A Potential-Based Node Selection Strategy for Influence Maximization in a Social Network

Yitong Wang[1] and Xiaojun Feng[2]

[1] Computer Science of Fudan University, Shanghai
ytwang@fudan.edu.cn
[2] Computer Science of Fudan University, Shanghai
072021144@fudan.edu.cn

Abstract. Social network often serves as a medium for the diffusion of ideas or innovations. The problem of influence maximization which was posed by Domingos and Richardson is stated as: if we can try to convince a subset of individuals to adopt a new product and the goal is to trigger a large cascade of further adoptions, which set of individuals should we target in order to achieve a maximized influence? In this work, we proposed a potential-based node selection strategy to solve this problem. Our work is based on the observation that local most-influential node-selection adopted in many works, which is very costly, does not always lead to better result. In particular, we investigate on how to set two parameters(θ_v and b_{uv}) appropriately. We conduct thorough experiments to evaluate effectiveness and efficiency of the proposed algorithm. Experimental results demonstrate that our approximation algorithm significantly outperforms local-optimal greedy strategy.

Keywords: social network, greedy algorithm, viral marketing, influence maximization, information diffusion.

1 Introduction

Social network is a social structure made of nodes that are tied by one or more specific types of relationship. In recent work, motivated by application to marketing, Domingos and Richardson posed a fundamental algorithmic problem for such systems [10], [20]. The premise of viral marketing is that by initially targeting a few "influential" members of the network, e.g. giving them free samples of the product and these "influential" members could trigger a cascading of influence since their friends will recommend the product to other friends and many individuals will try it as a result. So, now the problem is how should we choose k (the value of k is pre-defined) influential individuals as initial target set to maximize the process? Several node-selection strategies have been proposed based on node degree or characteristic of diffusion. Actually, we have to consider both factors during the process of node-selection: structure of network and diffusion model. There are two basic models: Linear Threshold Model(LTM) and

R. Huang et al. (Eds.): ADMA 2009, LNAI 5678, pp. 350–361, 2009.

Independent Cascade Model(ICM). We will introduce these models in Section 2. While a climbing-hill greedy algorithm that select the "most influential" node at each step has been proposed, we argue that local optimal (most-influential node which could provide the largest marginal increase at present) does not always lead to a global optimal and has an unacceptably high cost. Motivated by this observation, we propose a new algorithm by identifying some most potential nodes to maximize the spread of influence in a social network.

Section 2 introduces two most commonly used diffusion models and related work as well. In Section 3, we propose TW (Target Wise) greedy algorithm and present a new estimate b_{uv} of node influence. Randomized process is also described in Section 3. In Section 4, experiments are conducted on two datasets to evaluate the effectiveness of the proposed TW greedy algorithm. Finally, Section 5 draws some conclusions and discusses future works.

2 Background Knowledge

2.1 Two Fundamental Diffusion Models

We usually model the whole social network as a directed/undirected graph with each node representing an individual and edge between nodes representing some kind of relationship (friends, co-authorships etc.). Each node is marked active (an adoption of an idea or innovation) or inactive. When most adjacent nodes (neighbors) of one node are active, this node will also tend to be activated according to a pre-defined threshold.

Linear Threshold Model. LTM (linear Threshold Model) is at the core of many diffusion models based on node specific threshold [7], [8]. For a given social network that is modeled as an undirected graph $G(V, E)$, we define $N(v) = \{u|(u, v) \in E)\}$ as the neighbor set of node v and b_{uv} as influence of active node u on its inactive neighbor $v(\sum_{u \in N(v)} b_{uv} \leq 1)$. We also define $A(v)$ as set of active nodes in $N(v)(A(v) \subseteq N(v))$. Theoretically, for each node v, a node-specific threshold θ_v is defined. For a given node if $\sum_{u \in A(v)} b_{uv} \geq \theta_v$, node v becomes active. Intuitive meaning is that for an inactive node v, if total influence exerted by all its *active* neighbors exceeds a pre-defined threshold θ_v, node v becomes active. In turn, it will exert influence on its inactive neighbors and bring some inactive neighbors active again. This process will continue until no node can be activated. This diffusion model could be described briefly as follows: Given initial set A_0, b_{uv} and node-specific threshold θ_v, at step t of diffusion, activate nodes that meet the threshold based on A_{t-1}, and add these newly activated nodes to A_{t-1} to form A_t. The process continues until no more nodes can be activated. Obviously, θ_v is node-specific and highly dependent on each individual's personality. Moreover, each individual could exert different influence on his/her neighbors under various relation semantics (friends, co-authorship etc.).

Independent Cascading Model. Another diffusion model is Independent Cascade Model(ICM) based on interacting particle systems. It was first proposed

in [11] and [12], and the diffusion process is as follows: if node v is activated at step t and it then tries to activate all its inactive neighbors with success probability p_{vu} for each inactive neighbor u. If it is successful, then u will be active in step $t+1$, else, v failed and will no longer have chance to activate u. It is worth noting that p_{vu} is a system variable and is independent to other attempts to activate u made previously but failed. Each active node v has only one chance to activate its inactive neighbor u. David Kempe and Jon Kleinberg [14] think, however, p_{vu} should decrease with the diffusion process unfolding. That is, if node u has been attempted to be activated many times but all failed, the influence of newly active node v exerted on u will be weakened. And yet another new model named Decreasing Cascade Model was proposed. Influence maximization under the three models discussed so far are all NP-complete, see [12], [13]. While some other models are proposed [1], [2], [3], [4], [5], [6], [7], [8] and [9], they are all variations of the two core models we have introduced.

2.2 Related Work

A climbing-up Greedy Algorithm (KK Greedy Algorithm). For a predefined k value, we have to choose one node to place it into target set at each step. A naive approximate solution to influence maximization is to select the "most influential" node at each step. We use S to represent target set and define:

1) $S_0 = \emptyset$

2) $I(S)$: Set of active nodes throuth diffusion starting from target set S

3) $m(u|S) = |I(S \cup \{u\}| - |I(S)|$: node influence of u given S

Node influence of u given S is defined as *extra* nodes activated if adding u to S. $I(S)$ represents set of active nodes produced through diffusion starting from a given target set S. It is evident that "most influential" node is the node with the maximum node influence at the moment with current target set S. It is a local optimal. The goal of influence maximization is to choose S_k wisely with the goal of maximizing $I(S_k)$. Kempe and Kleinberg in [13] and [14] proposed a climbing-up greedy algorithm (we name the algorithm by initials of authors) based on the naive idea: choose most influential node at each step: starting from S_0, at step i, we choose node u as ith member of target set due to local optimal strategy: $u = \arg\max_v m(v|S_{i-1})$ and form S_i as $S_i = S_{i-1} \cup \{u\}$. S_k is the target set which could produce approximate maximized final influence. We say "greedy" since the most "influential" node we chose at each step is a local optimal in term of "influence" based on status of each S_i. However, it is very clear that the most influential node is very costly to obtain since we have to *compute* $I(S_i \cup \{u\})$ *for every inactive node* u *currently in the graph*. Even for a given node u, $I(S_i \cup \{u\})$ is also very costly since we need to trace the whole diffusion process.

Set Cover Greedy Algorithm and Shapley Value-Based Node Selection. Set Cover Greedy algorithm [19] was proposed for influence maximization under ICM. It kept choosing node with highest "uncovered degrees", once a node is

chosen, all its neighbors as well as itself are labeled as "covered". This procedure continues until k nodes are chosen. This algorithm is computationally fast and the underlying idea is that selected nodes cover the graph as much as possible. Obviously, "cover" does not imply "active". The paper [21] considers a special case of information maximization problem. For co-authorship network, given a value for k; we need to find a set of k researchers who have coauthored with maximum number of other researchers. The work in [21] does not involve diffusion process. The problem lies in the characteristic function, which is rather simple in this special case but computationally heavy in the general case. This makes it inapplicable to general cases. Heuristic approach is yet another node-selection strategy totally based on node degree.

3 Target Wise Greedy Algorithm

KK greedy algorithm is very costly. Computing of most influential node (local optimal) is extremely expensive since we have to consider all inactive nodes in G currently. Moreover, local optimal does not guarantee better results. We think it is more appropriate to choose some inactive nodes that might not be optimal at starting phase but could trigger more nodes in later stage of diffusion. Motivated by this understanding, we propose a novel algorithm named TW (Target wise) greedy algorithm under linear threshold diffusion model. We divide k steps of node selection in TW algorithm into two phases: "most potential" node selection (Phase 1) and "most influential" node selection (Phase 2). We introduce a parameter c ($[0, 1]$) to indicate the percentage of k steps in Phase 2. It is clear that TW greedy algorithm will be degenerated into KK greedy algorithm when c equals 1.

3.1 Potential-Based Node Selection Strategy

We define "potential" of node u as

$$p(u) \ = \sum_{v \in N(v),\ v \notin A(u)} b_{uv} \ . \tag{1}$$

$p(u)$ is the total possible influence that u could exert to all its inactive neighbors when u becomes active. The node with the maximum "potential value" is the most potential node. Obviously, the value $p(u)$ is determined by two factors: number of inactive neighbors of u at present as well as value of each b_{uv}. The more its inactive neighbors and the bigger each b_{uv}, node u has bigger "potential". As the title implies, we think the most potential node we selected in Phase 1 can accumulate some "influence" for future use since once it is chosen in target set, it could exert great influence on its inactive neighbors.

Target Wise Greedy Algorithm. We proposed target wise greedy algorithm based on potential-based node-selection strategy. The algorithm is depicted as Algorithm.1. The reason that we partition whole node-selection into two phases

Algorithm 1. Target Wise Greedy Algorithm

Input: Graph $G(V, E)$, threshold θ, influence b_{uv}, target set size k, parameter c
Initialize $S_0 = \emptyset$, $k_1 = k - \lceil ck \rceil$, $k_2 = k - k_1$
for $i = 1$ **to** k_1 **do**
  Choose node u, $u = \arg\max_v p(v)$
  $S_{i+1} = S_i \bigcup u$
  Update $p(v)$ for each v which has not been activated
end for
for $i = 1$ **to** k_2 **do**
  Choose node u, $u = \arg\max_v m(v|S_i)$
  make $S_{i+1} = S_i \bigcup u$
end for

is that: we want to select some nodes with great "potential" to bring much more nodes into active in later steps, while at present these "potential nodes" might not have largest influence. Exact running time of Target Wise greedy algorithm is difficult to obtain since it is related to how many nodes remained inactive at present as well as how to partition k steps into two phases in TW algorithm.

New Estimate of b_{uv}. Parameter b_{uv} is defined as the influence of active node u on inactive neighbor v and usually is estimated as 1/d(v) (d(v) is the degree of v), which means that for inactive node v, all its neighbors play the same roles and have the same influence on v. Obviously , this assumption does not hold in many real applications. We propose a new estimate of b_{uv} by taking account of not only how many neighbors of v but also how these neighbors connect to each other. We first review the concept of neighbor graph of a node: $NG(v)$ and then present the new estimate of b_{uv}. It is worth to be noting that degree of node in new estimate of b_{uv} is calculated based on neighbor graph.

$$NG(v) = G'(V', E'),\ V' = \{v\} \cup N(v),\ E' = \{(x, y)|x, y \in V',\ (x, y) \in E\}$$

$$b_{uv} = \frac{degree(u)}{\sum_{w \in N(v)} degree(w)}\ . \tag{2}$$

Fig. 1 gives neighbor graph of a node v. According to definition of b_{uv} given above, degree of u_1, u_2, u_3 in $NG(v)$ are 2, 2, 1, and accordingly, b_{u_1v}, b_{u_2v} and b_{u_3v} are 0.4, 0.4 and 0.2 respectively.

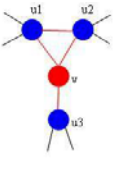

Fig. 1. An example of neighbor graph for node v

4 Experiments and Evaluations

4.1 Datasets

We conduct experiments on two datasets of social network and some statistics are shown in Table.1. We only consider an undirected graph in this paper. Dataset I is a medium-size dataset on yeast protein interaction, in which its social network effects are discussed in [15] and [16]. Dataset II [17] and [18] is a relatively large dataset. It can be seen from the table that dataset II is rather sparse while dataset I is much denser. We study influence maximization under linear threshold model. We first set b_{uv} as $1/d(v)$ and θ_v as $1/2$, which are commonly adopted in literatures. Parameter $c(0 \leq c \leq 1)$ indicates that Phase 2 has $\lceil ck \rceil$ steps and Phase 1 has $k - \lceil ck \rceil$ steps. We will evaluate experimental results in terms of final influence. As for running time, it is very clear that TW greedy algorithm is much faster than that of KK greedy algorithm since the cost of selecting "most potential" node is much lower than selecting "most influential" node as illustrated in their definition.

Table 1. Statistics of two data sets in Experiments

Data Set	node	edge	Average degree
Yeast protein interaction	2361(73 isolated point)	13292	11.6
Collaboration network in computational geometry	7343(1185 isolated point)	11898	3.9

4.2 Experimental Results

Joint Effects of c and k. We first investigate joint effect of parameter c and k based on final influence on two datasets and results are shown in Fig.2 and Fig.3. It is clear from Fig.2 that while KK greedy algorithm ($c = 1$) works better than the algorithm that are totally based on "potential" node selection($c = 0$), the proposed TW algorithm (including both most potential nodes and most

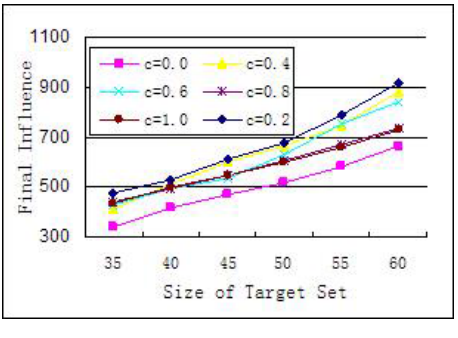

Fig. 2. Influence Curves with Various k and c values on dataset I

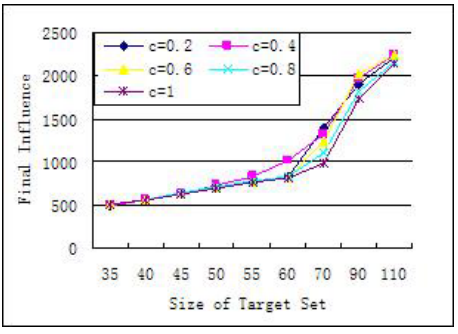

Fig. 3. Final Influence with difference c and k on dataset II

influential nodes) could beat KK greedy algorithm once c is positive and works much better when c is in the range 0.2 to 0.6.The proposed TW greedy algorithm achieves the best performance when c is around 0.2.

Fig.3 demonstrates almost the same phenomenon as that in Fig.2 except that the best performance is achieved when c is between 0.4 and 0.6. It indicates that when k is of appropriate medium-size, the proposed TW greedy algorithm outperforms KK greedy algorithm greatly even without the best c and the best performance could achieve around 15%-20% improvement in terms of final influence. Since dataset II is rather sparse, we could see from Fig.3 that when k is small, there is no big difference between TW and KK greedy algorithm and even for medium-size k, the improvement is also limited comparing with improvement in Fig.2.

Comparison between Different Algorithms. We compare TW greedy algorithms with other three related algorithms on dataset I. Our comparison is among four algorithms: TW greedy algorithm (c is 0.2), KK greedy algorithm ($c = 1$), heuristic and SCG algorithm (Set Cover Greedy) introduced in [19]. Heuristic

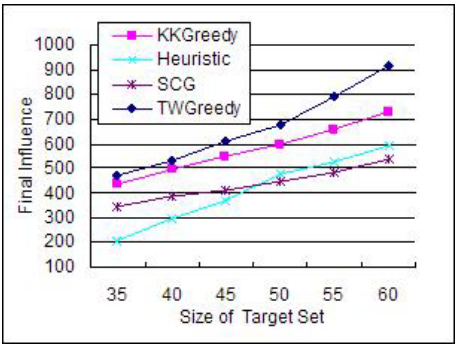

Fig. 4. Influence curves for different algorithms

algorithm is solely based on highest-degree node-selection. It was demonstrated in [14] that KK greedy algorithm is better than the heuristic algorithm since diffusion process should be also considered in addition to network feature. Our experimental results shown in Fig.4 also agree with this point.

We can see very clear from Fig.4 that TW greedy algorithm outperforms other three algorithms greatly. With increase of k (size of target set), advantages over other algorithms become even bigger and more evident.

Detailed Comparison between TW Greedy Algorithm and KK Greedy Algorithm. We investigate in detail about the difference between TW greedy algorithm and KK greedy algorithm to understand the merits of our approach more intuitively. The evaluation metric is average influence of set S(target set), which is defined as follow: $AI(S) = |I(S)|/|S|$, where AI represents "average influence". Average influence reveals how many active nodes could actually be triggered by each member of the target set on the average.

In TW greedy algorithm, we partition k steps of node-selection into two phases, $k - \lceil ck \rceil$ steps of most potential nodes selection and $\lceil ck \rceil$ steps of most influential nodes selection. We also partition k steps node-selection of KK greedy algorithm accordingly into $k-\lceil ck \rceil$ steps and $\lceil ck \rceil$ steps. We compare corresponding average influence of two target sets produced by two algorithms in terms of average influence at two phases. It is clear from Table.2 that given a fixed c (e.g. c is 0.1), average influence of TW greedy algorithm is a little bit lower than that of KK greedy algorithm at Phase 1(10.4 vs. 12.17), but at phases 2, average influence of TW greedy algorithm achieves much better (36.8 vs. 10.4). So in summary, TW greedy algorithm achieves much better performance since in Phase2, "influential" nodes could activate many more nodes due to "potential" accumulated in Phase1. We also found that average influence of TW greedy algorithm decreases with the increase of c. It is reasonable since the smaller c is; the bigger "potential" accumulated which will in turn influence more nodes.

Table 2. TW Greedy vs. KK Greedy based on average influence of two target sets (k is 50) on Dataset I

A(S)	TW Greedy		KK Greedy	
	Potential Accumulation	Potential Display	Corresponding Phase 1	Corresponding Phase2
c=0.1	10.4	36.8	12.17778	10.4
c=0.2	10.475	24.5	12.45	10.2
c=0.3	10.02941	20.9375	12.55882	10.8125
c=0.4	10.76667	17.15	12.8	10.8
c=0.5	10.48	13.16583	13.16	10.80769
c=0.6	10.84211	13.64516	13.26316	11.22581
c=0.7	11.8	12.28571	13.8	11.22857
c=0.8	12	12.175	14.6	11.35
c=0.9	13.6	11.82222	15.2	11.64444

Improvement Based on New b_{uv}. The effects of new estimate b_{uv} on two datasets are show in Fig.5 and Fig.6. We also present comparisons of TW greedy algorithm ($c = 0.5$) vs. KK greedy algorithm ($c = 1$) with old ($1/d(v)$) and new

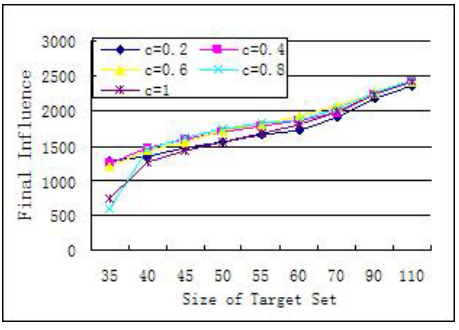

Fig. 5. TW greedy vs. KK greedy with new estimates of b_{uv} on dataset II

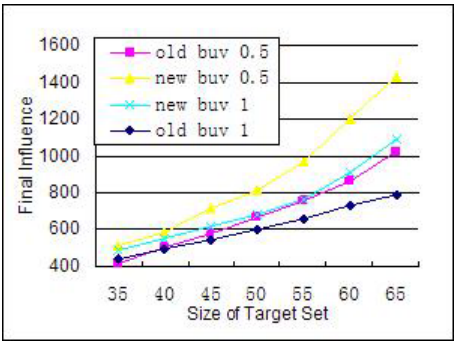

Fig. 6. TW greedy vs. KK greedy with two estimates of b_{uv} on dataset I

estimates (defined in section 3.2)of parameter b_{uv} on two datasets as shown in Fig.5 and Fig.6. It is very clear from the two figures as well as Fig.3 that the new estimate of b_{uv} is very effective on improvement of final influence. By combing Fig.3 and Fig.5, it is very clear that the proposed new b_{uv} is very effective on performance improvement. This is also true for Fig.6. Even for KK greedy algorithm, with the help of new estimate of b_{uv}, its performance also improves greatly. As we can see, the two datasets are very different in terms of network structure as well as data volume, however, the best performances are all achieved with c as 0.5 when new estimate b_{uv} is applied. Another message conveyed from Fig.5 and Fig.6 is that the improvement of TW over KK is more evident on dataset I than that of dataset II since structure of network plays a very important role during the diffusion process.

Randomize Node Specific Threshold θ_v. The randomized results shown in Fig.7 and Fig.8 are based on the average results of 5 attempts of randomizing. Combing Fig.3 and Fig.7, it is very clear that the effect of randomizing is very effective and double the final influence. Even for very small k, with randomized

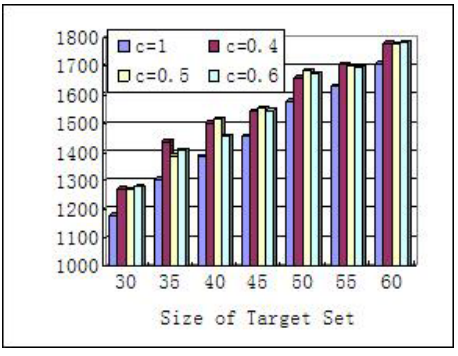

Fig. 7. TW greedy vs. KK greedy with randomized θ_v and old b_{uv} on dataset II

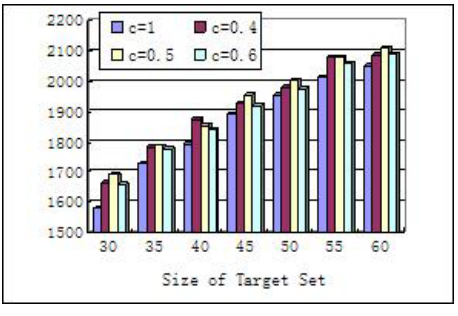

Fig. 8. TW greedy vs. KK greedy with randomized θ_v and new b_{uv} on dataset II

θ_v, the proposed TW achieves great results and outperforms KK greedy algorithm significantly.

It is evident from Fig.8 that while both randomized θ_v and new b_{uv} give better results independently, combing them could give much better results. Moreover, Fig.8 also conveys the message that the proposed TW algorithm is very stable and robust under new b_{uv} and randomized θ_v for different datasets.

5 Conclusion

In this paper, we proposed an approximate algorithm to influence maximization under a linear threshold model by identifying some "potential" nodes into the target set. While solutions to influence maximization under LTM and ICM models are NP-hard, we could only approximate the optimal result with various heuristics. KK greedy algorithm achieves 63% of the optimal result based on local optimal in terms of "influence", that is, at each step, it chooses the most "influential node" which could provide the largest marginal increase at present. However, by detailed check, we found that this local optimal is rather costly and not necessary. The proposed TW greedy algorithm is based on identifying

some most "potential" nodes at the starting stage of node selection. Moreover, selecting the most "potential" node is much cheaper and easier than selecting the most "influential node" as their definitions clearly show. We conducted experiments on two datasets to evaluate joint effects of parameters introduced (c, k). Comparison of the proposed TW greedy algorithm with other related algorithms demonstrates that our approach outperforms them significantly both in running time and final influence. We also investigate how to set appropriate parameters b_{uv} and θ_v. Our underlying idea is that b_{uv} is related not only to number of neighbors of v but also how these neighbors connect to each other. θ_v is highly node specific and should be randomized. We think the new estimate of b_{uv} and randomized are more reasonable and experimental results demonstrate that they are very effective on performance improvement independently and combine them can give a even better performance. While the proposed TW greedy algorithm is very effective and efficient, there is still much work needed to further investigate, such as how to extend influence maximization to a virtual social web (social network on the Web), where relationships between nodes are uncertain and community information are included.

Acknowledgement. This work is supported by China NSF Project (No. 60603042).

References

1. Granovetter, M.: Threshold models of collective behavior. American Journal of Sociology 83(6), 1420–1443 (1978)
2. Schelling, T.: Micromotives and Macrobehavior. Norton (1978)
3. Berger, E.: Dynamic Monopolies of Constant Size. Journal of Combinatorial Theory Series B 83, 191–200 (2001)
4. Morris, S.: Contagion. Review of Economic Studies 67 (2000)
5. Peleg, D.: Local Majority Voting, Small Coalitions, and Controlling Monopolies in Graphs: A Review. In: 3rd Colloq. On Structural Information and Communication (1996)
6. Macy, M., Willer, R.: From Factors to Actors: Computational Sociology and Agent-Based Modeling. Ann. Rev. Soc (2002)
7. Valente, T.: Network Models of the Diffusion of Innovations. Hampton Press (1995)
8. Peyton Young, H.: The Diffusion of Innovations in Social Networks. Santa Fe Institute Working Paper 02-04-018 (2002)
9. Watts, D.: A Simple Model of Global Cascades in Random Networks. Proc. Natl. Acad. Sci. 99, 5766–5771 (2002)
10. Domingos, P., Richardson, M.: Mining the Network Value of Customers. In: ICDM (2001)
11. Goldenberg, J., Libai, B., Muller, E.: Talk of the Network: A Complex Systems Look at the Underlying Process of Word-of-Mouth. Marketing Letters 12(3), 211–223 (2001)
12. Goldenberg, J., Libai, B., Muller, E.: Using Complex Systems Analysis to Advance Marketing Theory Development. Academy of Marketing Science Review (2001)

13. Kempe, D., Kleinberg, J., Tardos, E.: Influential nodes in a diffusion model for social networks. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005, vol. 3580, pp. 1127–1138. Springer, Heidelberg (2005)
14. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence in a social network. In: Proc. 9th Intl. Conf. on KDD, pp. 137–146 (2003)
15. Sun, S., Ling, L., Zhang, N., Li, G., Chen, R.: Topological structure analysis of the protein-protein interaction network in budding yeast. Nucleic Acids Research 31(9), 2443–2450 (2003)
16. Software package Protein Interaction Network PIN
17. Beebe, N.H.F.: Nelson H.F. Beebe's Bibliographies Page (2002)
18. Jones, B.: Computational Geometry Database (February 2002); FTP / HTTP
19. Estevez, P.A., Vera, P., Saito, K.: Selecting the Most Influential Nodes in Social Networks. In: Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17 (2007)
20. Richardson, M., Domingos, P.: Mining knowledge-sharing sites for Viral Marketing. In: Eighth Intl. Conf. on knowledge Discovery and Data Mining (2002)
21. Rama Suri, N., Narahari, Y.: Determining the Top-k Nodes in Social Networks using the Shapley Value (Short Paper). In: Padgham, Parkes, Müller, Parsons (eds.) AAMAS 2008, Estoril, Portugal, May 12-16, pp. 1509–1512 (2008)