Determining Top-k Nodes in Social Networks using Shapley Value

Research Supervisor: Prof. Y. Narahari

Ramasuri Narayanam
nrsuri@csa.iisc.ernet.in
Electronic Commerce Laboratory
Department of Computer Science and Automation
Indian Institute of Science
Bangalore, India

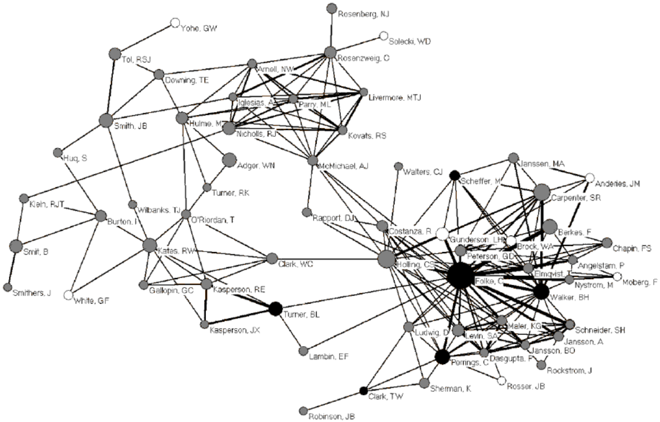
May 30, 2009

Outline of the Presentation

1. Social Networks : Introduction
1. Influential Nodes in Social Networks
1. Shapely Value based Algorithm for Top-k Nodes Problem
1. Experimental Results

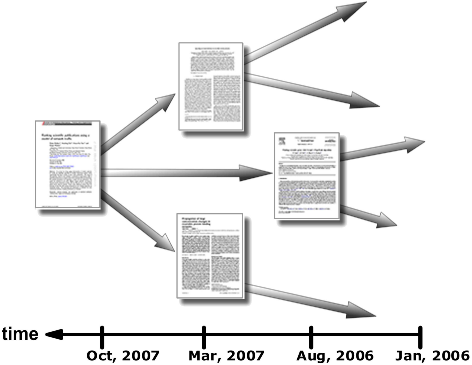
Social Networks

- *Social Networks:* A social structure made up of nodes that are tied by one or more specific types of relationships.
- Examples: Friendship networks, coauthorship networks, trade networks, etc.



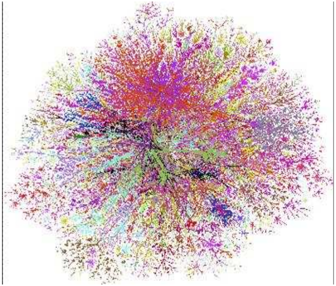
Social Networks

- *Social Networks:* A social structure made up of nodes that are tied by one or more specific types of relationships.
- Examples: Friendship networks, coauthorship networks, trade networks, etc.



Social Networks

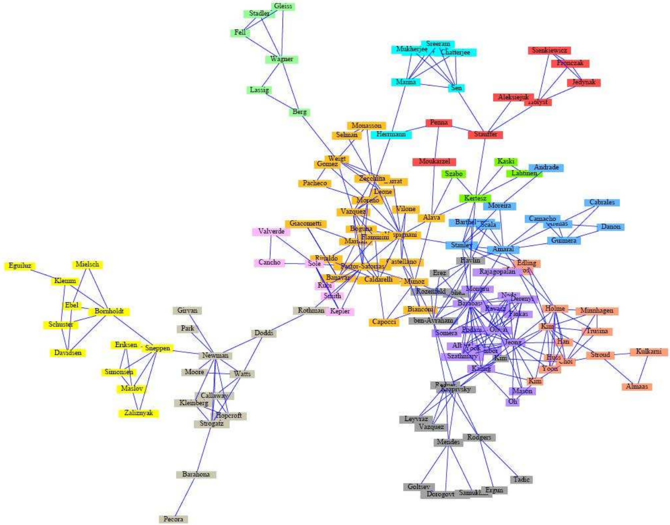
- *Social Networks:* A social structure made up of nodes that are tied by one or more specific types of relationships.
- Examples: Friendship networks, coauthorship networks, trade networks, etc.



- Real world social networks:

  Orkut, wikis, blogs, etc.
- Social networks are modeled using a graph where nodes represent individuals and edges represents the relationships between nodes

Features of Social Networks



Influential Nodes in Social Networks

Motivating Example 1: Diffusion of Information

- Social networks play a key role for the spread of an innovation or technology
- We would like to market a new product that we hope will be adopted by a large fraction of the network
- Which set of the individuals should we target for?
- Idea is to initially target a few influential individuals in the network who will recommend the product to other friends, and so on
- A natural question is to find a target set of desired cardinality consisting of influential nodes to maximize the volume of the information cascade

Motivating Example 2: Co-authorship Networks

- co-authorship network is concerned with the collaboration patterns among research communities
- nodes correspond to researchers and an edge exists if the two corresponding researchers collaborate in a paper
- interesting to find the most prolific researchers since they are most likely to be the trend setters for breakthrough

Linear Thresholds Model

- Call a node active if it has adopted the information
- Initially every node is inactive
- Let us consider a node i and represent its neighbors by the set $N(i)$
- Node i is influenced by a neighbor node j according to a weight w_{ij}. These weights are normalized in such a way that

$$\sum_{j \in N(i)} w_{ij} \leq 1.$$

- Further each node i chooses a threshold, say θ_i, uniformly at random from the interval [0,1]
- This threshold represents the weighted fraction of node i's neighbors that must become active in order for node i to become active

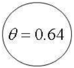
Given a random choice of thresholds and an initial set (call it S) of active nodes, the diffusion process propagates as follows:

- in time step t, all nodes that were active in step $(t-1)$ remain active
- we activate every node i for which the total weight of its active neighbors is at least θ_i
- if $A(i)$ is assumed to be the set of active neighbors of node i, then i gets activated if

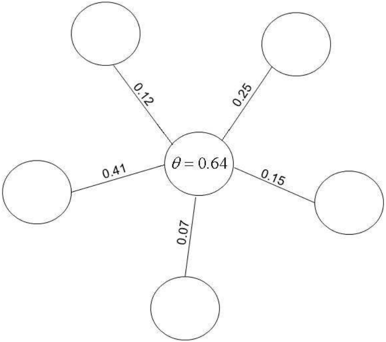
$$\sum_{j \in A(i)} w_{ij} \ \geq \ \theta_i .$$

- This process stops when there is no new active node in a particular time interval

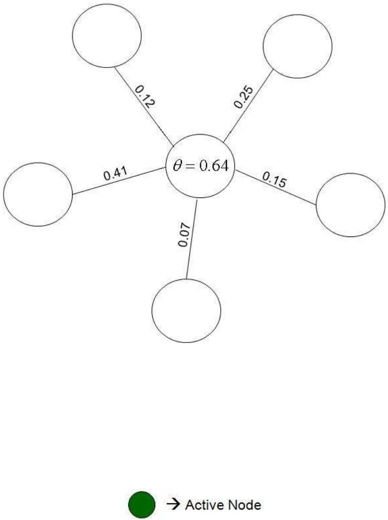
Illustrating Linear Threshold Model



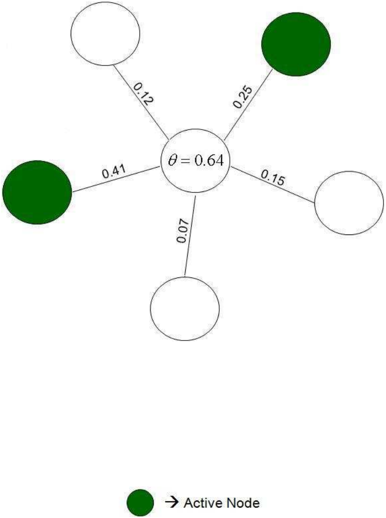
Illustrating Linear Threshold Model



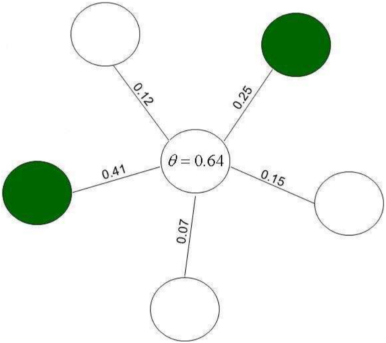
Illustrating Linear Threshold Model



Illustrating Linear Threshold Model



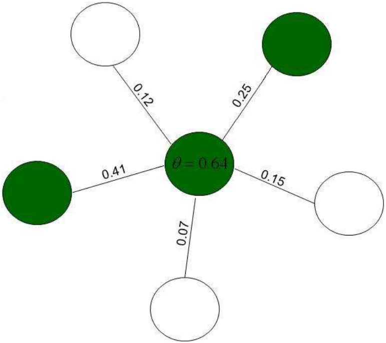
Illustrating Linear Threshold Model



0.41 + 0.25 > $\theta(=0.64)$

→ Active Node

Illustrating Linear Threshold Model



0.41 + 0.25 > $\theta(= 0.64)$

→ Active Node

Top-k Nodes Problem

- **Top-k Nodes Problem:**
  - Let us define an objective function $\sigma(.)$ to be the expected number of active nodes at the end of the diffusion process
  - If S is the initial set of target nodes, then $\sigma(S)$ is the expected number of active nodes at the end of the diffusion process
  - For economic reasons, we want to limit the size of the initial active set S
  - For a given constant k, the top-k nodes problem seeks to find a subset of nodes S of cardinality k that maximizes the expected value of $\sigma(S)$

Applications

- Viral Marketing
- Databases
- Water Distribution Networks
- Blogspace
- Newsgroups
- Virus propagation networks

- R. Akbarinia, F.E. Pacitti, and F.P. Valduriez. Best Position Algorithms for Top-k Queries. In VLDB, 2007.
- J. Leskovec, A. Krause, and C. Guestrin. Cost-effective outbreak detection in networks. In ACM KDD, 2007.
- N. Agarwal, H. Liu, L. Tang, and P.S. Yu. Identifying influential bloggers in a community. In WSDM, 2008.

Shapely Value based Algorithm for Top-k Nodes Problem

Our Algorithm

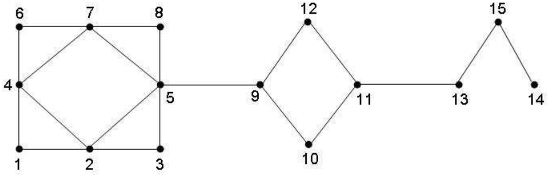
- *Influence of a Node:* expected number of other nodes that become active using this node
- we approach the top-k nodes problem using cooperative game theory
- we measure the influential capabilities of the nodes as provided by Shapley value
- our proposed algorithm is in two steps:
  1. construction of RankList[]
  2. choosing the top-k nodes from RankList[]

Construction of Ranklist[]

1. Let π_j be the j-th permutation in $\hat{\Omega}$.
2. **for** $j = 1$ to t **do**
3.   **for** $i = 1$ to n, **do**
4. $MC[i] \leftarrow MC[i] + v(S_i(\pi_j) \cup \{i\}) - v(S_i(\pi_j))$
5.   **end for**
6. **end for**
7. **for** $i = 1$ to n, **do**
8.   compute $\Phi[i] \leftarrow \frac{MC[i]}{t}$
9. **end for**
10. use an efficient sorting algorithm to sort the nodes in non-increasing order based on average marginal contribution values

Choosing Top-k Nodes

1. Naive approach is to choose the first k in the RankList[] as the top-k nodoes
2. *Drawback:* Nodes may be clustered
3. RankList[]={5,4,2,7,11,15,9,13,12,10,6,14,3,1,8}
4. Top 4 nodes are clustered
5. Choose nodes satisfying
    - ranking order of the nodes
    - spreading over the network



k value	Greedy Algorithm	Shapley Value Algorithm	MDH based Algorithm	HCH
1	4	4	4	2
2	8	7	7	4
3	10	10	8	6
4	12	12	8	7
5	13	13	10	8
6	14	14	13	8
7	15	15	13	8
8	15	15	13	8
9	15	15	13	10
10	15	15	13	11
11	15	15	13	13
12	15	15	13	13
13	15	15	14	14
14	15	15	15	15
15	15	15	15	15

Experimental Results

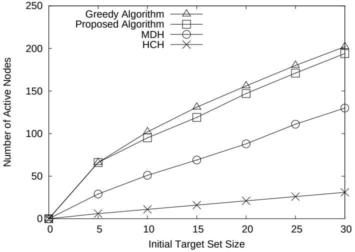
Benchmark Algorithms for Top-k Nodes

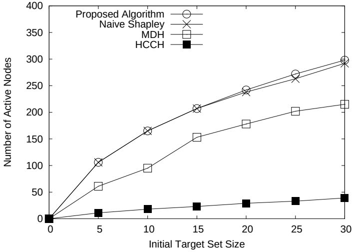
1. Greedy Algorithm
2. Maximum Degree Heuristic based Algorithm
3. High Clustering Coefficient based Algorithm

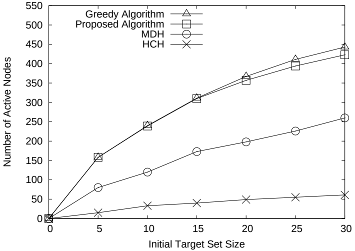
Network Datasets

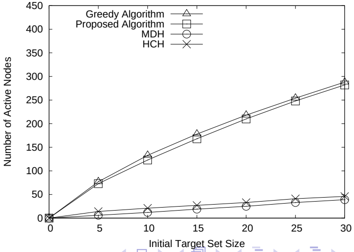
Dataset	Number of Nodes
Sparse Random Graph	500
Scale-free Graph	500
Jazz	198
NIPS	1061
Netscience	1589
HEP	10748

Experiments: Synthetic Datasets

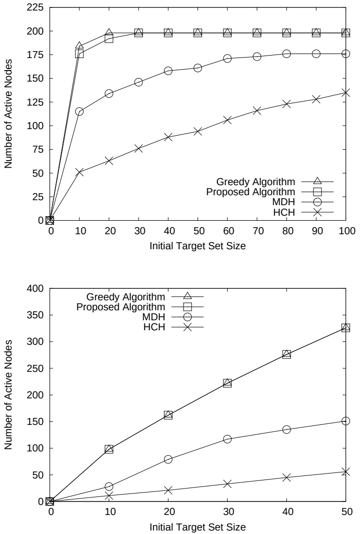


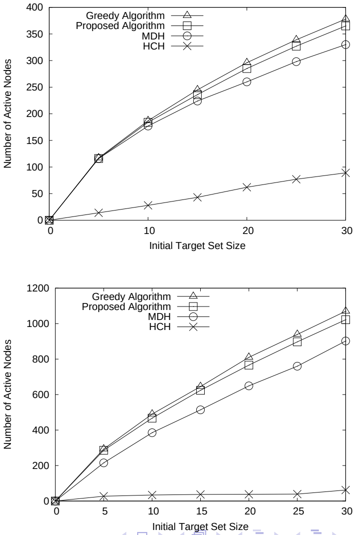




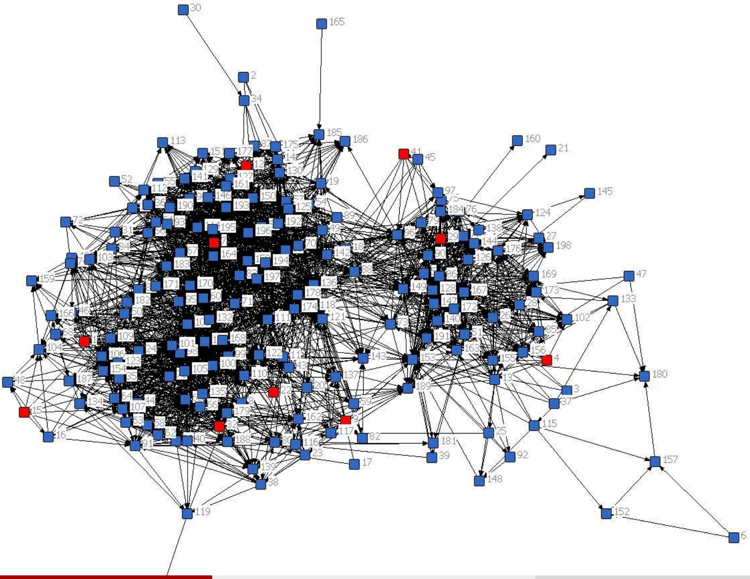


Experiments: Real World Datasets

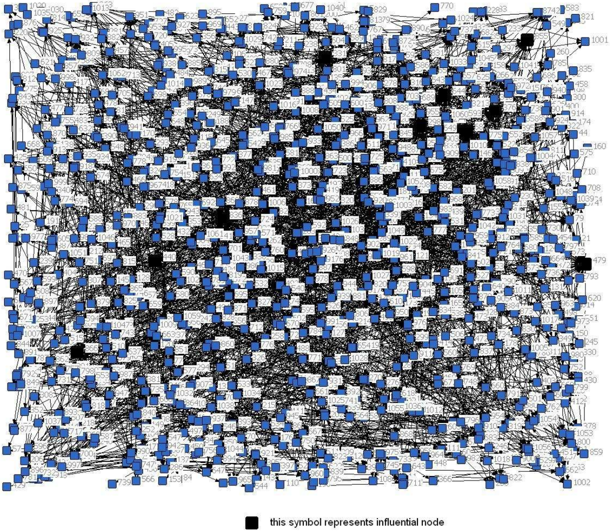




Visualization of Jazz Dataset



Visualization of NIPS Dataset



Thank You