

Affiliation Networks

Silvio Lattanzi

Sapienza University of Roma, Italy

D. Sivakumar

Google

Networks — a new extroverted paradigm in TCS

Classical applications

(explicitly constructed networks)

- Roads, maps, ...
- Traffic networks
- Computer networks

More recent uses

(implicitly extant networks)

- Friendships among people (email, Facebook, MySpace, orkut, hi5, ...)
- Citations among scientific articles
- Hyperlinked documents (Usenet articles, the web graph)

“... how social, technological, and natural worlds are connected”

— *Easley and Kleinberg*

Network models beyond Erdős–Rényi

Some 21st century concerns:

(Crovella et al., Faloutsos, et al., Broder et al.)

– *Evolving vs. static networks*

the web graph, citation networks, the Internet graph, etc. all grow over time

– *Non-uniform degree distributions*

heavy-tailed degree distributions, ‘rich-get-richer’ phenomena

– *Existence of locally dense and other interesting structures*

numerous copies of $K_{s,t}$ for small s, t on the web
clusteredness, or ‘friend-of-friend’ phenomena

Network models beyond Erdős–Rényi

Explicit degree-sequence Models (Aiello et al.)

– uniformly chosen graphs from among all graphs with a fixed degree sequence

Preferential attachment Models (Barabasi et al, Bollobás et al.)

– as network grows (and new vertices arrive), the probability that the next edge will be incident on a vertex is proportional to its degree

Edge-copying Models (Kumar et al.)

– new vertices pick existing vertices as ‘prototypes’ and copy some of their neighborhood

Small-world Models (Watts et al., Kleinberg)

– constant degree, most edges local, a few edges connect to long-range neighbors

Some successes of the post-ER models

Power-law degree distributions

‘Rich-get-richer’ or ‘centrality’ phenomenon (PA, EC)

Small ($O(\log n)$) diameter (PA, EC + some random edges)

Discoverable short paths aka. Milgram’s ‘six degrees of separation’ (SW)

Numerous bipartite cliques (EC)

Friend-of-Friend phenomenon in social networks (SW)

Empirically verifiable geographically ‘local’ and ‘long-range’ friendships
on *LiveJournal*

Limitations

No clear unifying model (for the web graph, citation networks, social networks...)

No known algorithmic benefits

A sample algorithmic problem (cf. Google's AdSense)

you're given a graph whose vertices are English phrases and whose edges indicate relatedness;

a subset C of commercially important phrases are highlighted;

you're allowed to preprocess the graph and construct some data structures;

given a small subset of vertices of the graph, can you quickly return the top few vertices in C that are close to the given vertices?

...what does this graph of relatedness look like?

But ships vanish at the horizon...

Leskovec, Faloutsos, and Kleinberg (2005)

Studied several real graphs (the Internet graph, citation networks, email networks, etc.) over time and reported:

Densification:

– average degree *increases* over time, graphs grow to have superlinear number of edges

Shrinking Diameter:

– effective diameter *decreases* over time, most pairs of vertices have bounded-length paths between them

A new slew of models (2005–2008)

Leskovec, Faloutsos, Kleinberg

‘Community Guided Attachment’ and ‘Forest Fire’ models

– some mathematical results, models too complex to admit tractable analyses

Leskovec et al., Mahdian–Xu

Models based on Kronecker multiplication

– mathematically flexible, unnatural

Leskovec, Backstrom, Kumar, Tomkins

Evolving models based on PA + ‘triangle closing’

– intuitively appealing, mathematically formidable

... which brings us to this work

Affiliation Networks — Highlights

Natural model rooted in sociology (Breiger, 1974)

- underlying bipartite graph tracks the growth of the network
- bipartite graph captures notion of *societies* of people

Mathematical analyses of all important properties reducible to analysis of degree distribution

- highlights role of high-degree ‘connectors’
- cf. Gladwell’s *The Tipping Point*

Explains all structural properties observed to date:

- heavy-tailed degree distributions, locally dense structures
- densification, shrinking/stabilizing diameter

Underlying sparse structure that preserves/approximates distances
algorithmically recoverable!

Social Networks from Affiliation Networks

$B(Q, U)$	$G(Q, E)$
<p>Fix two integers $c_q, c_u > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, the bipartite graph $B_0(Q, U)$ is a simple graph with at least $c_q c_u$ edges, where each node in Q has at least c_q edges and each node in U has at least c_u edges.</p>	

Social Networks from Affiliation Networks

$B(Q, U)$	$G(Q, E)$
<p>Fix two integers $c_q, c_u > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, the bipartite graph $B_0(Q, U)$ is a simple graph with at least $c_q c_u$ edges, where each node in Q has at least c_q edges and each node in U has at least c_u edges.</p>	<p>Fix integers $c_q, c_u, s > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, $G_0(Q, E)$ consists of the subset Q of the vertices of $B_0(Q, U)$, and two vertices have an edge between them for every neighbor in U that they have in common in $B_0(Q, U)$.</p>

Social Networks from Affiliation Networks

$B(Q, U)$	$G(Q, E)$
<p>Fix two integers $c_q, c_u > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, the bipartite graph $B_0(Q, U)$ is a simple graph with at least $c_q c_u$ edges, where each node in Q has at least c_q edges and each node in U has at least c_u edges.</p> <p>At time $t > 0$:</p> <p>(Evolution of Q) With probability β:</p> <p><i>(Arrival)</i> A new node q is added to Q.</p> <p><i>(Preferentially chosen Prototype)</i> A node $q' \in Q$ is chosen as <i>prototype</i> for the new node, with probability proportional to its degree.</p> <p><i>(Edge copying)</i> c_q edges are “copied” from q'; that is, c_q neighbors of q', denoted by u_1, \dots, u_{c_q}, are chosen uniformly at random (without replacement), and the edges $(q, u_1), \dots, (q, u_{c_q})$ are added to the graph.</p>	<p>Fix integers $c_q, c_u, s > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, $G_0(Q, E)$ consists of the subset Q of the vertices of $B_0(Q, U)$, and two vertices have an edge between them for every neighbor in U that they have in common in $B_0(Q, U)$.</p>

Social Networks from Affiliation Networks

$B(Q, U)$	$G(Q, E)$
<p>Fix two integers $c_q, c_u > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, the bipartite graph $B_0(Q, U)$ is a simple graph with at least $c_q c_u$ edges, where each node in Q has at least c_q edges and each node in U has at least c_u edges.</p> <p>At time $t > 0$:</p> <p>(Evolution of Q) With probability β:</p> <p><i>(Arrival)</i> A new node q is added to Q.</p> <p><i>(Preferentially chosen Prototype)</i> A node $q' \in Q$ is chosen as <i>prototype</i> for the new node, with probability proportional to its degree.</p> <p><i>(Edge copying)</i> c_q edges are “copied” from q'; that is, c_q neighbors of q', denoted by u_1, \dots, u_{c_q}, are chosen uniformly at random (without replacement), and the edges $(q, u_1), \dots, (q, u_{c_q})$ are added to the graph.</p>	<p>Fix integers $c_q, c_u, s > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, $G_0(Q, E)$ consists of the subset Q of the vertices of $B_0(Q, U)$, and two vertices have an edge between them for every neighbor in U that they have in common in $B_0(Q, U)$.</p> <p>At time $t > 0$:</p> <p>(Evolution of Q) With probability β:</p> <p><i>(Arrival)</i> A new node q is added to Q.</p> <p><i>(Edges via Prototype)</i> An edge between q and another node in Q is added for every neighbor that they have in common in $B(Q, U)$ (note that this is done after the edges for q are determined in B).</p>

Social Networks from Affiliation Networks

$B(Q, U)$	$G(Q, E)$
<p>Fix two integers $c_q, c_u > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, the bipartite graph $B_0(Q, U)$ is a simple graph with at least $c_q c_u$ edges, where each node in Q has at least c_q edges and each node in U has at least c_u edges.</p> <p>At time $t > 0$:</p> <p>(Evolution of Q) With probability β:</p> <p><i>(Arrival)</i> A new node q is added to Q.</p> <p><i>(Preferentially chosen Prototype)</i> A node $q' \in Q$ is chosen as <i>prototype</i> for the new node, with probability proportional to its degree.</p> <p><i>(Edge copying)</i> c_q edges are “copied” from q'; that is, c_q neighbors of q', denoted by u_1, \dots, u_{c_q}, are chosen uniformly at random (without replacement), and the edges $(q, u_1), \dots, (q, u_{c_q})$ are added to the graph.</p> <p>(Evolution of U) With probability $1 - \beta$, a new node u is added to U following a symmetrical process, adding c_u edges to u.</p>	<p>Fix integers $c_q, c_u, s > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, $G_0(Q, E)$ consists of the subset Q of the vertices of $B_0(Q, U)$, and two vertices have an edge between them for every neighbor in U that they have in common in $B_0(Q, U)$.</p> <p>At time $t > 0$:</p> <p>(Evolution of Q) With probability β:</p> <p><i>(Arrival)</i> A new node q is added to Q.</p> <p><i>(Edges via Prototype)</i> An edge between q and another node in Q is added for every neighbor that they have in common in $B(Q, U)$ (note that this is done after the edges for q are determined in B).</p> <p>(Edges via evolution of U)</p> <p>With probability $1 - \beta$:</p> <p>A new edge is added between two nodes q_1 and q_2 if the new node added to $u \in U$ is a neighbor of both q_1 and q_2 in $B(Q, U)$.</p>

Social Networks from Affiliation Networks

$B(Q, U)$	$G(Q, E)$
<p>Fix two integers $c_q, c_u > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, the bipartite graph $B_0(Q, U)$ is a simple graph with at least $c_q c_u$ edges, where each node in Q has at least c_q edges and each node in U has at least c_u edges.</p> <p>At time $t > 0$:</p> <p>(Evolution of Q) With probability β:</p> <p><i>(Arrival)</i> A new node q is added to Q.</p> <p><i>(Preferentially chosen Prototype)</i> A node $q' \in Q$ is chosen as <i>prototype</i> for the new node, with probability proportional to its degree.</p> <p><i>(Edge copying)</i> c_q edges are “copied” from q'; that is, c_q neighbors of q', denoted by u_1, \dots, u_{c_q}, are chosen uniformly at random (without replacement), and the edges $(q, u_1), \dots, (q, u_{c_q})$ are added to the graph.</p> <p>(Evolution of U) With probability $1 - \beta$, a new node u is added to U following a symmetrical process, adding c_u edges to u.</p>	<p>Fix integers $c_q, c_u, s > 0$, and let $\beta \in (0, 1)$.</p> <p>At time 0, $G_0(Q, E)$ consists of the subset Q of the vertices of $B_0(Q, U)$, and two vertices have an edge between them for every neighbor in U that they have in common in $B_0(Q, U)$.</p> <p>At time $t > 0$:</p> <p>(Evolution of Q) With probability β:</p> <p><i>(Arrival)</i> A new node q is added to Q.</p> <p><i>(Edges via Prototype)</i> An edge between q and another node in Q is added for every neighbor that they have in common in $B(Q, U)$ (note that this is done after the edges for q are determined in B).</p> <p>(Edges via evolution of U)</p> <p>With probability $1 - \beta$:</p> <p>A new edge is added between two nodes q_1 and q_2 if the new node added to $u \in U$ is a neighbor of both q_1 and q_2 in $B(Q, U)$.</p> <p>(Preferentially Chosen Edges) A set of s nodes q_{i_1}, \dots, q_{i_s} is chosen, each node independently of the others (with replacement), by choosing vertices with probability proportional to their degrees, and the edges $(q, q_{i_1}), \dots, (q, q_{i_s})$ are added to $G(Q, E)$.</p>

Some intuition

Which papers did I cite when I wrote this one?

Which papers did you cite in your most recent paper?

Which papers will be cited by the next FOCS paper written?

Who's on your IM List?

How many different 'circles' of friends do you have?

How often do you say 'So, how's so-and-so?' in your conversation when you meet someone at a conference reception?

Affiliation Networks – Model Summary

Social networks are unions of cliques on its set of users, plus a constant number of PA edges per vertex.

May be thought of as a natural generalization of Kleinberg's notions of 'local' and 'long-range' links:

- edges via folding are the local edges
- random edges via PA are the long-range edges

The Fundamental Constant & a Power Law for B

$$\text{Let } \Delta = \frac{c_u(1-\beta)}{c_q\beta}.$$

When studying $G(Q, E)$ it's interesting if $\Delta < 1$, so let's think of it as 0.99

Theorem

The degrees of vertices in U in $B(Q, U)$ follow a power law distribution with exponent $2 + \Delta$ for all degrees up to n^γ , where $\gamma = 1/(4 + 1/\Delta)$ (think $\gamma = 0.19$), that is,

$$\Pr_u[\text{degree}(u) = i] \sim i^{-(2+\Delta)}.$$

The Degree Lemmata

Lemmata

- (1) Large degree vertices in $B(Q, U)$ gain a constant fraction of their neighbors within φn steps for any $\varphi > 0$.
- (2) Large degree vertices in $B(Q, U)$ gain a constant fraction of their neighbors after φn steps for any $\varphi > 0$.
- (3) After φn steps, for any $\varphi > 0$, the number of edges in $B(Q, U)$ that touch a vertex in U of degree at least i is $\Theta(ni^{-\Delta})$, for any i up to n^γ .

Proofs all via recurrences, measure concentration, Lipschitz conditions, etc.

Heavy Tail Degree Distribution for G

Theorem

The degree distribution of $G(Q, E)$ dominates the cdf of a power law distribution with exponent $-2 - 1/\Delta$ for all degrees up to n^γ , where $\gamma = 1/(4 + 1/\Delta)$.

All but $o(n)$ vertices of $G(Q, E)$ have degree $\Theta(1)$.

Densification

Theorem

If $\Delta < 1$, the number of edges in $G(Q, E)$ is $\omega(n)$.

$$\begin{aligned} |E| &> \sum_{i=1}^{n^\gamma} (\text{num. vertices in } U \text{ of degree } i) \binom{i}{2} \\ &= \sum_{i=1}^{n^\gamma} \left(\left(\left(\frac{n}{\zeta(2+\Delta)} \frac{1}{i^{2+\Delta}} \right) (1 \pm o(1)) \right) \binom{i}{2} \right) \\ &\sim n^{1 + \frac{\Delta(1-\Delta)}{1+4\Delta}} \end{aligned}$$

Shrinking / Stabilizing Diameter

Theorem

If $\Delta < 1$, for any constant c , the c -effective diameter of $G(Q, E)$ shrinks or stabilizes after time φn , for some $\varphi > 0$.

Outline

Define $H \subseteq U$ in B to be the set of vertices of degree $> n^\alpha$, where $\alpha < \gamma$.

Choose $\epsilon < \varphi$ such that $(\varphi - \epsilon)^2 > 2c$.

Lower bound the number of edges in B with one edge in H at time ϵn by $\Omega(n^{1+\gamma(1-\Delta)})$.

Upper bound the number of edges in B with no end point in H at time φn by $O(n^{1+\alpha(1-\Delta)})$.

Thus any vertex that arrives between ϵn and φn will, whp., point to a vertex in H in B in its final s choices of PA edges.

Thus a c fraction of node pairs will have a path of length at most $\text{diam}(H) + 2$.

Sparsification

Intuitively, $G(Q, E)$ is highly sparsifiable, since it has an underlying sparse core, viz. $B(Q, U)$

However, the problem of computing ‘square roots’ of graphs is NP-Hard in general

For our models, it would be enough to compute any \hat{B} such that $\text{Fold}(\hat{B}) = G$ and $|\hat{B}| = O(n)$; however, this too is NP-Hard.

Nevertheless, we will show two sparsification techniques, both of which implicitly carry out the following idea: replace large cliques by stars.

Sparsification I: Pruning via BFS Trees

Given $G(Q, E)$ and a subset $R \subseteq Q$ of relevant nodes.

(1) Initially, label all edges *deletable*.

(2) For each node $q \in R$:

(a) compute the BFS tree starting from q , exploring the children of each node in increasing order of insertion;

(b) label all edges in the BFS tree from q as undeletable.

(3) Delete all edges still labeled deletable.

Theorem

If R is chosen uniformly at random from Q and $|R| = O(n/\log n)$, and if $\Delta < 1$, then whp. the algorithm produces a new graph in which the distance between every pair of nodes is preserved if at least one of them belongs to R .

Proof Idea for Sparsification I

The analysis is carried out ‘virtually’ on B , even though the algorithm is executed on G .

Key idea:

- for large degree vertices in U , a large fraction of its neighbors in Q do not participate in any BFS tree along any path to a relevant node
- these are the ‘latecomer’ nodes, typically of constant degree

Technical mess:

Getting the ‘high probability’ statement to work

Sparsification II: Stretching with Spanners

Definition

A k -spanner of a graph G is an edge-subgraph J of G such that for any pair of vertices in G , their distance in J is no more than k times their distance in G .

Algorithm (after Baswana and Sen)

Process edges of G one by one, and add it to J iff it does not close a cycle of length $2k$ or less in J .

Trivially produces a $2k$ -spanner.

otoh, since a graph with more than $n^{1+1/k}$ edges must have a cycle of length at most $2k$ edges, number of edges in J is at most $n^{1+1/k}$.

Sparsification II: Stretching with Spanners

Key idea: In any clique generated by any node in U of degree i , we will retain no more than $i^{1+1/k}$ edges. This implies that

$$|J| \leq \sum_{i=1}^n (\text{num. nodes of degree } i \text{ in } U) (i^{1+1/k}),$$

which turns out to be $\Theta(n)$ for suitable choice of k wrt. Δ .

Quick Recap + Some Extensions, Questions

Results still work if we replace t -cliques by $G(t, p)$ where $t = d(u)$ and $p = \Theta(1/d(u)^\alpha)$ for some $\alpha < 1$.

What if we change the rule for folding B to produce G as follows?

– place an edge between q_1 and q_2 iff there are at least two vertices $u_1 \neq u_2$ in U that are both common neighbors of q_1 and q_2 ?

Formalize the intuition of ‘local’ and ‘long-range’ edges with respect to the shortest distance metric on B and obtain simpler proofs of everything

Thank you!
Questions/comments welcome

Milgram Experiments

Algorithm	$\tau = 1$			$\tau = 5$			$\tau = 10$			$\tau = 15$		
	S	μ	M	S	μ	M	S	μ	M	S	μ	M
ShortestDistance	100	6.3	6.0	100	6.0	6.0	100	5.5	5.0	100	5.4	5.0
Lookahead-Expand	80	8.9	7.0	85	7.3	6.0	92	5.9	6.0	97	5.9	6.0
Lookahead-Monotone-Expand	64	6.7	6.0	75	6.3	6.0	91	5.9	6.0	94	5.9	6.0
Local-Expand	57	13.3	9.0	70	11.1	8.0	83	8.8	7.0	91	8.8	7.0
Lookahead	56	11.8	9.0	71	10.4	8.0	77	8.0	6.0	85	7.1	6.0
Lookahead-Monotone	33	6.8	7.0	46	6.5	6.0	63	5.9	6.0	74	5.8	6.0
Local	21	24.1	18.0	33	23.4	18.0	49	21.1	14.5	50	19.6	13.0
Local-Monotone-Expand	18	5.9	6.0	28	5.8	5.0	42	5.4	5.0	56	5.4	5.0
Local-Monotone	4	5.6	6.0	5	5.6	6.0	11	5.6	6.0	15	5.4	6.0

Table 1: Performance of routing methods on co-authorship network, sorted by success rate, computed using 575 random source–target pairs. Key: S = Success Rate of Algorithm / Success Rate of ShortestDistance, μ = Mean path length for successful routing, M = Median path length for successful routing.