Aalto University
School of Science

Mining temporal networks

Aristides Gionis

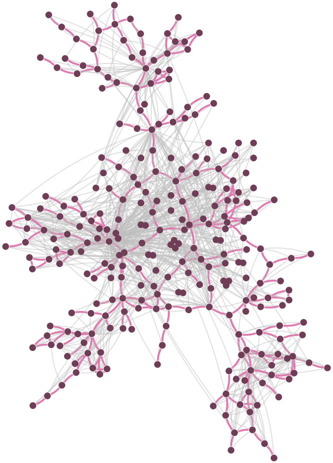
Department of Computer Science, Aalto University

users.ics.aalto.fi/gionis

Stochastic sauna
Dec 19, 2017

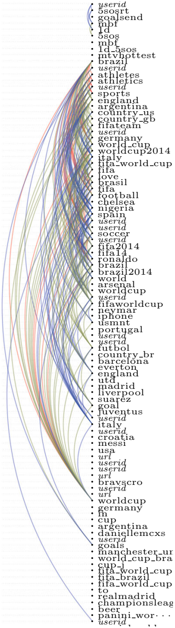
interconnected world

- networks model objects and their relations
- many different network types
  - social
  - informational
  - technological
  - biological
  - …



impact of network science

- online communication networks and social media
- implications in
  - knowledge creation
  - information sharing
  - education
  - democracy
  - society as a whole



research questions

- structure discovery
  - finding communities, events, roles of individuals
- study complex dynamic phenomena
  - evolution, information diffusion, opinion formation
- develop novel applications
- design efficient algorithms

traditional view

- networks represented as pure graph-theory objects

  no additional vertex / edge information

- emphasis on static networks

- dynamic settings model structural changes

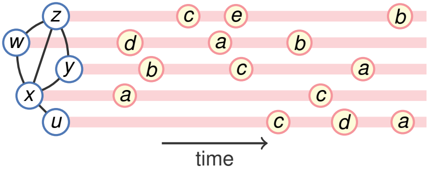
  vertex / edge additions / deletions

temporal networks

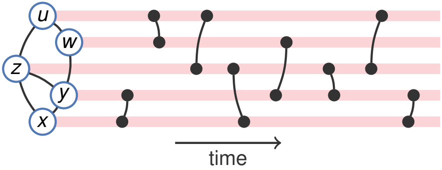
- ability to collect and store large volumes of network data
- available data have fine granularity
- lots of additional information associated to vertices/edges
- network topology is relatively stable, while
  lots of activity and interaction is taking place
- giving rise to new concepts, new problems, and
  new computational challenges

modeling activity in networks

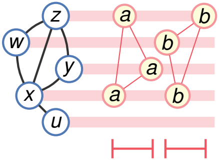
1. network nodes perform actions (e.g., posting messages)



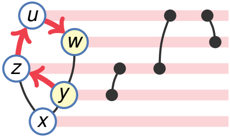
2. network nodes interact with each other
   (e.g., a “like”, a repost, or sending a message to each other)



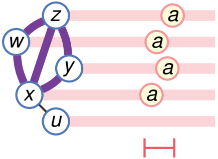
many novel and interesting concepts



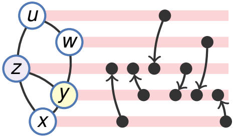
new pattern types



temporal information paths



new types of events



network evolution

temporal networks — objectives

- identify new concepts and new problems
- develop algorithmic solutions
- demonstrate revelance to real-world applications

agenda

tracking important nodes

- maintaining neighborhood profiles
- temporal PageRank

tracking important nodes

maintaining sliding-window neighborhood profiles

R. Kumar, T. Calders, A. Gionis, and N. Tatti, ECML PKDD 2015

distance distributions in graphs

- given graph G, a node u, and distance r :

  how many nodes of G are in distance r from u?

- fundamental graph-mining primitive
  - median distance, diameter, effective diameter
- related to small-world phenomena
- a measure of centrality for nodes of G

distance distributions in graphs

- exact solution requires all-pairs shortest path computation
  - Floyd-Warshall algorithm: $\mathcal{O}(n^3)$
  - or, BFS for unweighted graphs: $\mathcal{O}(nm)$
- clearly non scalable
- resort to approximations based on diffusion methods

diffusion-based computation

[Palmer et al., 2002]

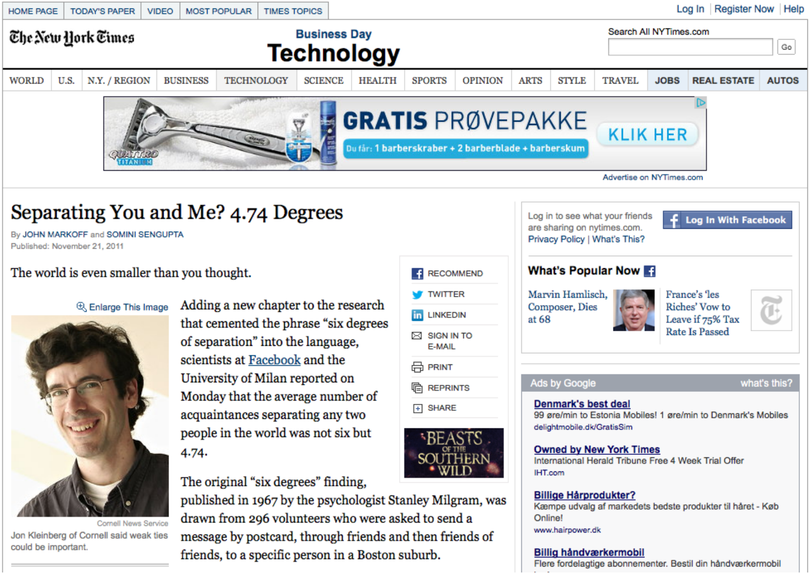
- let $B_t(x)$ be the ball of radius t around x (the set of nodes at distance $\leq t$ from x)
- clearly $B_0(x) = \{x\}$
- moreover $B_{t+1}(x) = \bigcup_{(x,y)} B_t(y) \bigcup \{x\}$
- so computing B_{t+1} from B_t just takes a single (sequential) scan of the graph

diffusion-based computation

- every set requires $O(n)$ bits, hence $O(n^2)$ bits overall
- amount of space is prohibitively large
- instead use sketching for counting distinct elements
- probabilistic counters require very small space ($\log\log$)
- HyperANF algorithm [Boldi et al., 2011]
  - uses HyperLogLog counters [Flajolet et al., 2007]
  - with 40 bits you can count up to 4 billion with standard deviation 6%

estimating the number of distinct values (F_0)

- [Flajolet and Martin, 1985]
- estimate distinct values seen in data stream $x_1, x_2, \ldots$
- consider a bit vector of length $O(\log n)$
- upon seen x_i, set:
  - the 1st bit with probability $1/2$
  - the 2nd bit with probability $1/4$
  - $\ldots$
  - the i-th bit with probability $1/2^i$
- important: bits are set deterministically for each x_j
- let R be the index of the largest bit set
- return $Y = 2^R$



The New York Times

Business Day
Technology

Separating You and Me? 4.74 Degrees

By JOHN MARKOFF and SOMINI SENGUPTA
Published: November 21, 2011

The world is even smaller than you thought.

Cornell News Service

Jon Kleinberg of Cornell said weak ties could be important.

Adding a new chapter to the research that cemented the phrase "six degrees of separation" into the language, scientists at Facebook and the University of Milan reported on Monday that the average number of acquaintances separating any two people in the world was not six but 4.74.

The original "six degrees" finding, published in 1967 by the psychologist Stanley Milgram, was drawn from 296 volunteers who were asked to send a message by postcard, through friends and then friends of friends, to a specific person in a Boston suburb.

Log in to see what your friends are sharing on nytimes.com. Privacy Policy | What's This?

What's Popular Now

Marvin Hamlisch, Composer, Dies at 68

France's 'les Riches' Vow to Leave if 75% Tax Rate Is Passed

extension to temporal networks

- limitations of existing solutions
  - consider static network
  - multi-pass algorithm
- in this work
  - extension to temporal networks
  - streaming algorithm for sliding-window model :

    consider only the most recent interactions (edges)

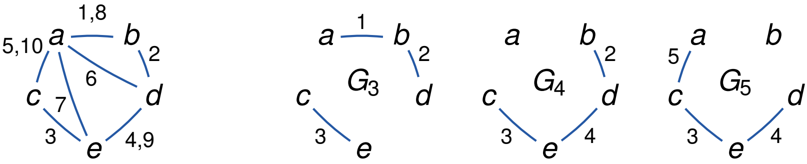
setting

- temporal network $G = (V, E)$
- stream of edges $E = \langle (u_1, v_1, t_1), (u_2, v_2, t_2), \ldots \rangle$ with $t_1 \leq t_2 \leq \ldots$
- sliding window length w
- snapshot network $G(t, w)$ at time t contains all edges with time-stamps in $(t - w, t]$

problem :

given node u, window length w, and distance r, how many nodes in $G(t, w)$ are within distance r from u at time t?

example



a toy example, 3 snapshot graphs with a window size of 3

proposed online algorithms

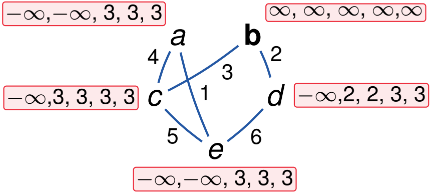
1. an exact but memory-inefficient streaming algorithm
2. an approximate memory-efficient streaming algorithm

– approximate algorithm uses logic of exact algorithm, combined with hyperloglog sketches

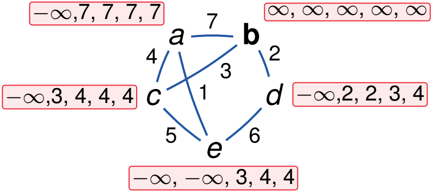
horizons

- path horizon : time-stamp of the oldest edge on the path
- $h(u, v, i)$: the horizon for length i between nodes u and v : the maximum horizon of any path of length at most i

example



$-\infty, -\infty, 3, 3, 3$
$\infty, \infty, \infty, \infty, \infty$
a
b
4
2
3
$-\infty, 3, 3, 3, 3$
c
1
d
$-\infty, 2, 2, 3, 3$
5
6
e
$-\infty, -\infty, 3, 3, 3$



two snapshot graphs along with $h(u, b, i)$ for $i = 0, \ldots, 4$

neighborhood summaries

- observation : if for a node u we know all horizons $h(u, v, i)$, for all distances i and all nodes v, we can give complete neighborhood profile for u for any window length
- neighborhood summary : $S_t^u = (S_t^u[0], \ldots, S_t^u[r])$ where $S_t^u[i] = \{(v, h_t(u, v, i)) \mid h_t(u, v, i) > -\infty\}$

updating neighborhood summaries

- edge deletion : simply delete entries from summaries
- edge addition : a change in summary at distance i for a node u will introduce a change in the summary of its neighbors at distance $i + 1$
  - updates propagate in a BFS fashion

exact algorithm

- update time : $\mathcal{O}(rmn \log n)$
- space complexity : $\mathcal{O}(rn^2)$

  where r an upper bound on max distance

- quadratic dependence not acceptable for large graphs
  - hence approximation algorithm

approximate algorithm

- sliding HyperLogLog sketch : extension of HyperLogLog to maintain a distinct set counter over sliding window
- if number of buckets in the HLL counter is k then the worst case complexity changes to

– update time :

$\mathcal{O}(rm2^k \log^2 n)$ from $\mathcal{O}(rmn \log n)$

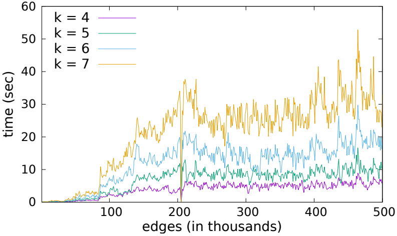
– space complexity :

$\mathcal{O}(rn2^k \log n)$ from $\mathcal{O}(rn^2)$

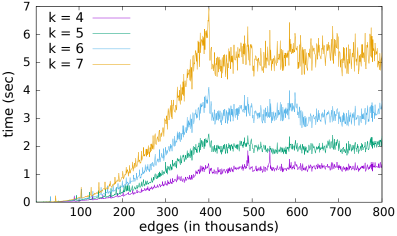
empirical evaluation — quality

dataset	nodes	dist edges	total edges	clus coef	diam	eff diam	avg rel error (k=7)
Facebook	4 039	88 234	88 234	0.60	8	4.7	0.08
Cit-HepTh	27 771	352 801	352 801	0.31	13	5.3	0.10
Higgs	166 840	249 030	500 000	0.19	10	4.7	0.14
DBLP	192 357	400 000	800 000	0.63	21	8.0	0.09

empirical evaluation — running time



(c) Higgs



(d) DBLP

contrast (DBLP)

- offline HyperANF : 3.6 sec / sliding window
- proposed approach : 0.003 sec / sliding window

tracking important nodes

temporal PageRank

P. Rozenshtein and A. Gionis, ECML PKDD 2016

PageRank

- classic approach for measuring node importance
- listed in the top-10 most important data-mining algorithms

  [Wu et al., 2008]
- numerous applications
  - ranking web pages
  - trust and distrust computation
  - finding experts in social networks
  - …

PageRank

- PageRank defined as the stationary distribution of a random walk in the graph

- inherently a static process

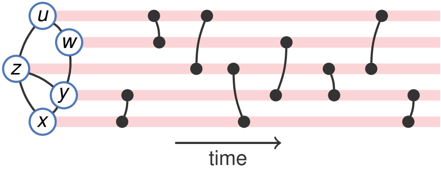
- however, many modern networks can be viewed as a sequence (stream) of edges
  - temporal network : $G = (V, E)$, with $E = \{(u, v, t)\}$
  - examples : twitter, instagram, IMs, email, ...

- what is an appropriate PageRank definition for temporal networks?

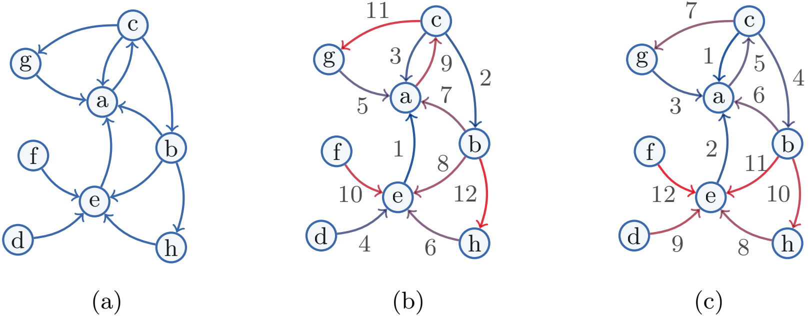
temporal networks

network nodes interact with each other

(e.g., a “like”, a repost, or sending a message to each other)



motivating example



(a)

static network

(b)

temporal network

(c)

temporal network

research questions and objectives

- extend PageRank to incorporate temporal information and network dynamics

- adapt PageRank to reflect changes in network dynamics and node importance

- estimate importance of a node u at any given time t

dynamic PageRank vs. temporal PageRank

- extensive work on dynamic PageRank
- dynamic PageRank computation :

  maintain correct PageRank during network updates

  - e.g., edge additions / deletions
- computation should return the static PageRank at a given network snapshot
- for edges present in a snapshot, order does not matter

static PageRank

- graph $G = (V, E)$
- corresponding row-stochastic matrix $P \in \mathbb{R}^{n \times n}$
- personalization vector $\mathbf{h} \in \mathbb{R}^n$
- PageRank is the stationary distribution of a random walk, with restart probability $(1 - \alpha)$

$$\pi(u) = \sum_{v \in V} \sum_{k=0}^{\infty} (1 - \alpha)\alpha^k \sum_{\substack{z \in \mathcal{Z}(v,u) \\ |z|=k}} h(v) \Pr[z \mid v]$$

where, $\mathcal{Z}(v, u)$ is the set of all paths from v to u

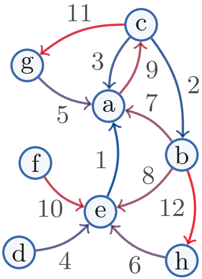
and $\Pr[z \mid v] = \prod_{(i,j) \in z} P(i, j)$

temporal PageRank

- make a random walk only on temporal paths

  e.g., time-respecting paths

  time-stamps increase along the path



$c \rightarrow b \rightarrow a \rightarrow c$: time respecting

$a \rightarrow c \rightarrow b \rightarrow a$: not time respecting

temporal PageRank

- intuition : probability of visiting node u at time t given a random walk on temporal paths
- need to model probability of following next temporal edge
  - we use an exponential distribution
- temporal PageRank definition

$$r(u,t) = \sum_{v \in V} \sum_{k=0}^{t} (1-\alpha)\alpha^k \sum_{\substack{z \in \mathcal{Z}^T(v,u|t) \\ |z|=k}} \Pr{}'[z \mid t]$$

$\mathcal{Z}^T(v, u \mid t)$ set of temporal paths from v to u until time t

computation

- simple online algorithm
- $r(u, t)$: temporal PageRank estimate of u at time t
- $s(u, t)$: count of active walks visiting u at time t

```
**input** : $E$, transition probability $\beta$, jumping probability $\alpha$
$\boldsymbol{r} = \mathbf{0}$, $\boldsymbol{s} = \mathbf{0}$;
**foreach** $(u, v, t) \in E$ **do**
    $\boldsymbol{r}(u) = \boldsymbol{r}(u) + (1 - \alpha)$;
    $\boldsymbol{r}(v) = \boldsymbol{r}(v) + (\boldsymbol{s}(u) + (1 - \alpha))\alpha$;
    $\boldsymbol{s}(v) = \boldsymbol{s}(v) + (\boldsymbol{s}(u) + (1 - \alpha))(1 - \beta)\alpha$;
    $\boldsymbol{s}(u) = (\boldsymbol{s}(u) + (1 - \alpha))\beta$;
normalize $\boldsymbol{r}$;
**return** $\boldsymbol{r}$;
```

static vs. temporal PageRank

- temporal PageRank is designed to capture changes in network dynamics and concept drifts
- what if the edge distribution is stable?

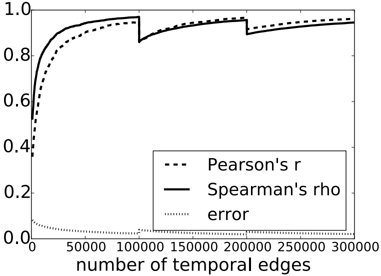
static vs. temporal PageRank

- consider static network $G_S = (V, E_S, w)$
- time period $[1, \ldots, T]$
- construct temporal network $G = (V, E)$ by sampling edges proportionally to their weight

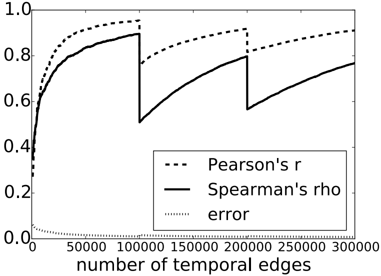
proposition :

as $T \to \infty$, the temporal PageRank on G
converges to the static PageRank on G_S,
with personalization vector equal to weighted out-degree

experiment — adaptation to concept drift



(a) *Facebook*



(b) *Twitter*

summary

- examples of mining temporal networks
  - maintaining sliding-window neighborhood profiles
  - temporal PageRank
- potential for new concepts, new problem definitions, new computational methods, and new applications

references

Boldi, P., Rosa, M., and Vigna, S. (2011).
HyperANF: approximating the neighborhood function of very large graphs on a budget.
In *WWW*.

Flajolet, F., Fusy, E., Gandouet, O., and Meunier, F. (2007).
Hyperloglog: the analysis of a near-optimal cardinality estimation algorithm.
In *Proceedings of the 13th conference on analysis of algorithm (AofA)*.

Flajolet, P. and Martin, N. G. (1985).
Probabilistic counting algorithms for data base applications.
Journal of Computer and System Sciences, 31(2):182–209.

Palmer, C. R., Gibbons, P. B., and Faloutsos, C. (2002).
ANF: a fast and scalable tool for data mining in massive graphs.
In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 81–90, New York, NY, USA. ACM Press.

references (cont.)

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Philip, S. Y., et al. (2008).

Top 10 algorithms in data mining.

KAIS.