Parallelization of Solving Influence Maximization Problem in Social Network

Zeyu Fu

Model of Influence

Independent Cascade Model (IC)

For a network G = (V, E), state of node $v_i \in$ V will only be active (1) or inactive (0)

(1) Initial state, all nodes are inactive
(2) At t = 0, an initial seed set X_0 are activated ($0 \rightarrow 1$)
(3) At t = n, the set of newly activated nodes at time n – 1, $v \in X_{n-1}$, will infect (activate) the inactive outgoing neighbors u with a probability $P_{u,v}$
(4) Propagation ends when no newly active node exists

Linear Threshold Model (LT)

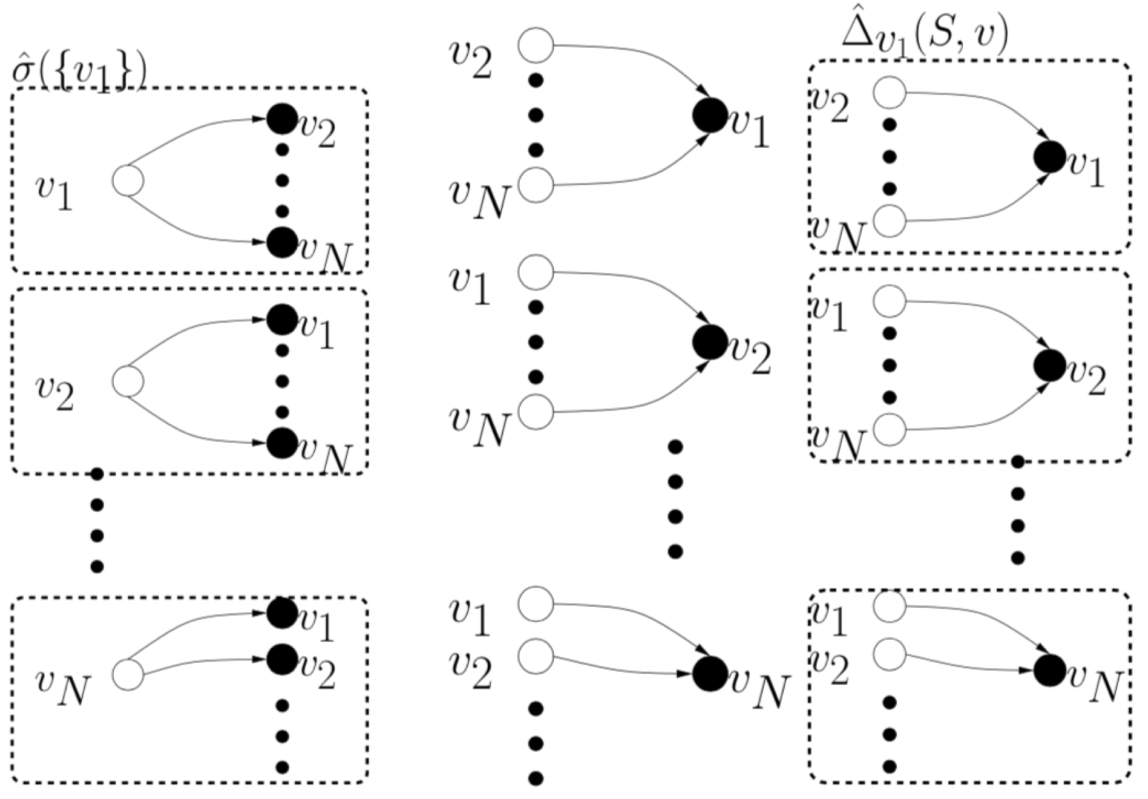
For a network G = (V, E), node $v_i \in$ V will have an activation threshold $\theta_v \in [0, 1]$ uniformly at random

(1) Same as IC model
(2) Same as IC model
(3) At t = n, node v become active if $\sum_{u\in\eta^{in}(v)\cap H_{t-1}} b(u, v) \geq \theta_v$, where $b(u, v) = 1 / L_{in}(v)$, $L_{in}(v)$ is the incoming degree of node v
(4) Same as IC model

Influence Maximization Problem

Given a network G(V, E) and a size k, under a specific influence model, we want to find a seed set S of k nodes such that the expected spread of influence is maximized

Independent Path Algorithm (IPA) : A parallel way to solve this problem



(a) Traversing (b) Re-organizing (c) Updating

Project goal:

1. Practice IPA algorithm (maybe traversing step of IPA)
2. Compare the performance with serial code

Ref: Kim, J., Kim, S. K., & Yu, H. (2013, April). Scalable and parallelizable processing of influence maximization for large-scale social networks?. In *2013 IEEE 29th International Conference on Data Engineering (ICDE)* (pp. 266-277). IEEE.

IPA Pseudo Code

Algorithm 2 traversing(G,θ)

Require: G : graph, θ : threshold
Ensure: R : set of $\hat{\sigma}_I(\{v\})$

$R \leftarrow \phi$
parallel for $v \in V$ **do**
  $\hat{\sigma}_I(\{v\}) \leftarrow 1$
  generate $P_{v \to V}$ by traversing G with θ
  for $u \in O_v$ **do**
    $\hat{\sigma}_I(\{v\}) \leftarrow \hat{\sigma}_I(\{v\}) + \hat{\sigma}_I^u(\{v\})$
  end for
  $R \leftarrow R \cup \{\hat{\sigma}_I(\{v\})\}$
end for
return R

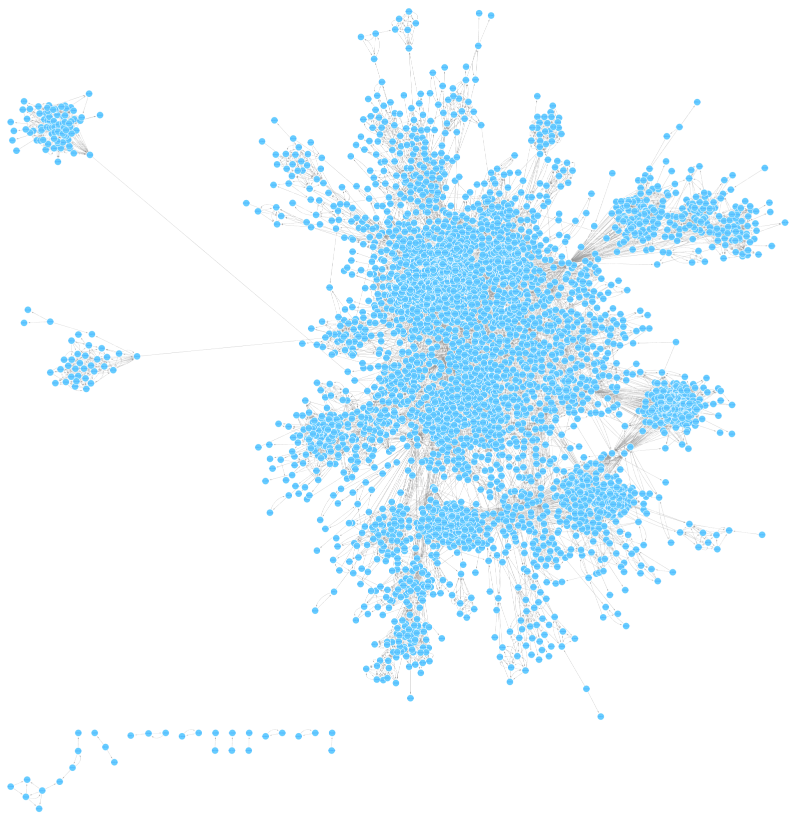
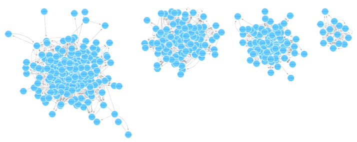
Algorithm 3 updating(S, v)

Require: S : current seed nodes, v : new seed node
Ensure: $\hat{\Delta}(S, v)$

$ret \leftarrow 1$
parallel for $u \in O_v \cup \{v\}$ **do**
  $new \leftarrow 1; old \leftarrow 1$
  for $p \in P_{v \to u}$ and $p \cap S = \phi$ **do**
    $new \leftarrow new \times (1 - ipp(p))$
  end for
  for $s \in S$ **do**
    for $p \in P_{s \to u}$ and $p \cap S \subseteq \{s, v\}$ **do**
      $old \leftarrow old \times (1 - ipp(p))$
      if $p \cap S = \{s\}$ **then**
        $new \leftarrow new \times (1 - ipp(p))$
      end if
    end for
  end for
  $ret \leftarrow ret + (1 - new) - (1 - old)$
end for
return ret

Dataset: Stanford Large Network Dataset

Twitter users relationship data (directed)

Will first work on a very small sub-graph
(maybe tens of nodes)

Partial network with 4514 nodes and 79600 edges