Mining Propagation Data
(in Social Networks)

Francesco Bonchi
Yahoo! Research Barcelona

http://francescobonchi.com/

Acknowledgments

Amit Goyal (Twitter)

Laks V.S. Lakshmanan (University of British Columbia, Vancouver, Canada)

Michael Mathioudakis (University of Toronto, Canada)

Giuseppe Manco (University of Calabria, Italy)

Aris Gionis (Aalto University, Finland)

Antti Ukkonen (Aalto University, Finland)

Carlos Castillo (QCRI, Doha, Qatar)

Tamir Tassa (The Open University of Israel)

Konstantin Kutzkov (IT University of Copenhagen, Denmark)

YAHOO!

The Web Mining Research group @Yahoo! Research Barcelona

We're hiring!

Post-doc or (Senior) Research Scientist positions available.

Summer Internship (application deadline Jan.15th)

intern-yrbcn@yahoo-inc.com

Overview

Background

Social influence

WOMM, Viral marketing

Influence maximization

Prior art

Propagation data

The global picture for influence maximization

Learning influence strength from propagation data

Why it is important, Why it is complicated

Direct mining of propagation data for influence maximization

Other mining problems with propagation data

Influence-preserving network sparsification

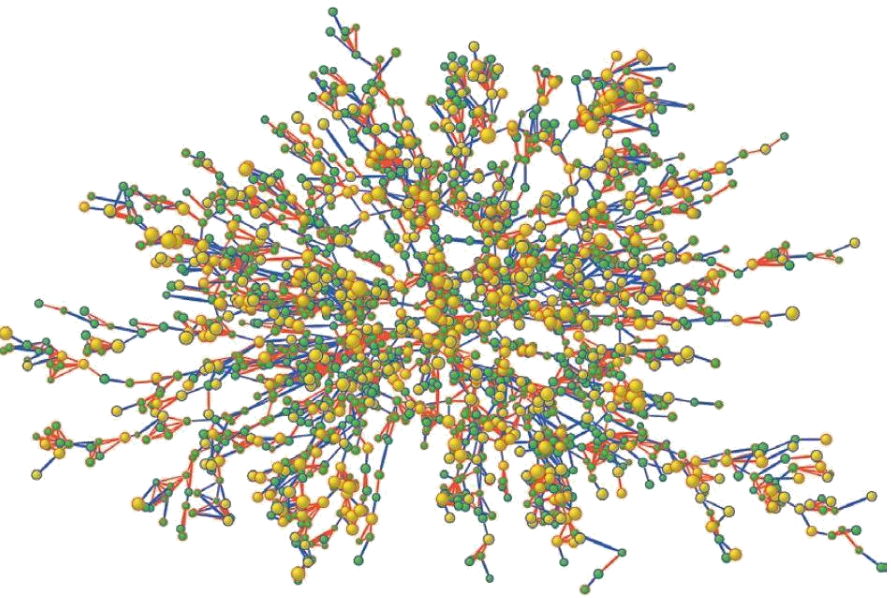
Cascade-based community detection

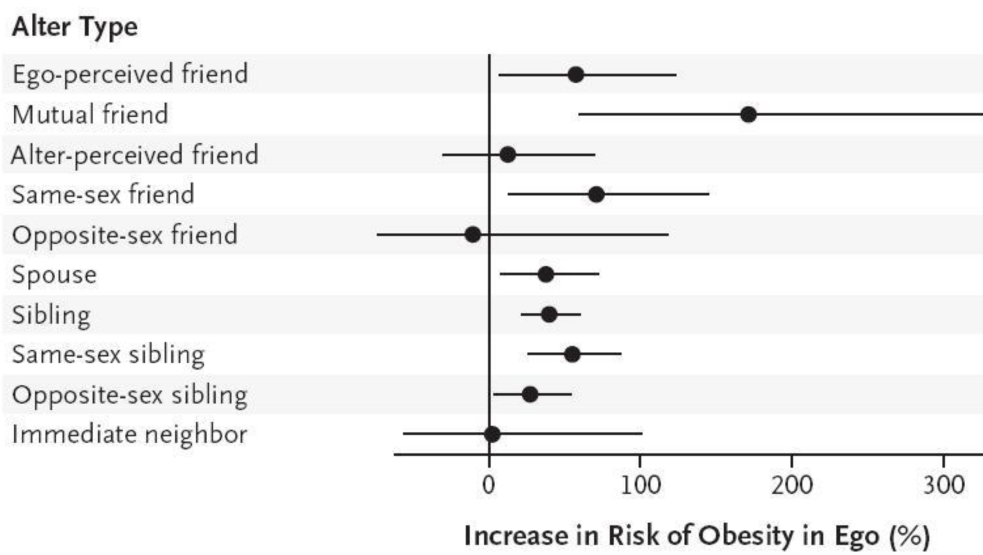
YAHOO!

The Spread of Obesity in a Large Social Network over 32 Years

Christakis and Fowler, New England Journal of Medicine, 2007

Data set: 12,067 people from 1971 to 2003, 50K links





Obese Friend → 57% increase in chances of obesity

Obese Sibling → 40% increase in chances of obesity

Obese Spouse → 37% increase in chances of obesity

YAHOO!

Influence or Homophily?

Homophily

tendency to stay together with people similar to you

"Birds of a feather flock together"

Social influence

a force that person A (i.e., the influencer) exerts on person B to introduce a change of the behavior and/or opinion of B

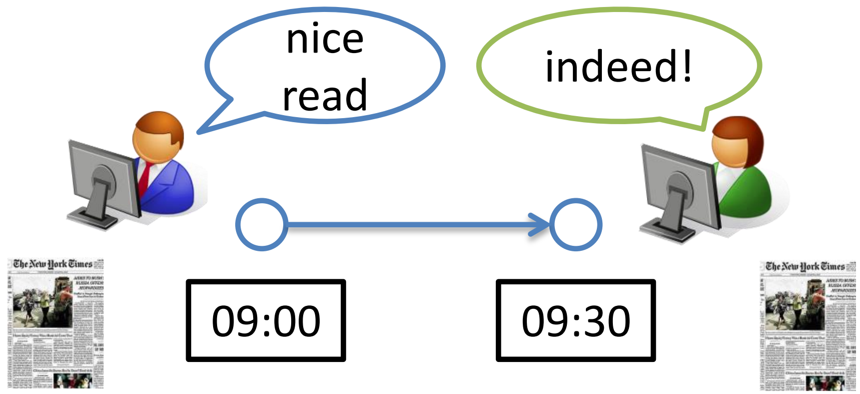
Influence is a causal process

Problem: How to distinguish social influence from homophily and other factors of correlation

Crandall et al. (KDD'08) *"Feedback Effects between Similarity and Social Influence in Online Communities"*
Anagnostopoulos et al. (KDD'08) *"Influence and correlation in social networks"*
Aral et al. (PNAS'09) *"Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks"*
Myers et al. (KDD'12) *"Information Diffusion and External Influence in Networks"*

YAHOO!

Influence-driven information propagation in on-line social networks



users perform actions

post messages, pictures, video

buy, comment, link, rate, share, like, retweet

users are connected with other users

interact, influence each other

actions propagate

YAHOO!

Opportunities

(science, society, technology and business)

studies and models of human interaction

innovation adoption, epidemics

social influence, homophily, interest, trust, referral

citizens engagement, awareness, law enforcement

citizens journalism, blogging and microblogging

outbreak detection, risk communication, coordination during emergencies

political campaigns

feed ranking, personalization, expert finding, “friends” recommendation

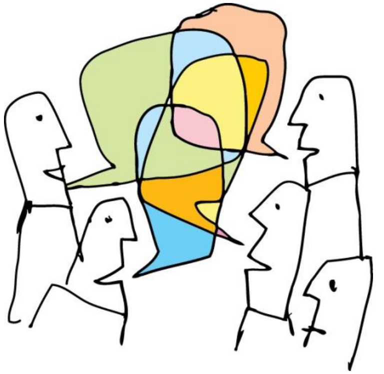
branding

behavioral targeting

WOMM, viral marketing

YAHOO!

Social Influence Marketing
Viral Marketing
WOMM



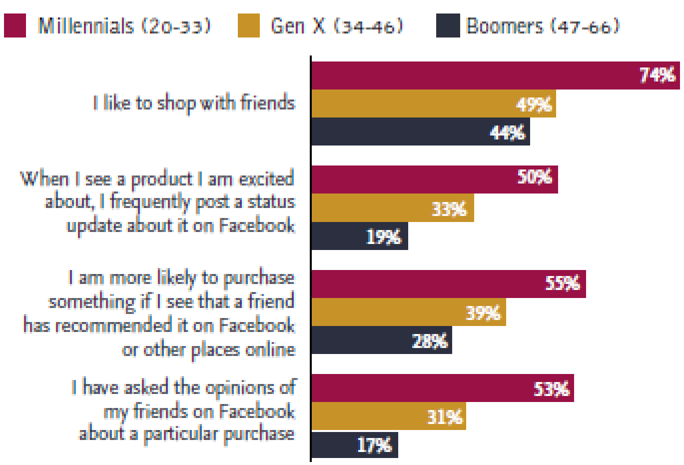
IDEA: exploit social influence for marketing

Basic assumption: word-of-mouth effect, thanks to which actions, opinions, buying behaviors, innovations and so on, propagate in a social network.

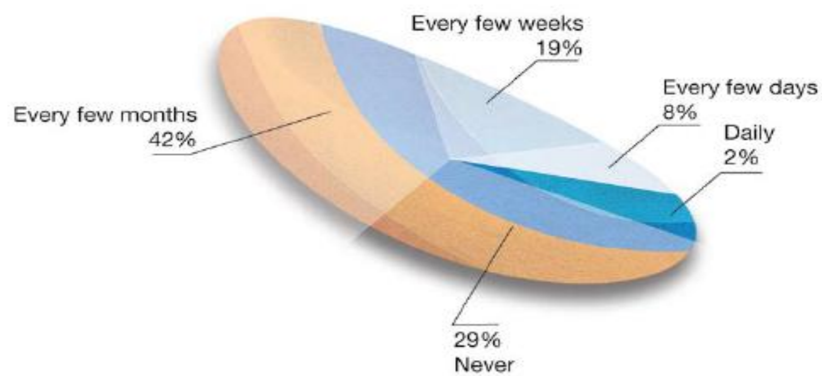
Target users who are likely to produce word-of-mouth diffusion, thus leading to additional reach, clicks, conversions, or brand awareness

Target the influencers

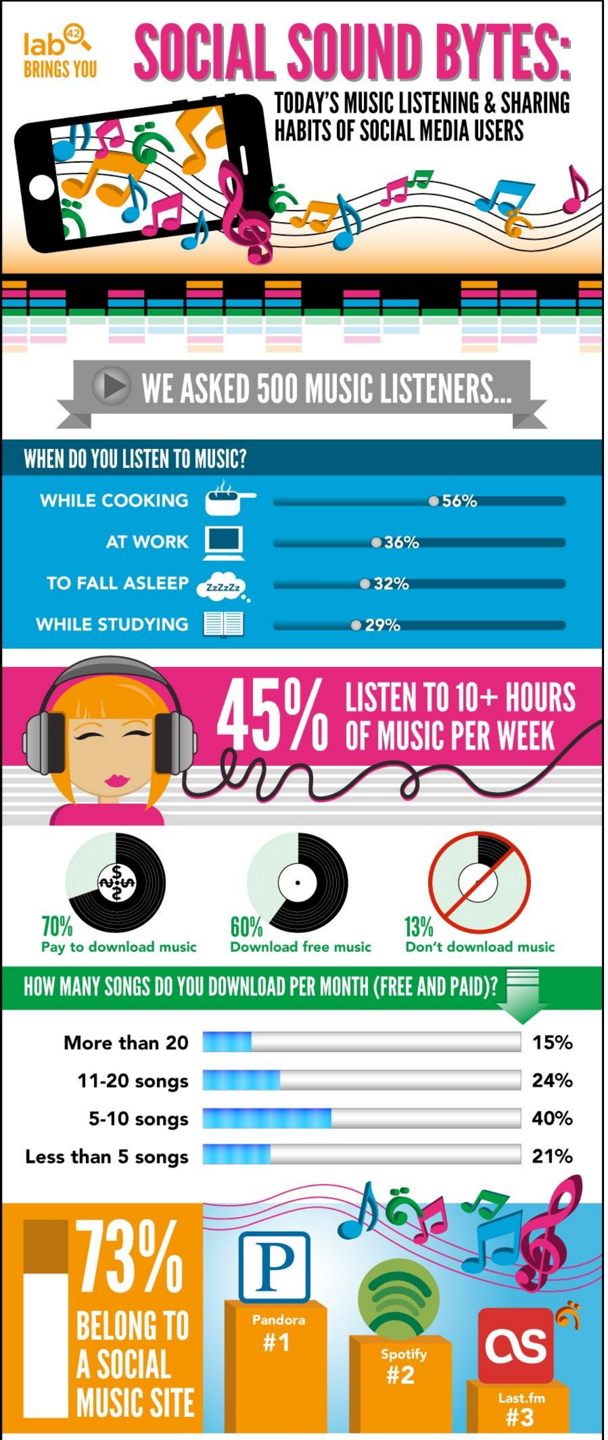
Sharing and social influence



How frequently do you share recommendations online?



YAHOO!



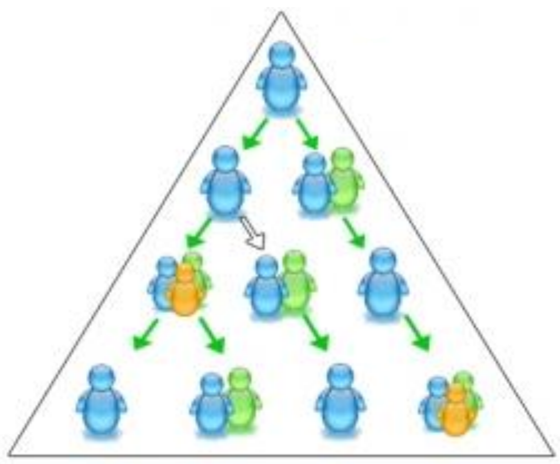
http://mashable.com/2012/08/12/social-music-listening/

YAHOO!

Viral Marketing and Influence Maximization

Business goal (Viral Marketing): exploit the "word-of-mouth" effect in a social network to achieve marketing objectives through self-replicating viral processes

Mining problem: find a seed-set of influential people such that by targeting them we maximize the spread of viral propagations



Hot topic in Data Mining research since 12 years:

Domingos and Richardson *"Mining the network value of customers"* (KDD'01)

Domingos and Richardson *"Mining knowledge-sharing sites for viral marketing"* (KDD'02)

Kempe et al. *"Maximizing the spread of influence through a social network"* (KDD'03)

YAHOO!

Influence Maximization Problem

following Kempe et al. (KDD'03) *"Maximizing the spread of influence through a social network"*

Given a propagation model *M,* define influence of node set *S,*
$\sigma_M(S)$ = expected size of propagation, if *S* is the initial set of active nodes

Problem: Given social network *G* with arcs probabilities/weights, budget k, find k-node set *S* that maximizes $\sigma_M(S)$

Two major propagation models considered:

independent cascade (IC) model

linear threshold (LT) model

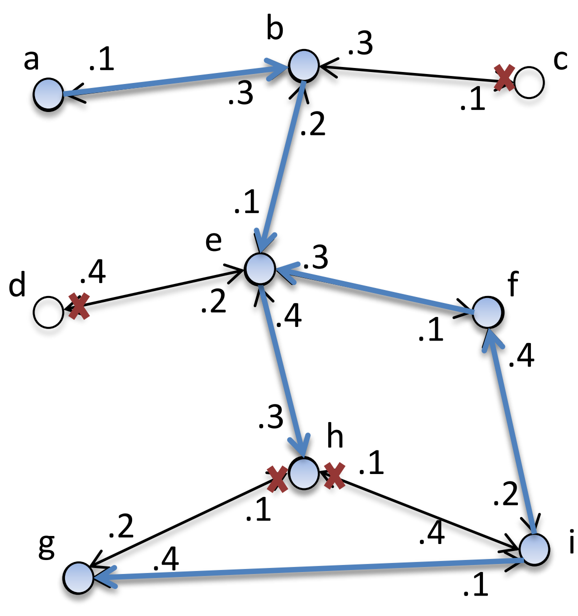
YAHOO!

Independent Cascade Model (IC)

Every arc *(u,v)* has associated the probability *p(u,v)* of *u* influencing *v*

Time proceeds in discrete steps

At time t, nodes that became active at t-1 try to activate their inactive neighbors, and succeed according to *p(u,v)*



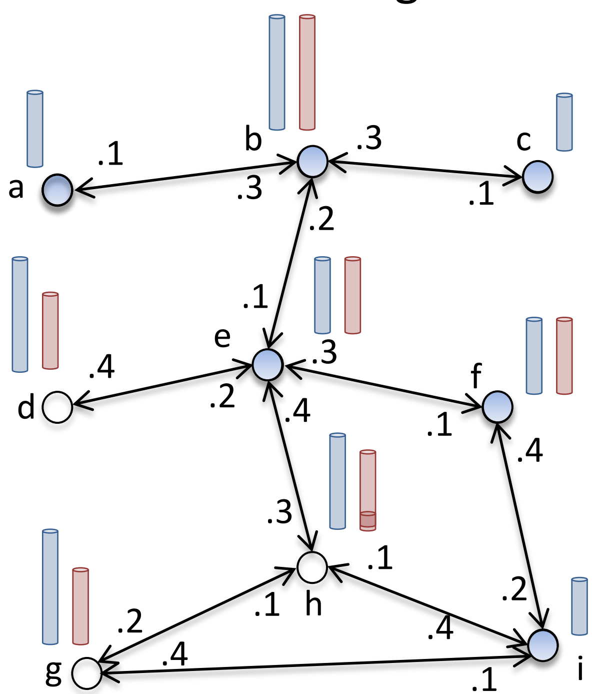
Linear Threshold Model (LT)

Every arc *(u,v)* has associated a weight *b(u,v)* such that the sum of incoming weights in each node is ≤ 1

Time proceeds in discrete steps

Each node *v* picks a random threshold $\vartheta_v \sim U[0,1]$

A node *v* becomes active when the sum of incoming weights from active neighbors reaches ϑ_v



Known Results

Bad news: NP-hard optimization problem for both IC and LT models

Good news: we can use Greedy algorithm

Algorithm 1 Greedy

Input: G, k, σ_m

Output: seed set S

1: $S \leftarrow \emptyset$
2: **while** $|S| < k$ **do**
3: select $u = \arg\max_{w \in V \setminus S}(\sigma_m(S \cup \{w\}) - \sigma_m(S))$
4: $S \leftarrow S \cup \{u\}$

$\sigma_M(S)$ is monotone and submodular

Theorem*: The resulting set S activates at least (1- 1/e) > 63% of the number of nodes that any size-k set could activate

Bad news: computing $\sigma_M(S)$ is #P-hard under both IC and LT models

step 3 of the Greedy Algorithm above can only be approximated by MC simulations

*Nemhauser et al. "An analysis of approximations for maximizing submodular set functions – (i)" (1978)

YAHOO!

Influence Maximization: prior art

Much work has been done following Kempe et al. mostly devoted to heuristichs to improve the efficiency of the Greedy algorithm:

E.g.,

Kimura and Saito (PKDD'06) *"Tractable models for information diffusion in social networks"*

Leskovec et al. (KDD'07) *"Cost-effective outbreak detection in networks"*

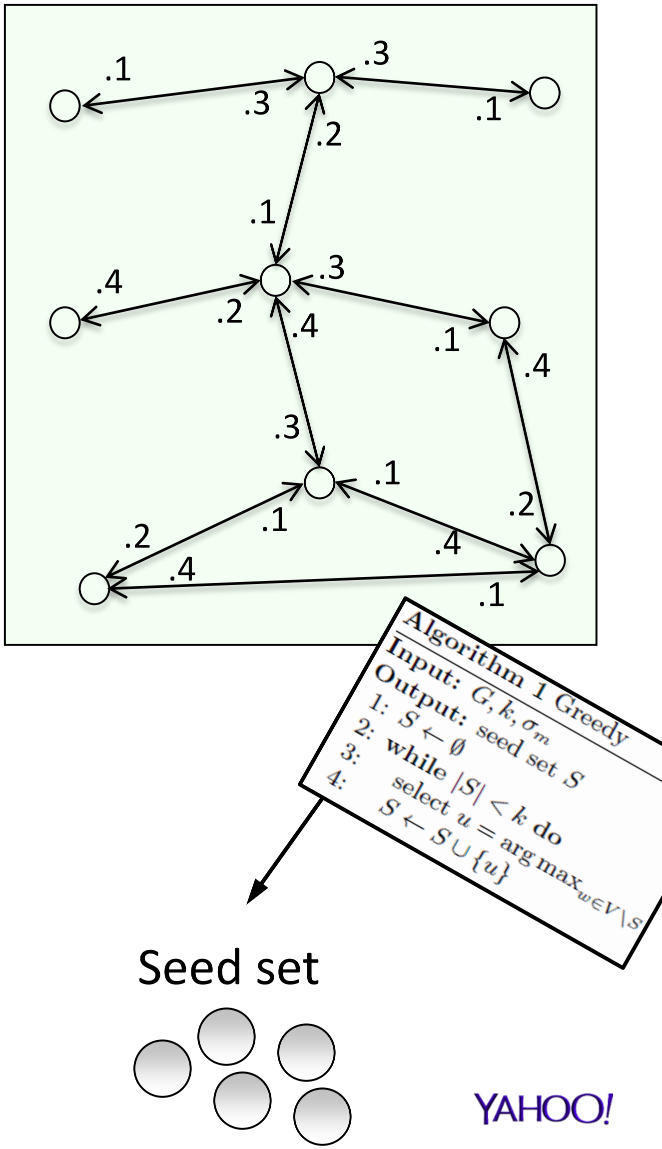
Chen et al. (KDD'09) *"Efficient influence maximization in social networks"*

Chen et al. (KDD'10) *"Scalable influence maximization for prevalent viral marketing in large-scale social networks"*

Chen et al. (ICDM'10) *"Scalable influence maximization in social networks under the linear threshold model"*

Goyal et al. (WWW'11) *"CELF++: optimizing the greedy algorithm for influence maximization in social networks"*

+ many more in 2011, 2012

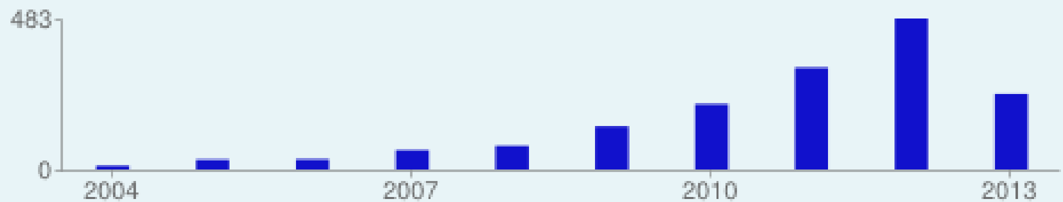


Problem: scalability of the Influence Maximization framework

Title	Maximizing the spread of influence through a social network
Authors	David Kempe, Jon Kleinberg, Éva Tardos
Publication date	2003/8/24
Conference name	Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining
Pages	137-146
Publisher	ACM
Description	Abstract Models for the processes by which ideas and influence propagate through a social network have been studied in a number of domains, including the diffusion of medical and technological innovations, the sudden and widespread adoption of various strategies in game-theoretic settings, and the effects of" word of mouth" in the promotion of new products. Recently, motivated by the design of viral marketing strategies, Domingos and Richardson posed a fundamental algorithmic problem for such social network processes: if we can try ...

Total citations: Cited by 1631

Citations per year



Scholar articles: Maximizing the spread of influence through a social network
D Kempe, J Kleinberg, É Tardos - Proceedings of the ninth ACM SIGKDD international ..., 2003
Cited by 1631 - Related articles - All 61 versions

YAHOO!

Information propagation data

Data! Data! Data!

We have 2 pieces of input data:
(1) social graph and (2) a log of past propagations

<u>Social graph $G = (V,E)$</u>

nodes are users

links represent social ties

can be explicit (i.e., declared friendship) or
implicit (e.g., derived on the basis of shared interests)

can be directed (e.g., I follow you) or undirected (e.g., we're friends)

when directed:

u_{45} is a follower of u_{12}

Data! Data! Data!

We have 2 pieces of input data:
(1) social graph and (2) a log of past propagations

Action	User	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

Propagation log

It's a relation *L(action,user,time)*

usual assumptions:

each user performs the same action only once

(if more than once, then we consider only the first occurrence)

the projection of L on the 2[nd] column is contained in V

Data! Data! Data!

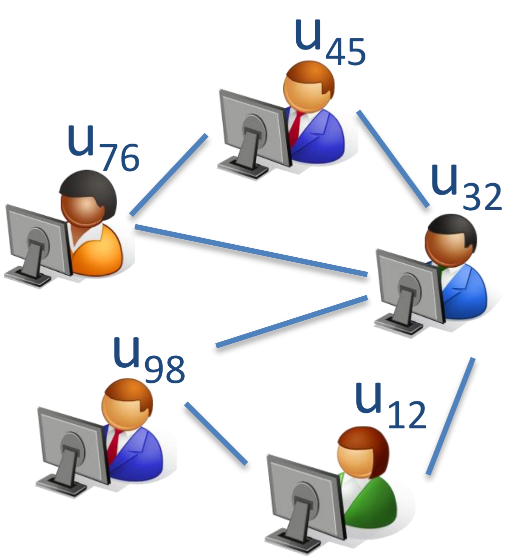
We have 2 pieces of input data:
(1) social graph and (2) a log of past propagations

Putting together (1) and (2) we can consider to have

a set of **DAGs**

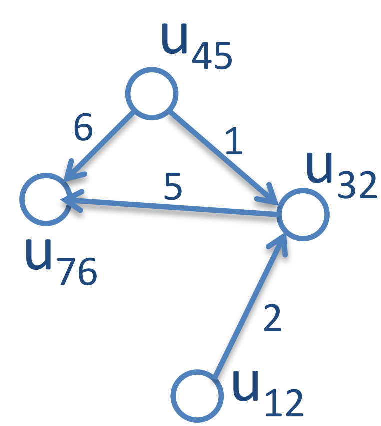
(sometimes a set of trees)

with arcs labeled with elapsed time between two actions



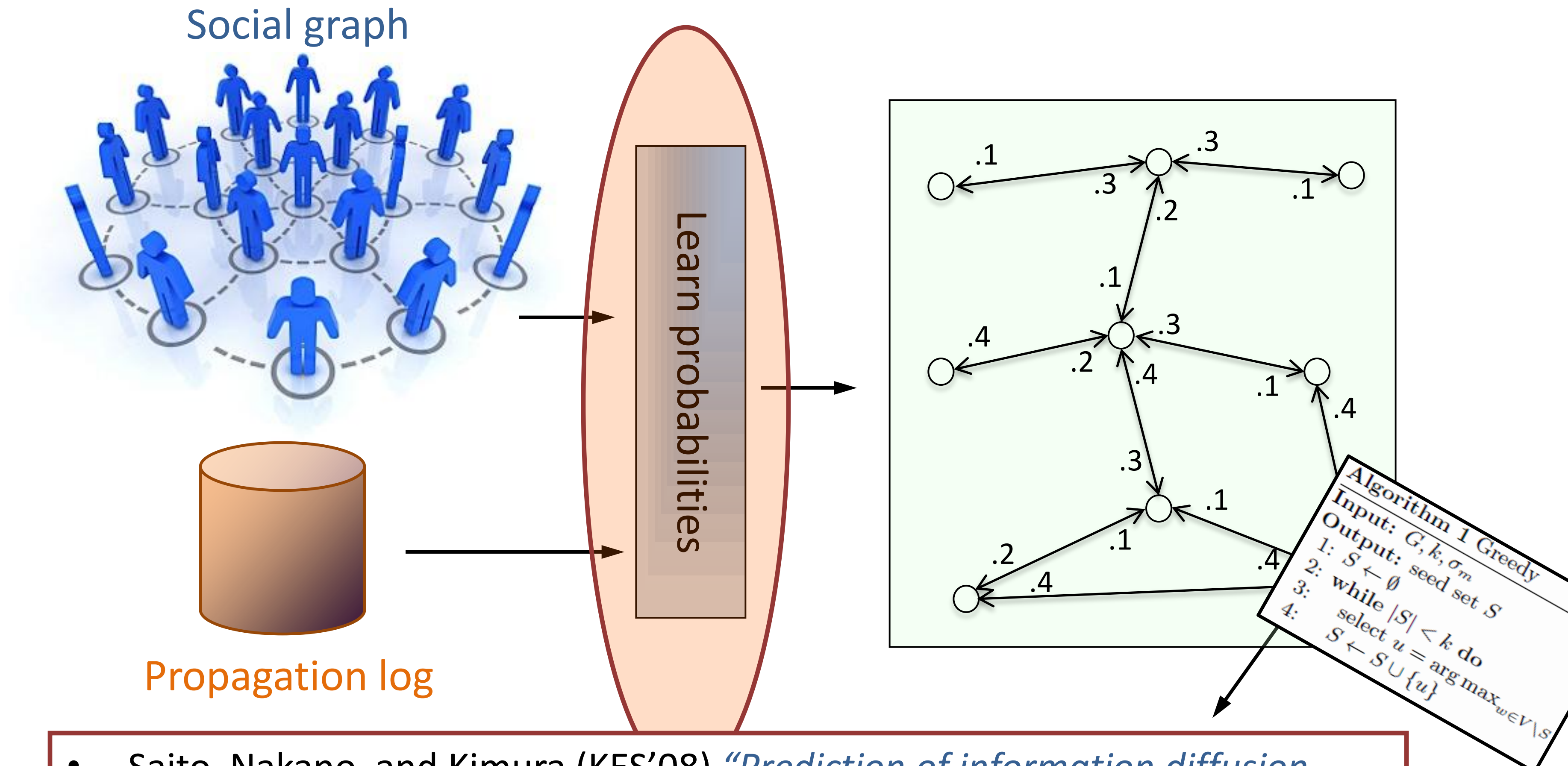
Action	User	Time
a	u_{12}	1
a	u_{45}	2
a	u_{32}	3
a	u_{76}	8
b	u_{32}	1
b	u_{45}	3
b	u_{98}	7

Action *a:*



YAHOO!

The global picture



- Saito, Nakano, and Kimura (KES'08) *"Prediction of information diffusion probabilities for independent cascade model"* $\rightarrow$ IC model
- Goyal, Bonchi, Lakshmanan (WSDM'10) *"Learning influence probabilities in social networks"* $\rightarrow$ General threshold model + **Time**
- Many more in 2010-2013

Learning influence strength from propagation data:

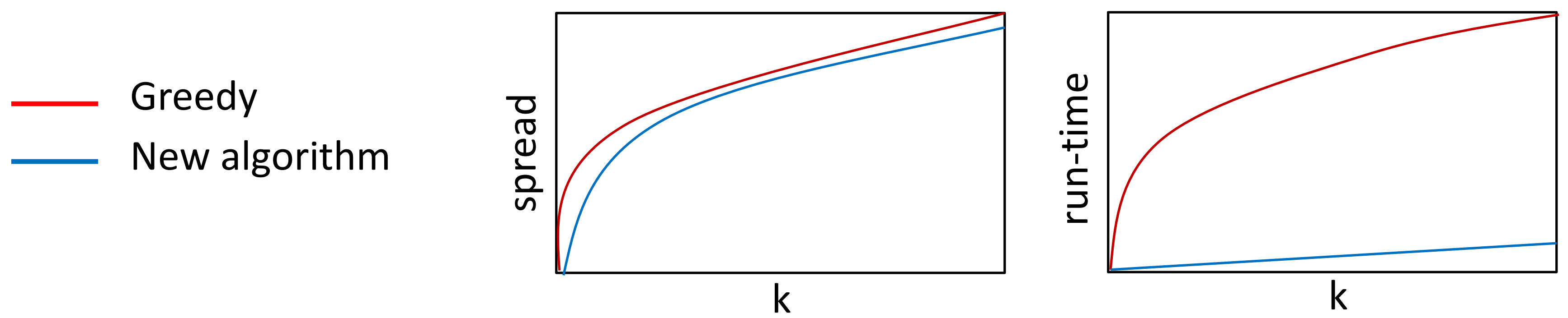
why it is important

Prior art typical experimental assessment

Assuming IC (or LT) model,

compare the influence spread achieved by seed sets selected by different algorithms

Spread computed by means of IC (or LT) propagation simulations (lack of ground truth!)



<u>Using simple methods of assigning probabilities:</u>

WC (weighted cascade) p(u,v) = 1/in_degree(v)

TV (trivalency) selected uniformly at random from the set *{0.1, 0.01, 0.001}*

UN (uniform) all edges have same probability (e.g. *p = 0.01)*

YAHOO!

Why learning from data matters – experiments

Goyal, Bonchi, Lakshmanan (VLDB'12)

- Methods compared (Greedy algorithm, IC model):
  - WC, TV, UN (no learning)
  - EM (learned from real data – Expectation Maximization method*)
  - PT (learned than perturbed ± 20%)
- Data:
  - 2 real-world datasets
  - social graph + propagation log

	Flixster Large	Flickr Large	Flixster Small	Flickr Small
$\#Nodes$	1M	1.32M	13K	14.8K
$\#Dir.\ Edges$	28M	81M	192.4K	1.17M
$Avg.degree$	28	61	14.8	79
$\#propagations$	49K	296K	25K	28.5K
$\#tuples$	8.2M	36M	1.84M	478K

  - On Flixster, we consider "rating a movie" as an action
  - On Flickr, we consider "joining a group" as an action
  - Split the data in training and test sets – 80:20
- Experiments:
  1. Seed sets intersection
  2. Given a seed set, we ask to the model to predict its spread (ground truth on the test set)

*Saito et al. (KES'08) *"Prediction of information diffusion probabilities for independent cascade model"*

YAHOO!

Why learning from data matters – experiments

1. <u>Seed sets intersection (*k = 50*)</u>

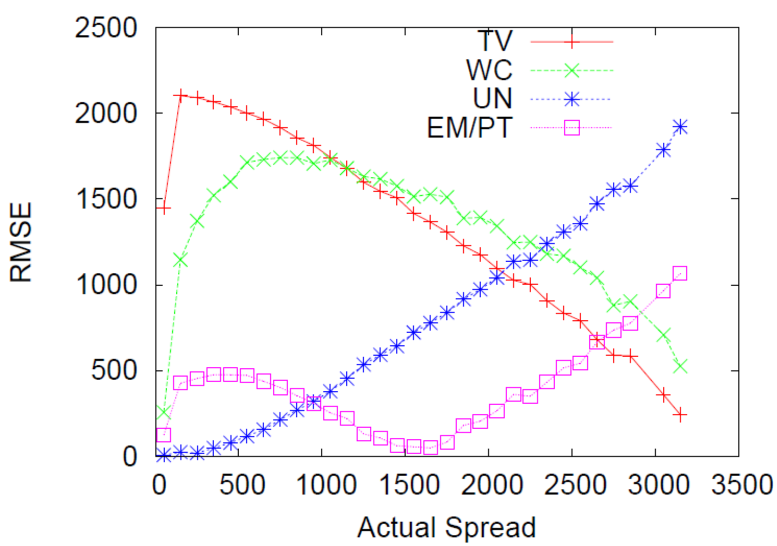
FLIXSTER_SMALL

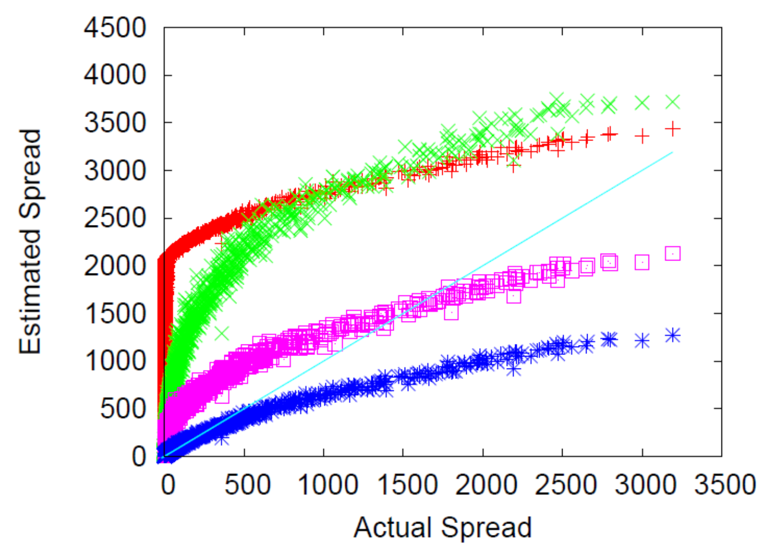
	UN	WC	TV	EM	PT
UN	50	25	5	6	6
WC		50	9	3	2
TV			50	3	2
EM				50	44
PT					50

FLICKR_SMALL

	PT	EM	TV	WC	UN
UN	0	0	44	19	50
WC	0	0	17	50	
TV	0	0	50		
EM	44	50			
PT	50				

2. <u>Given a seed set, we ask to the IC model to predict its spread (on the test set)</u>





YAHOO!

Learning influence strength from propagation data:

why it is complicated

(and some preliminary results)

Learning influence strength: some challenges

Privacy

social graph *G* proprietary and secret (e.g., Twitter)

propagation log *L* proprietary and secret (e.g., Amazon)

two different parties hold the two pieces of input

Scalability and streaming

we have |E| parameters to learn

propagation log *L* potentially huge and streaming

"STRIP: Stream Learning of Influence Probabilities"

Kutzkov, Bifet, Bonchi, Gionis (KDD 2013)

Overfitting

we have |E| parameters to learn

YAHOO!

Privacy-preserving learning of influence strength

(Tassa & Bonchi – submitted 2013)

How the 3 (or more) players can learn influence strength jointly without seeing each other data?

A typical Secure Multiparty Computation setting.
[Details in the paper… once published]

Topic-aware Social Influence Propagation Models

The bulk of the literature on Influence Maximization is **topic-blind**:
the characteristics of the item being propagated are not considered
(it is just one abstract item)

Users authoritativeness, expertise, trust and influence
are topic-dependent

Key observations:
users have different interests,
items have different characteristics,
similar items are likely to interest the same users.

Thus we take a topic-modeling perspective to jointly learn
items characteristics, users' interests and social influence.

The AIR propagation model

Authoritativeness of a user w.r.t. a topic
Interest of a user for a topic
Relevance of an item for a topic

$$P(i|u,t) = \sum_z \overbrace{P(z|u)}^{\text{Each user exhibits different degree of interest in different topics}} \underbrace{P(i|u,z,t)}_{\text{Likelihood of the activation on the item (i) when the topic is (z)}} \geq \theta_u$$

$$P(i|u,z,t) = \frac{\exp\left\{\overbrace{\sum_{v\in V} p_v^z f_v(i,u,t)}^{\text{Cumulative influence by neighbors}} + \overbrace{\varphi_i^z f(i,u,t)}^{\text{Item Selection Weight for the considered topic}}\right\}}{1+\exp\left\{\sum_{v\in V} p_v^z \underbrace{f_v(i,u,t)} + \varphi_i^z \underbrace{f(i,u,t)}\right\}}$$

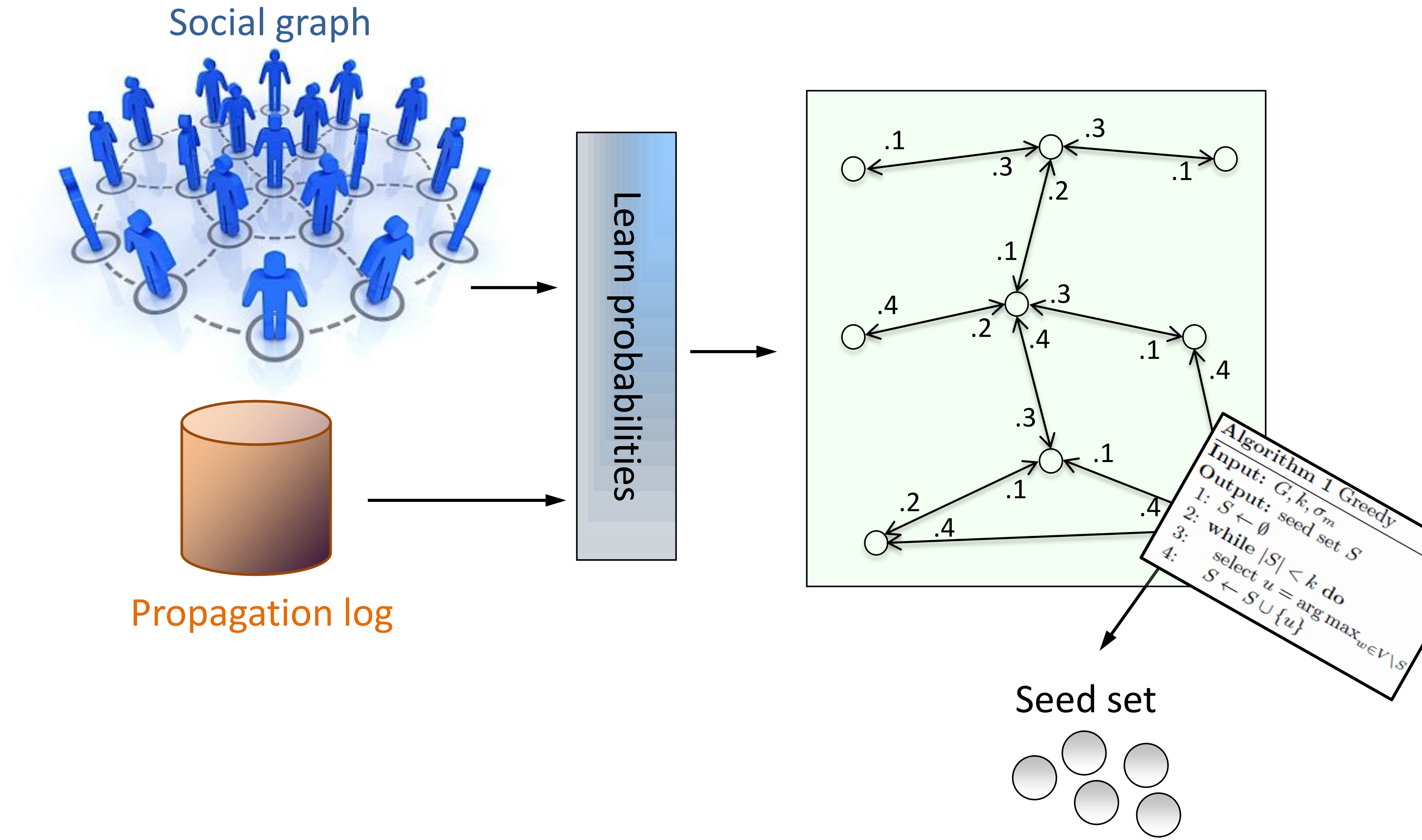
Selection scaling factors

[Learning the model parameters: see paper (!)]

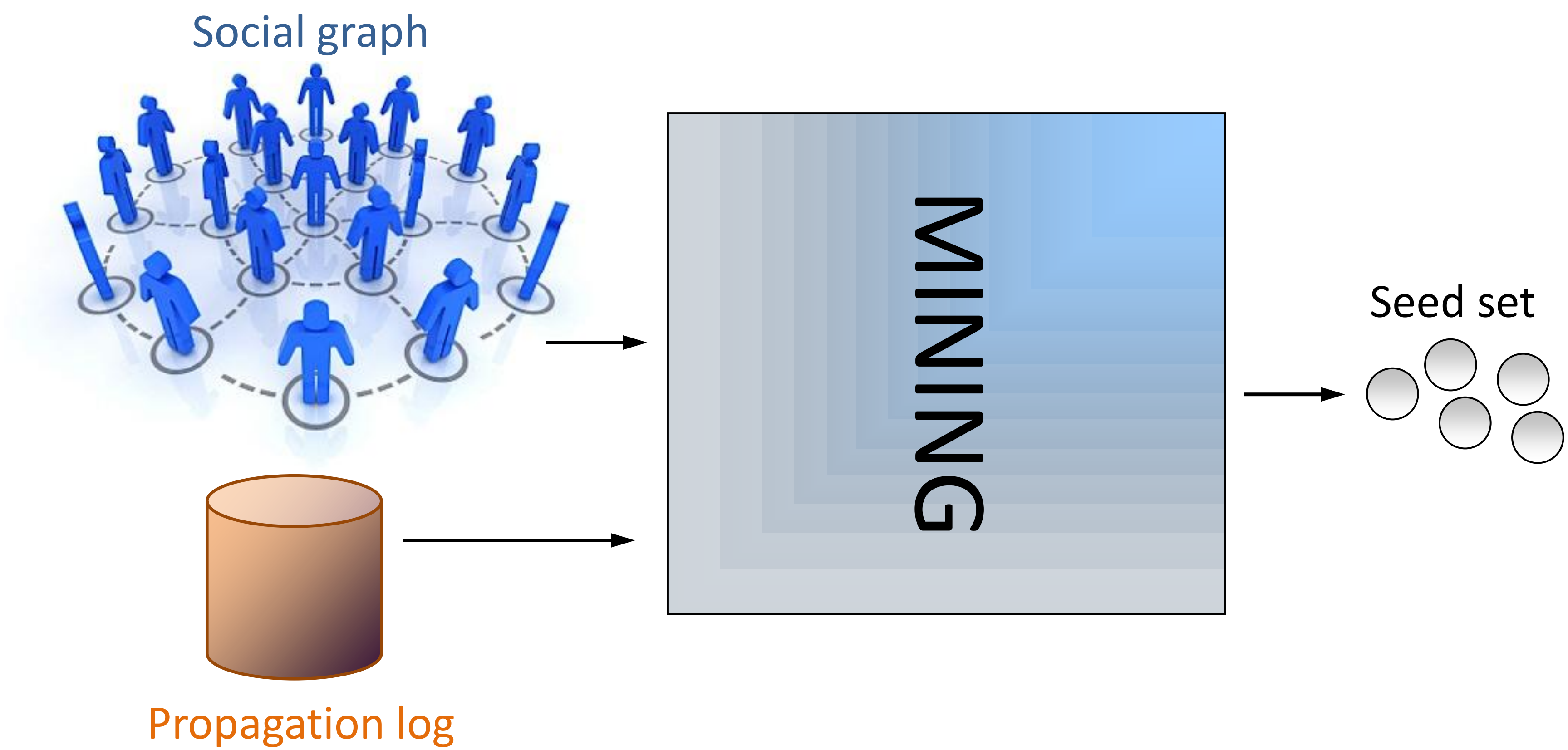
A Data-Based Approach to Social Influence Maximization

Goyal, Bonchi, Lakshmanan (VLDB'12)

The global picture for influence maximization



What we do in this work: direct mining!



YAHOO!

Expected spread: a different perspective

Instead of simulating propagations, use available propagations!

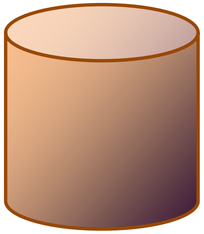
$$\sigma_m(S) = \sum_{X \in \mathbb{G}} Pr[X] \cdot \sigma_m^X(S)$$

sampling "possible worlds"
(MC simulations)

$$\sigma_m^X(S) = \sum_{u \in V} path_X(S, u)$$

$$\sigma_m(S) = \sum_{u \in V} \sum_{X \in \mathbb{G}} Pr[X] \, path_X(S, u)$$

$$\sigma_m(S) = \sum_{u \in V} E[path(S, u)] = \sum_{u \in V} \boxed{Pr[path(S, u) = 1]}$$



Estimate it in "available worlds"
(i.e., our propagation traces)

The sparsity issue

We can not estimate directly $Pr[path(S,u)=1]$ as:

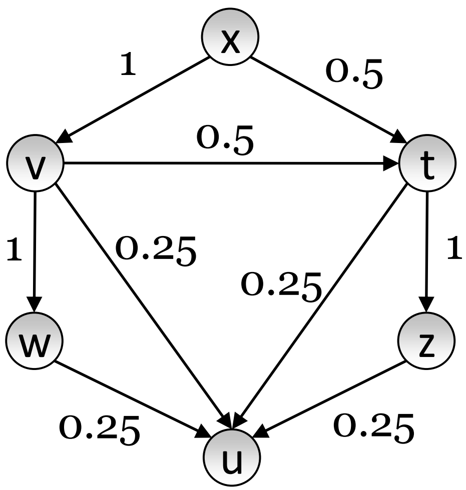
$$\frac{\#\text{ actions in which } S \text{ is the seed-set and } u \text{ participates}}{\#\text{ actions in which } S \text{ is the seed-set}}$$

Too few actions where *S* is effectively the seed set.

Take a u-centric perspective instead:

Each time *u* performs an action we distribute influence credit for this action, back to her anchestors

Credit distribution



Total credit: $\Gamma_{v,u}(a) = \sum_{w \in N_{\text{in}}(u,a)} \Gamma_{v,w}(a) \cdot \gamma_{w,u}(a)$

- Example: assume that for a given action *a* we uniformly split credit among the neighbors that performed the action before *u*: $\gamma_{v,u}(a) = 1/d_{in}(u,a)$

$$\begin{aligned}\Gamma_{v,u} &= \Gamma_{v,v} \cdot \gamma_{v,u} + \Gamma_{v,t} \cdot \gamma_{t,u} + \Gamma_{v,w} \cdot \gamma_{w,u} + \Gamma_{v,z} \cdot \gamma_{z,u} \\ &= 1 \cdot 0.25 + 0.5 \cdot 0.25 + 1 \cdot 0.25 + 0.5 \cdot 0.25 = 0.75.\end{aligned}$$

- For a group of nodes *S*: $\Gamma_{S,u}(a) = \begin{cases} 1 & \text{if } v \in S; \\ \sum_{w \in N_{\text{in}}(u,a)} \Gamma_{S,w}(a) \cdot \gamma_{w,u}(a) & \text{otherwise} \end{cases}$
- Example: $S = \{v, z\}$

$$\begin{aligned}\Gamma_{S,u} &= \Gamma_{S,w} \cdot \gamma_{w,u} + \Gamma_{S,v} \cdot \gamma_{v,u} + \Gamma_{S,t} \cdot \gamma_{t,u} + \Gamma_{S,z} \cdot \gamma_{z,u} \\ &= 1 \cdot 0.25 + 1 \cdot 0.25 + 0.5 \cdot 0.25 + 1 \cdot 0.25 = 0.875.\end{aligned}$$

YAHOO!

Basic credit attribution

different models can be plugged here

in this paper we experiment with

$$\gamma_{v,u}(a) = \frac{\mathit{infl}(u)}{N_{\text{in}}(u,a)} \cdot \exp\left(-\frac{t(u,a) - t(v,a)}{\tau_{v,u}}\right)$$

time-aware: influence decays exponentially over time

user influenceability:

different users have different level of influenceability.

We learn *infl(u)* as the fraction of actions that *u* performs under the influence of at least one neighbor

YAHOO!

Influence Maximization under credit distribution (CD) model

Influence of a set *S* on node *u*

$$\kappa_{S,u} = \frac{1}{\mathcal{A}_u} \sum_{a \in \mathcal{A}} \Gamma_{S,u}(a)$$

total influence of *S*

$$\sigma_{cd}(S) = \sum_{u \in V} \kappa_{S,u}$$

Problem: find S, |S| = k, s.t. $\sigma_{cd}(S)$ is maximum

NP-Hard

$\sigma_{cd}(S)$ is submodular and monotone

(see proofs of Theorem 1 and 2 in the paper)

Method

we can use the greedy algorithm…

Algorithm 1 Greedy

Input: G, k, σ_m
Output: seed set S
1: $S \leftarrow \emptyset$
2: **while** $|S| < k$ **do**
3: select $u = \arg\max_{w \in V \setminus S}(\sigma_m(S \cup \{w\}) - \sigma_m(S))$
4: $S \leftarrow S \cup \{u\}$

… however the greedy algorithm by itself does not guarantee efficiency!

we need an efficient way to compute

$$\sigma_{cd}(S \cup \{w\}) - \sigma_{cd}(S)$$

An efficient way to compute $\sigma_{cd}(S \cup \{w\}) - \sigma_{cd}(S)$

key theorem:

$$\boxed{\sigma_{cd}(S+x) - \sigma_{cd}(S) = \sum_{a \in \mathcal{A}} \left((1 - \Gamma_{S,x}(a)) \cdot \sum_{u \in V} \frac{1}{\mathcal{A}_u} \cdot \Gamma_{x,u}^{V-S}(a) \right)}$$

intuitively, the theorem says that the marginal gain of a node *x* equals the sum of normalized marginal gain of x on all actions

we can compute marginal gain analitically:
no need of MC simulations!

YAHOO!

Method

1. Scan action log once and compute $\Gamma_{v,u}(a)$ for all triplets *(v,u,a)*
2. Start greedy with CELF* optimization. To compute marginal gain use the theorem in the previous slide
3. Once a node is added to the seed set update $\Gamma_{v,u}^{V-S}(a)$ and $\Gamma_{S,x}(a)$ using Lemma 2 and 3.

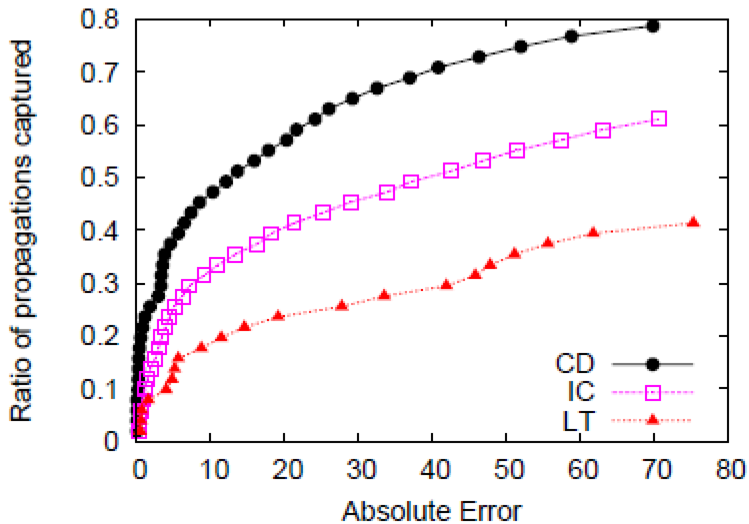
$$\text{LEMMA 2. } \Gamma_{v,u}^{W-x}(a) = \Gamma_{v,u}^{W}(a) - \Gamma_{v,x}^{W}(a) \cdot \Gamma_{x,u}^{W}(a)$$

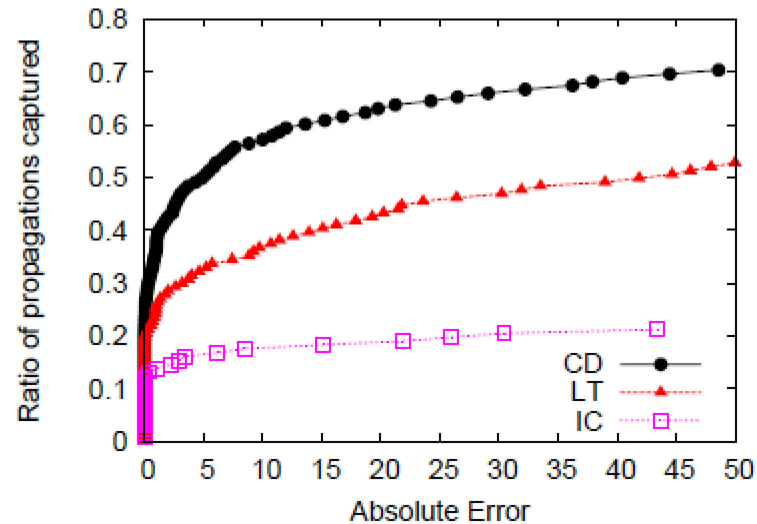
$$\text{LEMMA 3. } \Gamma_{S+x,u}(a) = \Gamma_{S,u}(a) + \Gamma_{x,u}^{V-S} \cdot (1 - \Gamma_{S,x}(a))$$

* Leskovec et al. (KDD'07) *"Cost-effective outbreak detection in networks"*

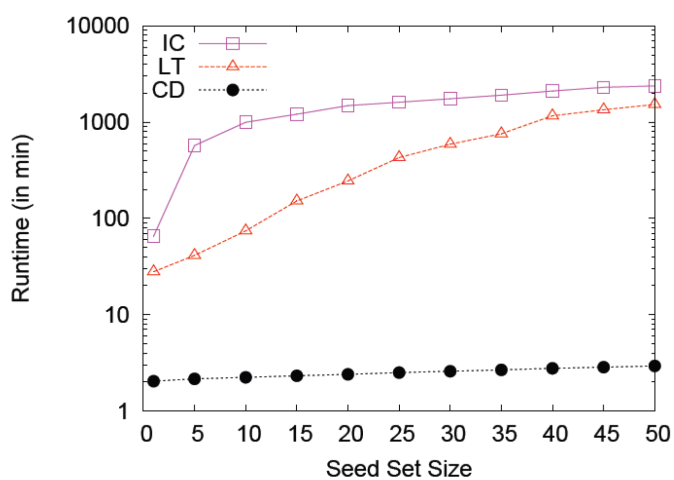
Experiments: quality and efficiency

Datasets: Flixster Flickr





Dataset: Flixster small



YAHOO!

Sparsification of Influence Networks

Mathioudakis, Bonchi, Castillo, Gionis, Ukkonen (KDD'11)

Sparsification of Influence Networks

which connections are most important
for the propagation of actions?

keep only important connections

data reduction
visualization
clustering
efficient graph analysis
find the backbone of influence/information networks

Sparsification

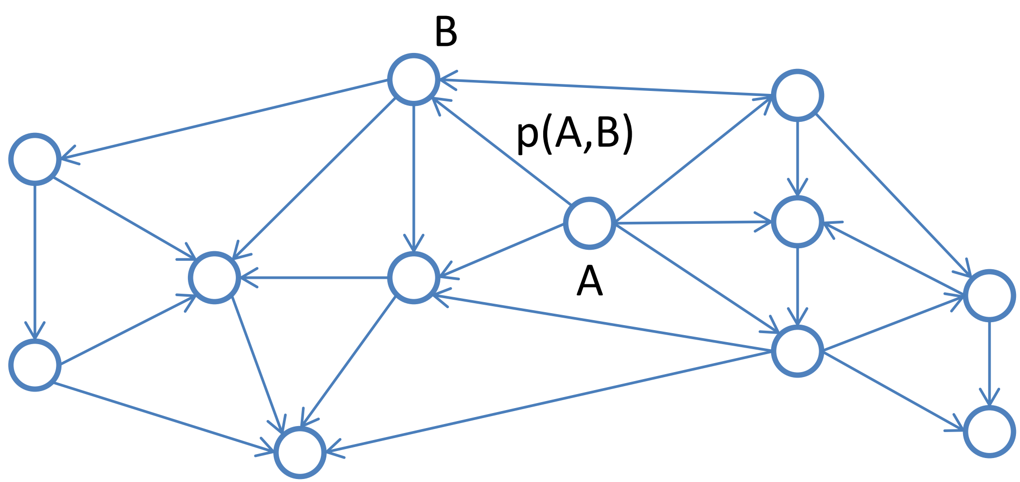
social network

set of propagations

p(A,B)

k arcs

most likely to
explain propagations
(assuming the Independent Cascade model)



Sparsification

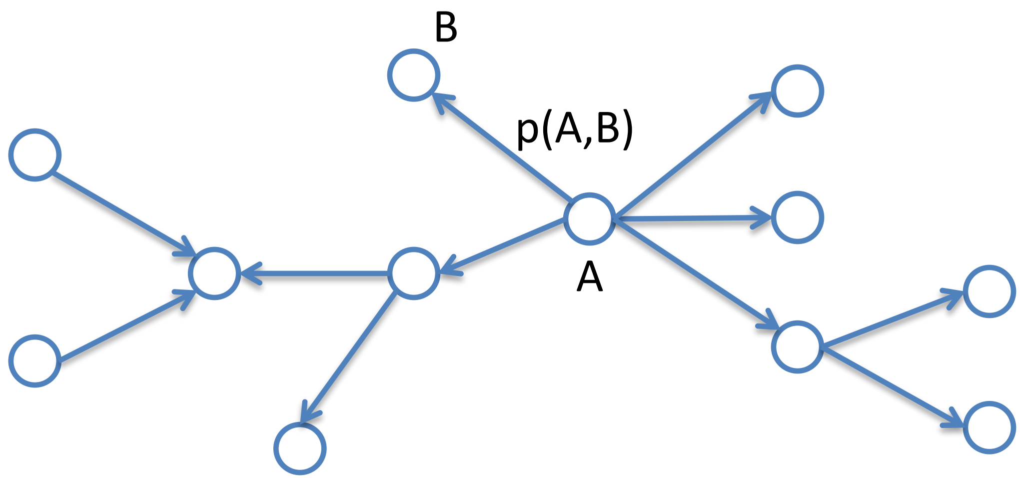
social network

set of propagations

p(A,B)

k arcs

most likely to
explain propagations
(assuming the Independent Cascade model)



Solution

not the k arcs with largest probabilities!

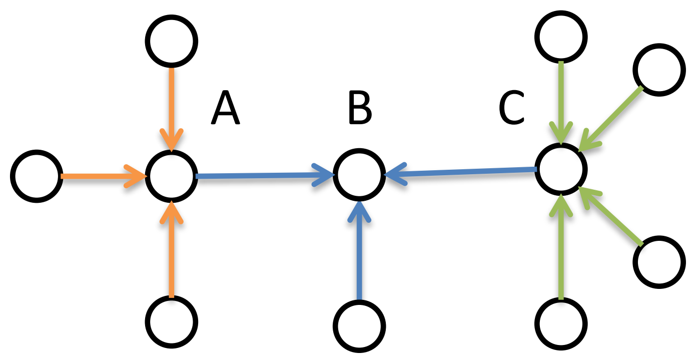
problem is NP-hard and inapproximable

sparsify separately incoming arcs of individual nodes

optimize corresponding likelihood

dynamic programming

optimal solution



$$k_A + k_B + k_C = k$$

Spine - **sp**arsification of **i**nfluence **ne**tworks

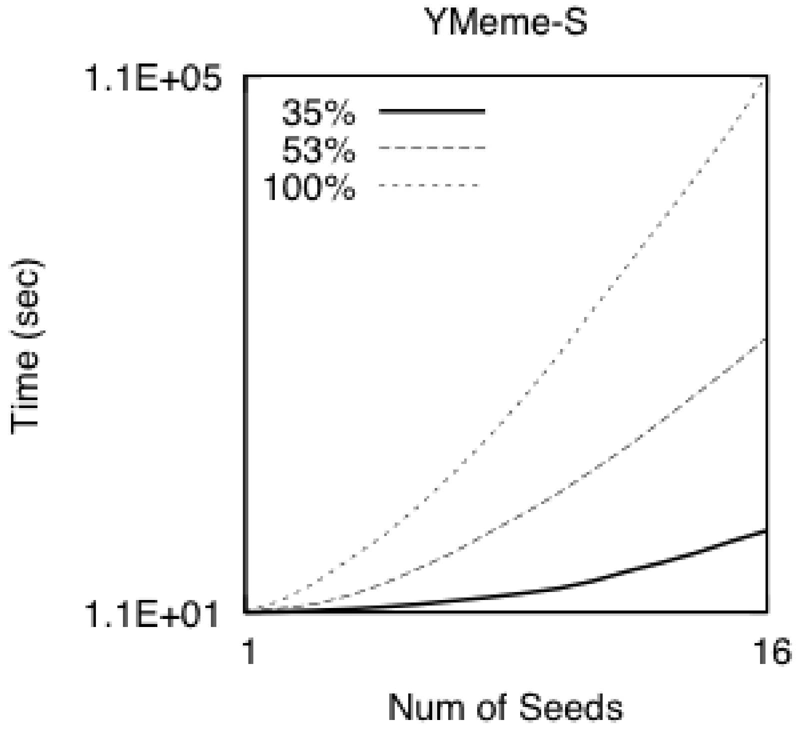
http://www.cs.toronto.edu/~mathiou/spine/

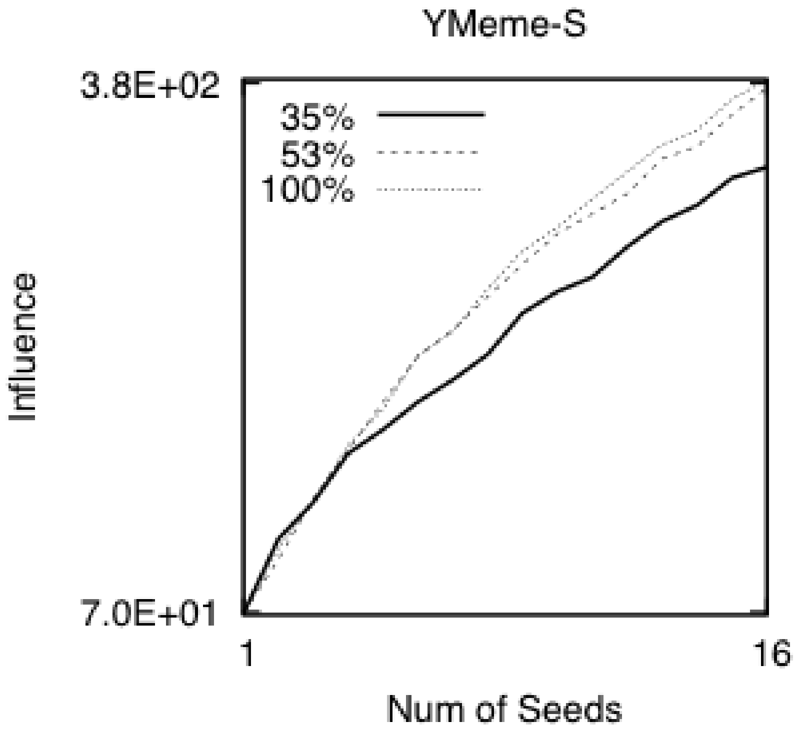
greedy algorithm
two phases

phase 1
obtain a non-zero-likelihood solution
(greedy algorithm for Hitting Set problem)

phase 2
add one arc at a time, the one that offers
largest increase in likelihood
(approximation guarantee for phase 2 thanks to submodularity)

Application to Influence Maximization





YAHOO!

Cascade-based Community Detection

Barbieri, Bonchi, Manco (WSDM'13)

State of the art

Social contagion

measuring social influence

distinguishing social influence from homophily in the data

analysis of influence-driven information propagation in social media

influence maximization

?

Community detection

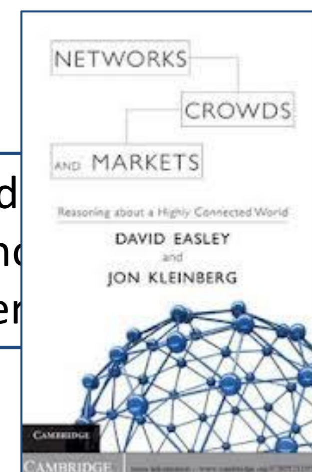
undirected Vs. directed graphs

disjoint Vs. overlapping communities

unlabeled Vs. labeled graphs

> *"...cascades and clusters truly are natural opposites: clusters block the spread of cascades, and whenever a cascade comes to a stop, there's a cluster that can be used to explain why."*

Easley and Kleinberg book [page 577]



Idea: to model the modular structure of SN and the phenomenon of social contagion *jointly*

Input:

directed social graph + a DB of past propagations over the graph

arc *(u,v)* means that *v* "follows" *u*

the DB of propagations is a set of tuples *(i,u,t)*

representing the fact that *u* adopted *i* at time *t*

Output:

overlapping communities of nodes, *that also explain the cascades*.

for each node we also learn the level of

active involvement (i.e., tendency to produce content)

and passive involvement (i.e., tendency to consume content)

in each community

How: by fitting a unique stochastic generative model to the observed social graph and propagations

assumption:

each observed action

forming a link (following somebody), tweeting (original content), re-tweeting

is the result of a stochastic process

observations:

(think about Twitter as an example)

one user belongs to multiple topics/communities of interest

with different levels of active/passive involvement

a link usually can be explained by one and only one community

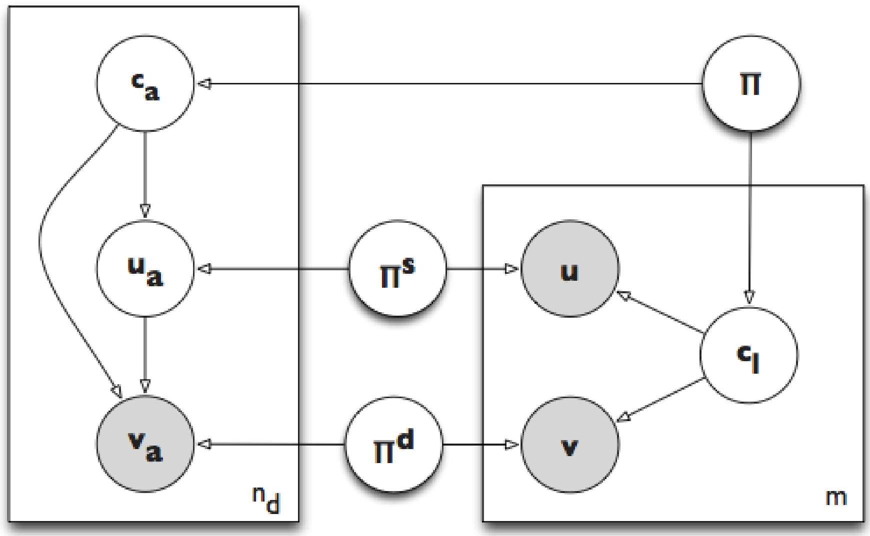
If I'm actively involved in a community I'm followed, and I tweet

If I'm passively involved in a community, I follow, I re-tweet,

but I'm not followed nor I tweet new content

The CCN Model

(communities, cascades, network)



3 prior components:

the probability Π to observe an action in a community

the level of active Π^s and passive Π^d interest of each user in each community

each observed action is explained by the 3 priors

The CCN Model (continued)

Probability of a link

(source)

$$\vartheta_u^k = \frac{\exp\left\{\pi_u^{k,s}\right\}}{\sum_{\overline{u}\in N} \exp\left\{\pi_{\overline{u}}^{k,s}\right\}}$$

(destination)

$$\varphi_u^k = \frac{\exp\left\{\pi_u^{k,d}\right\}}{\sum_{\overline{u}\in N} \exp\left\{\pi_{\overline{u}}^{k,d}\right\}}$$

Probability of an action being propagated

(influencer)

$$\theta_u^{k,a} = \frac{\exp\left\{\pi_u^{k,s}\right\}}{\sum_{u'\in \mathcal{F}_{i_a}(t_a)} \exp\left\{\pi_{u'}^{k,s}\right\}}$$

(influenced)

$$\phi_{u,v}^{k,a} = \frac{\exp\left\{\pi_v^{k,d}\right\}}{\sum_{v':(u,v')\in A, v'\notin C_{i_a}(t_a-1)} \exp\left\{\pi_{v'}^{k,d}\right\}}$$

Learning the model parameters

The non-linearity of the selection function makes it difficult to maximize the likelihood

Solution adopted

Generalized Expectation-Maximization + Improved Iterative Scaling

(details in the paper!)

Experimental evaluation: datasets

	Digg	Flixster	Meme	LastFm
Users	1,000	29,357	9,385	1.372
Social Relationships	24,842	425,228	1,144,932	14,708
Bidirectional	N	Y	N	N
Items	31,911	11,659	12,760	51,495
Overall Activations($\|\mathbb{L}\|$)	1,086,065	6,529,011	726,809	1,208,640
Influence Episodes ($\|\mathbb{D}\|$)	315,377	2,239,744	684,368	322,932

Digg: social news website

Action *(i,u,t)* means that user *u* voted story *i* at time *t*

Flixster: social movie consumption (ranting and rating)

Action *(i,u,t)* means that user *u* rated movie *i* at time *t*

Meme (discontinued): microblogging platforms

Action *(i,u,t)* means that user *u* posted meme *i* at time *t*

LastFM: social music consumption

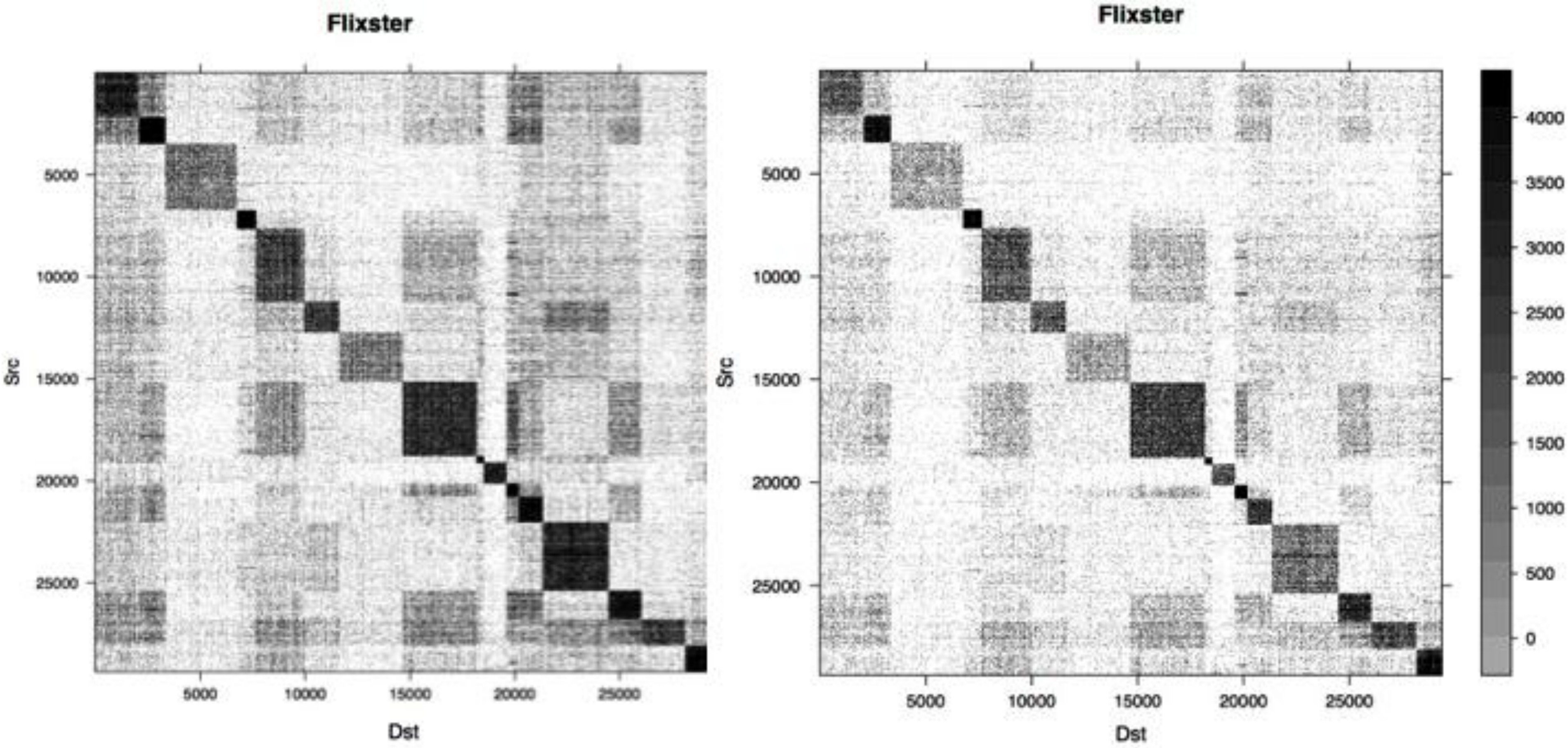
Action *(i,u,t)* means that user *u* listened to song *i* at time *t*

YAHOO!

Community structure within the graph and propagations DB

Adjacency matrix (left) and the influence matrix (right)

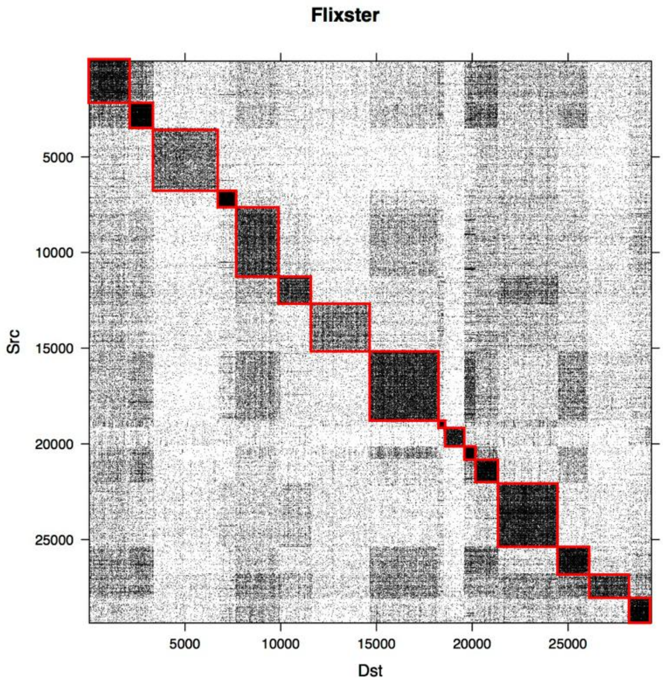
The influence matrix records for each cell *(u,v)* the number of actions for which the model infers that *u* triggered *v*'s activation



YAHOO!

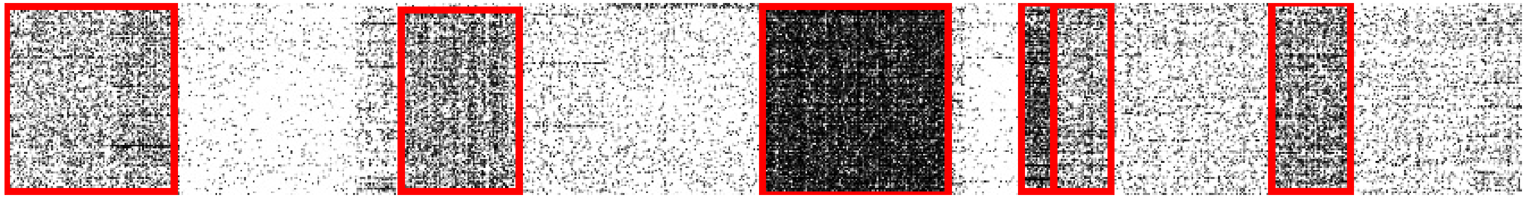
Observations

All the matrices reflect a community structure that is inferred by both he action log and the graph: blocks are clearly visible in both the adjacency and the influence matrices.



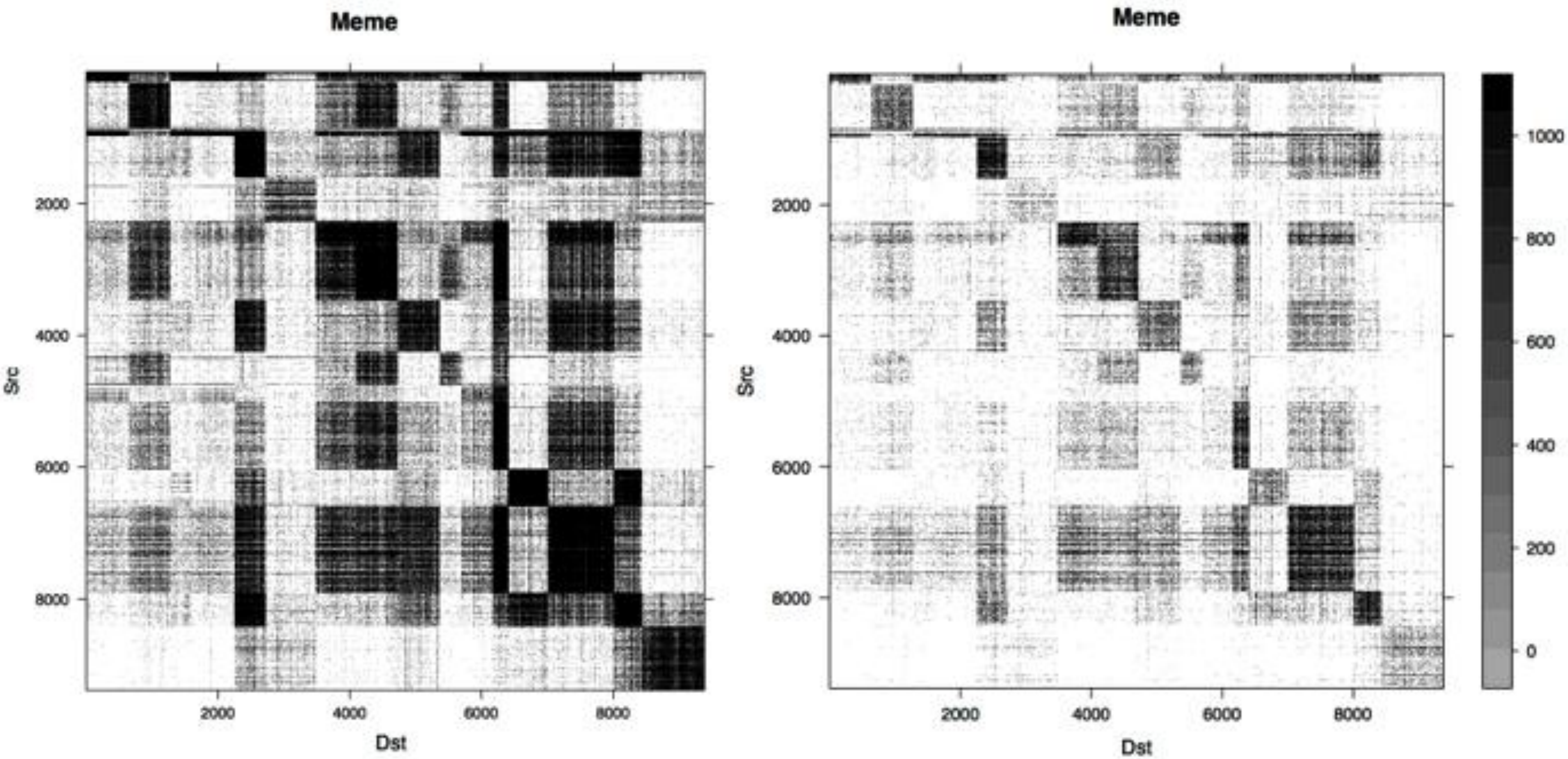
The matrices exhibit a diagonal structure, a clear indication that users are mainly bound to a single community.

Other blocks can be detected: since communities model links and actions, some users are likely to assume different roles in more than one community.



YAHOO!

Community structure within the graph and propagations DB



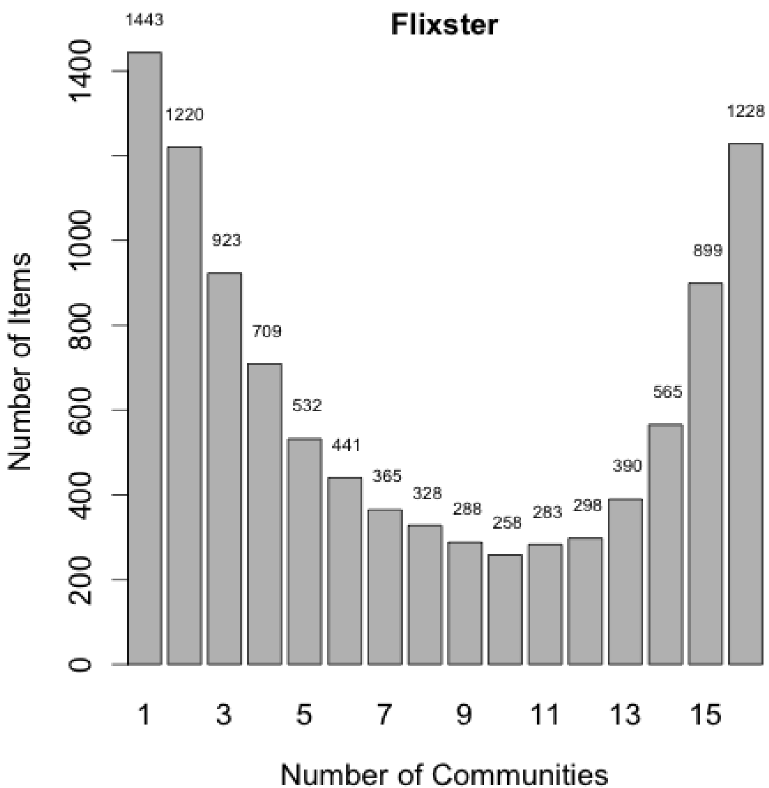
Although users tend to belong to different communities,
their influence is strong only in few of them

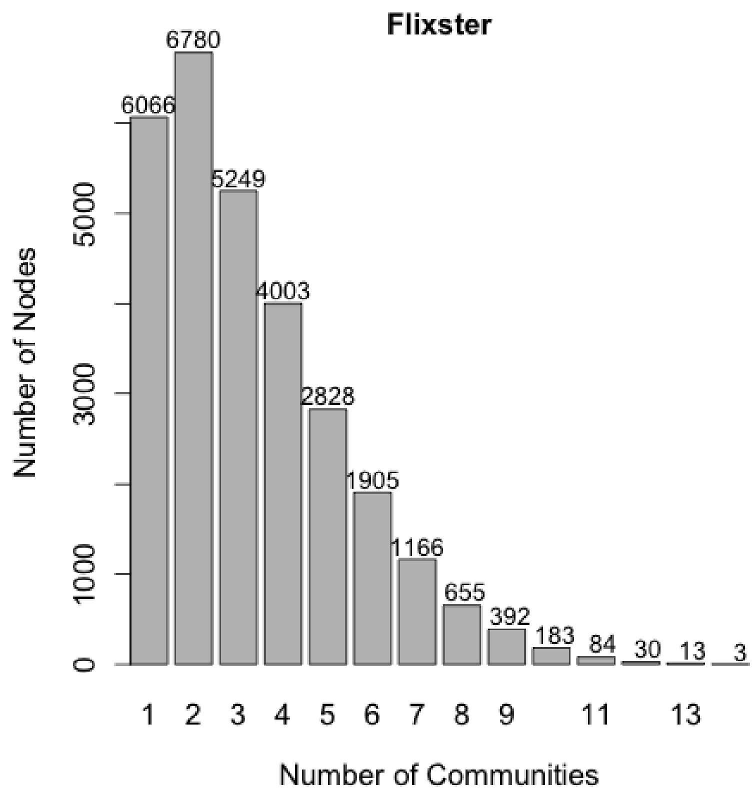
Characterizing the communities

In how many communities users and items tend to participate?

The participation in a community can be inferred by the parameter:

$$\eta_{u,a,k}(\Theta) = P(z_a^k, w_a^u | a \in \mathbb{D}, \Theta)$$





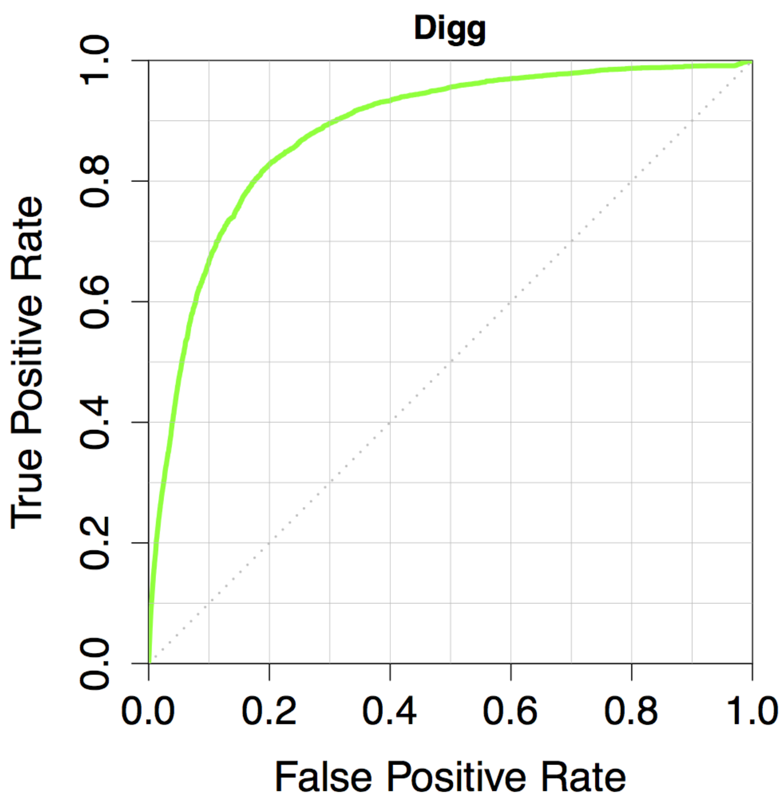
YAHOO!

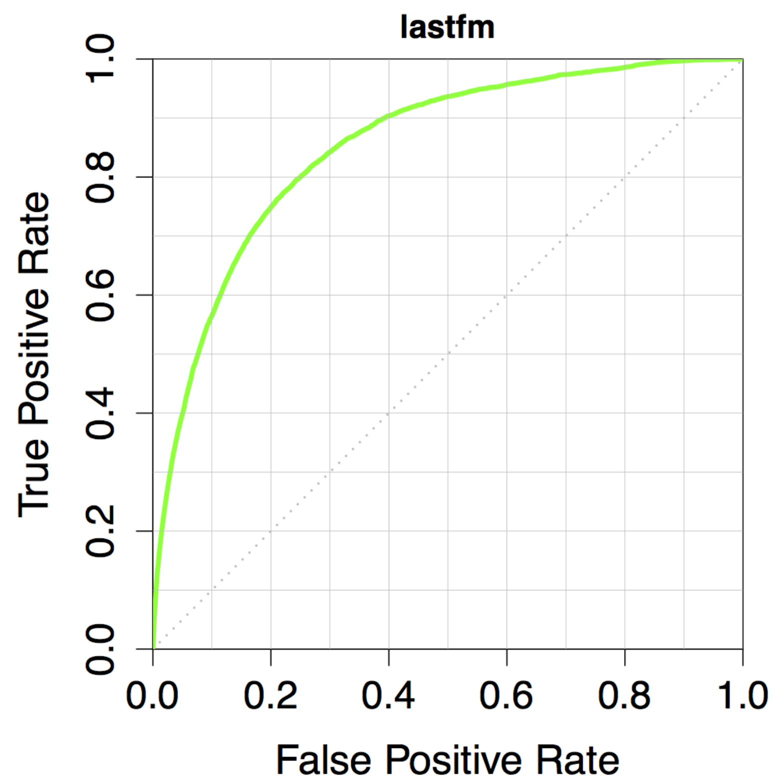
Link Prediction

(Preliminary results to be presented in the extended version)

CCN directly models links probabilities:

$$P(u, v|\Theta) = \sum_k \vartheta_u^k \varphi_v^k \pi_k$$





THANKS!

Twitter: @FrancescoBonchi
email: bonchi@yahoo-inc.com

http://francescobonchi.com/