in a community but also has high probability to influence the nodes outside the community. The above two observations serve as the support to our idea to select initial seeds with ground-truth community information.

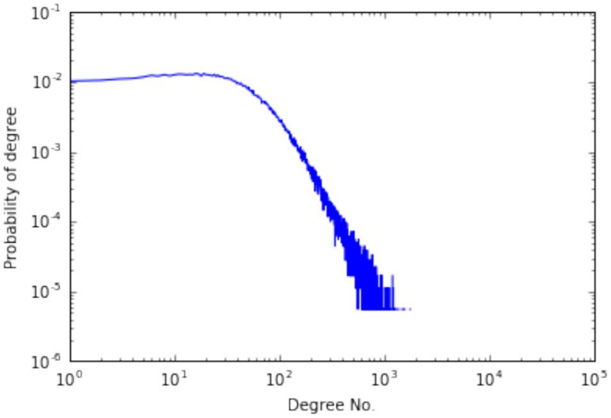


Figure 3: Distribution of the degree probability of the nodes in the largest community.

One interesting result we found is the degree probability of nodes in large communities. The degree here is the value from the whole graph. We have checked the top 10 largest communities. Figure 3 shows the degree probability distribution of the nodes in the largest community as an example. The following discussion also applies to the rest 9 communities. As shown in figure 3, there is a almost constant probability for low degree nodes. As degree increases, the degree probability slightly increases and reaches a maximum value of about 2% around degree 100. For the nodes with degree greater than 100, the degree probability decreases significantly, as degree increases.

In the experiment, we apply the independent cascade (IC) model and the linear threshold (LT) model. In the IC model, the probability of a node being activated by its neighbor is 0.01 [Kempe et al., 2003]. In the LT model, the threshold a node to be activated is chosen independently at random in $[0, 1]$. For both the IC and LT models, we use randomly selected seeds as baseline and we apply the high-degree heuristics, the Eigenvector-centrality heuristics, and the PageRank heuristics to choose the initial seeds. In all the simulations, due to the computational resource limitation, we stop the influence spreading simulation after 5 iterations.

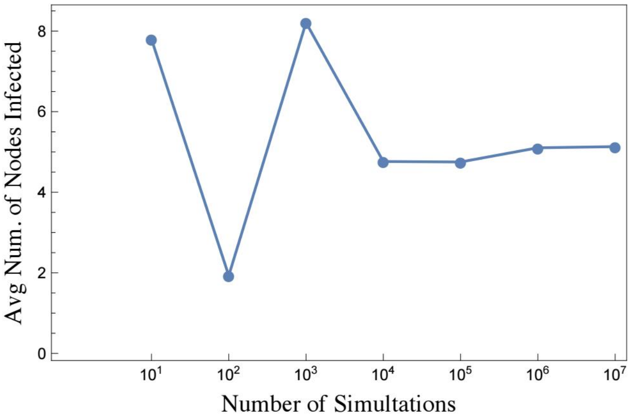


Figure 4: Number of influenced nodes of IC model with 1 initial random seed after 5-step simulation.

To determine the sufficient number of simulation to obtain statistically steady results for all experimental configurations, we first repeat the simulations 10^7 times with randomly-selected initial seeds. Figure 4 shows an example of IC model with 1 initial random seed. We observe that the average proportion of infection stabilizes around 5 after 10^4 simulations. Therefore, here and hereafter, all the simulations are repeated at least 10^4 times unless specified otherwise.

4.2 Initial Seed Selection Heuristics with Ground-Truth Community

Figure 5 shows the result of the IC model with three different heuristics—the degree centrality, Eigenvector-centrality, and PageRank heuristics—based on the whole graph or subgraphs. We also plot the result with the initial seeds from M3 algorithm and with the randomly-selected initial seeds as baseline. With the whole graph, the degree-centrality heuristics performs

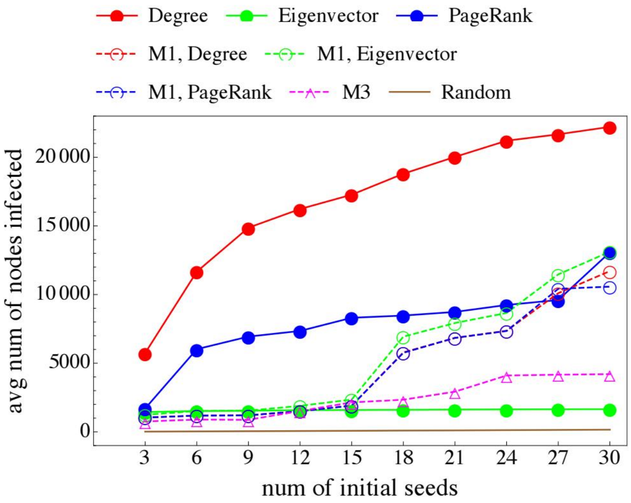


Figure 5: Independent Cascade model

better than the PageRank heuristics, and the Eigenvector-centrality heuristics influences least nodes. All the three heuristics perform much better than the baseline.

For the degree-centrality heuristics, the M1 result is only half of the one from the whole graph. As discussed in section 4.1, there are large amount of edges pointing outside the community. Therefore, the node with the highest degree in a community does not have to have the highest degree in the whole graph. For the PageRank heuristics, the M1 result is very similar to the whole graph result. It implies that for the PageRank heuristics, the M1 result is a good estimation for the whole graph. For the Eigenvector-centrality heuristics, interestingly, the M1 result significantly outperforms the result from the whole graph. We believe that the result can only be understood by checking the properties of the selected nodes, which can be part of the future work. With the M3 algorithm, the result only outperforms the Eigenvector-centrality heuristics with the whole graph and the baseline. For all the M1 results, when the number of initial seeds is greater than 15, the number of the influenced nodes increases significantly as the number of initial seeds increases comparing to the case when the number of initial seeds is less than 15. This phenomenon shows that the size of a community is not the only important factor to the influenced set. It is important to wisely select the communities to apply heuristics to have the largest influence.

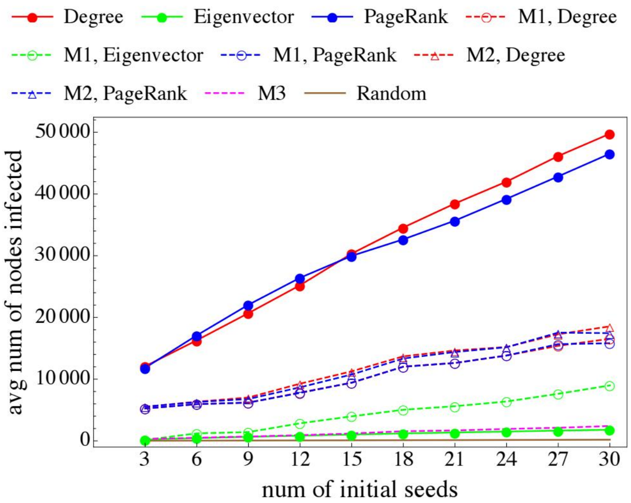


Figure 6: Linear Threshold model

Figure 6 plots the result of the LT model. For the result based on the whole graph, the variation of the result is similar to the one of the IC model, whereas the scale is different.

The comparison of the whole graph result and the M1 result shows that the M1 result of the Eigenvector-centrality heuristics is much better than the one of the whole graph, whereas the M1 results of both the degree-centrality and the PageRank heuristics are not. The behavior of the M3 algorithm is similar to that of the IC model.

In figure 6, we also plot the M2 result of the degree-centrality heuristics and the PageRank heuristics. Compared to the M1 result, the M2 algorithm slightly improves the result. However, the improvement is limited and there is still large different from the best result (i.e., the degree-centrality heuristics and the PageRank heuristics) of the whole graph.

Similar to the IC model, for all the M1 and M2 results, when the number of initial seeds is greater than 9, the number of the influenced nodes increases significantly as the number of initial seeds increases comparing to the case when the number of initial seeds is less than 9.

In summary, both the M1 and M2 algorithms have potential to find good initial seeds only based on the local community information. For example, the M1 Eigenvector-centrality heuristics result outperforms the result of the whole graph and the result of the M1 PageRank heuristics with the IC model can be similar to the whole graph one.

4.3 Community Selection Heuristics

In this section, we study the influence result based on the algorithms described in section 3.2. That is,

1. We first build up two "community" networks with the M4 and M5 algorithms, respectively.
2. Secondly, we apply the degree-centrality to the "community" networks built in step 1 and select the top communities.
3. Then, We build up local community sub-graphs with the M1 and M2 algorithms described in section 3.1.
4. Finally, we apply the degree-centrality heuristics, the Eigenvector-centrality heuristics, and the PageRank heuristics to the selected local community sub-graphs to select initial seeds for influence simulation.

To avoid confusion, we use the following naming rules. If a "community" graph is built with the M4 algorithm and a local community graph is built with the M1 algorithm, we name the result as "M4M1" result. So we have "M4M1", "M4M2", "M5M1", and "M5M2" algorithm combinations.

Figure 7 shows the variation of the averaged influence number of the M4M1 and M5M1 algorithms with the number of initial seeds. For comparison purpose, we also plot the M1 result. When the number of initial seeds is small, the result is similar among the M1, M4M1, and M5M1 algorithms. When the number of initial seeds is large, the difference is obvious. For both the degree-centrality heuristics and the Eigenvector-centrality heuristics, the M4M1 result is similar to the M1 result, whereas the M5M1 algorithm performs much better than the other two. Figure 8(a) shows the size of the influence set of the M1, M4M1, and M5M1 algorithms, when the number of initial seeds is 30. The comparison between the results of the M1 and M5M1 algorithms shows that the M5M1 result improves the influence size by about 30% in the degree-centrality heuristics and the PageRank heuristics and by about 12% in the Eigenvector-centrality heuristics.

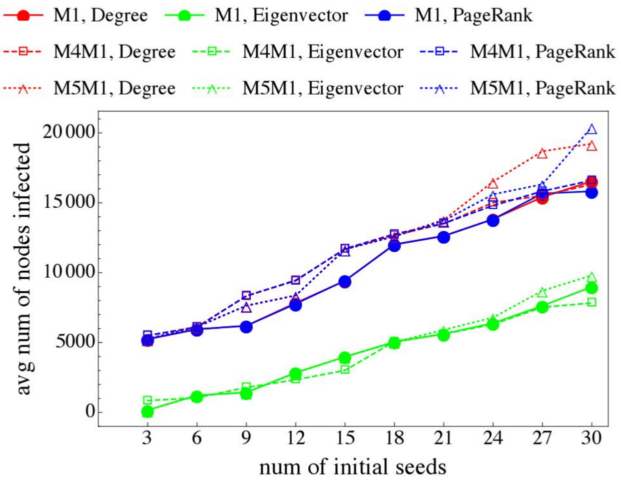
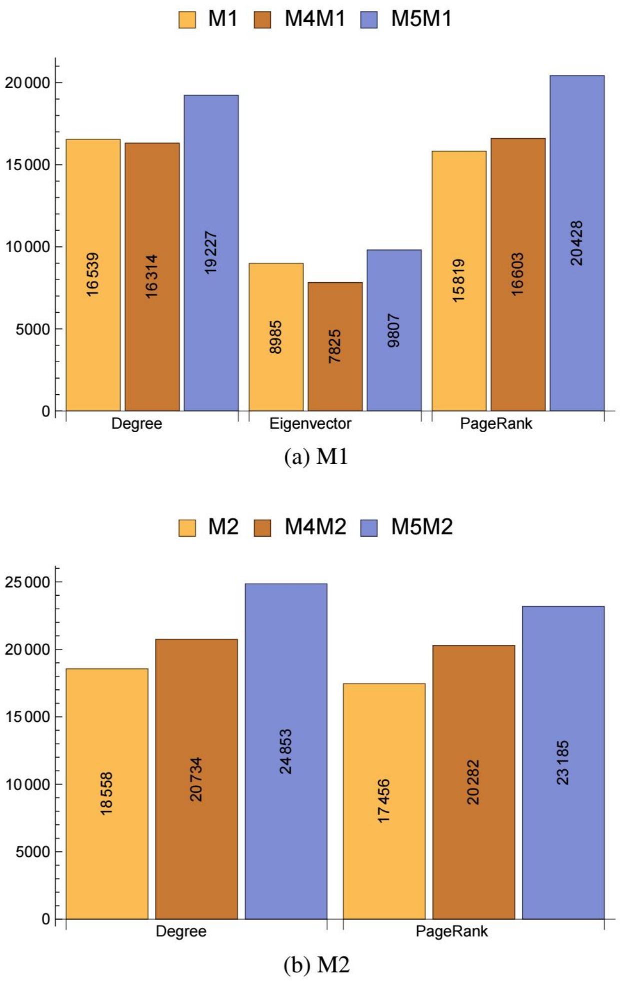


Figure 7: Variation of the averaged influenced node number of the M1, M4M1, and M5M1 algorithms with the number of initial seeds. The local community is built up with the M1 algorithm.

Figure 9 shows the variation of the averaged number of influence nodes of the M4M2 and M5M2 algorithms with the number of initial seeds. For comparison purpose, we also plot the M2 result. When the number of initial seeds is small, the result is similar among the M2, M4M2, and M5M2 algorithms. When the number of initial seeds is large, the difference is ob-



(a) M1

(b) M2

Figure 8: Number of the influenced nodes for the LT model with 30 initial seeds

vious. For both the degree-centrality heuristics and the Pagerank heurstics, the M5M2 algorithm perform the best, and the M4M2 algorithm performs better than the M2 algorithm. As shown in figure 8, when the number of initial seeds is 30, compared to the result of the M2 algorithm, the M5M2 algorithm improves the influence size by about 30% in both the degree-centrality heuristics and the PageRank heuristics.

In summary, the M5M2 algorithm outperforms the M1, M2, M4M1, M5M1, and M4M2 algorithms. This results shown in the present report provides the evidence that by improving the heuristics to select communities, an algorithm designed in the present project to select initial seeds may reach comparable or better results as the algorithm based on the whole graph. Although the result of the M5M2 algorithm still has some distance to the influence results of the high-degree centrality heuristics and the PageRank heuristics based on the whole graph, which are the best result of the LT model, the present work shines a light on the proposed algorithm. Since both the "community" graph and the local community sub-graph is much smaller than the whole graph, the running-time and memory of the initial seed selection heuristics can be improved significantly. Such algorithm also uses the ground-true community information which mimics the information spreading in the online social network.

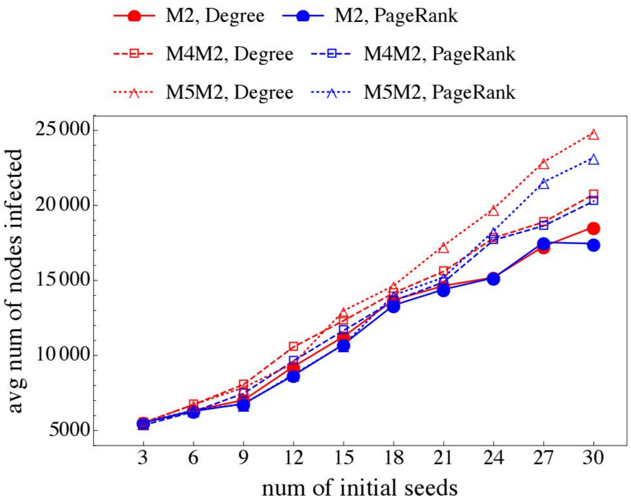


Figure 9: Variation of the averaged number of the influenced nodes of the M2, M4M2, and M5M2 algorithms with the number of initial seeds. The local community is built up with the M2 algorithm.

5 Conclusion and Future Work

In the present project, we proposed and studied the algorithms to use ground-true community information to select initial seeds for influence maximization problem. The philosophy of the algorithms is to first select the "influential" communities, then build local "community" sub-graphs with the nodes in or close to the community, and finally select the initial nodes from the "community" sub-graphs. We proposed 2 algorithms to build up local "community" sub-graphs, i.e., the M1 and M2 algorithms (section 3.1). We proposed 3 ways to select "influential" communities. One is based on the size of community, and the other two are the M4 and M5 algorithms (section 3.2). In addition to the above algorithms, we also propose to select initial nodes based on the number of communities they belong to.

By applying the proposed algorithms and comparing their performance, we conclude that by improving the heuristics to select the "influential" communities, the influence set can be improved significantly. Although the result we obtained still has some distance to the best result of the whole graph, the present study encourages future work to apply more algorithms of various centrality definitions for weighted graph to improve the aggregated influence.

Due to the time and computational resource limitation in the present project, we only apply the degree-centrality heuristics on both the unweighted "community" graph (built by the M4 algorithm) and the weighted "community" graphs (built by the M5 algorithm). Opsahl et al. [2010] proposed the definition of degree, closeness, and betweenness centralities for weighted graph. Their algorithms involve a tuning parameter for various research purposes. Applying the algorithms proposed in Opsahl et al. [2010] is one good direction to go in the future.

6 Contributions

XG : Coming up with the algorithms, coding up the algorithms, running experiment, writing up the report

NCC : Running experiment, plotting graphs during data analysis, revising the report

References

Chen, W., Yuan, Y., and Zhang, L. Scalable influence maximization in social networks under the linear threshold Model. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, pp. 88–97, 2010.

Gionis, A., Terzi, E., and Tsaparas, P. Opinion maximization in social networks. In *Siam International Conference on Data Mining*, 2013.

Kempe, D., Kleinberg, J., and Tardos, É. 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, 2003.

Kempe, D., Kleinberg, J., and Tardos, É. Maximizing the spread of influence through a social network. *Theory of Computing*, 11:105–147, 2015.

Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., VanBriesen, J., and Glance, N. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 420–429, 2007.

Mao, G.-J. and Zhang, J. A pagerank-based mining algorithm for user influences on micro-blogs. In *Proceedings of Pacific Asia Conference on Information Systems*, pp. 226, 2016.

Mossel, E. and Roch, S. Submodularity of influence in social networks: From local to global. *SIAM Journal on Computing*, 39:2176–2188, 2010.

Opsahl, T., Agneessens, F., and Skvoretz, J. Node centrality in weighted networks: Generalizing degree and shortest paths. *Social Networks*, 32: 245–251, 2010.

Singer, Y. How to win friends and influence people, truthfully: Influence maximization mechanisms for social networks. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pp. 733–742, 2012.

Song, C., Hsu, W., and Lee, M. L. Targeted influence maximization in social networks. In *Proceedings of Conference on Information and Knowledge Management*, 2016.

Yang, J. and Leskovec, J. Community-affiliation graph model for overlapping network community detection. In *Proceedings of 2012 IEEE International Conference on Data Mining*, pp. 1170–1175, 2012.