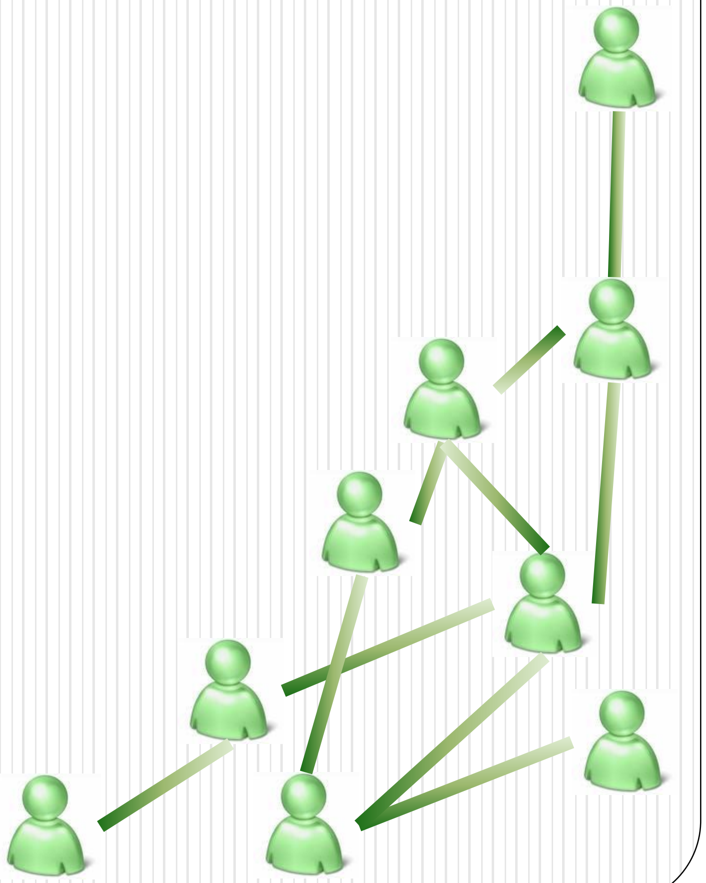
Influence Diffusion in Social Networks

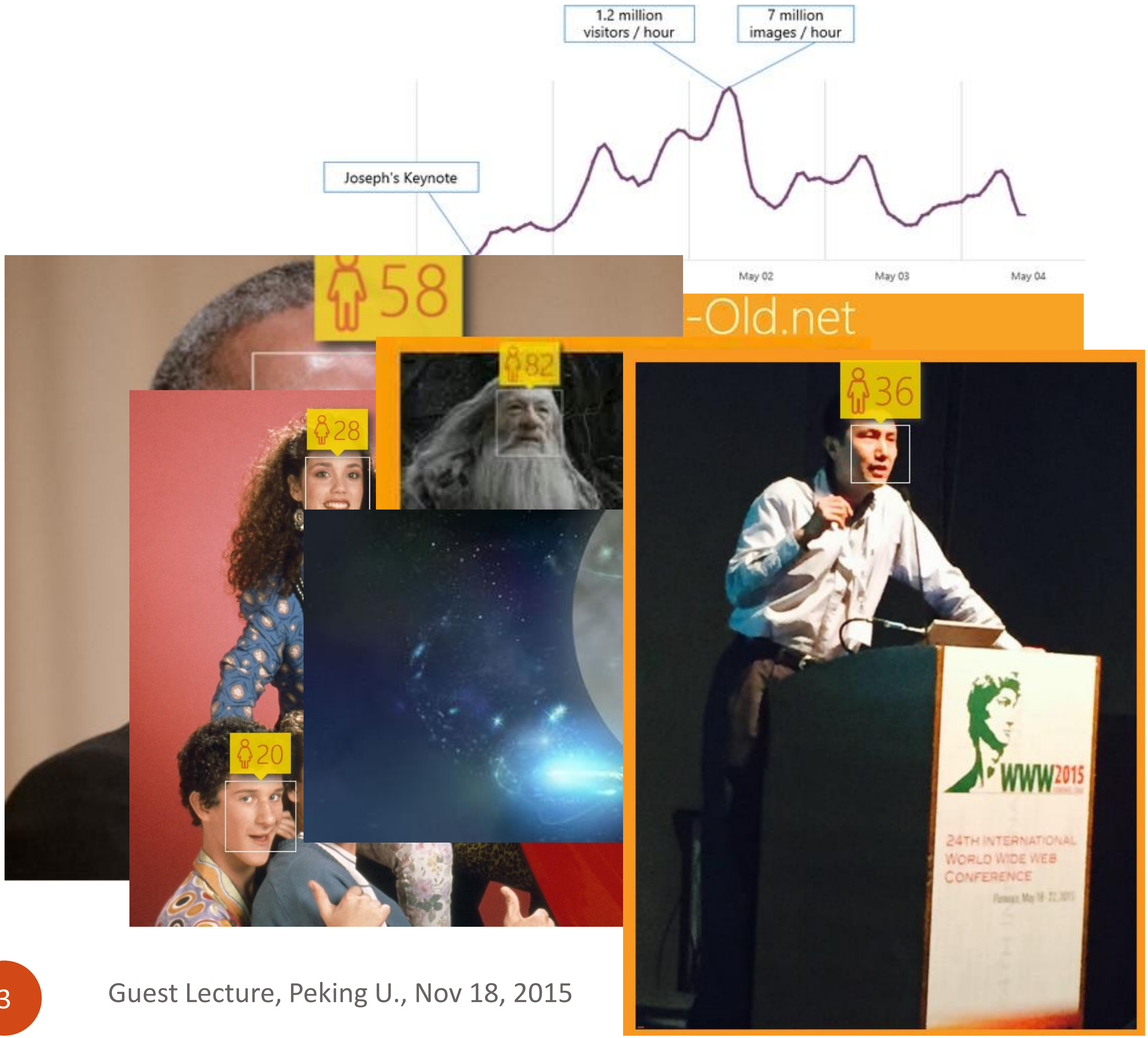
Wei Chen
陈卫
Microsoft Research Asia

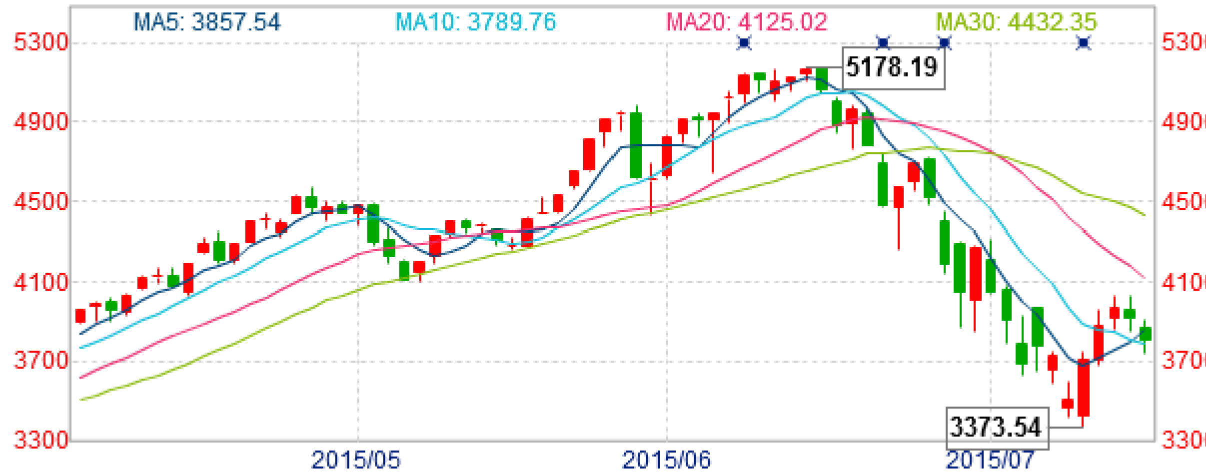


Guest Lecture, Peking U., Nov 18, 2015

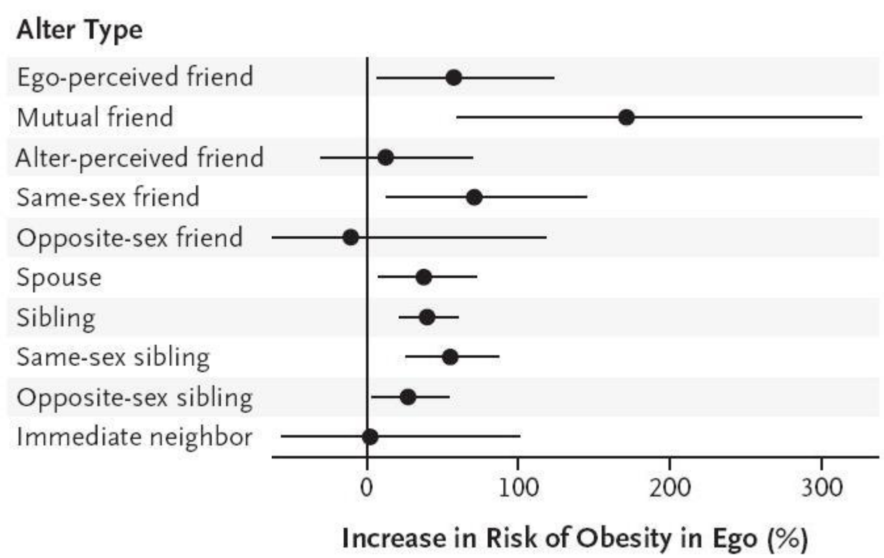
Social influence (人际影响力)

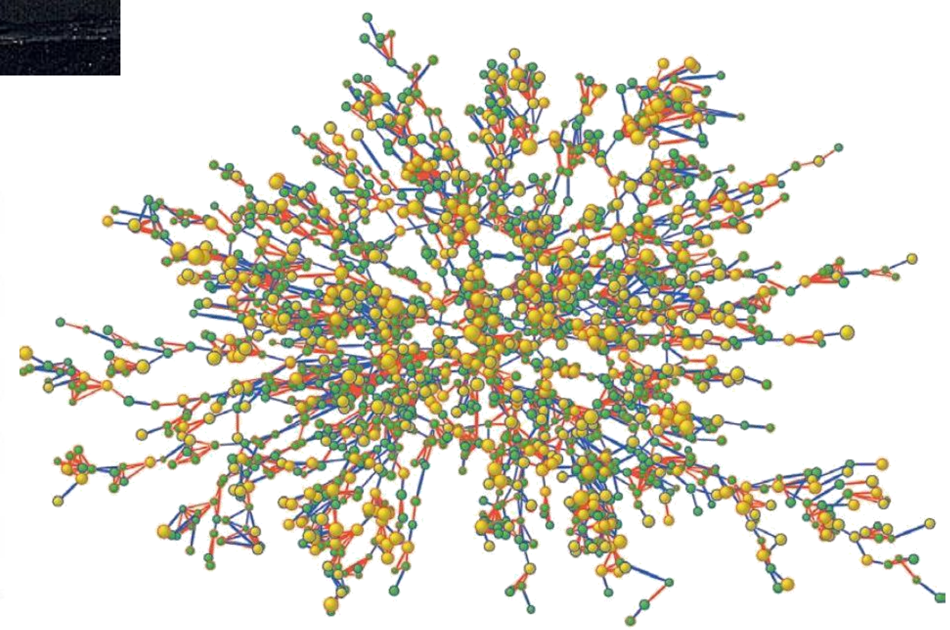
- **Social influence** occurs when one's emotions, opinions, or behaviors are affected by others.


1.2 million visitors / hour
7 million images / hour
Joseph's Keynote
May 02
May 03
May 04
58
-Old.net
82
28
36
20
WWW2015
24TH INTERNATIONAL WORLD WIDE WEB CONFERENCE



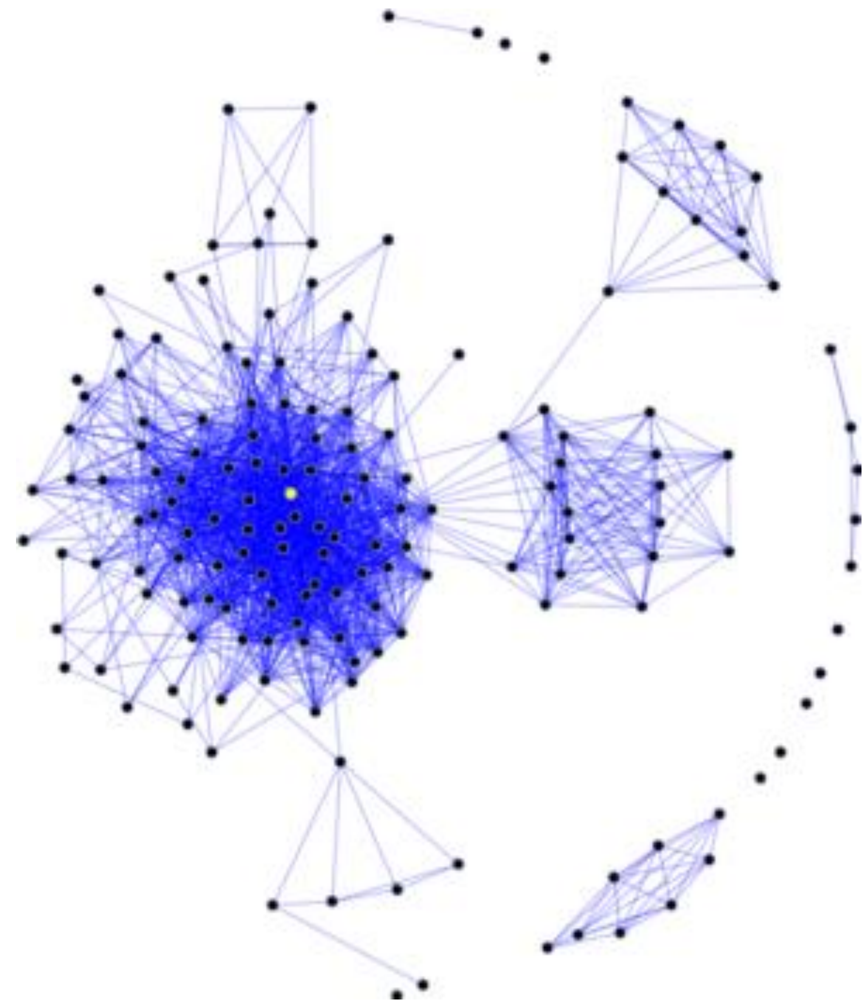
MA5: 3857.54
MA10: 3789.76
MA20: 4125.02
MA30: 4432.35
5300
4900
4500
4100
3700
3300
5178.19
3373.54
2015/05
2015/06
2015/07





[Christakis and Fowler, NEJM'07,08]

Booming of online social networks

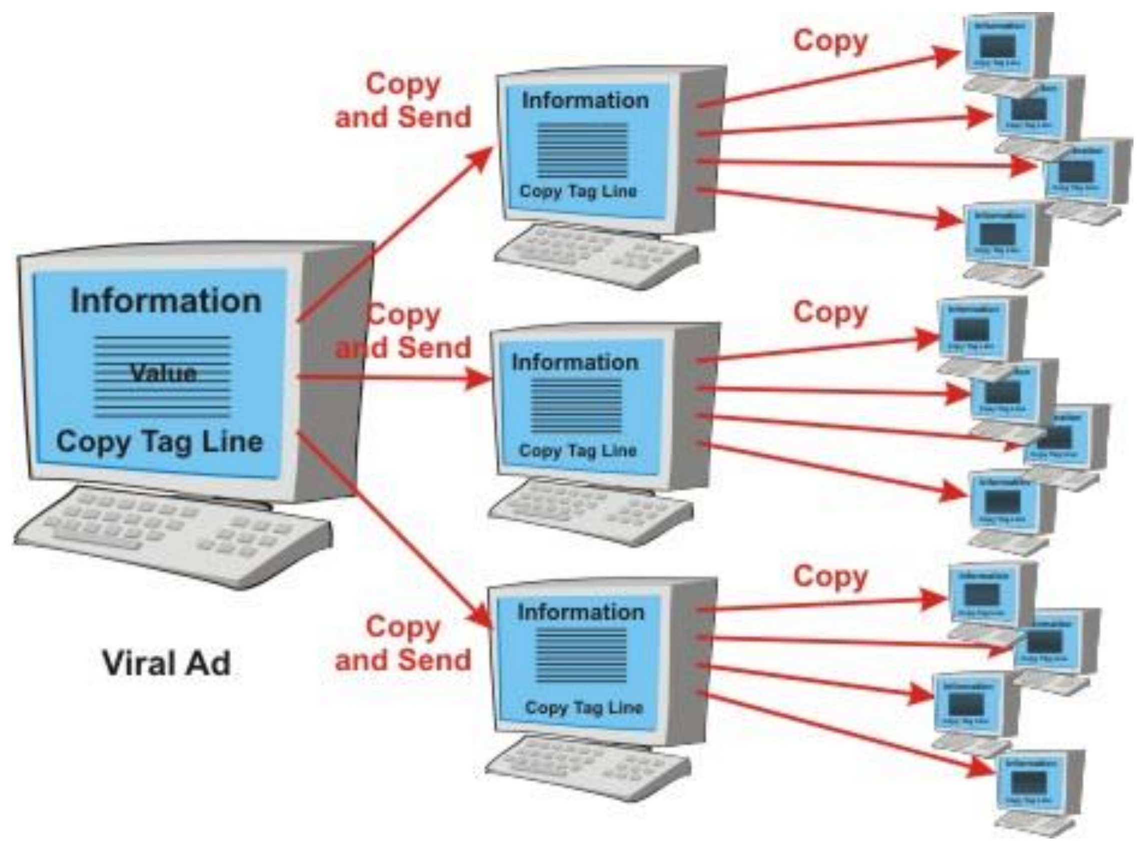


Hotmail: online viral marketing story

- Hotmail's viral climb to the top spot (**90s**): 8 million users in 18 months!
- Boosted brand awareness
- Far more effective than conventional advertising by rivals
  - … and far cheaper, too!

```
Join the world's largest e-mail service
with MSN Hotmail. http://www.hotmail.com
```

Simple message added to footer of every email message sent out



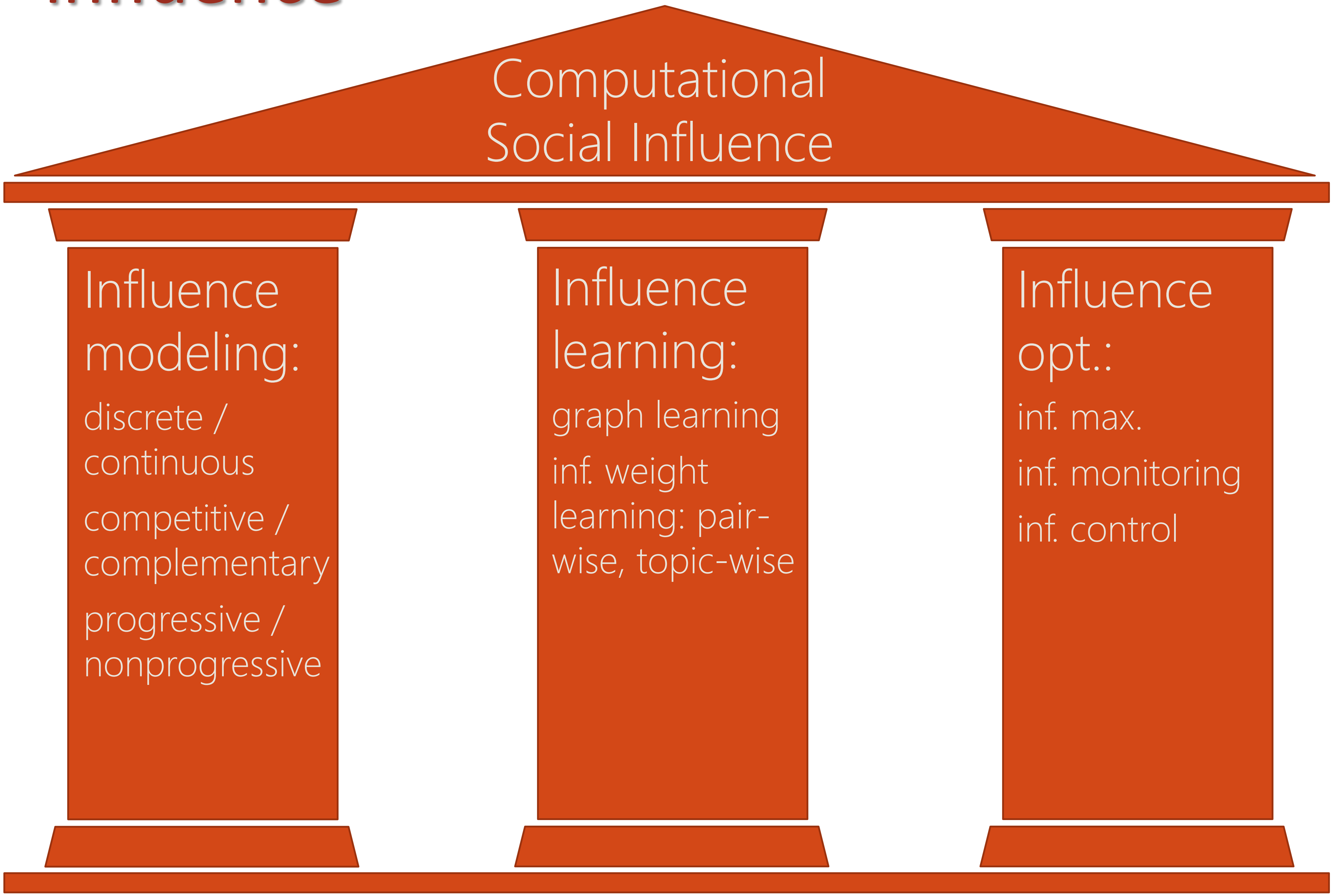
Voting mobilization: A Facebook study

- Voting mobilization [Bond et al, Nature'2012]
  - show a facebook msg. on voting day with faces of friends who voted
  - generate 340K additional votes due to this message, among 60M people tested

Opportunities for computational social influence research

- massive data set, real time, dynamic, open
- help social scientists to understand social interactions, influence, and their diffusion in grand scale
- help identifying influencers
- help health care, business, political, and economic decision making

Three pillars of computational social influence



Influence modeling

- Discrete-time models:
  - independent cascade (IC), linear threshold (LT), general cascade models [KKT'03]
  - topic-aware IC/LT models [BBM'12]
- Continuous-time models [GBS'11]
- Competitive diffusion models
  - competitive IC [BAA'11], competitive LT [HSCJ'12], etc.
- Competitive & complementary diffusion model [LCL'15]
- Others, epidemic models (SIS/SIR/SIRS...), voter model variants

Influence optimization

- Scalable inf. max.
  - Greedy approximation [KKT'03, LKGFVG'07, CWY'09, BBCL'14, TXS'14, TSX'15]
  - Fast heuristics [CWY'09, CWW'10, CYZ'10, GLL'11, JHC'12, CSHZC'13]
- Multi-item inf. max. [BAA'11, SCLWSZL'11, HSCJ'12, LBGL'13, LCL'15]
- Non-submodular inf. max. [GL'13, YHLC'13, ZCSWZ'14, CLLR'15]
- Topology change for inf. max. [TPTEFC'10,KDS'14]
- Inf. max with online learning [CWY'13, LMMCS'15]
- many others …

Influence learning

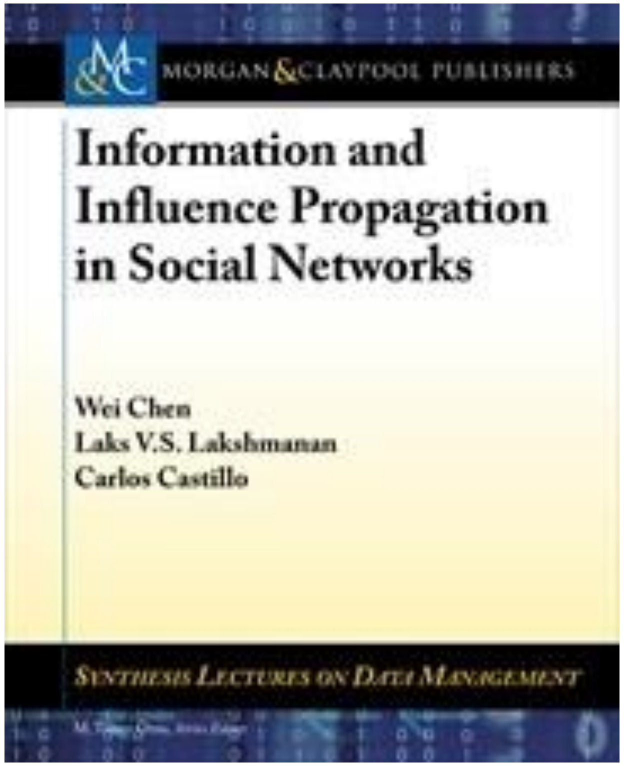
- Based on user action / adoption traces
- Learning the diffusion graph [GLK'10]
- Learning (the graph and) the parameters
  - frequentist method [GBL'10]
  - maximum likelihood [SNK'08]
  - MLE via convex optimization [ML'10,GBS'11,NS'12]

Outline of this lecture

- Introduction and motivation
- Stochastic diffusion models
- Influence maximization
- Scalable influence maximization
- Competitive influence dynamics and influence maximization tasks
- Influence model learning

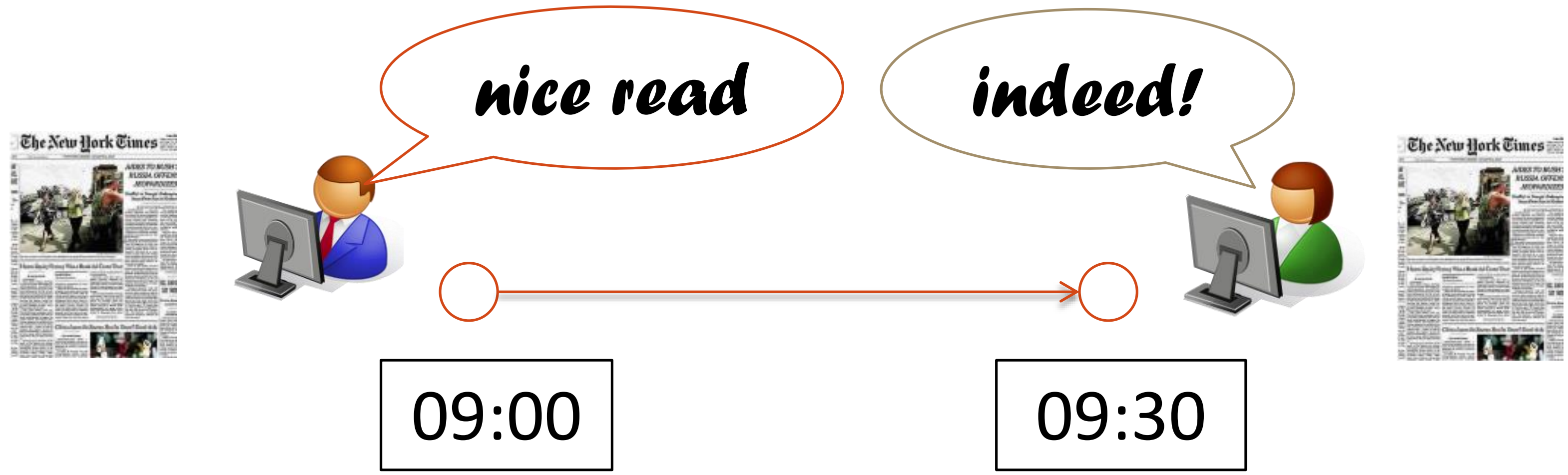
Reference Resources

- Search “Wei Chen Microsoft”
  - Monograph: “Information and Influence Propagation in Social Networks”, Morgan & Claypool, 2013
  - KDD’12 tutorial on influence spread in social networks
  - 社交网络影响力传播研究，大数据期刊，2015
  - my papers and talk slides



Stochastic Diffusion Models

Information/Influence Propagation



People are **<u>connected</u>** and perform **<u>actions</u>**

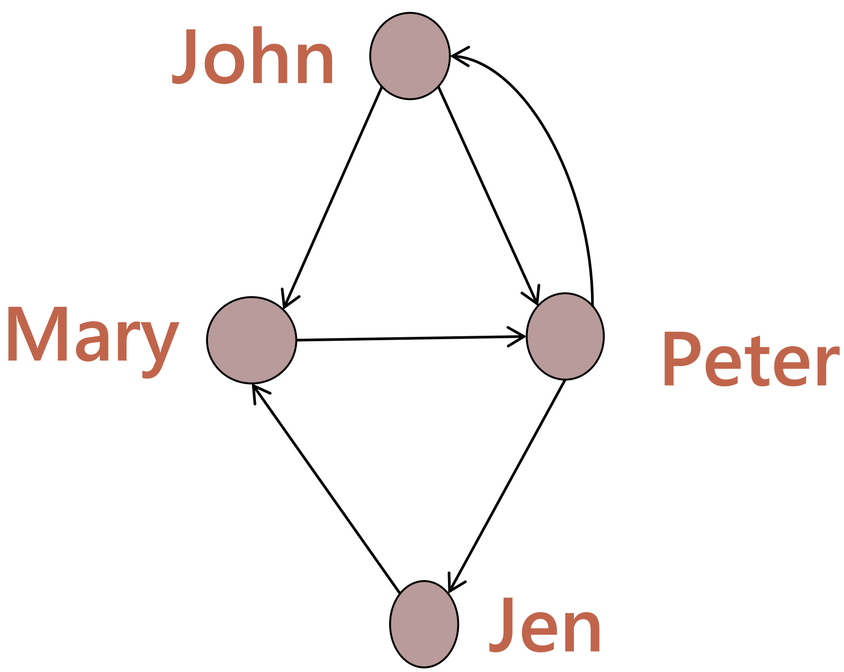
friends, fans, followers, etc.

comment, link, rate, like, retweet, post a message, photo, or video, etc.

Basic Data Model

Graph: users, links/ties

$$G = (V, E)$$



Log: user, action, time

$$A = \{\langle u_1, a_1, t_1 \rangle, \ldots\}$$

User	Action	Time
John	Rates with 5 stars *"The Artist"*	June 3^{rd}
Peter	Watches *"The Artist"*	June 5^{th}
Jen	…	…

Terminologies

- Directed graph $G = (V, E)$
  - Node $v \in V$ represents an individual
  - Arc (edge) $(u, v) \in E$ represents a (directed) influence relationship
- Discrete time t: $0, 1, 2, \ldots$
- Each node v has two states: *inactive* or *active*
- S_t: set of active nodes at time t
  - S_0: *seed set*, initially nodes selected to start the diffusion

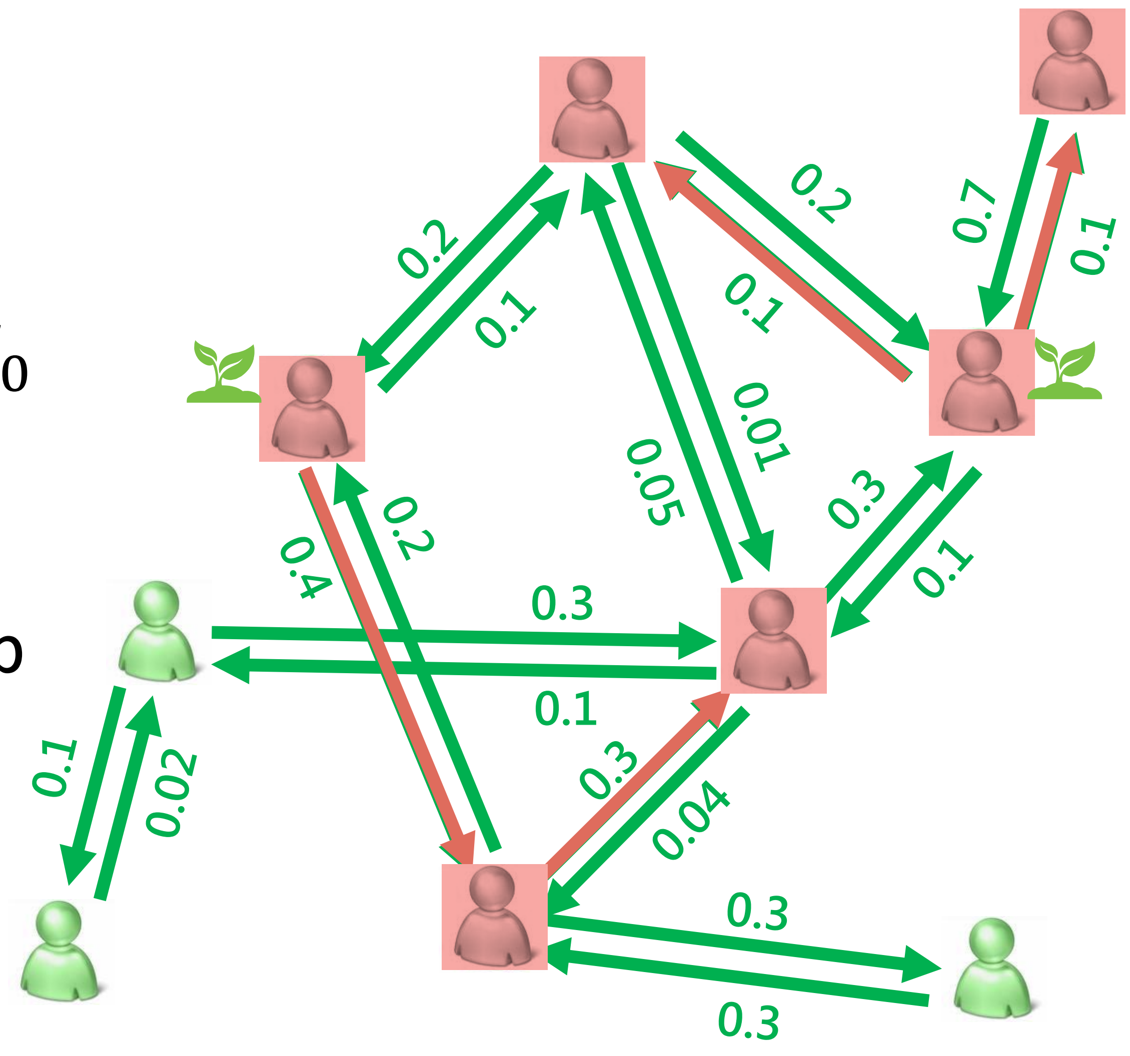
Stochastic diffusion models

Definition 2.1 Stochastic diffusion model. A *stochastic diffusion model (with discrete time steps)* for a social graph $G = (V, E)$ specifies the randomized process of generating active sets S_t for all $t \geq 1$ given the initial seed set S_0.

- *Progressive* models: for all $t \geq 1, S_{t-1} \subseteq S_t$
  - Once activated, always activated, e.g. once bought the product, cannot undo it
  - **Influence spread $\boldsymbol{\sigma(S)}$**: expected number of activated nodes when the diffusion process starting from the seed set S ends

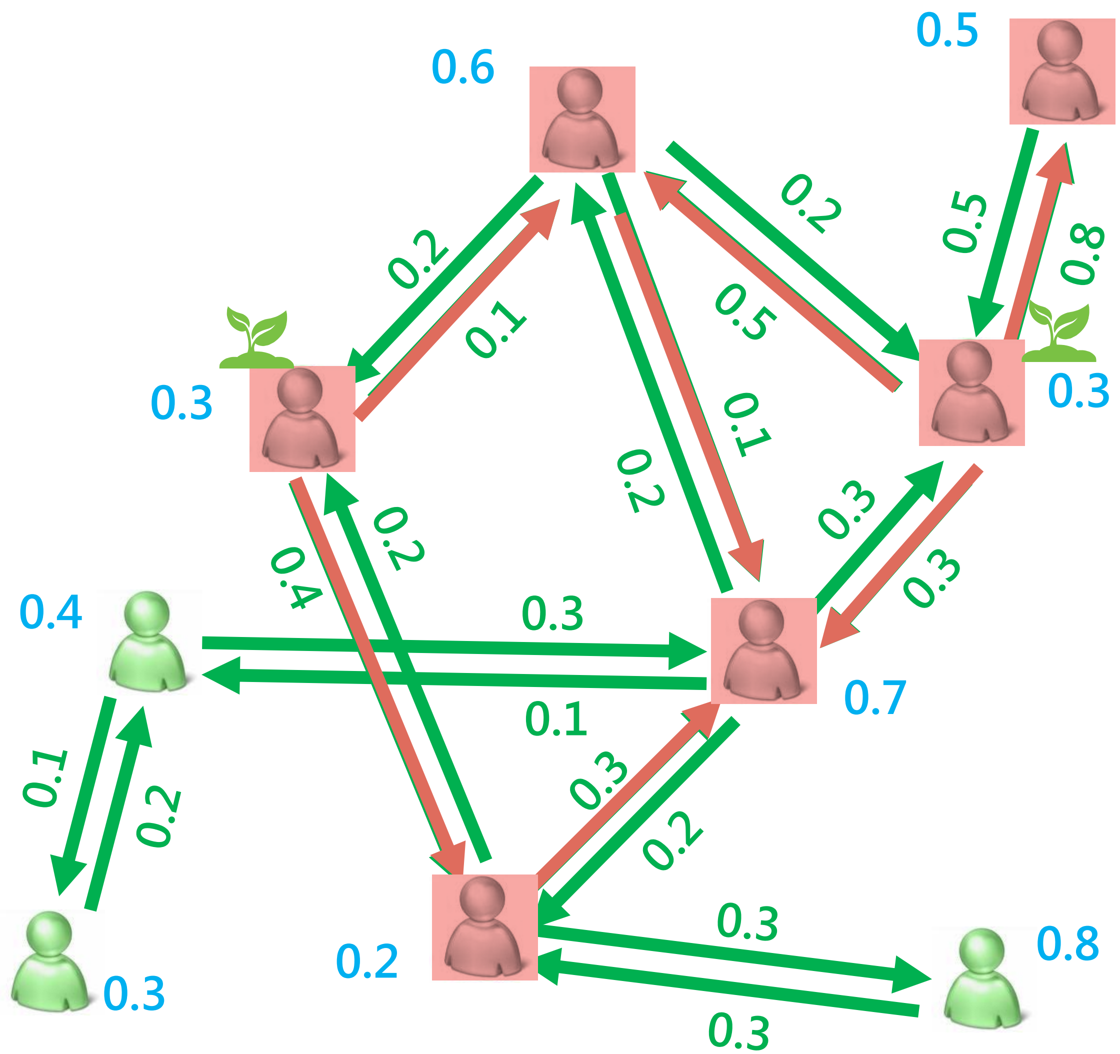
Independent cascade model

- Each edge (u, v) has a *influence probability* $p(u, v)$
- Initially seed nodes in S_0 are activated
- At each step t, each node u activated at step $t-1$ activates its neighbor v independently with probability $p(u, v)$



Linear threshold model

- Each edge (u, v) has a *influence weight* $w(u, v)$:
  - when $(u, v) \notin E, w(u, v) = 0$
  - $\sum_u w(u, v) \leq 1$
- Each node v selects a threshold $\theta_v \in [0,1]$ uniformly at random
- Initially seed nodes in S_0 are activated
- At each step, node v checks if the weighted sum of its active in-neighbors is greater than or equal to its threshold θ_v, if so v is activated



Interpretation of IC and LT models

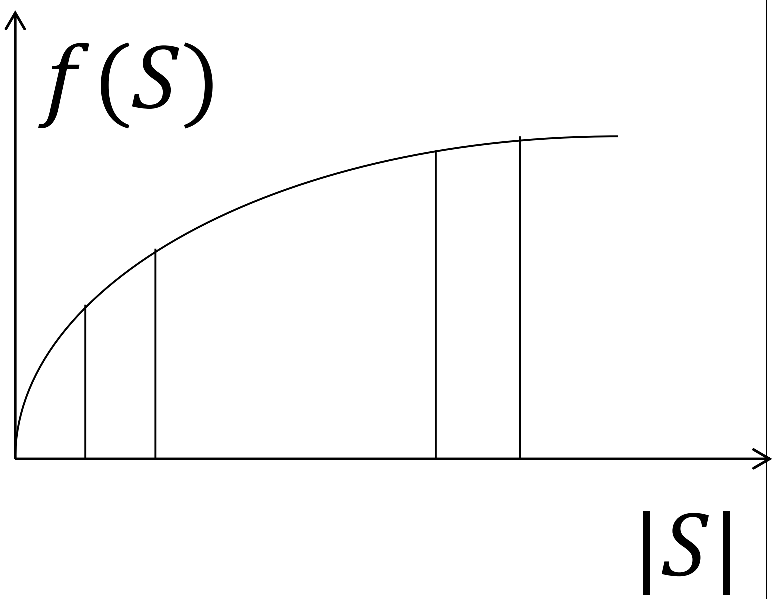
- IC model reflects simple contagion, e.g. information, virus
- LT model reflects complex contagion, e.g. product adoption, innovations (activation needs social affirmation from multiple sources [Centola and Macy, AJS 2007])

Influence maximization

- Given a social network, a diffusion model with given parameters, and a number k, find a seed set S of at most k nodes such that the influence spread of S is maximized.
- To be considered shortly
- Based on *submodular function* maximization

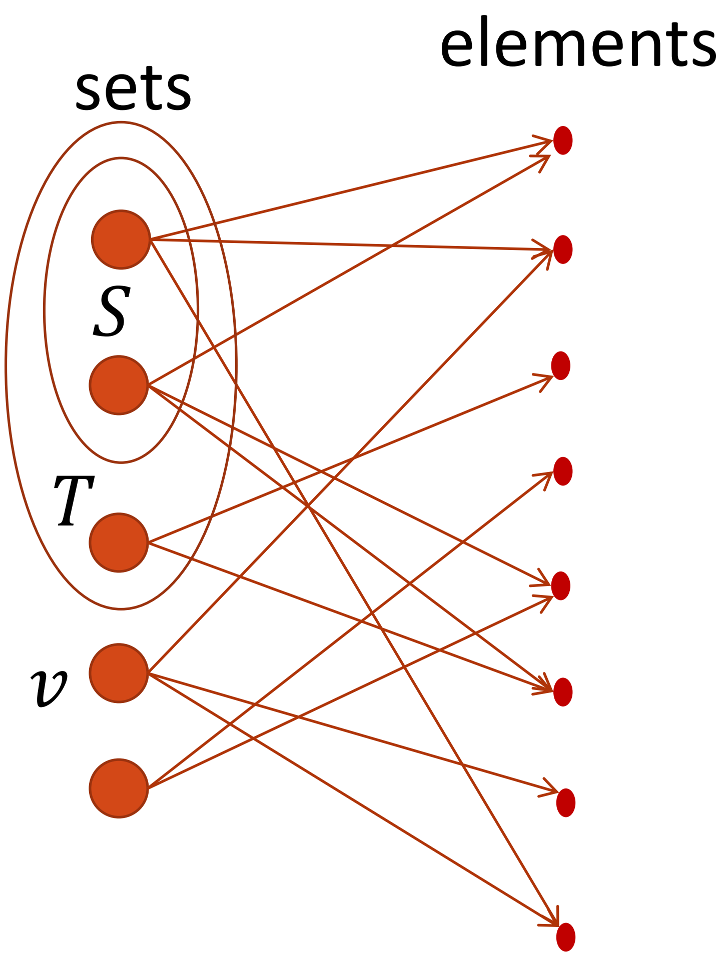
Submodular set functions

- **Sumodularity** of set functions $f: 2^V \to R$
  - for all $S \subseteq T \subseteq V$, all $v \in V \setminus T$,
    $f(S \cup \{v\}) - f(S)$
    $\geq f(T \cup \{v\}) - f(T)$
  - diminishing marginal return
  - an equivalent form: for all $S, T \subseteq V$
    $f(S \cup T) + f(S \cap T) \leq f(S) + f(T)$
- **Monotonicity** of set functions f: for all $S \subseteq T \subseteq V$,
  $$f(S) \leq f(T)$$



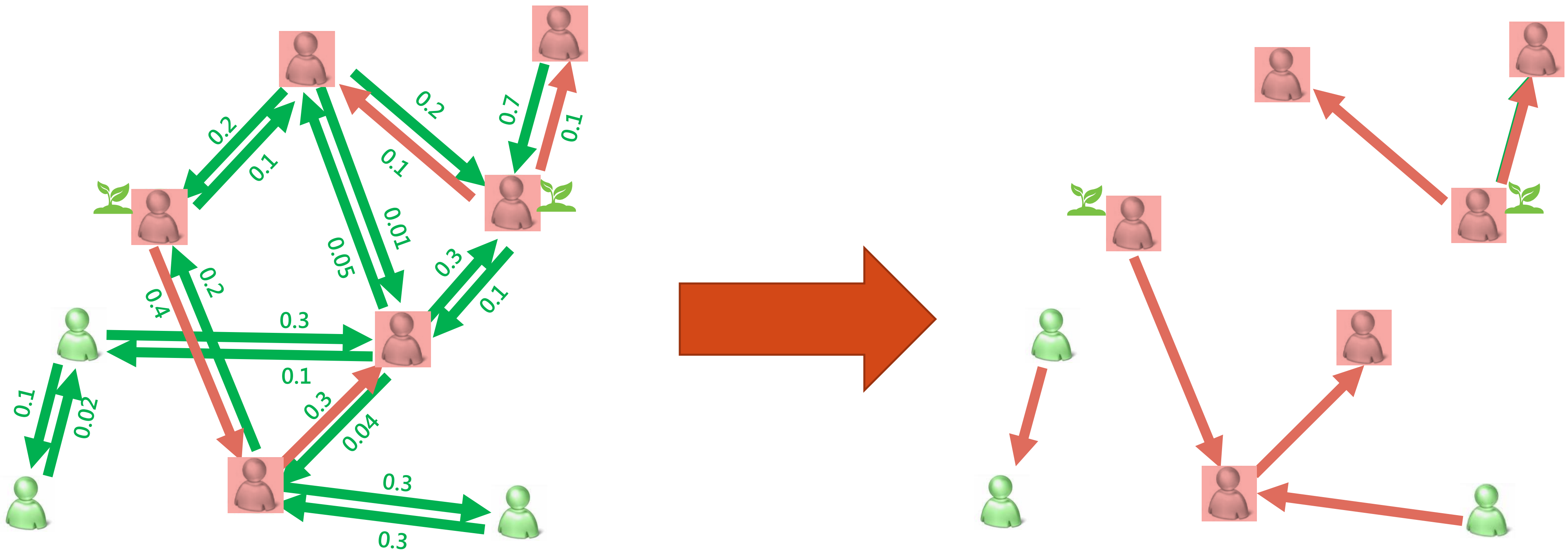
Example of a submodular function and its maximization problem

- set coverage
  - each entry u is a subset of some base elements
  - coverage $f(S) = |\cup_{u \in S} u|$
  - $f(S \cup \{v\}) - f(S)$: additional coverage of v on top of S
- k-max cover problem
  - find k subsets that maximizes their total coverage
  - NP-hard
  - special case of IM problem in IC model



Submodularity of influence diffusion models

- Based on equivalent live-edge graphs



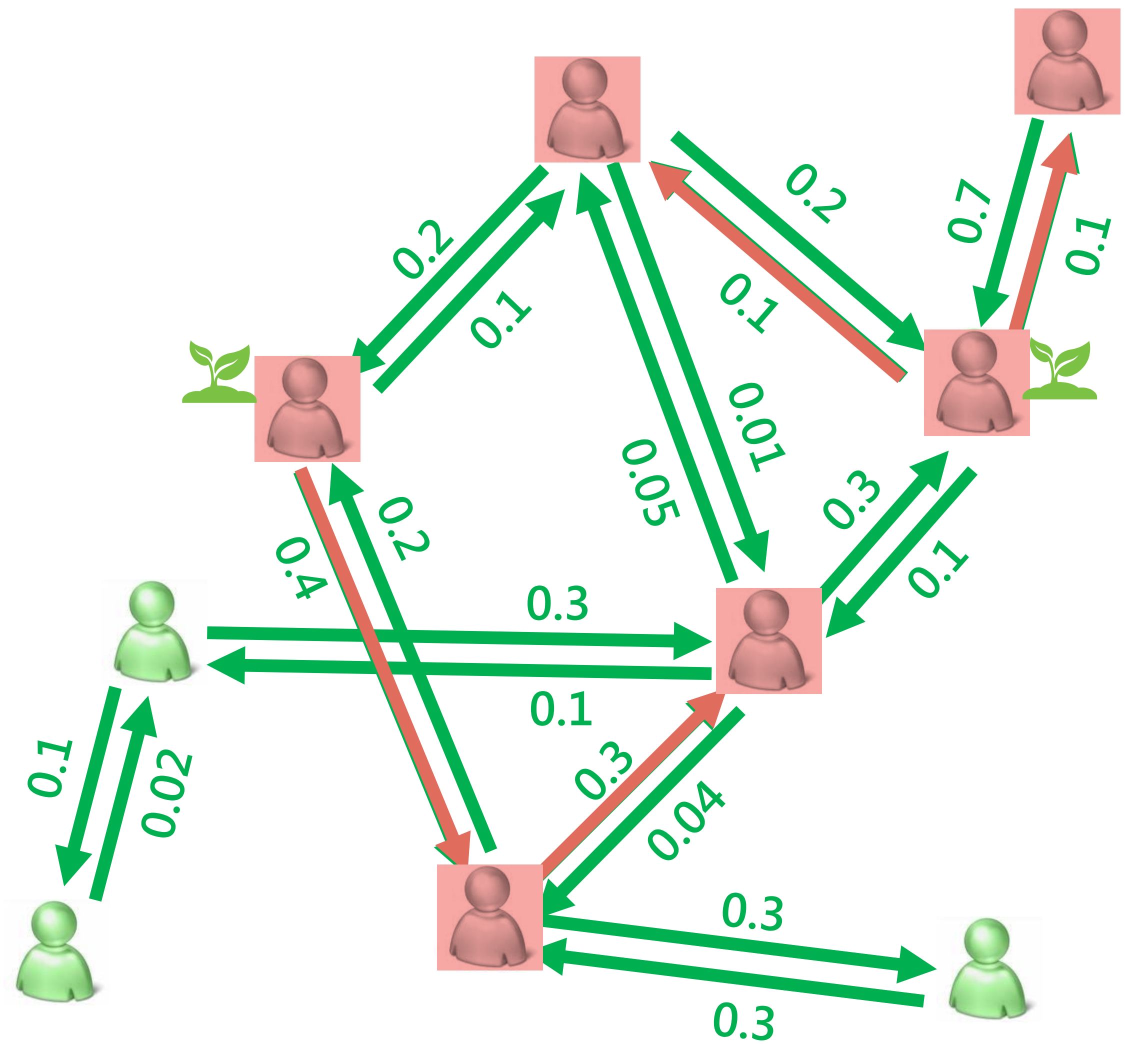
diffusion dynamic

random live-edge graph: edges are randomly removed

Pr(set A is activated given seed set S) = Pr(set A is reachable from S in random live-ledge graph)

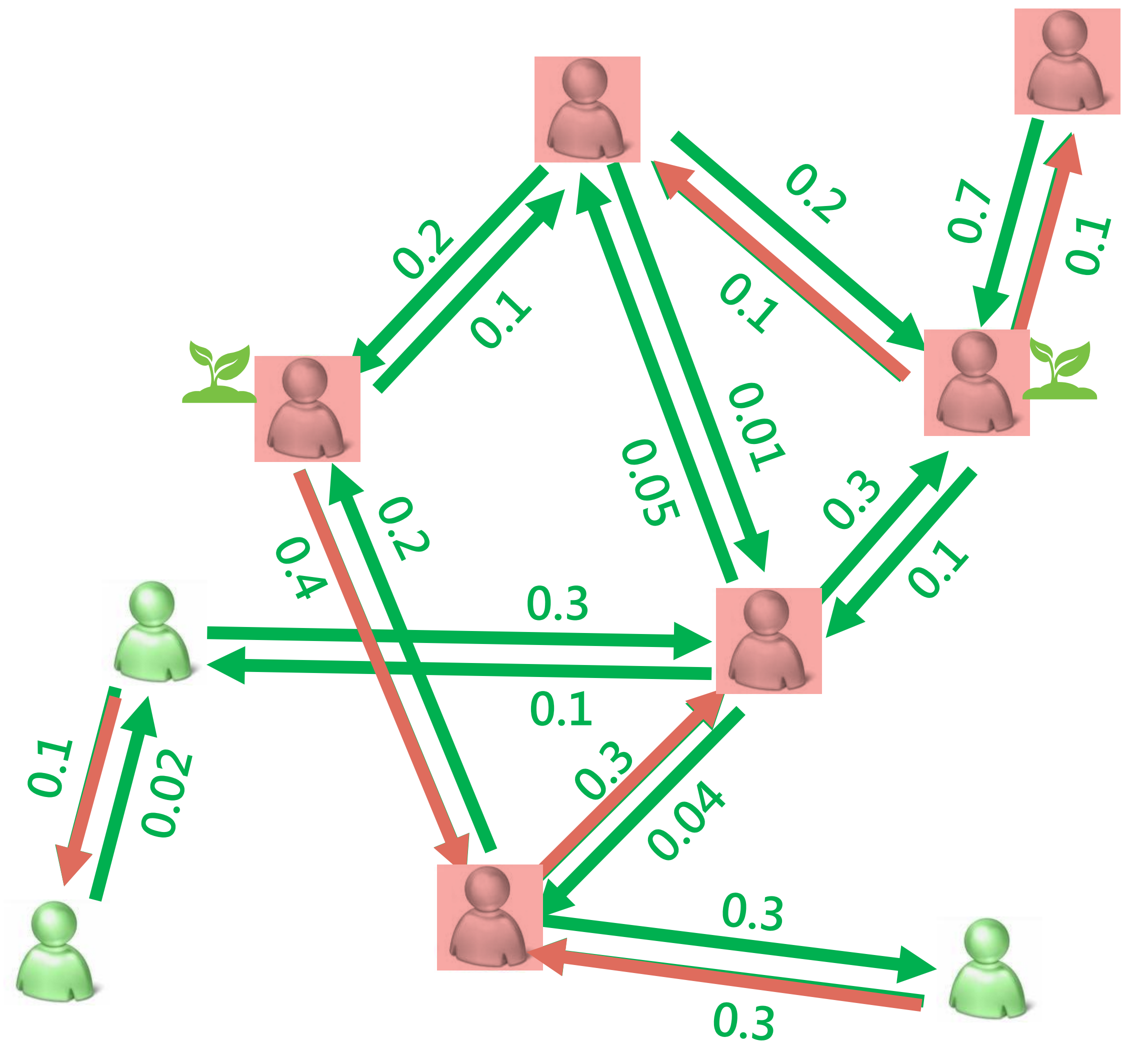
(Recall) active node set via IC diffusion process

- Pink node set is the active node set after the diffusion process in the independent cascade model



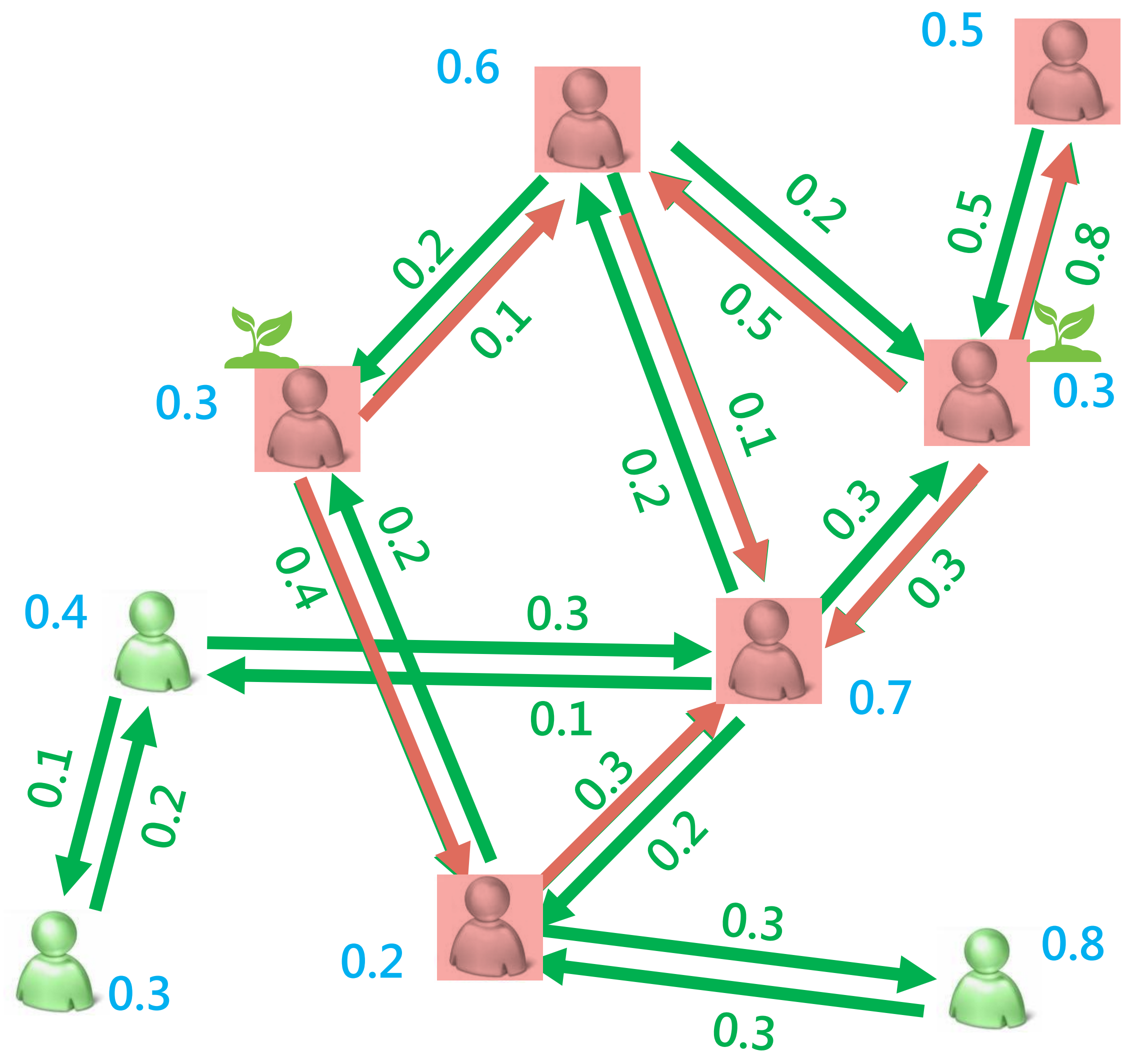
Random live-edge graph for the IC model and its reachable node set

- Random live-edge graph in the IC model
  - each edge is independently selected as live with its influence probability
- Pink node set is the active node set reachable from the seed set in a random live-edge graph
- Equivalence is straightforward



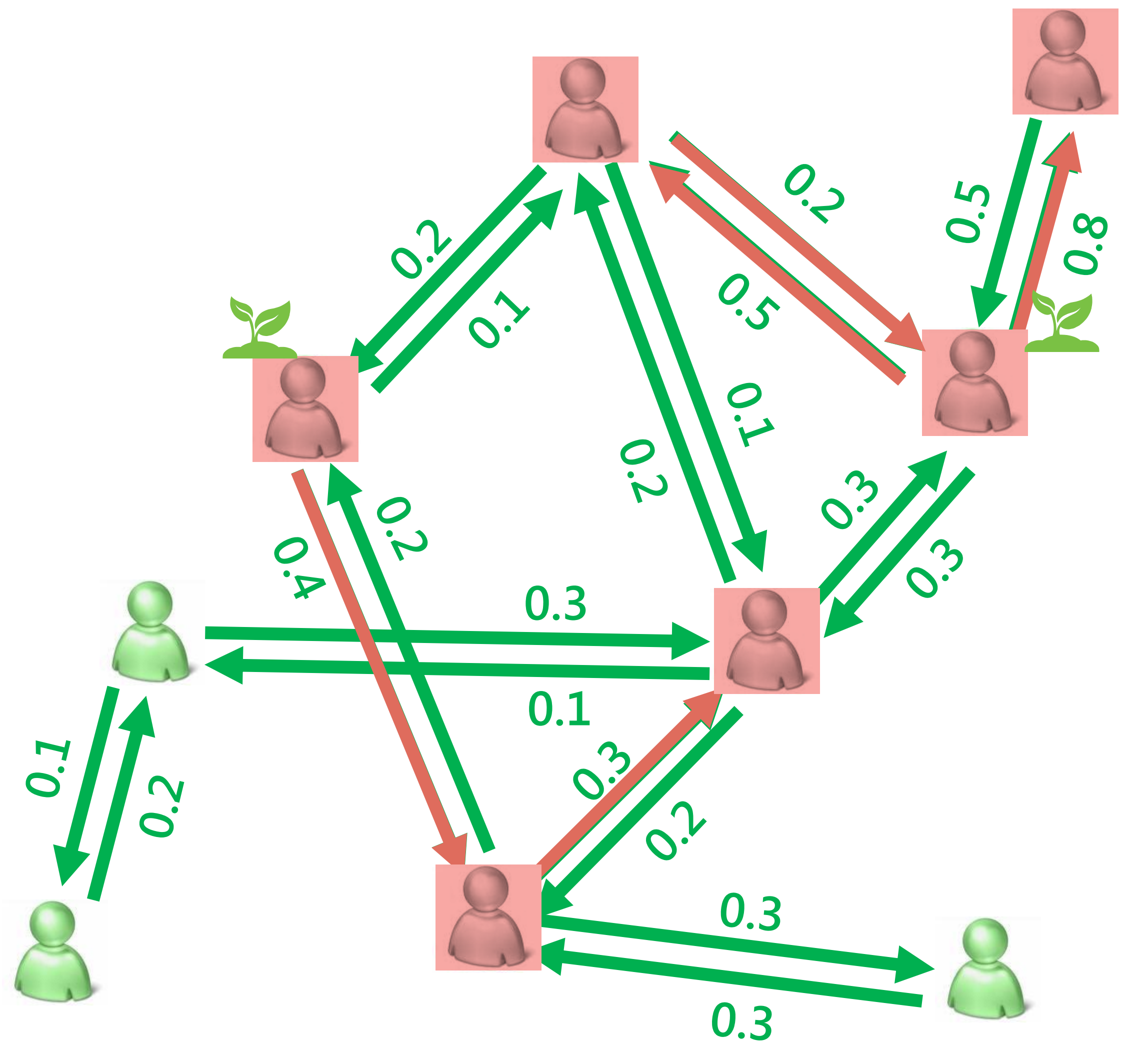
(Recall) active node set via LT diffusion process

- Pink node set is the active node set after the diffusion process in the linear threshold model



Random live-edge graph for the LT model and its reachable node set

- Random live-edge graph in the LT model
  - each node select at most one incoming edge, with probability proportional to its influence weight
- Pink node set is the active node set reachable from the seed set in a random live-edge graph
- Equivalence is based on uniform threshold selection from [0,1], and linear weight addition



Submodularity of influence diffusion models (cont'd)

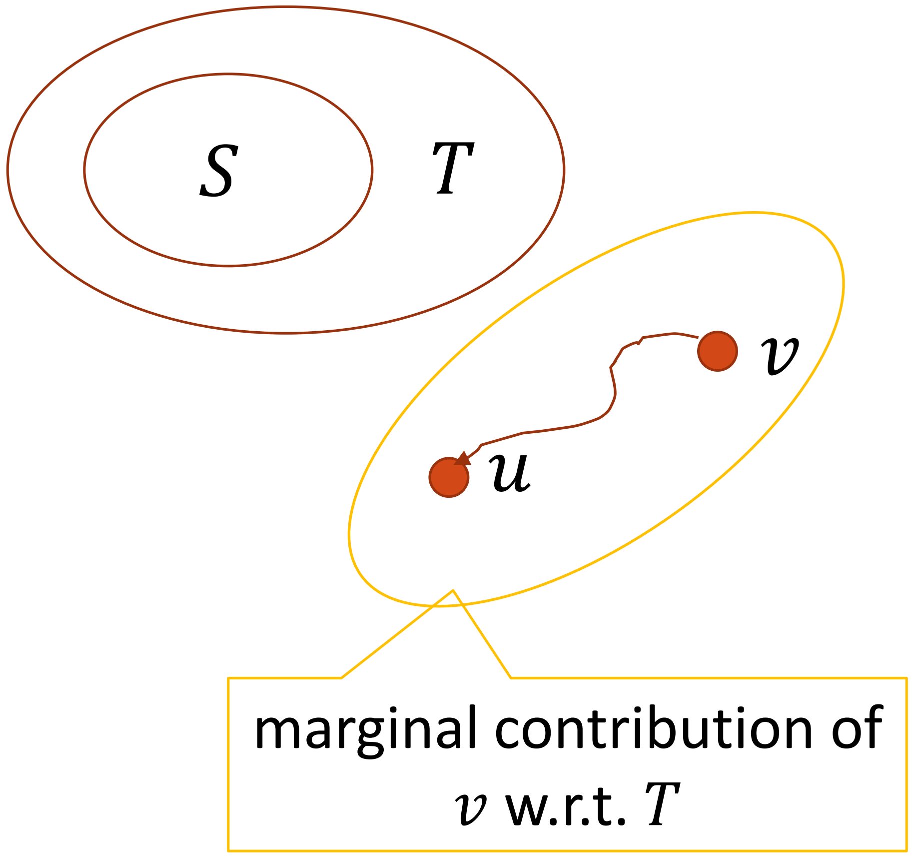
- Influence spread of seed set S, $\sigma(S)$:

$$\sigma(S) = \sum_{G_L} \Pr(G_L)\,|R(S, G_L)|,$$

  - G_L: a random live-edge graph
  - $\Pr(G_L)$: probability of G_L being generated
  - $R(S, G_L)$: set of nodes reachable from S in G_L
- To prove that $\sigma(S)$ is submodular, only need to show that $|R(\cdot, G_L)|$ is submodular for any G_L
  - sumodularity is maintained through linear combinations with non-negative coefficients

Submodularity of influence diffusion models (cont'd)

- Submodularity of $|R(\cdot, G_L)|$
  - for any $S \subseteq T \subseteq V,\ v \in V \setminus T$,
  - if u is reachable from v but not from T, then
  - u is reachable from v but not from S
  - Hence, $|R(\cdot, G_L)|$ is submodular
- Therefore, influence spread $\sigma(S)$ is submodular in both IC and LT models



General threshold model

- Each node v has a threshold function
$$f_v: 2^V \rightarrow [0,1]$$
- Each node v selects a threshold $\theta_v \in [0,1]$ uniformly at random
- If the set of active nodes at the end of step $t-1$ is S, and $f_v(S) \geq \theta_v$, v is activated at step t
- reward function $r(A(S))$: if $A(S)$ is the final set of active nodes given seed set S, $r(A(S))$ is the reward from this set
- generalized influence spread:
$$\sigma(S) = E[r(A(S))]$$

IC and LT as special cases of general threshold model

- LT model
  - $f_v(S) = \sum_{u \in S} w(u,v)$
  - $r(S) \; = \; |S|$
- IC model
  - $f_v(S) = 1 - \prod_{u \in S}(1 - p(u,v))$
  - $r(S) \; = \; |S|$

Submodularity in the general threshold model

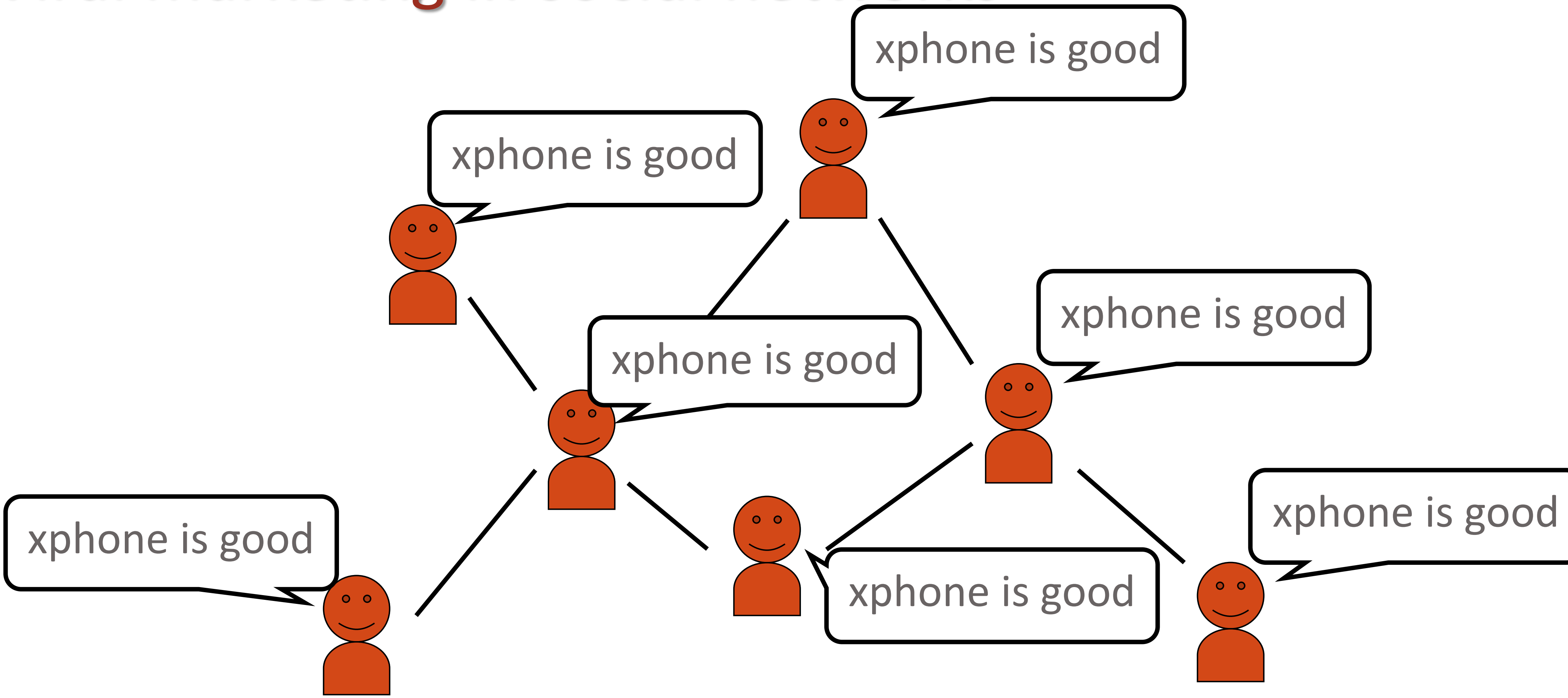
- Theorem [Mossel & Roch STOC 2007]:
  - In the general threshold model,
  - if for every $v \in V$, $f_v(\cdot)$ is monotone and submodular with $f_v(\emptyset) = 0$,
  - and the reward function $r(\cdot)$ is monotone and submodular,
  - then the general influence spread function $\sigma(\cdot)$ is monotone and submodular.
- Local submodularity implies global submodularity

Summary of diffusion models

- Main progressive models
  - IC and LT models
- Main properties: submodularity and monotonicity
- Other diffusion models:
  - Epidemic models: SI, SIR, SIS, SIRS, etc.
  - Voter model
  - Markov random field model
  - Percolation theory

Influence Maximization

Viral marketing in social networks



- Viral effect (word-of-mouth effect) is believed to be a promising advertising strategy.
- Increasing popularity of online social networks may enable large scale viral marketing

Influence maximization

- Given a social network, a diffusion model with given parameters, and a number k, find a seed set S of at most k nodes such that the influence spread of S is maximized.
- May be further generalized:
  - Instead of k, given a budget constraint and each node has a cost of being selected as a seed
  - Instead of maximizing influence spread, maximizing a (submodular) function of the set of activated nodes

Hardness of influence maximization

- Influence maximization under both IC and LT models are NP hard
  - IC model: reduced from k-max cover problem
  - LT model: reduced from vertex cover problem
- Need approximation algorithms

Greedy algorithm for submodular function maximization

```
initialize S = ∅;
for i = 1 to k do
    select u = argmax_{w∈V\S}[f(S ∪ {w}) − f(S)]
    S = S ∪ {u}
end for
output S
```

Property of the greedy algorithm

- Theorem: If the set function f is monotone and submodular with $f(\emptyset) = 0$, then the greedy algorithm achieves $(1 - 1/e)$ approximation ratio, that is, the solution S found by the greedy algorithm satisfies:
  - $f(S) \geq \left(1 - \frac{1}{e}\right) \max_{S' \subseteq V, |S'| = k} f(S')$

Proof of the theorem

$S_0^* = S_0^g = \emptyset$ s_i: i-th entry found by algo; $S_i^g = S_{i-1}^g \cup \{s_i\}$

S^*: optimal set; $S^* = \{s_1^*, \ldots, s_k^*\}$; $S_j^* = \{s_1^*, \ldots, s_j^*\}$, for $1 \leq j \leq k$

$f(S^*) \leq f(S_i^g \cup S^*)$ /* by monotonicity */

$\leq f\left(S_i^g \cup \{s_k^*\}\right) - f(S_i^g) + f(S_i^g \cup S_{k-1}^*)$ /* by submodularity */

$\leq f(S_{i+1}^g) - f(S_i^g) + f(S_i^g \cup S_{k-1}^*)$ /* by greedy algorithm*/

$\leq k(f\left(S_{i+1}^g\right) - f(S_i^g)) + f(S_i^g)$ /* by repeating the above k times */

Rearranging the inequality: $f\left(S_{i+1}^g\right) \geq \left(1 - \frac{1}{k}\right) f\left(S_i^g\right) + \frac{f(S^*)}{k}$.

Multiplying by $\left(1 - \frac{1}{k}\right)^{k-i-1}$ on both sides, and adding up all inequalities:

$f\left(S_k^g\right) \geq \sum_{i=0}^{k-1} \left(1 - \frac{1}{k}\right)^{k-i-1} \cdot \frac{f(S^*)}{k} = \left(1 - \left(1 - \frac{1}{k}\right)^k\right) f(S^*) \geq \left(1 - \frac{1}{e}\right) f(S^*)$.

Influence computation is hard

- In IC and LT models, computing influence spread $\sigma(S)$ for any given S is #P-hard.
  - IC model: reduction from the s-t connectedness counting problem.
  - LT model: reduction from simple path counting problem.

MC-Greedy: Estimating influence spread via Monte Carlo simulations

- For any given S
- Simulate the diffusion process from S for R times (R should be large)
- Use the average of the number of active nodes in R simulations as the estimate of $\sigma(S)$
- Can estimate $\sigma(S)$ to arbitrary accuracy, but require large R
  - Theoretical bound can be obtained using Chernoff bound.

Theorems on MC-Greedy algorithm

Theorem 3.6 *Let* $S^* = \text{argmax}_{|S|\leq k} f(S)$ *be the set maximizing* $f(S)$ *among all sets with size at most* k*, where* f *is monotone and submodular, and* $f(\emptyset) = 0$*. For any* $\varepsilon > 0$*, for any* γ *with* $0 < \gamma \leq \frac{\varepsilon/k}{2+\varepsilon/k}$*, for any set function estimate* $\hat{f}$ *that is a multiplicative* γ*-error estimate of set function* f*, the output* S^g *of* Greedy$(k, \hat{f})$ *guarantees*

$$f(S^g) \geq \left(1 - \frac{1}{e} - \varepsilon\right) f(S^*).$$

Theorem 3.7 *With probability* $1 - 1/n$*, algorithm* MC-Greedy(G, k) *achieves* $(1 - 1/e - \varepsilon)$ *approximation ratio in time* $O(\varepsilon^{-2}k^3n^2m\log n)$*, for both IC and LT models.*

- Polynomial, but could be very slow

Empirical evaluation of MC-Greedy

- Use a network NetHEPT
  - Collaboration network in arXiv, High Energy Physics-Theory section, 1991-2003
  - Edge: two authors have a co-authored paper
  - Allow duplicated edges

Number of nodes	15233
Number of edges with duplicated edges	58891
Number of edges	31398
Average degree	4.12
Maximal degree	64
number of connected components	1781
Largest component size	6794
Average component size	8.55

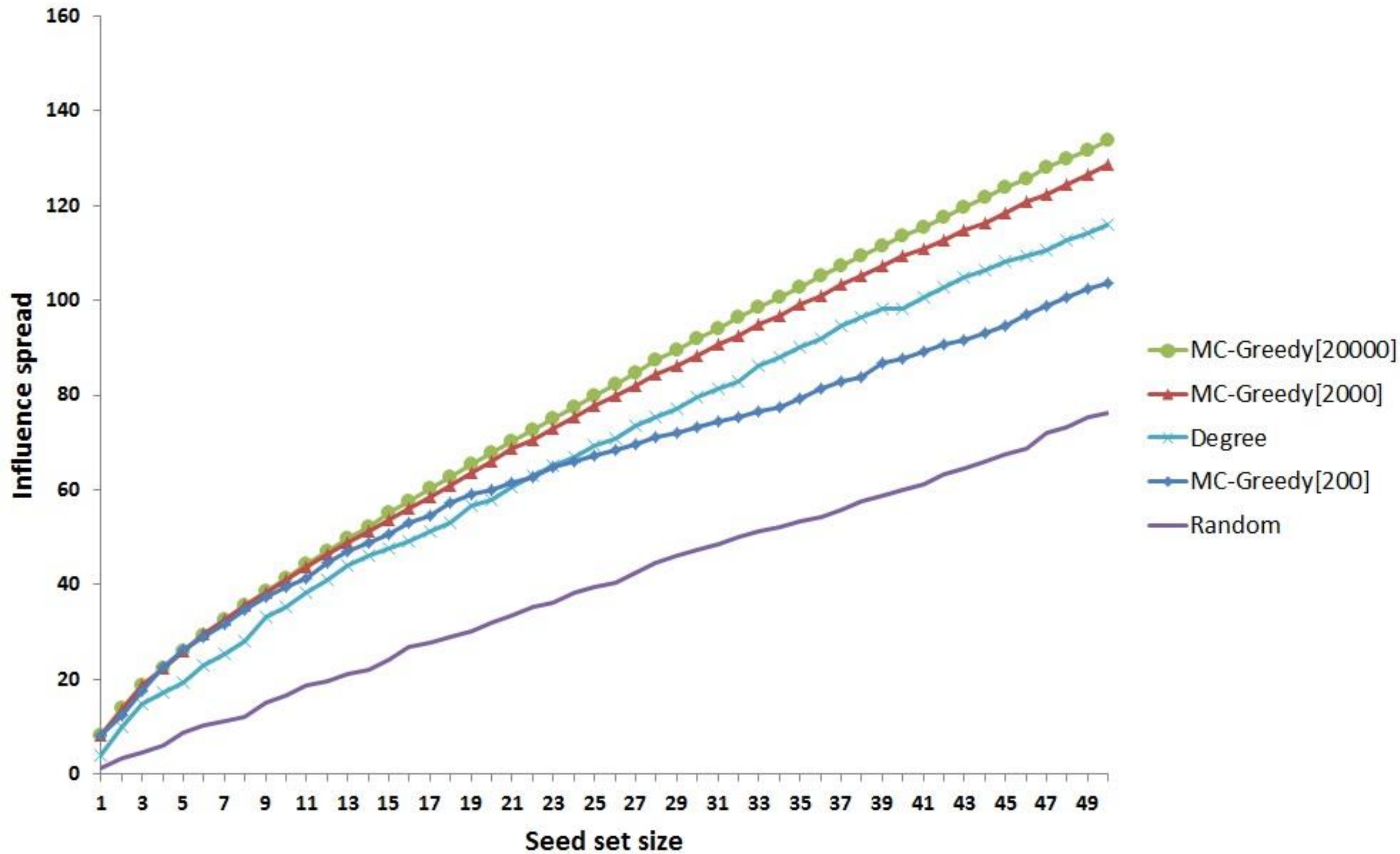
Algorithms to compare

- MC-Greedy[R]: Monte Carlo greedy algorithm with R simulations
- Degree: high-degree heuristic
- Random: random selection

Parameter setting

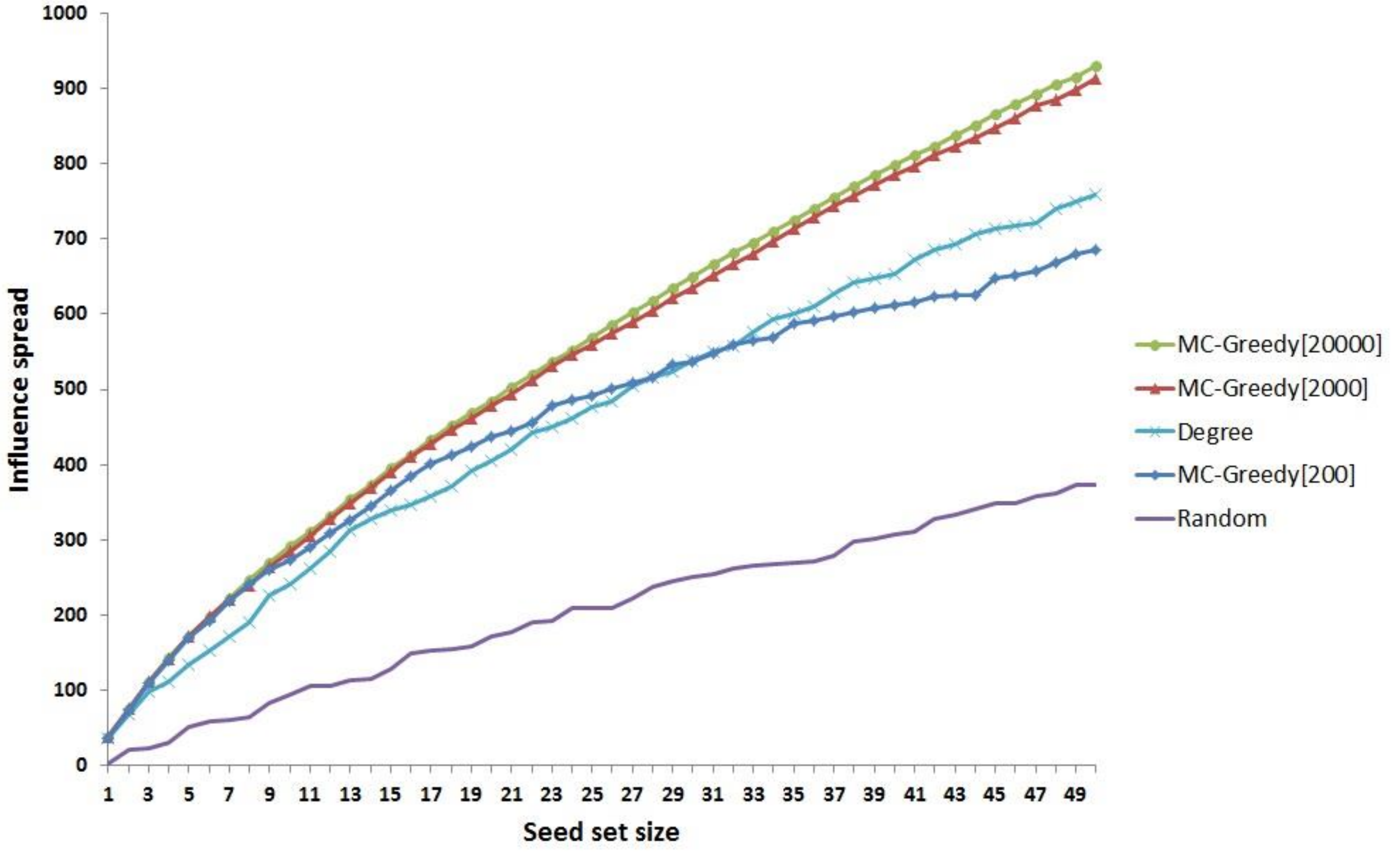
- Edge weights
  - IC-UP[0.01]: IC model, each edge has probability 0.01.
  - IC-WC: IC model with weighted cascade probabilities
    - each in-coming edge has probability $1/d(v)$, where $d(v)$ is the in-degree of v.
  - LT-UW: LT model with uniform weights
    - Each in-coming edge of v has weight $1/d(v)$
  - All parameters above are before removing duplicates
- Number of MC simulations R = 200, 2000, 20000
- Influence spread computed with 20000 simulations

IC-UP[0.01] Influence spread result



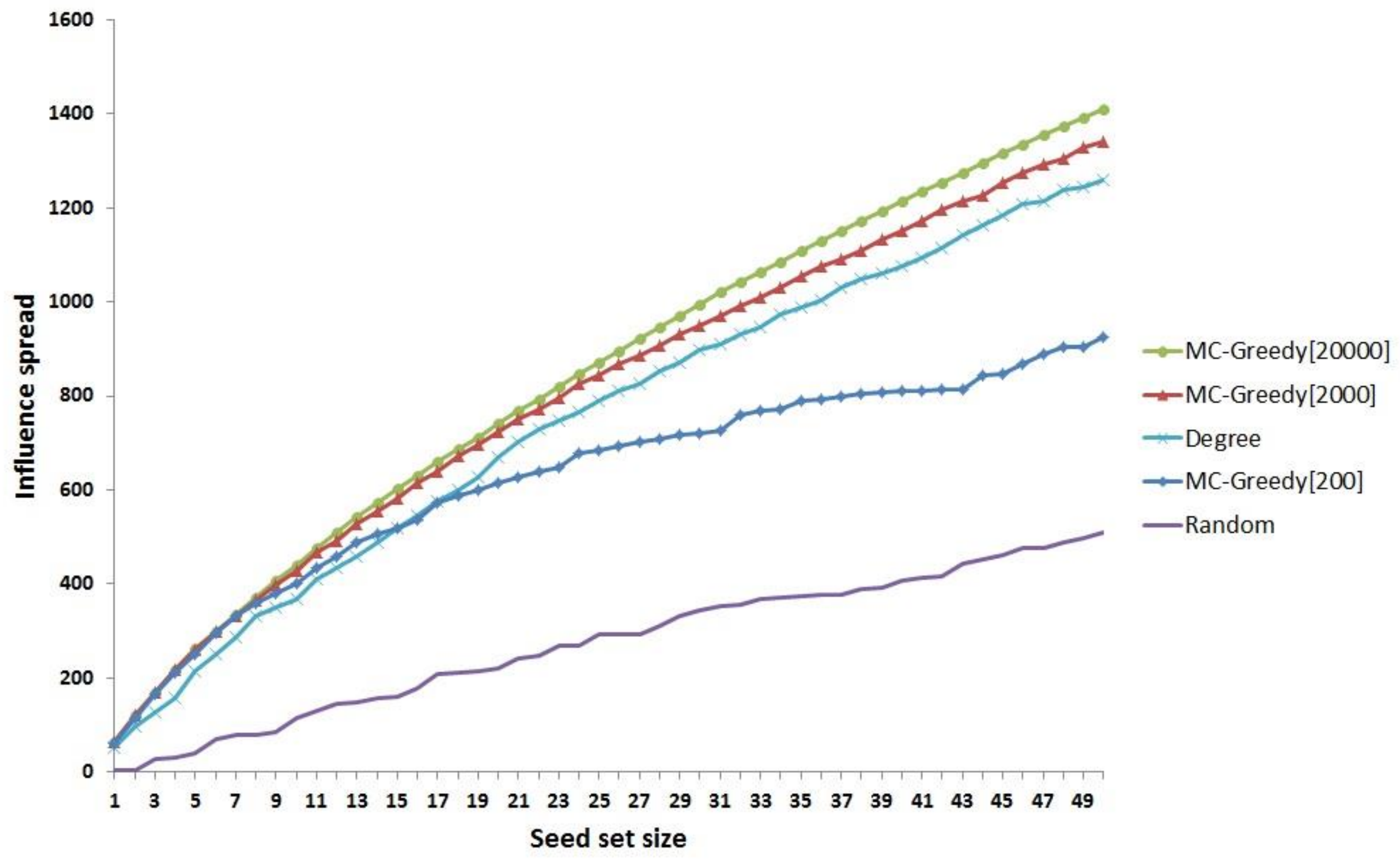
- MC-Greedy[20000] is the best
- MC-Greedy[200] is worse than Degree
- Random is the worst

IC-WC result



- MC-Greedy[20000] is the best
- MC-Greedy[200] is worse than Degree
- Random is the worst

LT-UW result



- MC-Greedy[20000] is the best
- MC-Greedy[200] is worse than Degree
- Random is the worst

Scalable Influence Maximization

Drawback of MC-Greedy

- Very slow: on NetHEPT with ICUP[0.01], finding 50 seeds
  - MC-Greedy[2000] takes 73.6 hours
  - MC-Greedy[200] takes 6.6 hours
- Two sources of inefficiency:
  - Too many influence spread ($\sigma(S)$) evaluations
  - Monte Carlo simulation for each $\sigma(S)$ is slow

Ways to improve scalability

- Reduce the number of influence spread evaluations
  - Lazy evaluation
- Avoid Monte Carlo simulations
  - MIA heuristic for IC model

Lazy evaluation

- Exploit submodularity of influence spread function
- For any submodular set function f, $f(u|S) = f(S \cup \{u\}) - f(S)$, u's marginal contribution under S
- In greedy algorithm, the i-th iteration found seed set S_i^g
- Then: $f\left(u|S_i^g\right) \leq f(u|S_j^g)$ for all $i > j$
- Lazy evaluation: at i-th iteration, $i > j$, for two nodes u and v, if $f\left(u\middle|S_j^g\right) \leq f(v|S_i^g)$, then $f\left(u|S_i^g\right)$ does not need to be evaluated at the i-th iteration

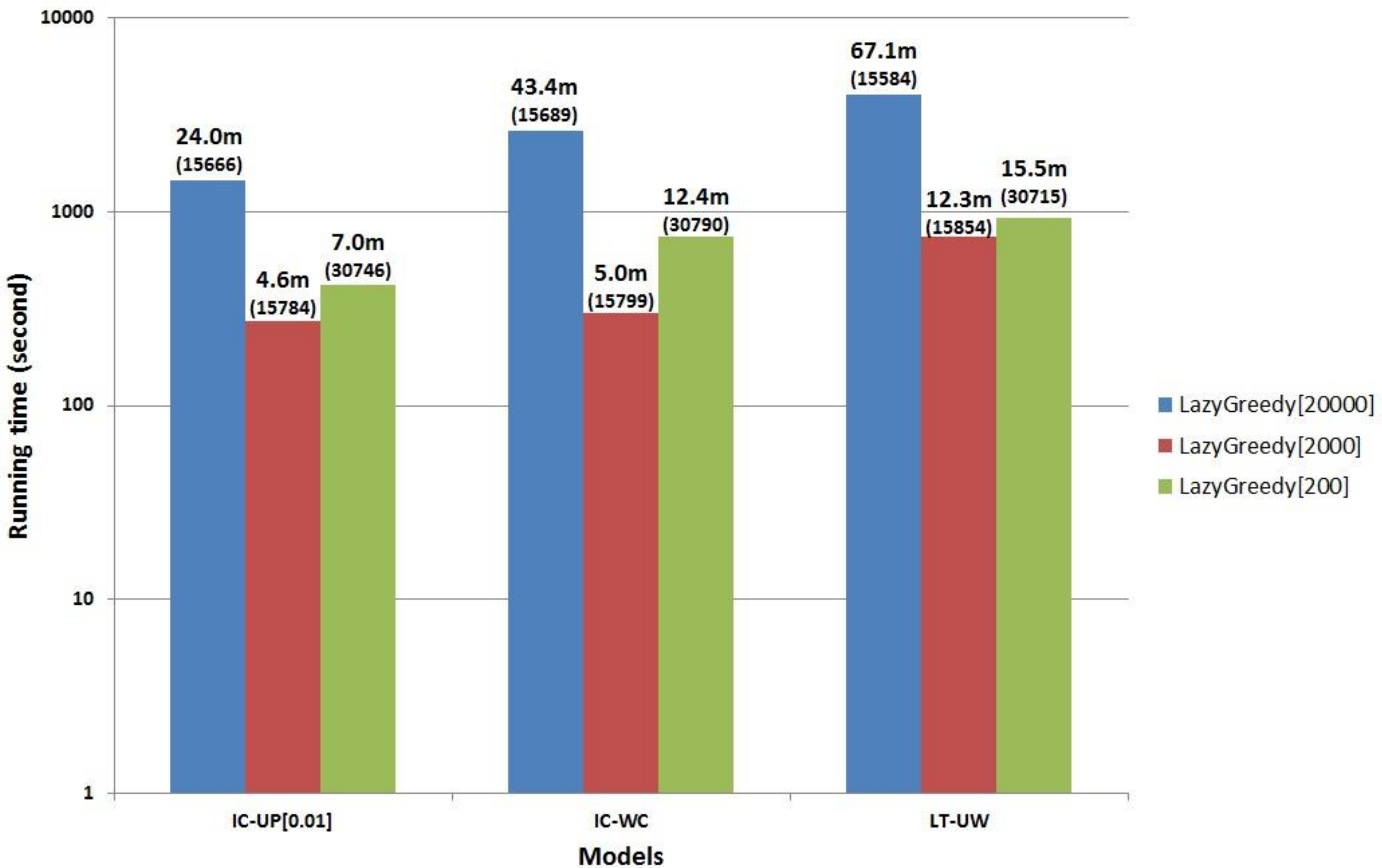
Algorithm 3 LazyGreedy(k, f): general greedy algorithm with lazy evaluations.

Input: k: size of returned set; f: monotone and submodular set function

Output: selected subset

initialize $S \leftarrow \emptyset$; priority queue $Q \leftarrow \emptyset$; $iteration \leftarrow 1$
for $i = 1$ to n **do**
    $u.mg \leftarrow f(u \mid \emptyset)$; $u.i \leftarrow 1$
    insert element u into Q with $u.mg$ as the key
end for
while $iteration \leq k$ **do**
    extract top (max) element u of Q
    if $u.i = iteration$ **then**
        $S \leftarrow S \cup \{u\}$; $iteration \leftarrow iteration + 1$;
    else
        $u.mg \leftarrow f(u \mid S)$; $u.i \leftarrow iteration$
        re-insert u into Q
    end if
end while
return S

Running time of Lazy-Greedy



Fast heuristics

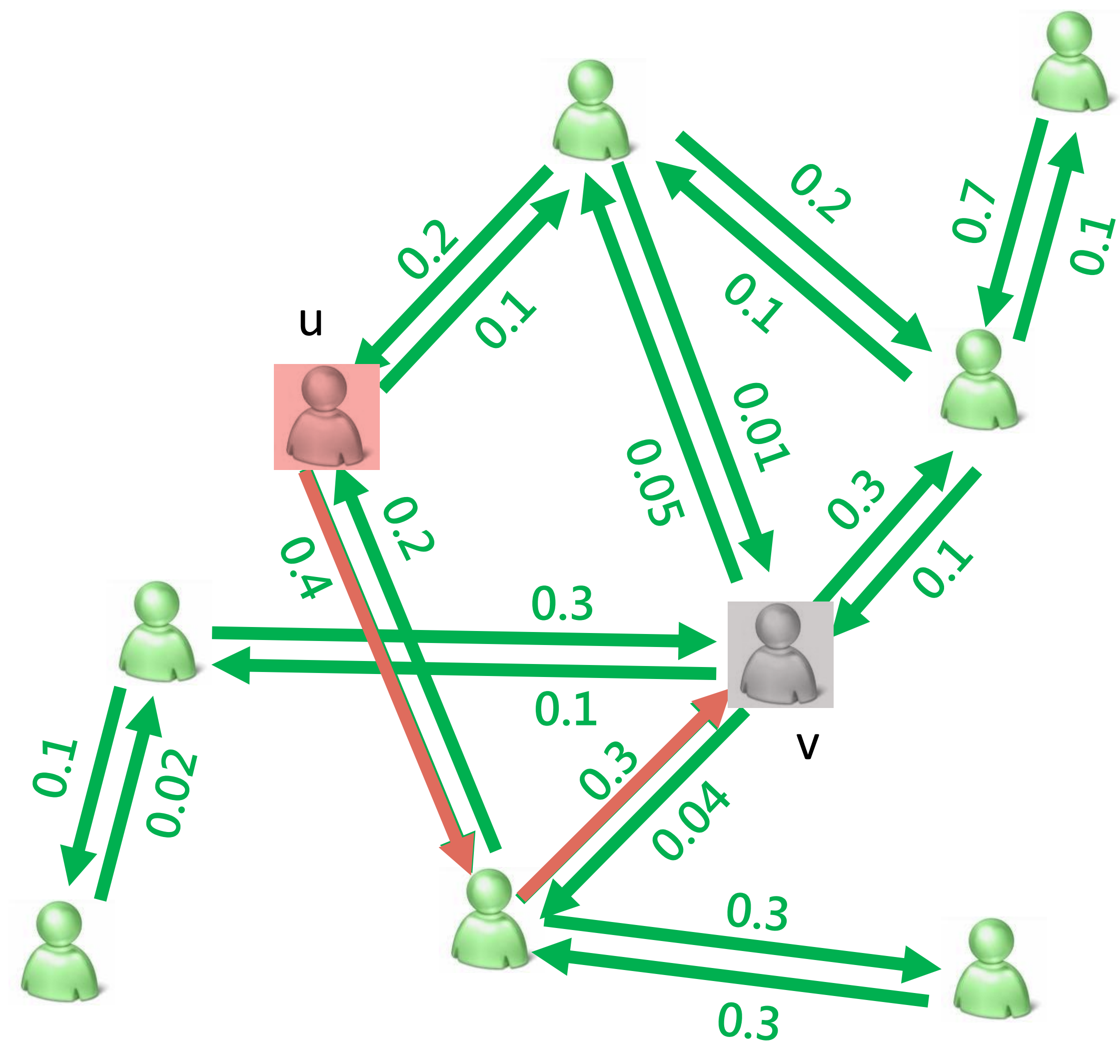
- The running time of Lazy-Greedy is still slow, and not scalable to large graphs (millions of nodes and edges)
- Need faster heuristic to avoid Monte Carlo simulations

Our work

- Exact influence computation is #P hard, for both IC and LT models --- computation bottleneck [KDD'10, ICDM'10]
- Design new heuristics
  - MIA for general IC model [KDD'10]
    - 10^3 speedup --- from hours to seconds
    - influence spread close to that of the greedy algorithm of [KKT'03]
  - Degree discount heuristic for uniform IC model [KDD'09]
    - 10^6 speedup --- from hours to milliseconds
  - LDAG for LT model [ICDM'10]
    - 10^3 speedup --- from hours to seconds
  - IRIE for IC model [ICDM'12]
    - further improvement with time and space savings
- Extend to time-critical influence maximization [AAAI'12]

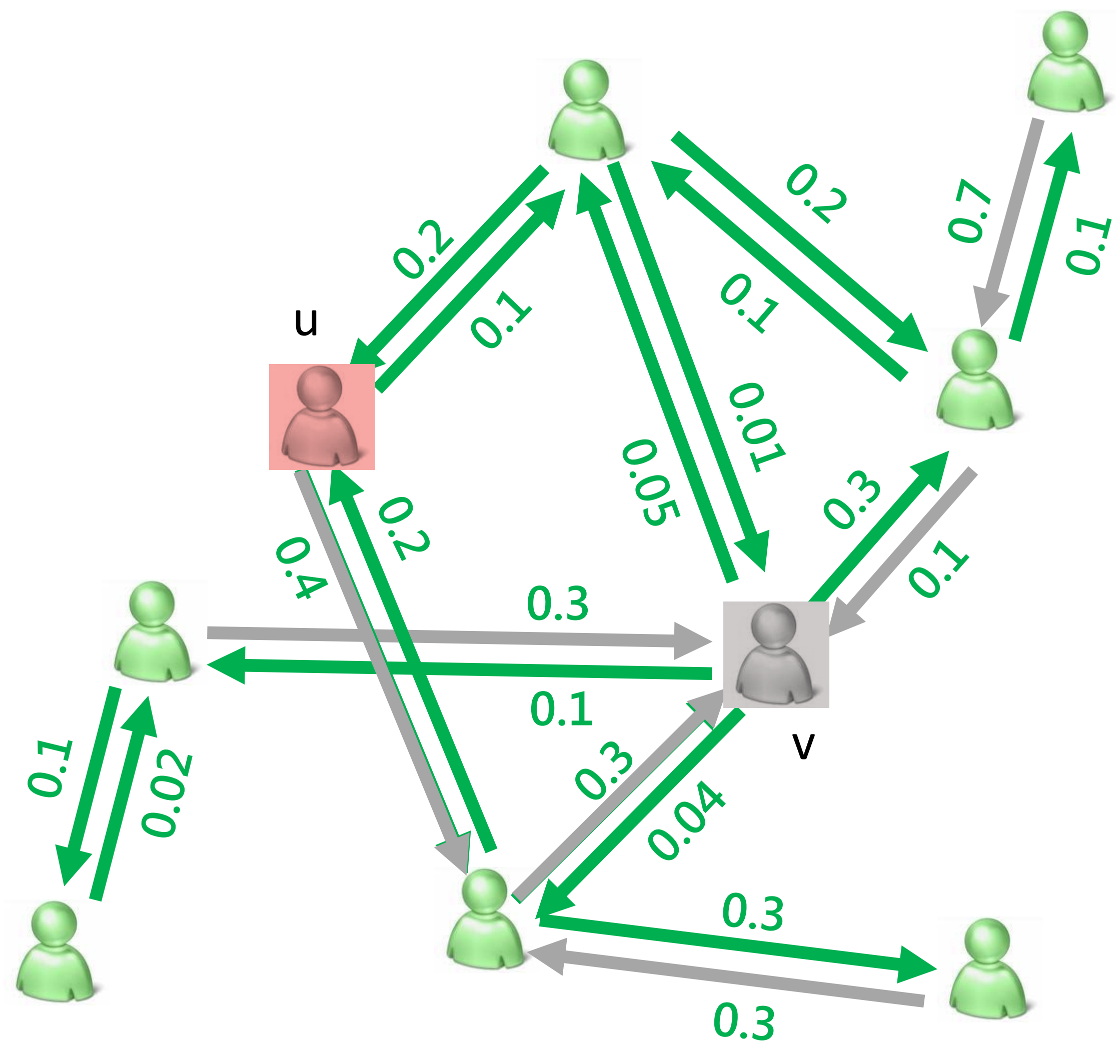
Maximum Influence Arborescence (MIA) Heuristic

- For any pair of nodes u and v, find the maximum influence path (MIP) from u to v
- ignore MIPs with too small probabilities (< parameter θ)



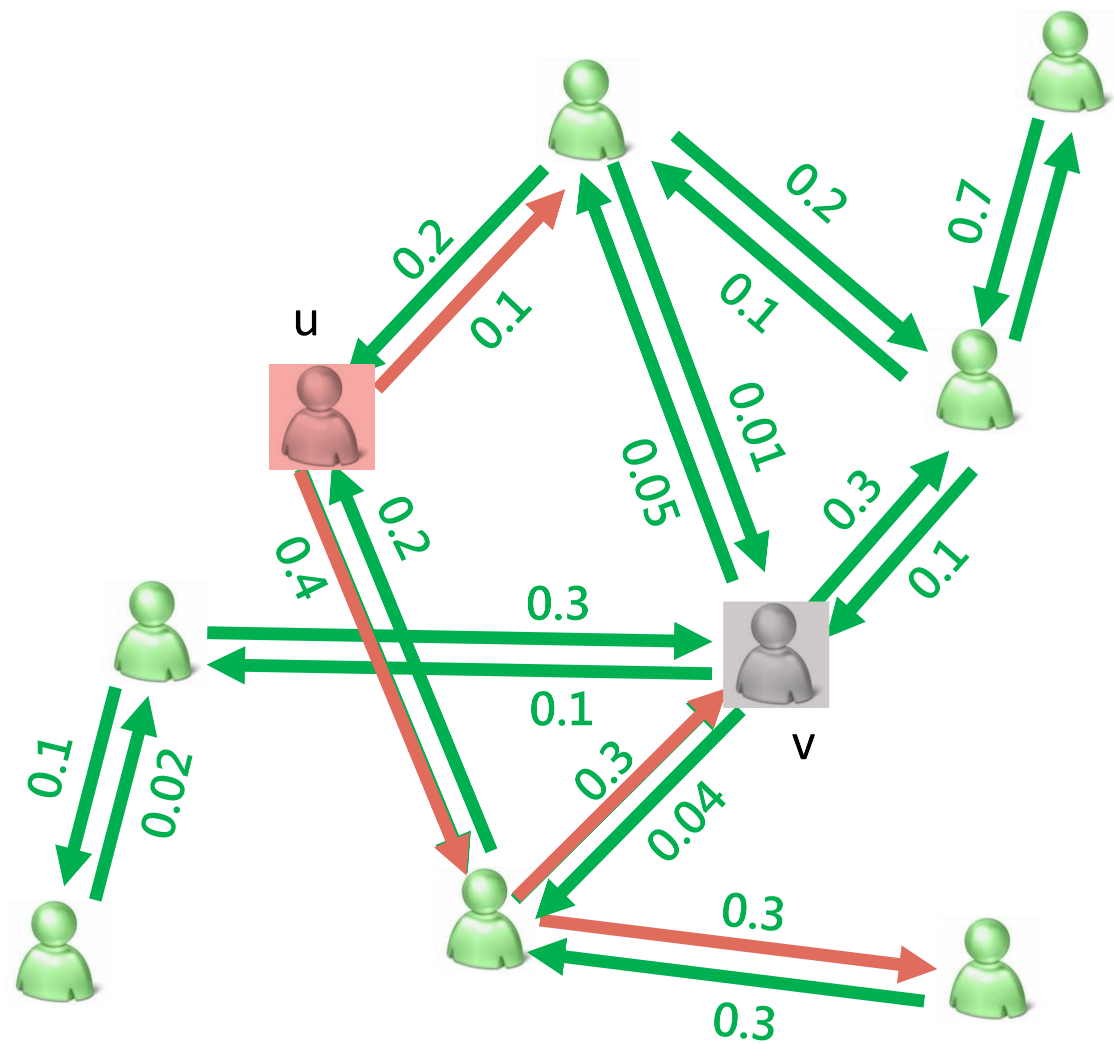
MIA Heuristic (cont'd)

- Local influence regions
  - for every node v, all MIPs to v form its maximum influence in-arborescence (MIIA)



MIA Heuristic (cont'd)

- Local influence regions
  - for every node v, all MIPs to v form its maximum influence in-arborescence (MIIA)
  - for every node u, all MIPs from u form its maximum influence out-arborescence (MIOA)
  - computing MIAs and the influence through MIAs is fast



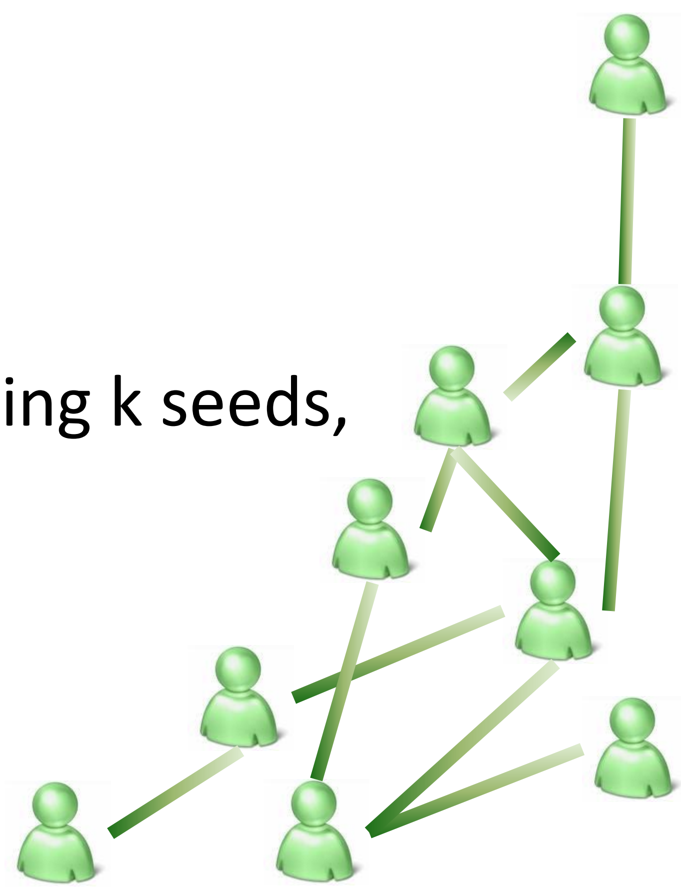
MIA Heuristic III: Computing Influence through the MIA structure

- Recursive computation of activation probability ap(u) of a node u in its in-arborescence, given a seed set S

Algorithm 2 $ap(u, S, MIIA(v, \theta))$

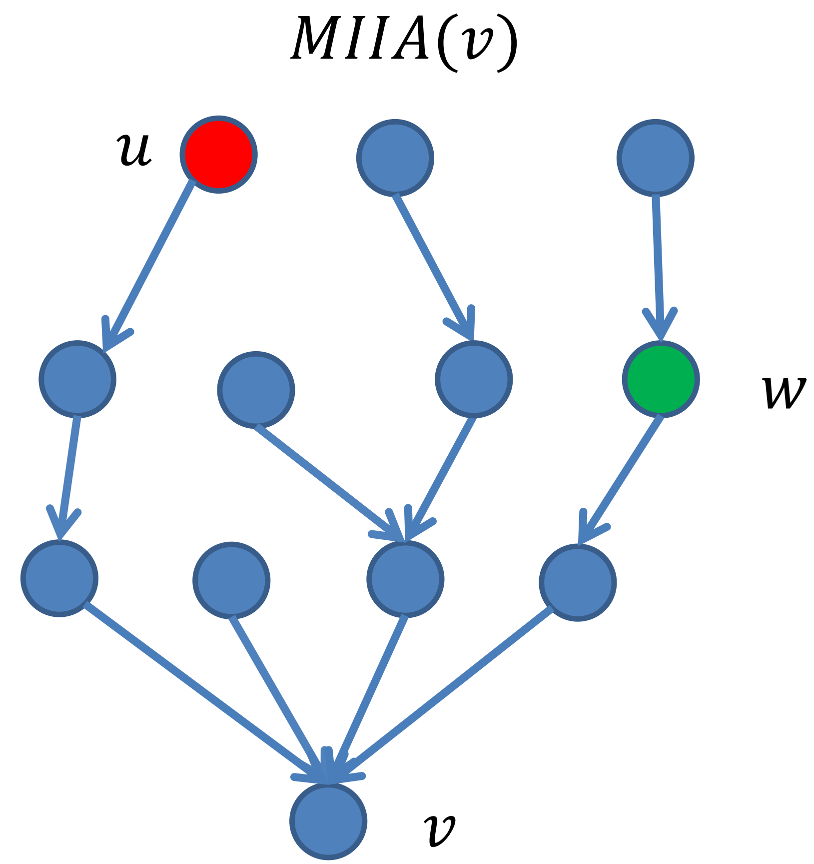
if $u \in S$ **then**
    $ap(u) = 1$
else if $Ch(u) = \emptyset$ **then**
    $ap(u) = 0$
else
    $ap(u) = 1 - \Pi_{w \in Ch(u)}(1 - ap(w) \cdot pp(w, u))$
end if

- Can be used in the greedy algorithm for selecting k seeds, but not efficient enough



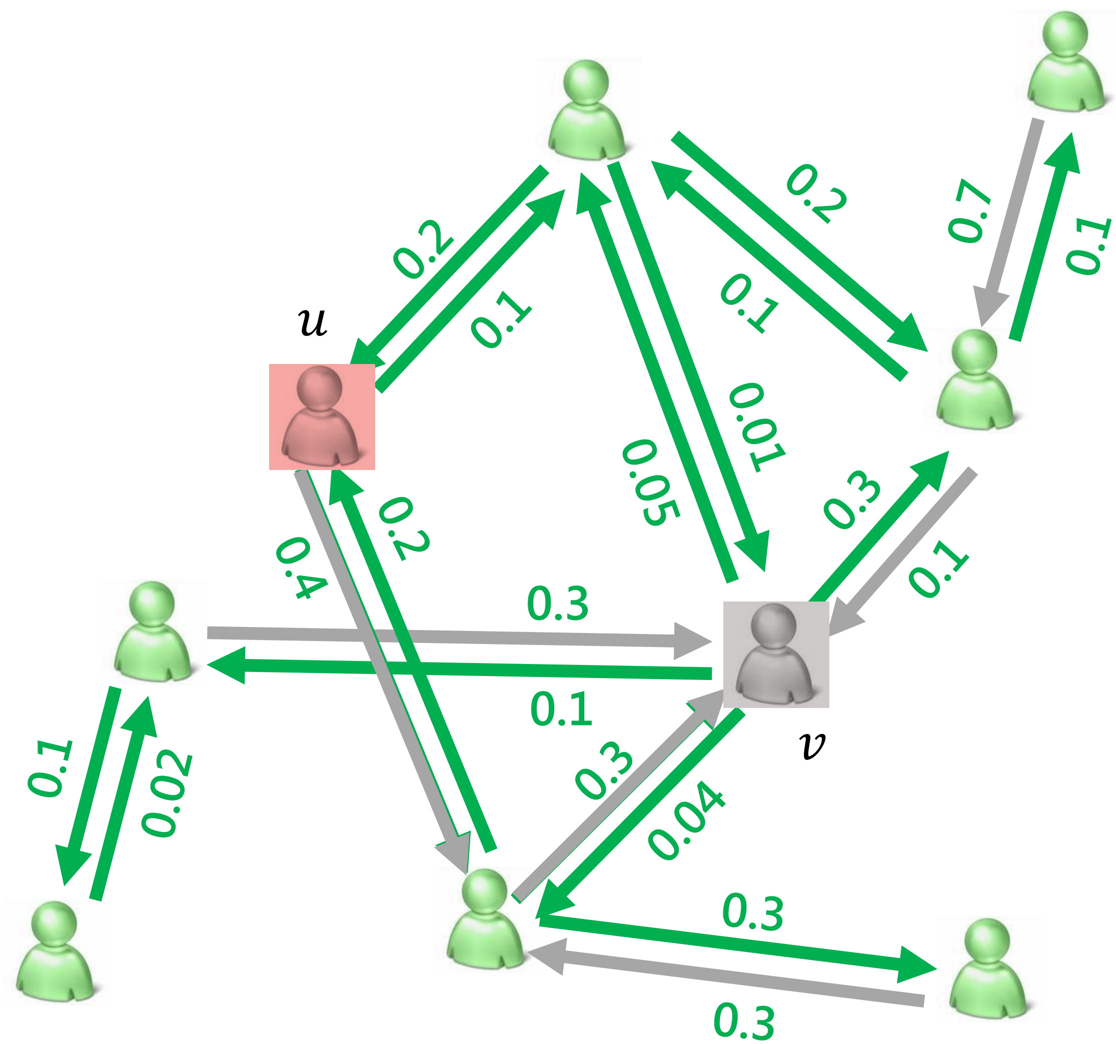
MIA Heuristic IV: Efficient updates on incremental activation probabilities

- u is the new seed in $MIIA(v)$
- Naive update: for each candidate w, redo the computation in the previous page to compute w's incremental influence to v
  - $O(|MIIA(v)|^2)$
- Fast update: based on linear relationship of activation probabilities between any node w and root v, update incremental influence of all w's to v in two passes
  - $O(|MIIA(v)|)$

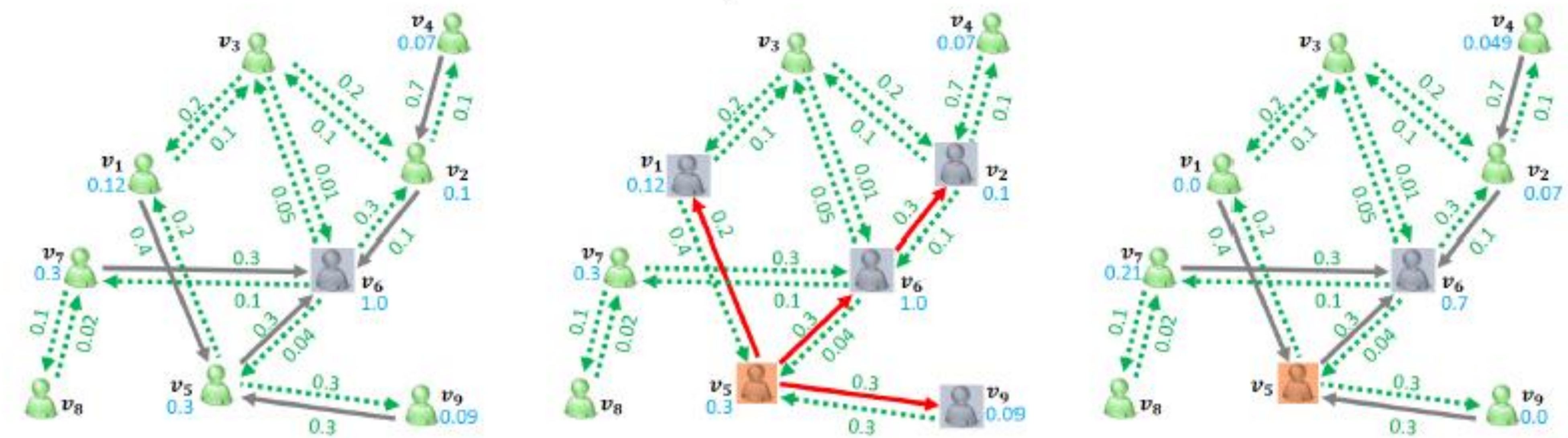


Summary: features of Maximum Influence Arborescence (MIA) heuristic

- Based on greedy approach
- Localize computation
- Use local tree structure
  - easy to compute
- linear batch update on marginal influence spread



An example of MIA run



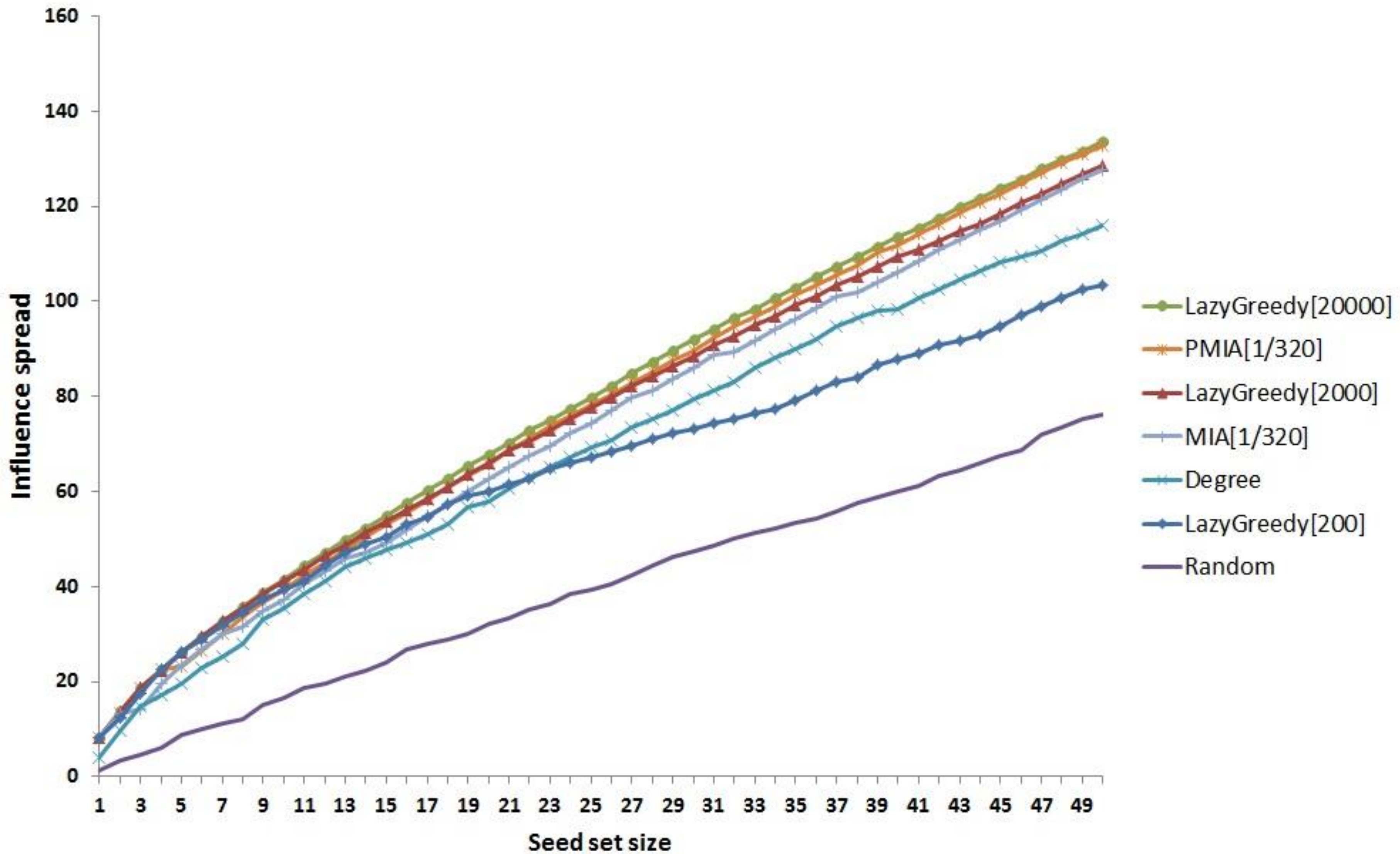
(a) $MIIA(v_6, 0.05)$, and $IncInf(w, v_6)$ for all $w \in MIIA(v_6, 0.05)$.

(b) The set of gray nodes is $InfSet(v_5) \setminus \{v_5\}$.

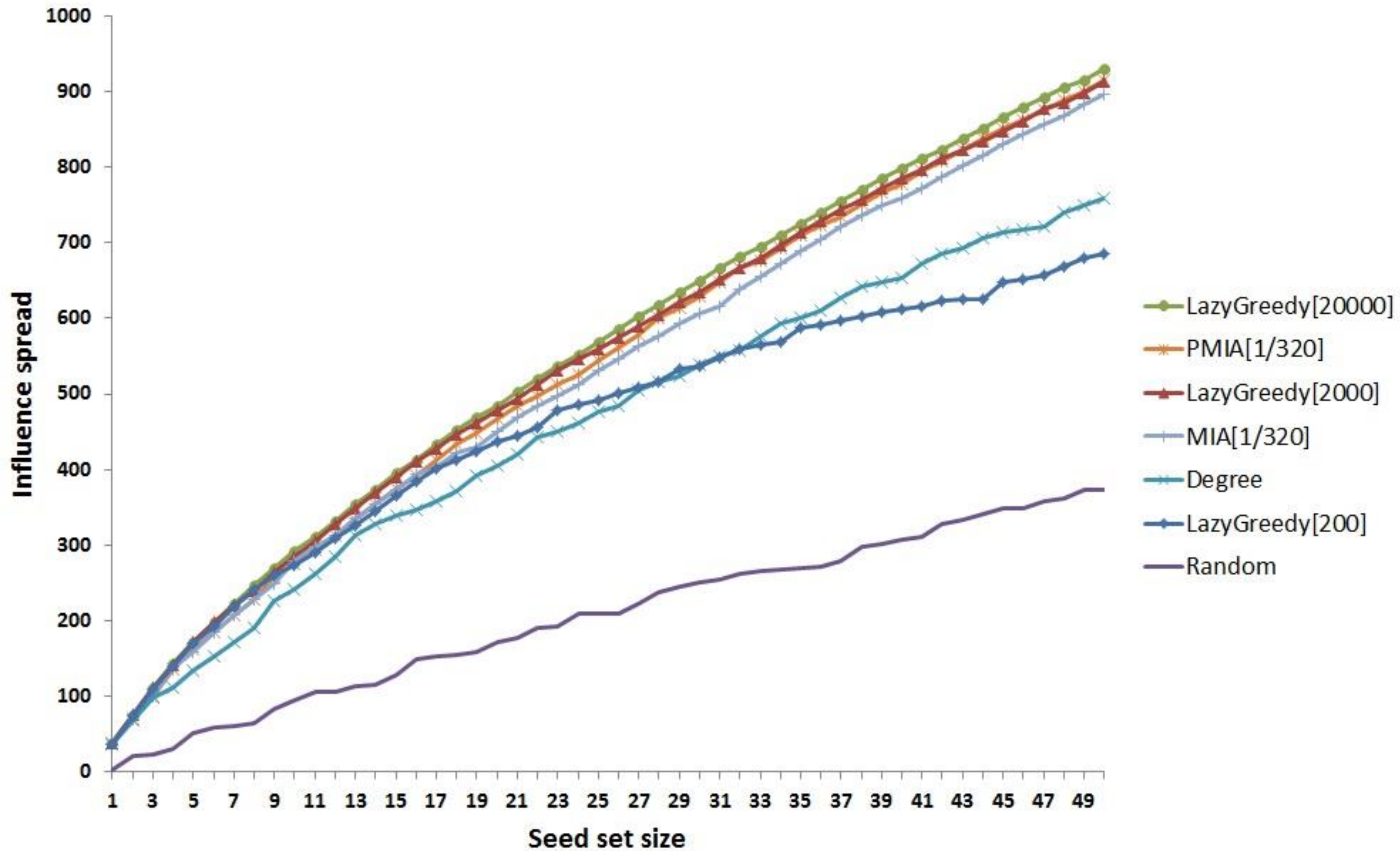
(c) Updated $IncInf(w, v_6)$ for all $w \in MIIA(v_6, 0.05)$, after v_5 is selected as a seed.

Figure 3.6: An example of computation of MIA algorithm. The blue number under a node w is $IncInf(w, v_6)$.

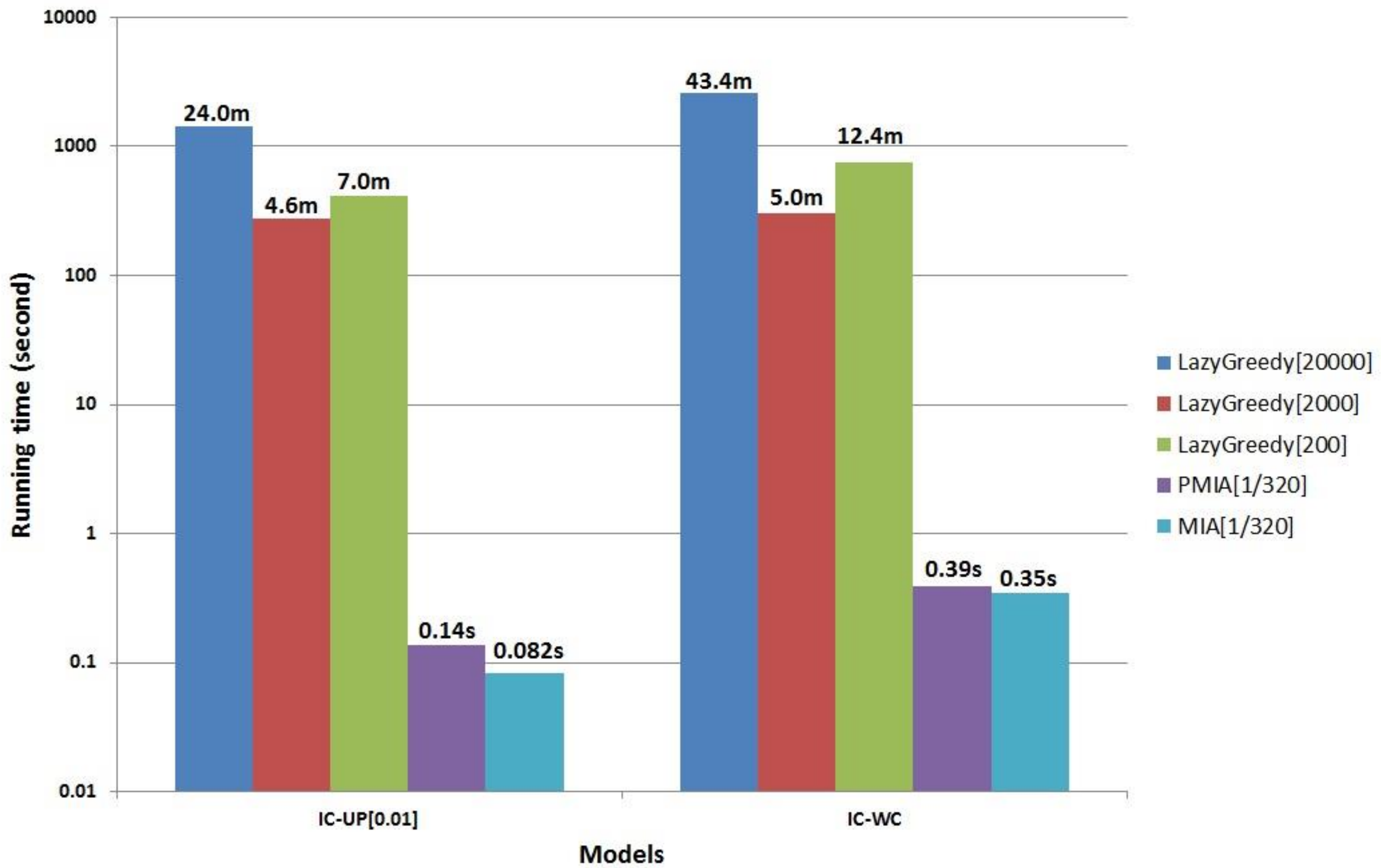
Influence spread in IC-UP[0.01] model



Influence spread in IC-WC model



Running time comparison



Experimental result summary

- MIA heuristic achieves almost the same influence spread as the greedy algorithm
- MIA heuristic is 3 orders of magnitude faster than the greedy algorithm
- MIA can scale to large graphs with millions of nodes and edges

Summary

- Scalable influence maximization algorithms
  - MixedGreedy and DegreeDiscount [KDD'09]
  - PMIA for the IC model [KDD'10]
  - LDAG for the LT model [ICDM'10]
  - IRIE for the IC model [ICDM'12]: further savings on time and space
  - MIA-M for IC-M model [AAAI'12]: include time delay and maximization within a short deadline
- PMIA/LDAG have become state-of-the-art benchmark algorithms for influence maximization
- Many followup work further improves the performance

Multi-item / Competitive Influence diffusion

Motivations

- Multiple items (ideas, information, opinions, product adoptions) are being propagated in the social network
- Items often have competing nature
  - One user adopted iPhone will not likely to adopt another Android phone
- How to model multi-item diffusion?
- What are the optimization problems in multi-item diffusion? And how to do them?

Terminologies

- Consider two item diffusion: positive opinion and negative opinion
- Each node v has three states: *inactive, positive,* and *negative* (positive and negative are both *active*)
  - Progressive model: once active, do not change state
- $S_t^+ (S_t^-)$: set of positive (negative) nodes at time t
  - $S_0^+ (S_0^-)$: *positive (negative) seed set*, $S_0^+ \cap S_0^- = \emptyset$ (can be relaxed)

Competitive independent cascade (CIC) model

- Positive/negative influence probabilities $p^+(u,v)/p^-(u,v)$
- At every step t, a newly activated u makes an attempt to active each of its inactive out-neighbor v
  - $A_t^+(v)/A_t^-(v)$: positive/negative successful attempt set
    - $u \in A_t^+(v)$ if u is positive and u's attempt of activating v at time t (with independent probability $p^+(u,v)$) is successful
    - $u \in A_t^-(v)$ if u is negative and u's attempt of activating v at time t (with independent probability $p^-(u,v)$) is successful
  - If $A_t^+(v) \neq \emptyset \wedge A_t^-(v) = \emptyset$: $v \in S_t^+$
  - If $A_t^-(v) \neq \emptyset \wedge A_t^+(v) = \emptyset$: $v \in S_t^-$
  - If $A_t^+(v) \neq \emptyset \wedge A_t^-(v) \neq \emptyset$: tie-breaking rule

Tie-breaking rule

- Applied when both positive and negative in-neighbors of v have successful activation attempts at the same step
- Fixed-probability tie-breaking rule TB-FP(ϕ): v is positive with probability ϕ, and negative with probability $1-\phi$.
  - TB-FP(1)/ TB-FP(0): positive/negative dominance
- Proportional probability tie-breaking rule TB-PP: v is positive with probability $\frac{|A_t^+(v)|}{|A_t^+(v)|+|A_t^-(v)|}$, negative with probability $\frac{|A_t^-(v)|}{|A_t^+(v)|+|A_t^-(v)|}$.

Equivalent tie-breaking rule to TB-PP

- Randomly permute all of v's in-neighbors (an priority ordering)
- When need a tie-breaking, check the priority order, the node $u \in A_t^+(v) \cup A_t^-(v)$ that is order first wins, and v takes the state of u.

Competitive linear threshold (CLT) model

- Positive/negative influence weights $w^+(u,v)$/ $w^-(u,v)$
- Initially, each node v selects a positive threshold θ_v^+ and a negative threshold θ_v^- independently from $[0,1]$
- At each step, first propagate positive influence and negative influence separately, using respective weights and threshold
  - If both successful, use fixed probability tie-breaking rule

Summary of competitive diffusion models

- Extensions of single-item diffusion models
- Each item diffusion follows single-item diffusion rules
- Each node only adopts one state
  - First adoption wins
  - Tie-breaking rule is used for simultaneous activation
- Other variants are possible

Influence maximization for a competitive diffusion model

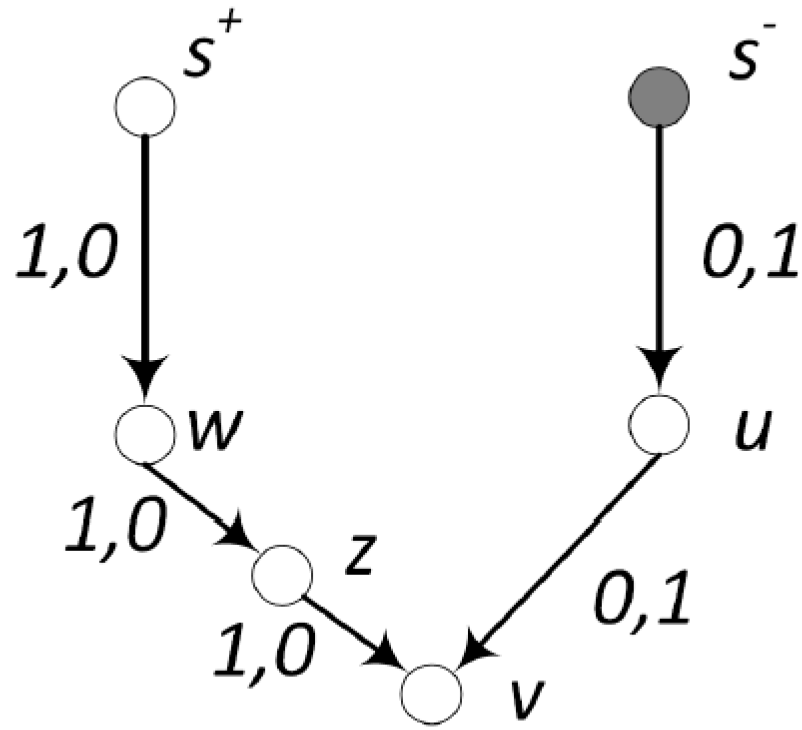
Problem 4.6 Influence maximization under a competitive diffusion model Given a social graph G, a competitive diffusion model on G for positive and negative opinions, a negative seed set S_0^-, and an integer k, the *influence maximization* problem under this competitive diffusion model is to find a positive seed set $S_0^+ \subseteq V \setminus S_0^-$ with at most k seeds, such that the positive influence spread of S_0^+ given negative seeds S_0^-, $\sigma^+(S_0^+, S_0^-)$, is maximized. That is, compute set $S_0^{+*} \subseteq V \setminus S_0^-$ such that

$$S_0^{+*} = \operatorname*{argmax}_{S_0^+ \subseteq V \setminus S_0^-, |S_0^+| = k} \sigma^+(S_0^+, S_0^-).$$

- When $S_0^- = \emptyset$, reduced to the original problem
- Thus, still NP hard for CIC and CLT models
- $\sigma^+(\cdot, S_0^-)$ is monotone for CIC and CLT

Submodularity of $\sigma^+(\cdot, S_0^-)$

- $\sigma^+(\cdot, S_0^-)$ is not submodular for general CIC and CLT models
- s^- is the negative seed
- $\emptyset, \{s^+\}, \{u\}, \{s^+, u\}$ are positive seed sets
- Key: the blocking effect of u

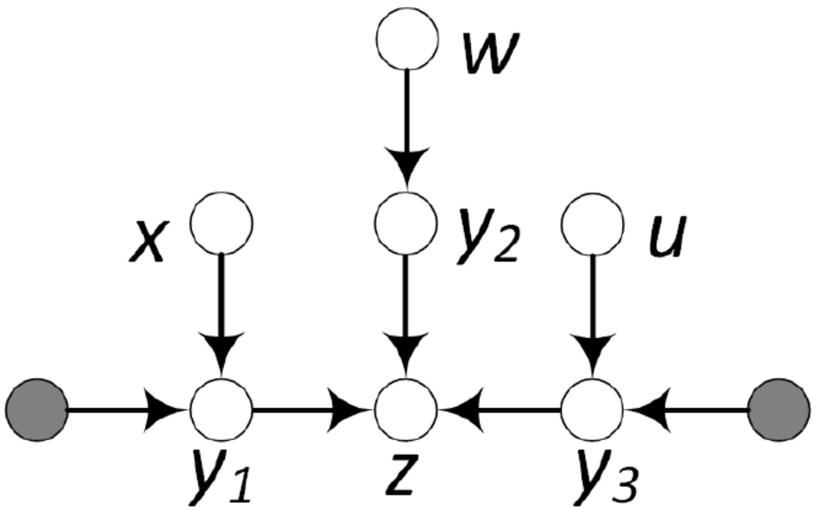


Homogeneous CIC model

- $p^+(u,v) = p^-(u,v)$ for all $(u,v) \in E$
- In homogeneous CIC model with positive dominance or negative dominance or proportional probability tie-breaking rule, $\sigma^+(\cdot, S_0^-)$ is submodular.
  - Use live-arc graph model
  - Each edge is sampled once, since only one item propagates through each edge
  - For positive/negative dominance rule, use distance argument
  - For TB-PP, pre-determine the priority order
    - Proof more complicated

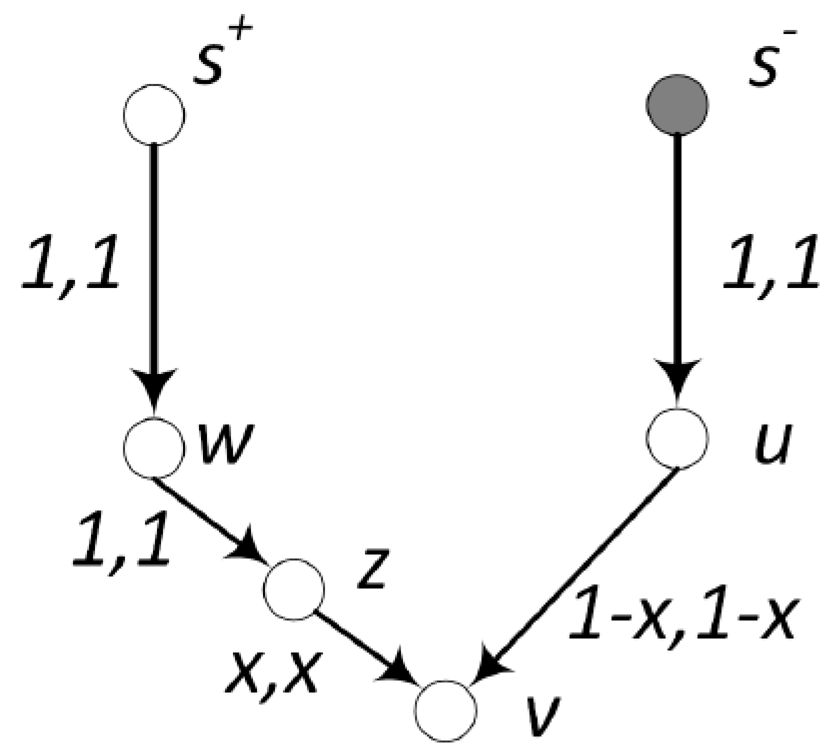
Homogeneous CIC with TB-FP(ϕ), $0 < \phi < 1$

- Not submodular
- Gray nodes are negative seeds
- $\{w\}, \{w, x\}, \{w, u\}$, $\{w, x, u\}$ are positive seed sets
- Same example shows that if nodes have difference dominance rules, then not submodular



Homogeneous CLT model

- Not submodular



Influence blocking maximization

- New objective function --- negative influence reduction:
  - $\rho^-(S_0^+, S_0^-) = \sigma^-(\emptyset, S_0^-) - \sigma^-(S_0^+, S_0^-)$

Problem 4.12 Influence-blocking maximization under a competitive diffusion model Given a social graph G, a competitive diffusion model on G for positive and negative opinions, a negative seed set S_0^-, and an integer k, the *influence-blocking maximization* problem under this competitive diffusion model is to find a positive seed set $S_0^+ \subseteq V \setminus S_0^-$ with at most k seeds, such that the negative influence reduction of S_0^+ given negative seeds S_0^-, $\rho^-(S_0^+, S_0^-)$, is maximized. That is, compute set $S_0^{+*} \subseteq V \setminus S_0^-$ such that

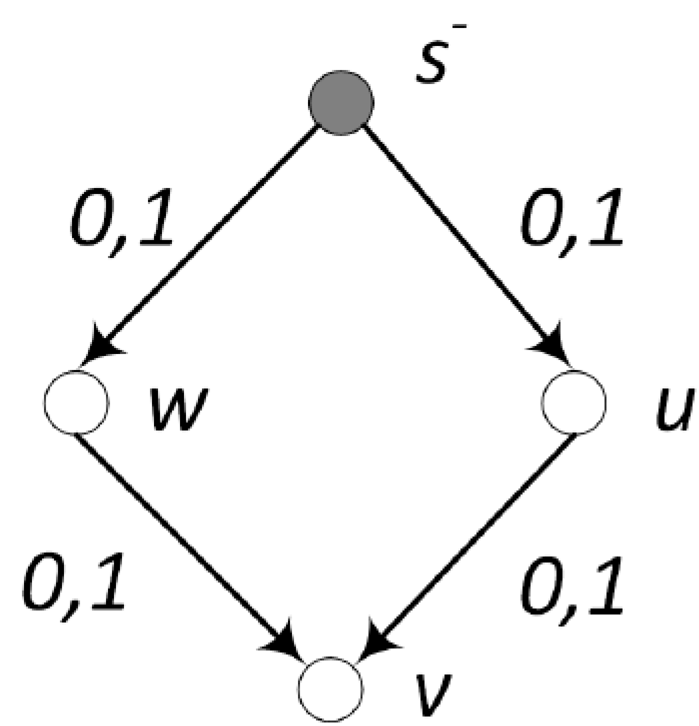
$$S_0^{+*} = \underset{S_0^+ \subseteq V \setminus S_0^-, |S_0^+| = k}{\operatorname{argmax}} \rho^-(S_0^+, S_0^-).$$

Motivation of influence blocking maximization

- Stop rumor spreading
- Immunization
  - Special case: positive seeds (nodes getting vaccination) do not spread positive influence

Solving IBM problem

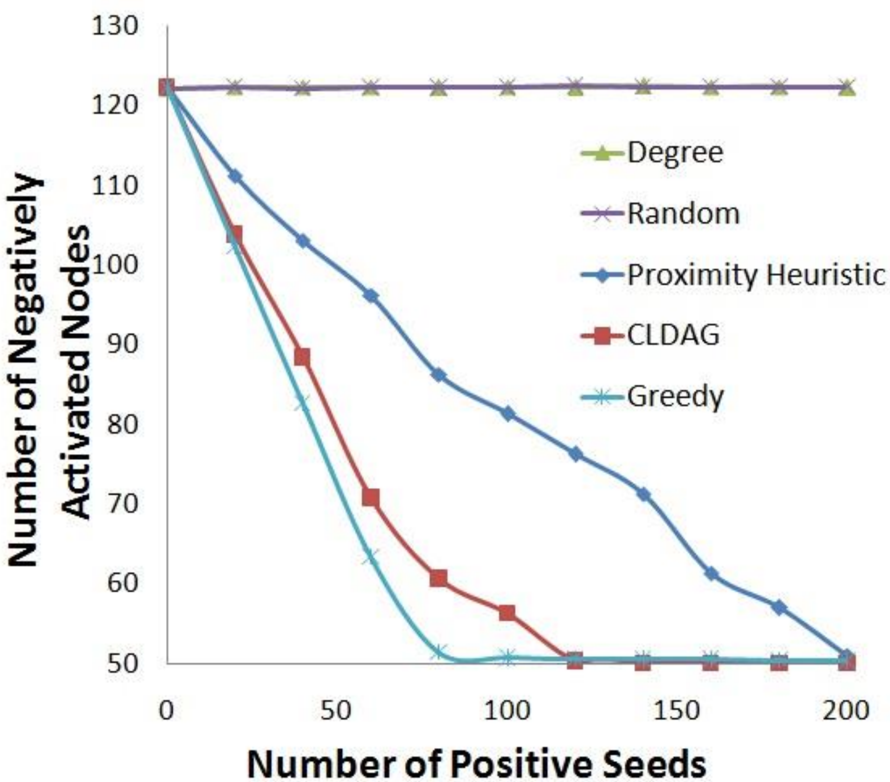
- IBM is NP-hard in both CIC and CLT models
- Negative influence reduction $\rho^{-}(\cdot, S_0^{-})$ is monotone submodular in CLT models, and homogeneous CIC models with TB-FP(0), TB-FP(1), or TB-PP rules.
- Non-homogeneous CIC is not submodular (right example)
  - Key blocking effect
- Homogeneous CIC with TB-FP(ϕ), $0 < \phi < 1$, is not submodular

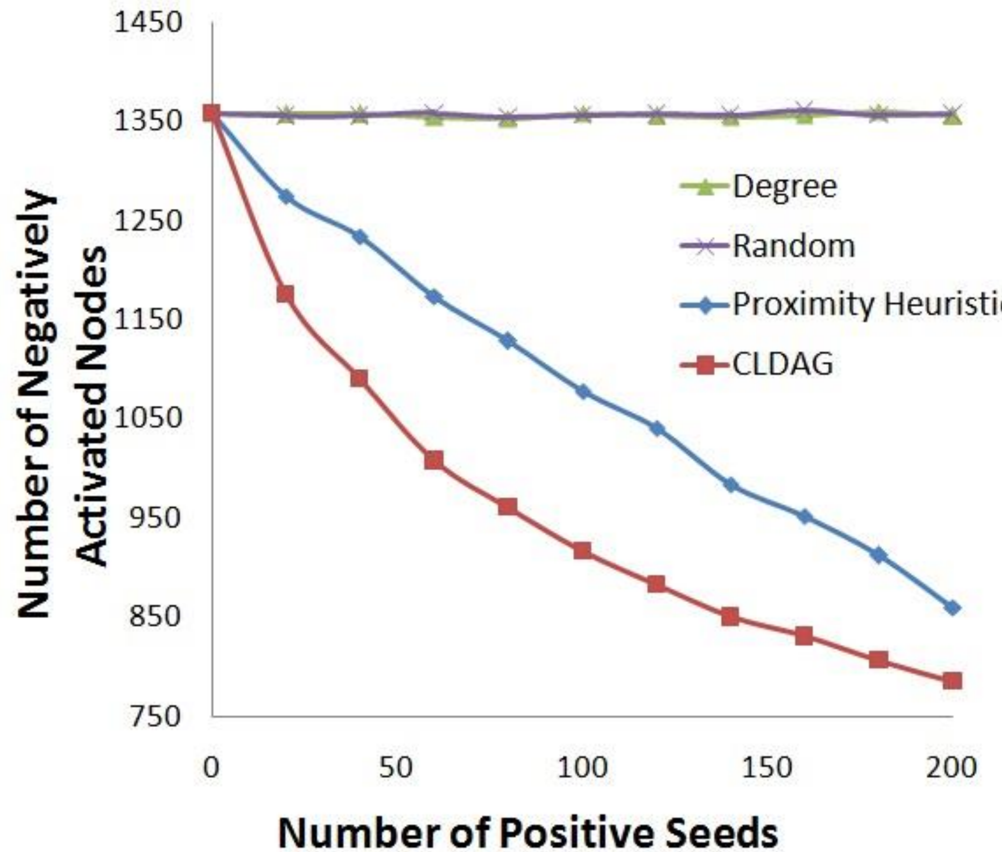


IBM in CLT model [He, Song, C., Jiang 2012]

- Negative influence reduction is submodular
- Allows greedy approximation algorithm
- Fast heuristic CLDAG:
  - reduce influence computation on local DAGs
  - use dynamic programming for LDAG computations

Performance of the CLDAG

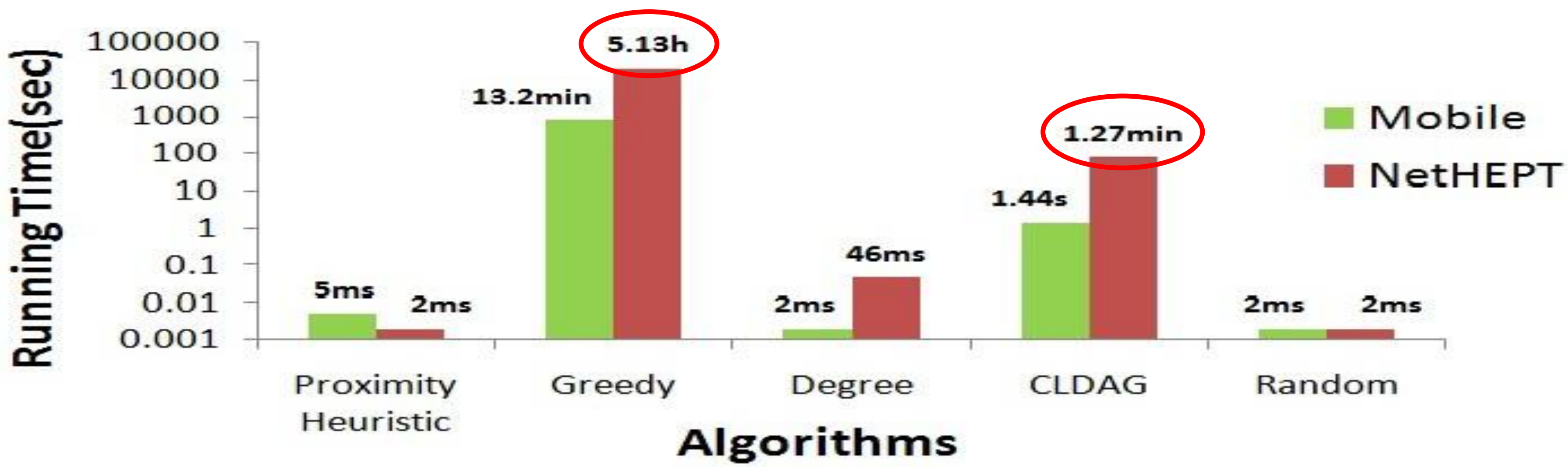




- with Greedy algorithm
- 1000 node sampled from a mobile network dataset
- 50 negative seeds with max degrees

- without Greedy algorithm
- 15K node NetHEPT, collaboration network in arxiv
- 50 negative seeds with max degrees

Scalability—Real dataset



Scalability Result for subgraph with greedy algorithm

Other studies on multi-item diffusion

- Endogenous competition: bad opinions about a product due to product defect competes with positive opinions [C., et al., 2011]
- Influence diffusion in networks with positive and negative relationships [Li, C., Wang, Zhang, 2013]
- Participation maximization: seed allocation of multiple diffusions maximizing total influence [Sun, et al., 2012]
- Fair seed allocation: seed allocation to guarantee fairness in influence [Lu, Bonchi, Goyal, Lakshmanan, 2012]
- From competition to complementarity [Lu, C., Lakshmanan, 2016]
- Etc.

Summary on multi-item diffusion

- Multi-item diffusion models often need to accommodate competitions
- Submodularity may no longer hold
  - Model dependent
  - Whether collective behavior is greater than the sum of its parts
- More models need to be considered
- Need data validation

Influence Model Learning

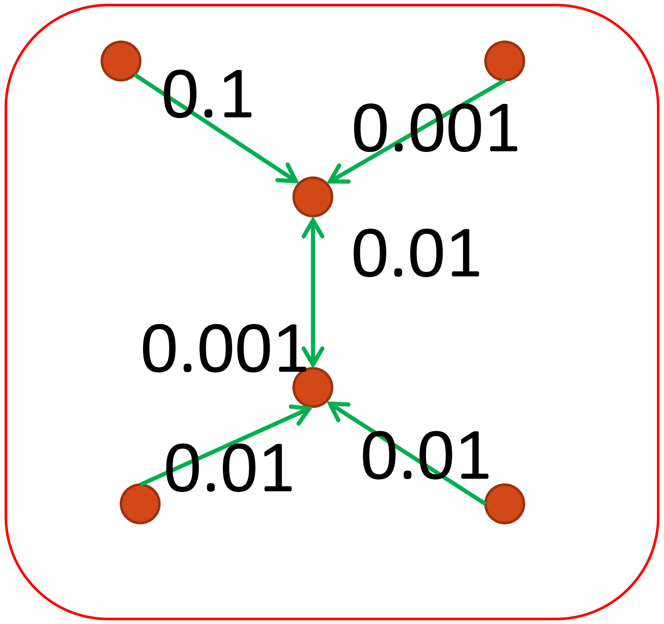
Where do the numbers come from?

Learning influence models

- Where do **influence probabilities** come from?
  - Real world social networks don't have probabilities!
  - Can we learn the probabilities from action logs?
  - Sometimes we don't even know the social network
  - Can we learn the social network, too?

Where do the weights come from?

- Influence Maximization – Gen 0: academic collaboration networks (real) with weights assigned arbitrarily using some models:
  - Trivalency: weights chosen uniformly at random from {0.1, 0.01, 0.001}.

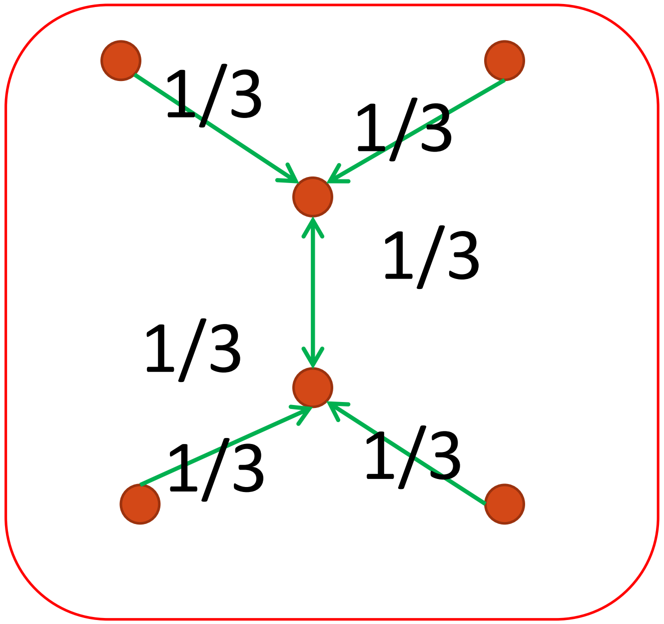


Where do the weights come from?

- Influence Maximization – Gen 0: academic collaboration networks (real) with weights assigned arbitrarily using some models:
  - Weighted Cascade: $w_{uv} = \frac{1}{d_v^{in}}$.

Other variants: uniform (constant),
WC with parallel edges.

Weight assignment not backed by real data. ☹



Inference problems

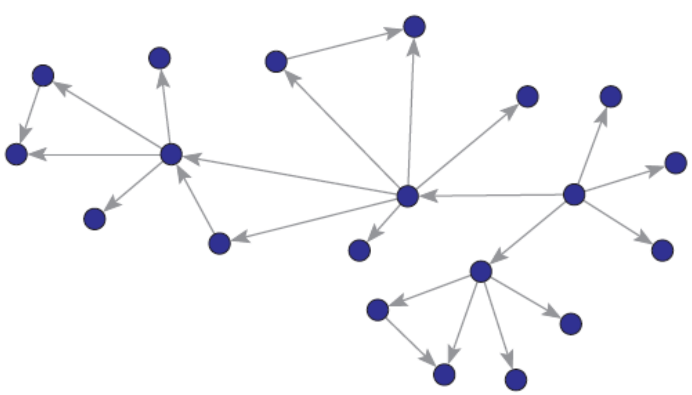
- Given a log $A = \{\langle u_1, a_1, t_1 \rangle, \ldots\}$
- P1. Social network not given
  - Infer network and edge weights
- P2. Social network given
  - Infer edge weights
- P3. Social network and attribution given
  - Explicit "trackbacks" to parent user

$$A = \{\langle u_1, a_1, t_1, p_1 \rangle, \ldots\}$$

  - Simple counting

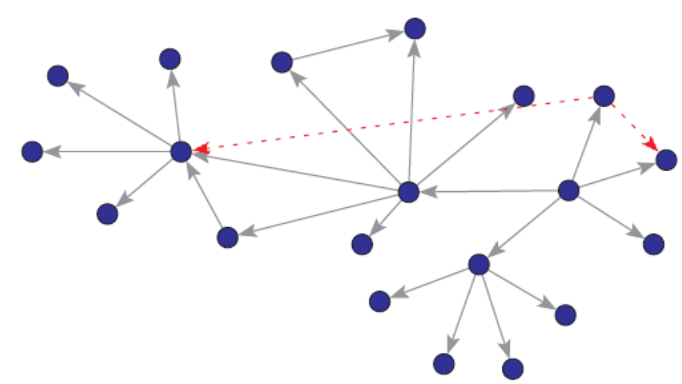
P1. Social network not given

- Observe activation times, assume probability of a successful activation decays (e.g., exponentially) with time



Actual network

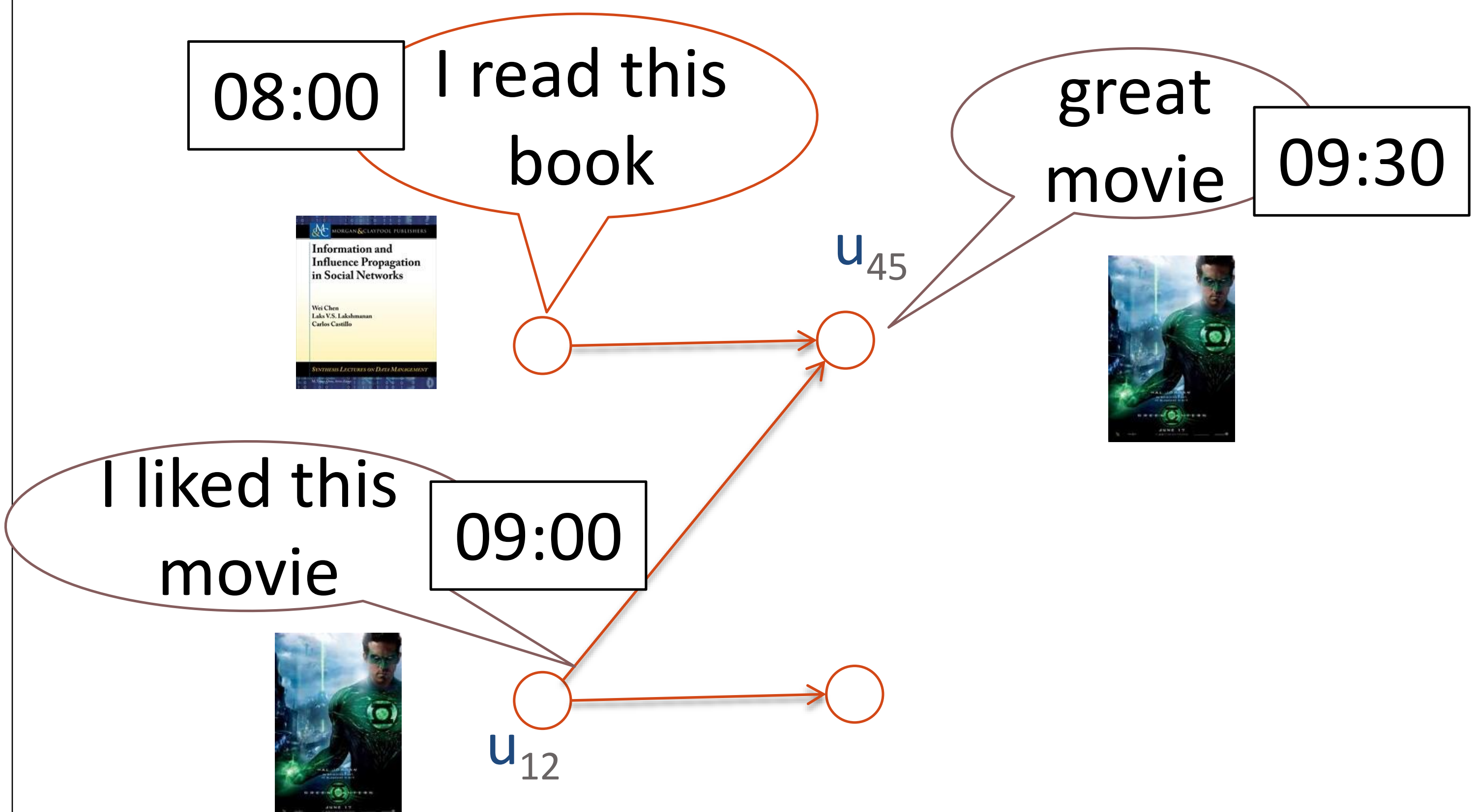
$$\langle u_1, a_1, t_1 \rangle,$$
$$\langle u_2, a_2, t_2 \rangle,$$
$$\langle u_3, a_3, t_3 \rangle,$$
$$\langle u_4, a_4, t_4 \rangle,$$
$$\ldots$$



Learned network

P2. Social network given

Input data: (1) social graph and (2) action log of past propagations



Action	Node	Time
a	u_{12}	1
a	u_{45}	2
a	u_{32}	3
a	u_{76}	8
b	u_{32}	1
b	u_{45}	3
b	u_{98}	7

u_{45} follows u_{12}

P2. Social network given

- D(0), D(1), … → D(t) nodes that acted at time t.
- $C(t) = \bigcup_{\tau \leq t} D(\tau)$. → cumulative.
- $P_w(t+1) = 1 - \Pi_{v \in N^{in}(w) \cap D(t)} (1 - \kappa_{vw})$.
- Find $\theta = \{\kappa_{vw}\}$ that maximizes likelihood

$$L(\theta; D) = \left(\Pi_{t=0}^{T-1} \Pi_{w \in D(t+1)} P_w(t+1)\right) - \quad \leftarrow \text{success}$$

$$\left(\Pi_{t=0}^{T-1} \Pi_{v \in D(t)} \Pi_{w \in N^{out}(v) \setminus C(t+1)} (1 - \kappa_{vw})\right) \quad \leftarrow \text{failure}$$

Very expensive (not scalable)

Assumes influence weights remain constant over time

Summary on model learning

- Other more efficient learning methods available
- Data sparsity is a big problem
  - By clustering?
- Influence propagation is topic-aware
- How to validate data analysis with real-world influence?

Conclusion

Ongoing and future research directions

- Model validation and influence analysis from real data
- Online and adaptive algorithms
- Game theoretic settings for competitive diffusion
- Incentives for information / influence diffusions
- Influence maximization with non-submodular objective functions

Grand challenge

- Understand from data the true peer influence and viral diffusion scenarios, online and offline
- Apply social influence research to explain, predict, and control influence and viral phenomena
- Network and diffusion dynamics would be focus of network science in the next decade

Thanks and Questions?