Rumor Has It...An Algorithm Could Scope Out Gossip!

...

By Vamsi Mokkapati

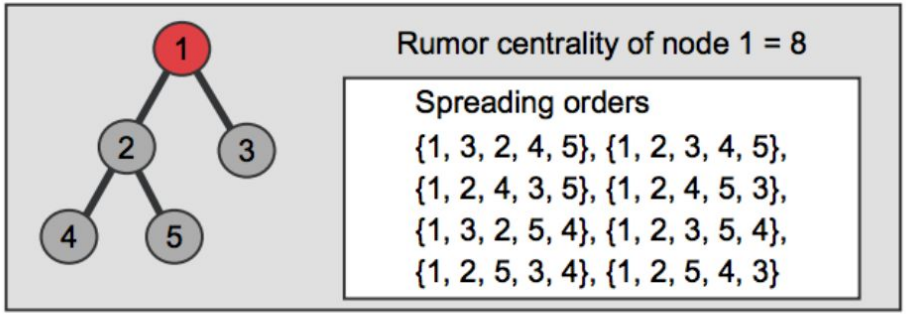
Where did the Rumor Originate From?

- This information can now be known using only information regarding **WHO** has heard the rumor within a sample group (not when)
- Complicated networks are more conducive to having algorithms be effective on them
  - A system where every individual knows all the other individuals or every person knows only one person is too simple, and makes it impossible to figure out where the rumor originated from
- This kind of network analysis is useful for other situations too, such as determining the origin of memes, social media trends, fashion trends, or even the origins of epidemics or computer viruses

Using a Tree Network to Collect Rumor Spreading Data

- According to the 2016 research paper done by Devavrat Shah and Tauhid Zaman of MIT, there's a generally accurate algorithmic method using trees to show the probability of which people started a rumor.
- **Rumor centrality parameter** at various nodes using a linear time search:
- Assumptions: equal likelihood of branching, and exponential spreading time.

Figure 1. (Color online) Example of rumor centrality calculation for a 5 node network.



Note. The rumor centrality of node 1 is 8 because there are 8 spreading orders that it can originate, which are shown in the figure.

The Rumor Centrality Parameter (RCP)

- In the previous graph, the nodes of the tree represent people, or our objects of interest
- From observation, we can find the following from the previous simple example:
  - At node 1, the RCP is 8
  - At node 2, the RCP is 2
  - At nodes 3, 4, and 5, the RCP is 1
- Since the RCP is a maximum likelihood estimator, we know the node with the largest RCP has the highest likelihood of being the rumor source.
- In the sample case above, node 1 is clearly seen to have the highest chance of being the rumor source.

The Formula to Calculate the RCP at Any Node

- We now need to generalize our method to find the RCP for any tree, with any number of nodes; we can do so using the formula derived by Shah and Zaman
- Given a graph G, the set of nodes V, and set of subtrees T_w, and the node u, we can find the RCP with the following formula:

$$R(u, G) = \frac{|V|!}{\prod_{w \in V} T_w^u}$$

- Given this formula for RCP, it can be easily seen that this calculation is not useful for networks with exceedingly easy complexities, such as a linear network.

Examining Shah and Zaman's Research

- In their research paper, it is evident that the rumor centrality parameter is used as a basis for detecting the likelihood of a node as a source; there is no theoretical guarantee that the node selected is correct
- However, after applying various probabilistic methods and techniques, they found that the probability of the true source being further than k hops away from the estimated source **exponentially** decreases
  - This validates the usage of the RCP in a more broad, general usage, since it was originally developed for the specific setting of regular trees in which all branches have an equal likelihood to be set.
- Also, popularity biases have to be taken into account; since according to the RCP method, a person with more "friends" would be more likely to be the source of the rumor

Ying and Zhu's Short-Fat Tree (SFT) Algorithm

- Works under BOTH regular trees and Erdos-Renyi random graphs
  - Erdos-Renyi is just either of two models for generating random graphs
- Identifies the information source node as the one with the minimum depth and the most leaf nodes

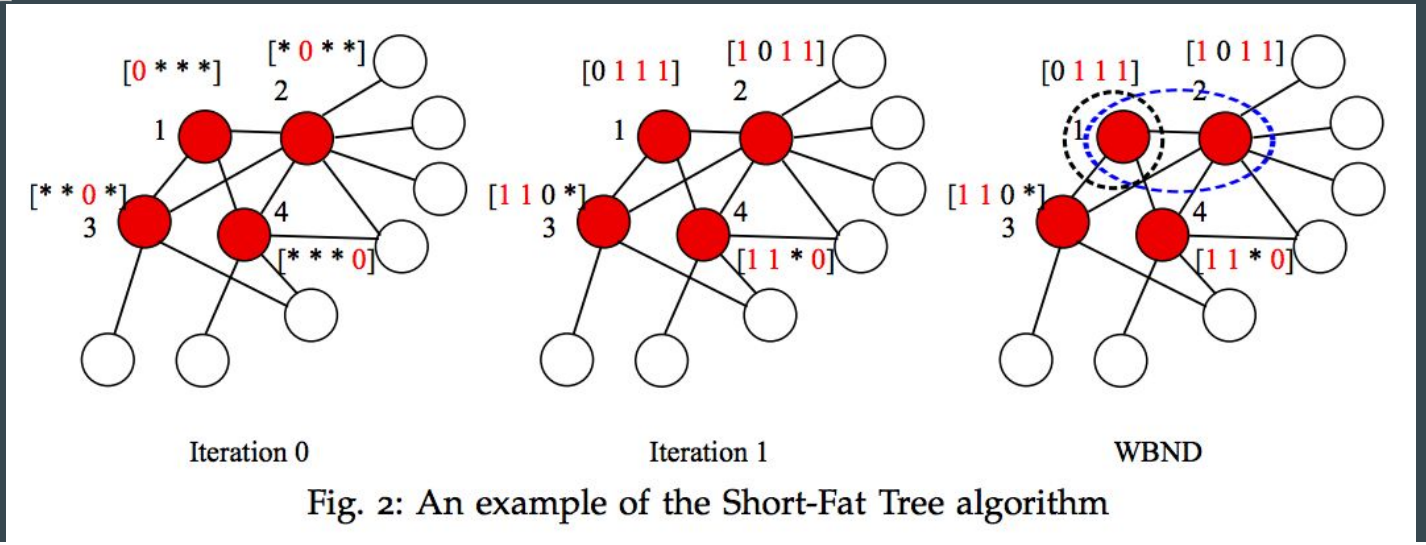


Fig. 2: An example of the Short-Fat Tree algorithm

SFT Algorithm Cont.

- UNLIKE Shah and Zaman's algorithm, the SFT method takes into account the probability that one node is infected by another node
- This probability is used as the basis to arrive at a weighted boundary node degree (WBND) measure, which is calculated as follows (q_{uw} is the infection probability):

$$\sum_{(u,w)\in\mathcal{F}_v'} |\log(1-q_{uw})|,$$

- The WBND values are used to determine the estimator for the information source node.

Algorithm 1: The Short-Fat Tree Algorithm

Input: $\mathcal{I}, g$;
Output: $v^{\dagger}$ (the estimator of information source)
Set subgraph g_i to be a subgraph of g induced by node set $\mathcal{I}$.
for $v \in \mathcal{I}$ **do**
  Initialize an empty dictionary D_v associating with node v.
  Set $D_v[v] = 0$.
end
Each node receives its own node ID at time slot 0.
Set time slot $t = 1$.
do
  for $v \in \mathcal{I}$ **do**
    if *v received new node IDs in $t-1$ time slot, where "new" IDs means node v did not receive them before time slot $t-1$* **then**
      v broadcasts the new node IDs to its neighbors in g_i.
    end
  end
  for $v \in \mathcal{I}$ **do**
    if *v receives a new node ID u which is not in D_v.* **then**
      Set $D_v[u] = t$.
    end
  end
  $t = t + 1$.
while *No node receives $|\mathcal{I}|$ distinct node IDs;*
Set $\mathcal{S}$ to be the set of nodes who receive $|\mathcal{I}|$ distinct node IDs.
for $v \in \mathcal{S}$ **do**
  Compute WBND of T_v using Algorithm 2.
end
return $v^{\dagger} \in \mathcal{S}$ with the maximum WBND.

Pseudocode for SFT and WBND

Algorithm 2: The WBND Algorithm

Input: v,D_v (Dictionary of distance from v to other nodes), g, $\mathcal{I}$, t;
Output: WBND(v)
Set $\mathcal{B}$ to be empty.
for *u in the keys of D_v* **do**
  if $D_v[u] = t$ **then**
    Add u to $\mathcal{B}$.
  end
end
Set $x = 0$;
for $w \in \mathcal{B}$ **do**
  Find the neighbor u of w such that $D_v[u] = t - 1$.
  Set $x = x + \sum_{y \in \text{neighbors}(w)} |\log(1 - q_{wy})| - |\log(1 - q_{wu})|$.
end
return x.

Conclusion

- Source localization techniques like the SFT Algorithm and Rumor Centrality Parameter data have been found to be fairly accurate indicators of the information source node, which tells us where the rumor originated from
- Such techniques are indispensable for helping find the solutions to many kinds of problems, such as the origins of computer viruses, epidemiology, and locating original news sources to analyze news credibility
- Research is still being done in this cutting-edge field, with current scientists looking at how to find the origin when we don’t know how much the rumor has spread.

Works Cited

1. Woo, Marcus. "Rumor Has It An Algorithm Could Scope Out Gossip."*Inside Science*. Inside Science, 11 Mar. 2016. Web. 05 May 2016. <https://www.insidescience.org/content/rumor-has-it-algorithm-could-scope-out-gossip/3756>.
2. Shah, Devavrat, and Tauhid Zaman. "Finding Rumor Sources on Random Trees." *Operations Research* (2016): n. pag. Web. 5 May 2016.
3. Zhu, Kai, and Lei Ying. "Source Localization in Networks: Trees and Beyond." (n.d.): n. pag. Web. 5 May 2016.