A Greedy Algorithm Approach for Mobile Social Network

Smita Bhosale
Department Of Computer Engineering
Dr. D.Y.Patil College Of Engineering, Ambi, Pune

Dhanshree Kulkarni
Department Of Information Technology
Dr. D.Y.Patil College Of Engineering, Ambi, Pune.

ABSTRACT

With the proliferation of mobile devices and wireless technologies, mobile social network systems used more. A mobile social network has important role in social network. The Process of finding influential nodes is NP-hard. Greedy rule with demonstrable approximation guarantees will provide smart approximation. A divide-and-conquer method with parallel computing mechanism has been used. Community-based Greedy rule for mining top-K influential nodes is used first. It has two parts: dividing the large- scale mobile social network into many communities by taking under consideration data diffusion. Communities select influential nodes by a dynamic programming. Performance is to be increased by considering the influence propagation supported communities and take into account the influence propagation crossing communities. Experiments on real large-scale mobile social networks show that the proposed algorithm is quicker than previous algorithms.

General Terms

Mobile social network, influence maximization, PCA greedy algorithm,

Keywords

PCA - Parallelized Community-based Algorithm

1. INTRODUCTION

Today mobile social network systems are more and more used. Mobile social network is important role of data and influence. It has main issue to search out a set of cogent people in a very mobile social network. The corporate plans to at first target low variety of “Influential” people of the network by giving them free samples of the merchandise. Corporate hopes that at first users can advocate the product to their friends; their friends can influence their friends’. The most cogent nodes are referred to as NP-hard. Greedy algorithmic rule is known as associate degree algorithm rule. Community-based Greedy algorithmic rule is used for mining top-K cogent nodes. It divides into 2 components: dividing the large-scale mobile social network into many communities by taking into consideration data diffusion and choosing communities to search out cogent nodes by a dynamic programming.

The most cogent nodes are referred to as NP-hard. Greedy algorithmic rule with demonstrable approximation guarantees will offer sensible approximation. This paper, a divide-and-conquer strategy with parallel computing mechanism has been used. I first used associate degree algorithmic rule known as Community-based Greedy algorithmic rule for mining top-K cogent nodes. It encompasses two components: dividing the large-scale mobile social network into many communities by taking into consideration data diffusion and choosing communities to search out cogent nodes by a dynamic programming.Experiments on real large-scale mobile social networks show that the projected ways ar abundant quicker than previous algorithms, meanwhile, with high accuracy. Connecting people or organizations, social network seems a system, like email networks, on-line Facebook, and scientific collaboration networks etc. With the shape of spoken data or influence spreads upon the social network, that may be a elementary issue in social network analysis.

An organization would really like to promote a replacement product, hoping it'll be adopted by an outsized fraction of the network. At first target a tiny low variety of “influential” people of the network is considered. first users can advocate for product to their friends. Next because of their friends can influence their friends. New product adopt through the powerful spoken impact or known as infectious agent selling.

“Divide-and-conquer” strategy and “Parallelized computation” is used associate degree algorithmic rule for mining top-K cogent nodes, known as Community-based Greedy algorithmic rule CGA. These networks are weighted directed graphs. And therefore people in a very community can influence one another within the variety of “words”. It’s not possible to calculate the influence in every community at the same time. New algorithm with parallel computing is used. Parallel computing algorithm is known as Parallelized Community-based algorithmic rule (PCA).

2. RELATED WORK

Independent cascade model and linear threshold model are two extensively studied influence diffusions models originally summarized by Kempe et al. It proves that the generalized versions of these two models are equivalent. Based on the IC and LT model, Kempe et al propose a greedy algorithm to solve the influence maximization problem to maximize the spreading of a single piece of ideas, innovations, etc. under these two models. Efficient algorithm design is base for to find efficient local graph structures to speed up the computation. CLDAG algorithm is similar to the LDAG algorithm, which is also based on the DAG structure. CLDAG algorithm is novel in dealing with competitive influence diffusion using the dynamic programming method.Kostka et al study competitive rumor spreading on a more restricted model than IC and LT, and focused on showing the hardness of computing the optimal solution for the two competing parties. Pathak et al. study a model of multiple cascades, which is an extension of a different diffusion model called the voter model.

The social influence analysis problem poses a unique set of challenges: First, how to leverage both node-specific topic distribution and network structure to quantify social influence? In another word, a user’s influence on others not only depends on their own topic distribution, but also depends on what kinds of social relationships they have with others. The goal is to design approach to utilize both the local attributes (topic distribution) and the global structure (network information) for social influence analysis.

In addition, we can use statistical topic modeling to automatically extract topics from the social networking data. In this paper, we focus on the topic modeling approach to initialize the topic distribution of each node. Topical Affinity Propagation (TAP): Based on the input network and topic distribution on the nodes, we formalize the social influence problem in a topical factor graph model and propose at opical affinity propagation on the fact or graph to automatically identify the topic specific social influence. Our main idea is to leverage affinity propagation at the topic- level for social influence identification. The approach is based on the theory of factor graph, in which the observation data are cohesive on both local attributes and relationships. In our setting, the node corresponds to the observation data in the factor graph and the social relationship corresponds to edge between the observation data in the graph. Finally, we propose two different propagation rules: one based on message passing on graphical models, the other one is a parallel update rule that is suitable for Map-Reduce framework.

3. METHODOLOGY

Our community detection algorithm consists of two steps, partition and combination.

1) Partition. We extend the algorithm with the information influence mechanism based on Independent Cascade model. The algorithm is a nearly linear algorithm for community detection. It is designed for undirected and also for unweighted graph, and thus is not directly applicable.

2) Combination. The generated communities by the partition step are very small and dispersed; we develop a method to combine communities such that the difference between influence degree of a node in its community and its influence degree in the whole network is restricted.

CGA Implementation

- Mining TOP-L Infulential Nodes
- Top-K Influential Nodes Mining
- Precision Analysis of CGA

PCA Implementation

- Community-based Parallelization Mechanism
- Influence Propagation between Crossing Communities
- Precision Analysis of PCA

4. A GREEGY ALGORITHM APPROACH

Given a mobile social network G = (V, E, W), we aim to mine a set of top-K Influential nodes I on the network such that R (I) is maximized using the extended Independent Cascade information diffusion model. It has been proved that the optimization problem is NP- hard [5]. A greedy algorithm can approximate the optimum to within a factor of (1− 1 e). However, the greedy algorithm is expensive for solving the influence maximization problem on a large-scale network. So we propose a community based greedy algorithm which mine the Influential nodes in each community rather than the whole network

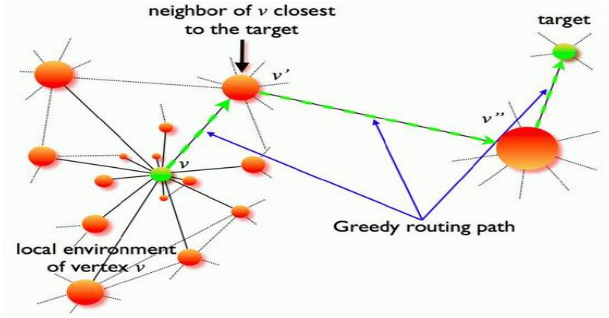


Fig 1: Greedy Algorithm Illustration

4.1 Algorithm and parameters

In the experiments, we compare our algorithms (CGA,PCA) with existing representative algorithms for influence maximization. We listed as follow:

• **MixGreedy**: We take MixGreedy as the benchmark to evaluate the proposed algorithm CGA for two reasons. First, MixGreedy is the state-of-the-art Greedy Algorithm for influence maximization, and it is shown that MixGreedy outperforms previously proposed Greedy Algorithms, such as GA and CELF. Second, CGA adopts MixGreedy to find influential nodes within communities, and thus a performance comparison between them will reveal their pros and cons.

• **DegreeDiscount**: It is defined as degree discount. This is developed by Chen et al for the uniform IC model.

• **SA**: Unlike previous algorithms, it takes a totally different approach by using artificial intelligence.

• **MIA**: The most representative algorithm is proposed by Chen et at., which uses local arborescence structures of each node to approximate the influence propagation.

• **Random**: simply select K random nodes in the graph.

• **CGA**: It is the community-based greedy algorithm proposed in this paper.

• **PCGA**: PCGA is CGA in parallelization. It only uses LTP algorithm to balance the computation.

• **PCA**: It is the algorithm that mining Top-K influential nodes with considering the propagation of crossing communities based on parallelism in this paper.

4.2 Community Based Parallelization Mechanism

We used balance computation assignment for parallelize the influence in every community. Size and density of community are basic components of community. The complexity of community is dependent of above factor. Computational weight *C.W* for each community is as follows

$$C.W = C.N * \beta_1 + C.E * \beta_2$$

Where *C.N* and *C.E* is the number of nodes and number of edges in community *C* separately.$\beta 1$ and $\beta 2$ are tuning parameters to between the size and density. Balance computation assignment is used for parallelize the influence in every community. Size and density of community

4.3 PCA algorithm

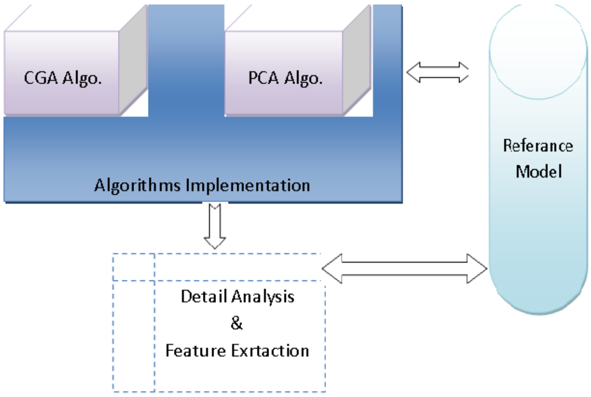
.**Input**: G = (V,ε,⊒), M communities;

```
Output: I, Top-K influential nodes;
1: I = I1 = I2 = ··· = IM = ∅ ◃ I and Ij are the set of K influential
nodes in the whole network and community Cj respectively
2: for k = 1 to K do
3:for m = 1 to M do
4: Cm.W = Cm.N ∗β1 + Cm.E ∗β2
5: end for
6: Sort(C,1,M)
7: for m = 1 to M do
```

```
8: i = argminpi∈P (workload[pi]) ◃ pi is ith loaded processor
9: threadwork[i].push back(Cm)
10: end for ◃ The following is the parallelization step
11: omp set num threads(P) ◃ The number of threads
12: ♯ pragma omp parallel ◃ Parallel mark of OpenMP
13: compute ∆Rm
14: vmax = argmaxvj∈Cj (Rj(Ij∪{vi})−Rj(Ij))
15: Ij = Ij∪{vmax}
16:I = I∪vmax
17: end for
```

In Algorithm, line 3-5 computes the computational weight of each community, and assign them to each loaded processor(line 7-10); Line 11-13 parallelize the influence spread by computing ∆Rm of each community m. Specifically, we compute ∆Rm with consideration of influence propagation crossing communities, which will be detailed in Section 4.2.2. In line 14-16, we find the node vmax that maximizes Rj(Ij∪{vi})−R(Ij) in community Cj. In addition, we optimize the above algorithm by the fol- lowing technique: if the kth influential node and its influence spread are not in community Cm, the weight of this community is not computed repeatedly.

5. SYSTEM DIAGRAM



CGA is known as Community-based Greedy Algorithm.

It divides the network into communities to find top K influential nodes. Influence degree of nodes within a community can be as close as whole network. It proposes a dynamic programming algorithm to choose which community to find kth influential node.

Performance of CGA is calculated by analyzing approximation ratio of CGA.

PCA is known as Parallelized community based algorithm.

Serial computation is not enough for large scale social network. PCA is community based algorithm which considers the information propagation across communities. It calculates computational weight C.W for each community.

6. CONCLUSION

In this paper, we have a tendency to propose the economical algorithms known as Community-based Greedy algorithm(CGA) and Parallelized Community-based algorithm(PCA) for mining top-K authoritative nodes in an exceedingly MSN. We have a tendency to initial extend the essential free lance Cascade model to require weight fringe of MSN into thought. CGA has 2 main elements, AN formula for detective work communities by taking under consideration data diffusion, and a dynamic programming formula for choosing communities to find authoritative nodes. Then, to any improve the efficiency and accuracy, we have a tendency to set the influence unfold in every community with a balance computation assignment and take into account the influence propagation crossing communities by removing less authoritative nodes through a rigorous established regulation. We conjointly offer demonstrable approximation guarantees for CGA and PCA.

7. REFERENCES

[1] F.Bass, “A new product growth model for consumer durables,” Manage- ment Science, vol. 15, pp. 215–227, 1969.

[2] J.Brown and P.Reinegen, “Social ties and word-of-mouth referral be- havior,” Journal of Consumer research, vol. 14, no. 3, pp. 350–362, 1987.

[3] J.Goldenberg, B.Libai, and E.Muller, “Talk of the network: A complex systems look at the underlying process of word-of-mouth,” Marketing Letters, vol. 12, no. 3, pp. 211–223, 2001.

[4] V.Mahajan, E.Muller, and F.Bass, “New product diffusion models in marketing: A review and directions for research,” Journal of Marketing, vol. 54, no. 1, pp. 1–26, 1999.

[5] D. Kempe, J. Kleinberg, and E. 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, 2003, pp. 137–146.

[6] D.Kempe, J.Kleinberg, and E.Tardos, “Influential nodes in a diffusion model for social networks,” In ternational colloquium on automata, languages and programming, no. 32, pp. 112–1138, 2005.

[7] W. Chen, Y. Wang, and S. Yang, “Efficient influence maximization in social networks,” in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, 2009, pp. 199–208.

[8] 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, 2007, pp. 420–429.

[9] M. Girvan and M. E. J. Newman, “Community structure in social and biological networks,” Proceedings of the National Academy of Sciences, vol. 99, no. 12, pp. 7821–7826, 2002.

[10] W.Yu, G.Cong, G.Song, and K.Xie, “Community-based greedy algo- rithm for mining top-k influential nodes in mobile social networks,” in KDD, 2010, pp. 1039–1048.

[11] P. Domingos and M. Richardson, “Mining the network value of cus- tomers,” in Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, 2001, pp. 57–66.