ISSN: 0975-833X

Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 8, Issue, 07, pp.34443-34446, July, 2016

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

RESEARCH ARTICLE

MODIFIED GREEDY ALGORITHM FOR MAXIMUM INFLUENCE IN WEIGHTED GRAPHS

***Mini Singh Ahuja and Navpreet Kaur**

Guru Nanak Dev University (Regional Campus Gurdaspur)

ARTICLE INFO

Article History:
Received 18th April, 2016
Received in revised form
10th May, 2016
Accepted 25th June, 2016
Published online 16th July, 2016

Key words:

Influence maximization, Social network, Spread of influence, Independent cascade model and Greedy algorithm.

ABSTRACT

Now a day's many researchers have been researching on how to find most influential nodes in a network, especially in social networks. If we are given a social network where neighbors can influence other nodes of a network then to identify some such seed nodes in social network through which we can maximize the spread of influence is recent research topic. In this paper we are working on weighted greedy algorithm and modifying it to get better results.

Citation: Mini Singh Ahuja and Navpreet Kaur, 2016. "Modified greedy algorithm for maximum influence in weighted graphs", *International Journal of Current Research*, 8, (07), 34443-34446.

INTRODUCTION

Influence maximization is the problem in which we find a small set of seeds which have maximum influence in the network so that their aggregated influence can be maximized in a network. Influence maximization is connected to social network where it has application in viral marketing, where companies try to promote their products by propagating through word-of-mouth. Some social networking sites such as facebook, napster, linked Inetc are attracting millions of people as these websites provide platform for effective viral marketing. Consider an example in which we can understand that why we need to maximize the influence. A small company develops a mobile application let's say an online shopping application. Now, this company wants traffic on this application. But it has limited budget so it can only select small number of initial users and motivates them to use application by providing them special offers (like heavy discounts or by giving them gifts). The company desires that the initial users would start influencing their friends on social networking sites to use the application and their friends would influence their friends' friends and so on, and thus through the word-of-mouth effect a large population in the social network would adopt the application (Wei Chen *et al*., 2010; Amit Goyal *et al*., 2012). The problem is who to select as seed users such that influence is on large population. This problem is referred as influence maximization. This problem is of great interest to companies because they can promote their product through this problem. Kempe *et al*. has proposed two basic influence cascade models, the independent cascade (IC) model and linear threshold (LT) model. In both models, social network is modeled as directed graph G = (V, E), where V represent vertices means individuals of a graph and E represents edges means relationship between vertices. In the independent cascade model, the selection procedure from one node to another is based on a probabilistic way; individuals can successfully choose their neighbors with certain probabilities. In LT model each edge has weight attached to it. It is considered that our bond with each friend varies and our behavior with them is according to closeness of our relationship with them (Wei Chen *et al*., 2009). If we have to ask a friend to use certain kind of product then it will be more effective to say to as friend who is our roommate rather than to a friend who lives far apart.

That's why weights are attached to the edges to show that which friend can be more influenced. In this paper we are working on LT model. We are implementing weighted greedy algorithm and then modifying it so as to get maximum influence in social network.

****Corresponding author: Mini Singh Ahuja,***
Guru Nanak Dev University (Regional Campus Gurdaspur)

Related work

In 2011 Zaixin Lu *et al*. studied the influence maximization problem in deterministic linear threshold model. They showed that in deterministic linear threshold model there is no polynomial time. They also showed that that the exact computation of the final influence given a seed set can be solved in linear time in the deterministic linear. They also discussed the Least Seed Set problem, which aims to find a seed set with least number of people to activate all the required people in a given social network.
In 2012 SauravPandit *et al*. presented a simple and scalable algorithm that outperforms the existing state-of the-art, and its success does not depend significantly on any kind of tuning parameter. It was then compared with the existing algorithms and output set of k nodes is calculated. The output produced by their algorithm is much better in case of spread of influence.
In 2013 Huiyuan Zhang *et al*. proposed a model called opinion based cascading model. They formulate an opinion maximization problem in which opinion of individual is taken into consideration as well as capture the change of opinions. They designed an efficient algorithm to compute total positive influence based on this model.
In 2014 Shengfu Zhou *et al* says that traditional greedy algorithm is not very efficient for large networks. They proposed a more efficient greedy algorithm. They named it LNG algorithm which works for linear thresh hold model. They performed experiments for large network on their algorithm and showed that in LNG the time consumed is very less and spread of influence is better in their proposed algorithm rather thanin classic greedy algorithm.

Weighted Greedy Algorithm

Greedy algorithm is used to maximize the influence of social networking. In case of weighed greedy algorithm weights are attached to each edge which tells how much influence those two nodes can have on each other. In case of weighted greedy algorithm nodes which have maximum sum of weights of children is picked. The algorithm for weighted greedy algorithm for maximizing the influence is as given below.

1. Read adjacency matrix
2. totalseeds<- 5
3. seeds<-(25)
4. coverednodes<-(25)
5. weights = 0
6. Repeat for i from 1 to totalseeds-1
7. iseed=seeds(i)
8. Get children of iseed in array ar.
9. Repeat for j from 1 to length of ar
10. Find weight wof (iseed,ar(j))
11. weights = weights + w.
12. Endloop
13. coverednodes=(coverednodes,ar)
14. len=length(ar)
15. Repeat for j from 1 to len
16. el=adjmat(ar(j),:)
17. count(j)=length(el)
18. endloop
19. (xmaxymax)=max(count) //where ymax is index of maximum value
20. seeds(i+1)=ar(ymax)
21. endloop

In above algorithm first adjacency matrix is read. Then total seeds to pick, initial seed and covered nodes (nodes that will be covered or on which nodes influence will be made with seed nodes) are initialized. Then we start a loop in which loop from initial seed and add its children to covered array and then we loop on its children and find a node whose aggregate weight of children is maximum. Then we add that node to seed node and repeat the process until we get total seeds.

Weighted Greedy algorithm modified

In our paper we have modified greedy algorithm so that we can maximize the influence or we can say so that we can cover maximum nodes with seed nodes. In this algorithm, instead of choosing node with maximum children we have chosen node with maximum unique nodes. The algorithm for modified greedy algorithm for maximizing the influence is as given below.

1. Read adjacency matrix
2. totalseeds<- 5
3. seeds<-(25)
4. coverednodes<-(25)
5. weights =0
6. Repeat for i from 1 to totalseeds-1
7. iseed=seeds(i)
8. Get children of iseed in array ar.
9. Repeat for j from 1 to length of ar
10. Find weight w of (iseed,ar(j))
11. weights = weights + w.
12. Endloop
13. coverednodes=(coverednodes,ar)
14. len=length(ar)
15. Repeat for j from 1 to len
16. el=adjmat(ar(j),:)
17. el(ismember(el,coverednodes))=( )
18. count(j)=length(el)
19. endloop
20. (xmaxymax)=max(count) //where ymax is index of maximum value
21. seeds(i+1)=ar(ymax)
22. endloop

Similarly, in above algorithm first adjacency matrix is read. Then total seeds to pick, initial seed and covered nodes (nodes that will be covered or on which nodes influence will be made with seed nodes) are initialized. Then we start a loop in which we loop from first seed and find adjacency list of first seed then we added children of seed into covered nodes then for each child (covered node) of seed we find children which are not covered before and then we find node with maximum un covered children and maximum aggregate weight of children and add that node into seed node.

Experimentation

In our experiment we took a graph of 25 nodes. We made an adjacency list of these nodes. Table 1 shows the adjacency list of 25 nodes.

Table 1. Adjacency list of 25 nodes

Node	Adjacency List
1	2 3 4 10 11
2	1 3 5 6 7
3	1 2 5 7 9 10
4	1 5 8 9 10 11 12 13 16
5	2 3 4 9 10 16 21
6	2 8 9 10 11 13
7	2 3 8 10 12 13 16
8	4 6 7 12 16
9	3 4 5 6 12
10	1 3 4 5 6 7 14 15 24
11	1 4 6 15 16
12	4 7 8 9 13 17
13	4 6 7 12 15 17 18 19
14	10 15 18 19 20
15	10 11 13 14 18 19 20 21 22
16	4 5 7 8 11 19
17	12 13 20 23
18	13 14 15 20 25
19	13 14 15 16
20	14 15 17 18 21
21	5 15 20 22
22	15 21 24 25
23	17 24 25
24	10 22 23
25	18 22 23

Then we took weights of each edge as:

Table 2. Weights of each edge

1 ->2 = 1	5->16= 3	13->17=1.8
1-> 3= 1.5	5->21=2.7	13->18=1
1-> 4= 2.5	6->8= 4	13->19=2
1-> 10= 3	6-> 9= 4.5	14->15=2.5
1-> 11= 3.5	6->10=1.7	14->18=3
2-> 3= 0.5	6->11=2	14->19=3.2
2-> 5 =0.4	6->13=2.6	14->20=2.2
2-> 6= 2	7->8=3	15->18=3
2-> 7 =2	7->10=3.7	15->19=2
3-> 5= 2.5	7->12=4	15->20=3.6
3-> 7= 3	7->13=3	15->21=3
3-> 9= 3.5	7->16=3.5	15->22=4
3-> 10= 4	8->12=4	16->19=4.2
4 ->5= 5	8->16= 4.7	17->20=4
4-> 8= 4.5	9->12=1.7	17->23=3.5
4 ->9= 4.7	10->14= 2	18->20=3
4->10= 0.3	10->15=2.7	18->25=2.8
4-> 11= 2	10->24= 2	20->21=2
4->12= 2.4	11->15=1.8	21->22=3
4 ->13= 2.7	11->16=2	22->24=3.7
4-> 16=3	12->13=3	22->25=2
5->9=3.9	12->17=2.7	23->24=1.8
5->10=1	13->15=3.7	23->25=1

After taking 25 nodes and weights we did our experiment for both weighted greedy and modified weighted greedy algorithm. Outputs of these are shown in Figure 1 and Figure 2.

Figure 1 shows that if we took 5 nodes as seed nodes then total nodes that will be covered by these nodes will be 19 and aggregate weight of these nodes will be 41 in case of weighted greedy algorithm.

Figure 2 shows that if we take 5 nodes as seed nodes then total nodes that will be covered by these nodes will be 24 and aggregate weight will be 52.90 in case of modified weighted greedy algorithm.

```
>> WeightedGreedyAlgo
Graph is undirected
Seeds for this graph are:
   25   18   15   10    4

total number of nodes covered by these seeds
ans =

   22

Nodes covered by these seeds are:
  Columns 1 through 19

    1    2    3    4    5    6    7    9   10   11   13
  14   15   16   18   19   20   21   22

  Columns 20 through 22

   23   24   25

Weight of covered nodes is
   41
>>
```

Figure 1. Output of weighted Greedy algorithm for 5 seeds

```
>> WeightedGreedyAlgoMod
Graph is undirected
Seeds for this graph are:
   25   18   13    4    5
total number of nodes covered by these seeds
ans =

   24
Nodes covered by these seeds are:
  Columns 1 through 11
    1    2    3    4    5    6    7    8    9   10   11
Columns 12 through 19
   12   13   14   15   16   17   18   19

  Columns 20 through 24

   20   21   22   23   25
Weight of covered nodes is
   52.9000
>>
```

Figure 2. Output of modified weighted greedy algorithm for 5 seeds

Table 3 shows the tabular form of output (aggregate weight) for weighted greedy algorithm and modified weighted greedy algorithm in case we select 5 nodes as seed nodes.

Table 3. Aggregate weights with weighted greedy algorithm and modified weighted greedy algorithm for 5 seed nodes

Aggregate weight for weighted Greedy Algorithm	Aggregate weight for modified weighted greedy algorithm
41	52.9

Graphical form of Table 3 is shown in Figure 3.

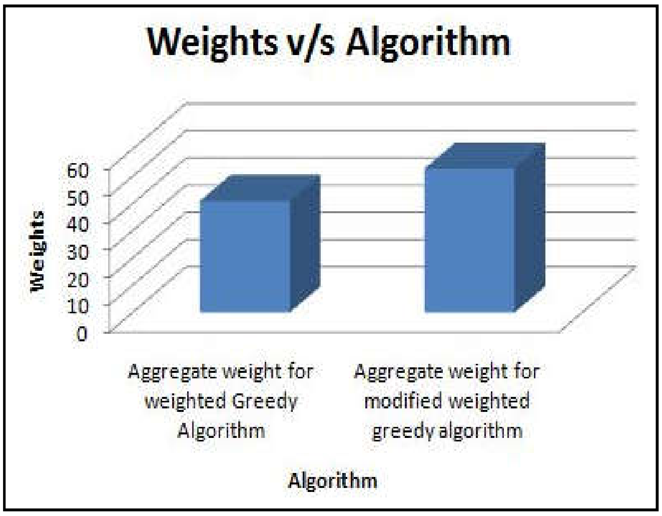

Figure 3. Aggregate weights with weighted greedy algorithm and modified weighted greedy algorithm for 5 seed nodes

Conclusion and Future Scope

The influence maximization is a problem in which we find set of seed nodes such that these seeds can spread maximum influence. In this paper we have worked on LT model for maximizing the influence. We have first studied weighted greedy algorithm which picks node with maximum aggregate weight of children. We have modified this algorithm in which we have picked nodes with maximum aggregate weight of unique children rather than only maximum weight of children. We have also tested our algorithm which shows that modified algorithm gives better results. In future further modification will be done on this algorithm to improve results. While selection of next seed; we can see children of all seed nodes rather children of selected seed only.

REFERENCES

Amit Goyal, Francesco Bonchi and Laks V. S. Lakshmanan, 2012. "A DataBased Approach to Social Influence Maximization", Proceedings of the VLDB Endowment, Vol. 5, No. 1, pp 73-84.

Huiyuan Zhang, Thang N. Dinh, and My T. Thai, 2013. "Maximizing the Spread of Positive Influence in Online Social Networks", IEEE 33rd International Conference on Distributed Computing Systems, pp. 317-326, Philadelphia, PA.

Kempe D., J. M. Kleinberg, and ´E. Tardos, 2003. "Maximizing the spread of influence through a social network," in KDD.

SauravPandit, Yang Yang and Nitesh V. Chawla, 2012. "Maximizing Information Spread Through Influence Structures in Social Networks", IEEE 12th International Conference on Data Mining Workshops, pp 258-265, Brussels.

Shengfu Zhou, Kun Yue *et al*. 2014. "An Efficient Algorithm for Influence Maximization under Linear Threshold Model", Control and Decision Conference (CCDC), The 26th Chinese, 2014, pp 5352 - 5357, Changsha.

Wei Chen, Yajun Wang and SiyuYang, 2009. "Efficient Influence Maximization in Social Networks", ACM SIGKDD international conference on Knowledge discovery and data mining, Pages 199-208, New York, NY, USA

Wei Chen, Yifei Yuan and Li Zhang, 2010. "Scalable Influence Maximization in Social Networks under the Linear Threshold Model", ICDM '10 Proceedings of the 2010 IEEE International Conference on Data Mining, Pages 88-97, Washington, DC, USA

Zaixin LU, Wei Zhang *et al*. 2011. "Approximation and Inapproximation for The Influence Maximization Problem inSocial Networks under Deterministic Linear Threshold Model", International Conference on Distributed Computing Systems Workshops, pp 160-165, Minneapolis, MN.

****