Topic-based Targeted Influence Maximization

Balaji V Srinivasan*, Anandhavelu N*, Aseem Dalal†,
Madhavi Yenugula‡, Prashanth Srikanthan‡ and Arijit Layek§
*Adobe Research India Labs, Bangalore
†Indian Institute of Technology, Delhi
‡Indian Institute of Technology, Madras
§Indian Institute of Technology, Kharagpur

***Abstract*—Social Networks play a very important role as a medium to propagate information among people. Marketers use this to campaign for their products and influence customers. However, it is not practically possible for a marketer to reach out to each and every individual prospective/existing customer due to the sheer size of the networks (in the orders of millions or billions). Therefore, marketers reach out to a small set of people (influencers) who have the potential to further influence/reach out to the targeted customers. Practically, it is not just enough if these influencers have a large following, they also need to have to be able to influence people in the topic that is relevant to the marketer and the influencer must be able to address the target segment that the marketer is targeting. In this paper, we first analyze various edge weighting mechanisms to incorporate influencing probability and utilize this to propose an algorithm to find influencers to maximize the spread to a specified set of targets.**

I. Introduction

Social media has changed the way people interact with each other and has contributed greatly towards shrinking the online world. The advent of platforms such as Facebook, Twitter, Pinterest, Linkedin have provided excellent platforms for people to voice their opinions and disseminate information. It has been estimated that over 27% of time spent online is on social media. This has encouraged business enterprises to leverage the customer presence on social media for brand marketing. Social platforms are increasingly being used by marketers to campaign for their products and influence customers on brand value. However, due to its sheer size, it is not practically possible for a marketer to reach out to each and every individual prospective/existing customer due to the sheer size of the networks (in the orders of millions or billions). Under these scenarios, marketers reach out to a small set of people (influencers) who have the potential to further influence/reach out to the targeted customers. This problem has been widely studied in the literature as the "Influence Maximization". The problem itself is NP-hard and existing literature deals with several approximations to the core optimization problem.

The foremost challenge in this problem is the scalability of the algorithms, since the underlying networks span several millions of users. The influence maximization algorithms can be categorized into two; one which accesses only the local properties of a user to assess his influence and another that accesses the position and role of the user in the network and its activities to determine his/her influence. The former obviously is less intense in computations; but the algorithms are far from what is expected in terms of the identified influencers. The latter on the other hand provides better solutions at the expense of increased computational cost. The work in this paper is based out of the approach due to Chen et al. [1] which addresses the computational cost to some extent.

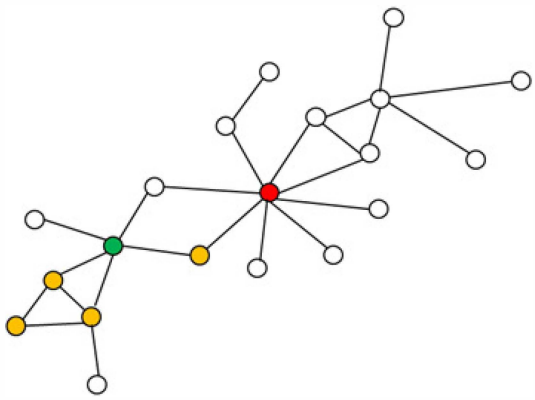


Fig. 1. A sample network with connections. If an algorithm has to select 1-influencer to maximize the reach across the network, the solution could possibly be the node colored in "red". However, if the objective is to maximize the reach to a specific set of targets colored in "yellow", the solution should be the "green" node and not the "red" node.

A common assumption in this domain is the availability of a network with edges indicating the "probability of influence". Most of the existing literature assume a uniform influence acros all the followers. This leads to a case where the influencers are based on the number of connections and remain the same irrespective of what the subject to be influenced is. However, marketers have various tasks to market, and each one will have different influencers. For example, one cannot expect a movie star to promote a software product, and even if the star does, the star's credibility on that is questionable. It is therefore not enough if these influencers have a large following, they also need to have to be able to influence people in the topic that is relevant to the marketer. To address this, it is necessary to incorporate these factors into the edge weights. Intuitive factors of influence are often misleading due to their statistical insignificance. In this paper, we study this problem and use an edge weight that can incorporate the topical credibility/interest in the edge weights.

Another challenge is to identify the influencers for a specific set of targets. Consider the network in Fig. 1. If an algorithm has to select 1-influencer to maximize the reach across the network, the solution could possibly be the node

978-1-4799-3635-9/14/$31.00 ©2014 IEEE

colored in "red". However, if the objective is to maximize the reach to a specific set of targets colored in "yellow", the solution should be the "green" node and not the "red" node. This leads to the primary objective of this paper: identify credible influencers to maximize the reach to a specified set of targets. Note that, the influencer to maximize the reach to a set of targets **need not necessarily be a part of the targets**.

For example, suppose a software company wants to run a promotion to their customers in California, its "targets" are the customers in California, and message is the "Software Promotion". It is necessary to find someone who will be listened to by the community on the software promotion, a random celebrity will not solve the problem. Further, the influencer to the community on Software may well be a blogger from New York (and may not belong to the identified targets). However, if the blogger is influential enough, it is necessary to use him to seed the promotions.

The paper is organized as follows. We discuss the related work and the existing gaps in Section II. We describe the Maximum Influence Arborescence algorithm from [1] in Section III and extend it to identify influencers for a specified targets in Section IV. We elaborate our edge weighting mechanism in Section V before discussing our experimental observations in Section VI. Section VII concludes the paper.

II. Related Work

Finding influencers in a network has been a widely studied problem. There are two versions of this problem, one where the top-k nodes to maximize the spread of a message is identified, and other where a set of nodes that will spread the message to λ of the audience is identified. The former is the "Top-k node" selection problem and the latter is the "λ coverage" problem.

Domingos and Richarson [2] first studied the problem of assigning a influence score to customers for designing viral marketing strategies. They modeled social networks as Markov random fields where the probability of an individual adopting a technology (or buying a product) is a function of both the intrinsic value of the technology (or the product) to the individual and the influence of neighbors. Their objective was to find a subset of nodes that would best trigger cascade of further adoptions and found it effective for viral marketing. Influence maximization has since been extensively studied in literature however, scalability of the algorithms is a major concern.

Kempe et al [3] posed this problem as a discrete optimization problem. Given a directed graph $G = (V, E, w)$ and an input k, the influence maximization problem is to find a subset $S^* \subseteq V$ such that $|S^*| = k$ and $\sigma_1(S^*) = \max\{\sigma_1(S) \mid |S| = k, S \subseteq V\}$, *i.e.* to maximize the spread of a information if the information is seeded to them. They showed it is NP-hard and provide approximation guarantees for efficient algorithms. They show that natural greedy algorithm approximates the solution with $(1 - \frac{1}{e} - \epsilon)$. Leskovec et al. [4] develop an efficient algorithm based on the submodularity of the underlying objective function by utilizing the underlying sparsity and by reducing the number of function evaluations using the submodularity of the influence functions. Chen et al. [1] present an efficient algorithm to find the top nodes in a social network and improves upon the [3] and [4]. We use this as the basis of our work in this paper.

The $\lambda - -$coverage problem is also widely studied [5][6]. Researchers have worked on the problems of finding the minimum number of influencers to influence at least J people [7] and finding the minimum number of influencers to influence a percentage λ of the nodes in the network [8]. Lappas et al. [9] define the k-effectors problem which selects a set of nodes called "effectors" to be activated so that a given activation state of the network may be achieved, which is a modified version of the λ coverage problem. They prove this problem to be NP-hard to solve optimally and also NP-hard to approximate for a general graph.

Although, influence maximization is studied widely in different flavors, the influence probabilities are not dealt with widely. Tang et. al [10] introduce the problem of topic-based social influence analysis. Given a social network and a topic distribution for each user, the problem is to find topic-specific subnetworks, and topic-specific influence weights between members of the subnetworks by using a Topical Affinity Propagation (TAP) approach using a graphical probabilistic model. Saito et al. [11] have studied the influence probabilities in the context of the Independent Cascade Model (ICM) of propagation in a Expectation-Maximization (EM) framework. While their formulation is elegant, it is not scalable to huge datasets.

III. Maximum Influence Arborescence

The Maximum Influence Arborescence (MIA) Algorithm was proposed by Chen. et. al [1] and has been shown to be computationally better than most algorithms. We have extended this algorithm to solve the influence maximization to a specified set of targets. We will briefly describe the MIA algorithm along with a few notations for the sake of completeness, for more details the reader is referred to [1].

DEFINITION 1: For a path $P_{uv} = \{p_1, p_2, \ldots, p_m\}$ from u to v, propagation probability of the path, $pp(P_{uv})$, is defined as

$$pp(P_{uv}) = \prod_{i=1}^{m-1} pp(p_i, p_{i+1})$$

This defines the probability that u activates v through path P_{uv} is $pp(P_{uv})$, as it needs to activate all nodes along the path. $P(G, u, v)$ denotes the set of all paths from u to v in a graph G.

DEFINITION 2: For a graph G, the maximum influence path $MIP_G(u, , v)$ from u to v in G as

$$MIP_G(u, v) = arg \max_{P_{uv}} \{pp(P_{uv}) | P_{uv} \in P(G, u, v)\}$$

For each edge (u, v) in the graph, if the propagation probability $pp(u, v)$ is translated to a distance weight $-\log pp(u, v)$ on the edge, then $MIP_G(u, v)$ is simply the shortest paths and shortest-path arborescences, and thus the Dijkstra algorithm can be used to compute them.

DEFINITION 3: For a given node v in the graph, the maximum influence in-arborescence (MIIA) is the union of the maximum influence paths to v, to estimate the influence to

v from other nodes in the network. An influence threshold θ is used to eliminate MIPs that have too small propagation probabilities. Symmetrically, maximum influence out-arborescence (MIOA) is defined to estimate the influence of v to other nodes. For an influence threshold θ, the maximum influence in-arborescence of a node $v \epsilon V$, $MIIA(v, \theta)$, is

$$MIIA(v, \theta) = \cup_{(u}$$

The maximum influence out-arborescence $MIOA(v, \theta)$ is:

$$MIOA(v, \theta) = \cup_{(u}$$

Given a set of seeds S in G and the in-arborescence $MIIA(v, \theta)$ for some $v \notin S$, Chen et al.[1] approximated the Independent Cascade Model (ICM) by assuming that the influence from S to v is only propagated through edges in $MIIA(v, \theta)$. With this approximation, the probability that v is activated given S is computed. The activation probability of any node u in $MIIA(v, \theta)$ denoted as $ap(u, S, MIIA(v, \theta)$ is defined as the probability that u is activated when the seed set is S and influence is propagated in $MIIA(v, \theta)$ and $N^{in}(u, MIIA(v, \theta))$ be the set of in-neighbours of u in $MIIA(v, \theta)$. Algorithm 1 computes $ap(u, S, MIIA(v, \theta)$ recursively.

Algorithm 1 Activation Probability$ap(u, S, MIIA(v, \theta))$

1: **if** $u \in S$, **then**
2: $ap(u) = 1$
3: **else if** $N^{in}(u) = \phi$ **then**
4: $ap(u) = 0$
5: **else**
6: $ap(u) = 1 - \prod_{w \in N^{in}(u}$
7: **end if**

An important step in the greedy algorithm is to select the next seed that gives the largest incremental influence spread. For a maximum influence in-arborescence $MIIA(v, \theta)$ of size t and a given seed set S, to select the next seed u, we need to compute the activation probability $ap(u, S \cup \{w\}, MIIA(v, \theta))$ for every $w \in MIIA(v, \theta)$, which takes $O(t^2)$ time by simply using Algorithm 1 to compute every $ap(v, S \cup \{w\}, MIIA(v, \theta))$. However, if the linear relationship between $ap(u)$ and $ap(v)$ in $MIIA(v, \theta)$ is utilized, $ap(v, S \cup \{w\}, MIIA(v, \theta))$ can be computed in $O(t)$ time by using the following lemma in Algorithm 2.

LEMMA 1: (INFLUENCE LINEARITY:) Consider $MIIA(v, \theta)$ and a node u in it. If we treat the activation probabilities $ap(u)$ and $ap(v)$ as variables and other $ap(w)'s$ as constants, where w is any node in $MIIA(v, \theta)$ other than u and v, then $ap(v) = \alpha(v, u).ap(u) + \beta(v, u)$, where $\alpha(v, u)$, $\beta(v, u)$ are constants independent of $ap(u)$

This algorithm performs the best when the arborescences are small. This typically occurs for a reasonable range of θ values, when the graph is sparse and the propagation probabilities on edges are usually small, which is typically the case for social networks; thus achieving significant scalability.

IV. Targeted Maximum Influence Arborescence

We have extended the MIA Algorithm [1] to maximize the influence to work in the case when targets to be reached are specified. We extended the algorithm to consider targets by considering the spread of influence from the other nodes only into the target nodes and constructed the trees of influence, called the MIA structures. Given a social graph G, edge weights w_{ij}, nodes T to be targeted and a positive integer k, the targeted influence maximization problem is to find a set of k nodes such that the expected influence on the set T is maximum.

Algorithm 2 Compute $\alpha(v, u)$ with $MIIA(v, \theta)$ and S, after $ap(u, S, MIIA(v, \theta))$ for all u in $MIIA(v, \theta)$ are known.

1: /* the following is computed recursively */
2: **if** $u = v$ **then**
3: $\alpha(u, v) = 1$
4: **else**
5: set w to be the out-neighbor of u
6: **if** $w \in S$ **then**
7: $\alpha(v, u) = 0$
8: **else**
9: $\alpha(v, u) = \alpha(v, w).pp(u, w).\prod_{u' \in N^{in}(} - ap(u').pp(u', w))$
10: **end if**
11: **end if**

For a threshold θ, $MIIA(v, \theta)$ and $MIOA(v, \theta)$ are the maximum influence in-arborescence and out-arborescence tree respectively as defined in [1]. Since we are dealing with a target set T, we want a set of seeds who have the maximum influence on these target nodes. So, we compute $MIIA(v, \theta)$ for only $v \epsilon T$ and $MIIA(v, \theta) = \varnothing \; \forall \; v \epsilon V \setminus T$. The modified algorithm is provided in Algorithm 3.

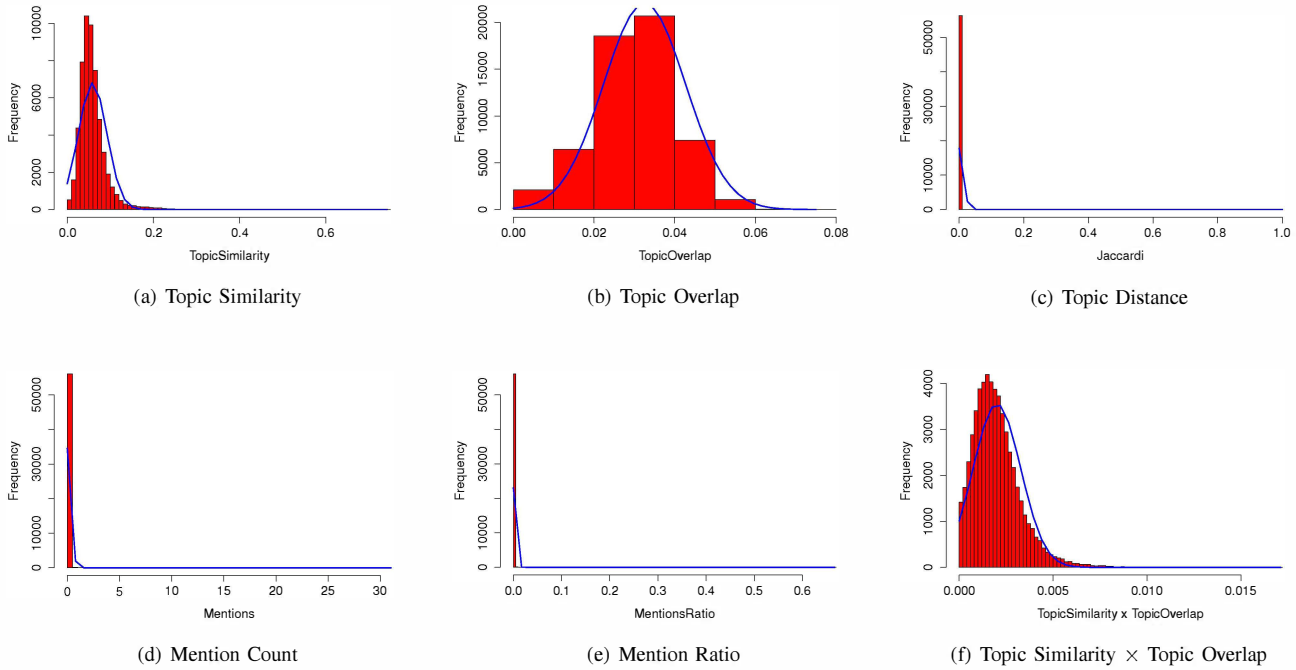
V. Edge weight computation

One of the important problems that has not been addressed well in the literature so far is the computation of edge weights in the graph. Edge weight of a directed edge from Node A to Node B is the probability with which A will be able to influence B into discussing about any given message. What social marketers have with them potentially is a huge set of public conversations on social networks like twitter. It turns out that it is pretty hard to compute the edge weights in a graph, given just the public conversations.

Further, the features that make intuitive sense to be good predictors of edge weight might have a distribution that is extremely skewed that would make it virtually impossible to use them for edge weight computation. Using a measure that has a skewed distribution like number of mentions can end up creating a sparse graph which is not usable for any further processing. In this section we study the distribution of features that make intuitive sense to be predictors of edge weight, if they can be computed from data and if they can be used for computing edge weight.

Our dataset consisted of the connections and tweets of a network formed by the followers of a particular twitter handle. Based on the connections, a social graph is constructed, where an edge in the graph indicates the presence of a connection. Our dataset had ~ 3000 nodes and ~ 56000 edges.

We consider two sets of features for computation of edge weight - ones that directly indicate the fact that a node has



(a) Topic Similarity (b) Topic Overlap (c) Topic Distance

(d) Mention Count (e) Mention Ratio (f) Topic Similarity $\times$ Topic Overlap

Fig. 2. Distributions of candidate features for computation of edge weight

engaged in a conversation with another and the ones that are based on topical interest of the nodes.

1) For the first set of features we took mentions and mentions ratio (the ratio of mentions of node A by node B to the total number of mentions by node A within a particular window of time).
2) For computing topical interest based metrics we ran Latent Dirichlet Allocation on the tweets with the set of tweets by each user as a document to obtain the latent topic proportions for each nodes. Three similarity measures were calculated from this - viz-a-viz., topic similarity, topic overlap and topic distance.
   a) Topic similarity of two nodes is the cosine distance between the topic proportions of those users.
   b) Topic overlap of two nodes is the fraction of the number of common topics (determined based on a threshold on the topic proportions) present in both the nodes' tweets to the total number of topics.
   c) Topic distance is the Jacardi-distance between the nodes' topics (determined based on a threshold on the topic proportions).

Fig.2 shows the distribution of the features we considered. As we can see from the figures, even though mentions and mention count are intuitively the obvious measures of edge weight it is not possible to use them as edge weights as the distribution is extremely skewed and will end up in creating a very sparse graph. Of the topical features, topic similarity and topic overlap have a distribution that can be used for computation of edge weights where as the topic distance is again skewed thereby making it unsuitable for computing edge weights. For our experiments, we used the product of the topic similarity and topic overlap as the edge weights. The resulting distribution is shown in Fig 2(f).

A. *Incorporating message-specificity to edge-weights*

In order to incorporate message-specificity, we marginalize the topic proportion of each user T_u with the topic proportion of the message that the marketer wants to spread T_m. If $T_u = \{t_{u1}, t_{u2}, \ldots, t_{uN}\}$ and $T_m = \{t_{m1}, t_{m2}, \ldots, t_{mN}\}$, the message specific topic proportion T_{mu} is given by,

$$T_{mu} = \{\frac{t_{u1}t_{m1}}{\sum_i^N t_{ui}t_{mi}}, \frac{t_{u2}t_{m2}}{\sum_i^N t_{ui}t_{mi}}, \ldots, \frac{t_{uN}t_{mN}}{\sum_i^N t_{ui}t_{mi}}\} \quad (1)$$

By incorporating this message specific edge-weights in the network, we will essentially get different influencers for different message. For example, Fig. 3 show the influencers in the network for a message on "Software" (Fig 3(a)) and for a message on "Business" (Fig 3(b)). It is evident that the influencers identified are different for the two message.

By this approach, we get an improvement of $10\% - -25\%$ across various topics (topics experimented include "Software", "Business"; each message is represented by a set of associated keywords for which the topic proportions are extracted). Fig 4 shows the spread for one case and it is evident that our approach gives significantly better spread than a centrality-based influencers.

VI. Performance of Targeted MIA

The performance of the targeted MIA was measured against MIA by simulating over the network with an independent

Algorithm 3 Targeted Influence

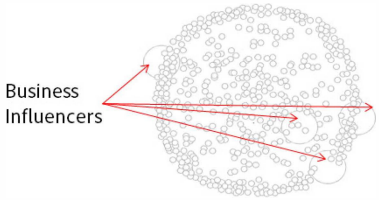
Require: $G = (V, E, pp)$
$S := \phi$
$IncInf(v) := 0 \ \forall v \in V$
for $v \in V$ **do**
  Compute $MIOA(v, \theta)$
end for
for $v \in T$ **do**
  Compute $MIIA(v, \theta)$
  $ap(u, S, MIIA(v, \theta) := 0 \ \forall u \in MIIA(v, \theta)$
  Compute $\alpha(v, u) \ \forall u \in MIIA(v, \theta)$
  for $u \in MIIA(v, \theta)$ **do**
    $IncInf(u) += \alpha(v, u)(1 - ap(u, S, MIIA(v, \theta)))$
  end for
end for
for $i = 1 \rightarrow k$ **do**
  Pick $u = argmax_{v \in}$
  for $v \in MIIA(u, \theta) \backslash S$ **do**
    for $w \in MIIA(v, \theta) \backslash S$ **do**
      $IncInf(w) - = \alpha(v, w)(1 - ap(w, S, MIIA(v, \theta)))$
    end for
  end for
  $S = S \cup \{u\}$
  for $v \in MIOA(u, \theta)$ **do**
    Compute $ap(w, S, MIIA(v, \theta)) \ \forall w \in MIIA(v, \theta)$
    Compute $\alpha(v, w) \ \forall w \in MIIA(v, \theta)$
    for $w \in MIIA(v, \theta) \backslash S$ **do**
      $IncInf(w) + = \alpha(v, w).(1 - ap(w, S, MIIA(v, \theta)))$
    end for
  end for
end for
return S

cascade model for several randomly chosen target sets. Fig.5 shows a comparison of the mean percentage of target nodes covered by each of these algorithms across multiple trials. As we can see, targeted MIA performs a lot better than MIA in effecting the spread across the set of targets, suggesting the need for targeted influencers for viral marketing.

VII. Conclusions

We have studied the distributions of different metrics that could be used for computing edge weights and applied them to a realistic social marketing scenario. We have also done a modification to the maximum arborescence to suit the real world needs and have shown that its performance is better than MIA.

(a) Software



(b) Business

Fig. 3. Influencers for different messages

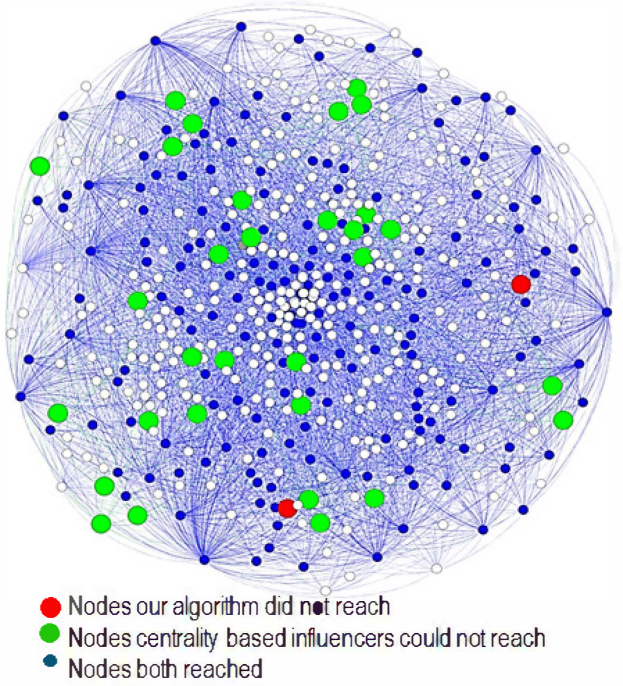


Fig. 4. Information Spread by influencers based on our approach vs based on centrality measures for "Software"

References

[1] W. Chen, C. Wang, and Y. Wang, "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*. ACM, 2010, pp. 1029–1038.

[2] P. Domingos and M. Richardson, "Mining the network value of customers," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2001, pp. 57–66.

[3] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2003, pp. 137–146.

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

[5] T. Broxton, Y. Interian, J. Vaver, and M. Wattenhofer, "Catching a viral video," in *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*. IEEE, 2010, pp. 296–304.

[6] S. Datta, A. Majumder, and N. Shrivastava, "Viral marketing for multiple products," in *Data Mining (ICDM), 2010 IEEE 10th International*

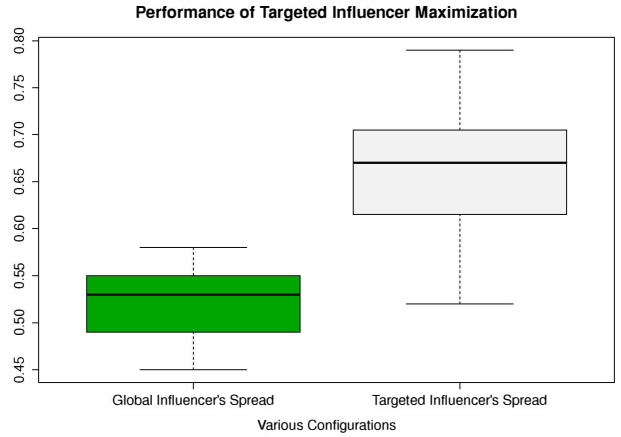


Fig. 5. Comparison of targeted MIA and MIA

Conference on. IEEE, 2010, pp. 118–127.

[7] C. Long and R.-W. Wong, "Minimizing seed set for viral marketing," in *Data Mining (ICDM), 2011 IEEE 11th International Conference on*. IEEE, 2011, pp. 427–436.

[8] R. Narayanam and Y. Narahari, "A shapley value-based approach to discover influential nodes in social networks," *IEEE Transactions on Automation Science and Engineering*, no. 99, pp. 1–18, 2010.

[9] T. Lappas, E. Terzi, D. Gunopulos, and H. Mannila, "Finding effectors in social networks," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 1059–1068.

[10] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social influence analysis in large-scale networks," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2009, pp. 807–816.

[11] K. Saito, R. Nakano, and M. Kimura, "Prediction of information diffusion probabilities for independent cascade model," in *Knowledge-Based Intelligent Information and Engineering Systems*. Springer, 2008, pp. 67–75.