Diffusion and Clustering on Large Graphs

Alexander Tsiatas
Thesis Proposal / Advancement Exam
8 December 2011
Introduction

Graphs are omnipresent in the real world – both natural and man-made

Examples of large graphs:

– Technological networks
– Social networks
– Communication networks
– Transportation networks
– Biological networks
Introduction

Real world graphs are **large**

- Internet: billions of webpages
- Facebook: over 350M active users
- Millions of road miles in the USA

Computations become intractable

Important to have rigorous analysis
Proposal outline

Background on diffusion and clustering
  Introductory example: network epidemics

Preliminary work
  Global graph partitioning using PageRank
  Graph drawing using PageRank

Ongoing work
  Planted partition model
  Spectral analysis of truncated communication networks

Future work
  Hierarchical clustering
Network epidemics: a motivating example

- Graph $G$ given
- $t = 0$: infection starts
- Spreads along edges
- Infected agents cure according to antidote
Network epidemics: a motivating example

How to track: **diffusion**

Distributing antidote: find a **cluster**
Random walk: a tool for studying diffusion

Model of diffusion on $G$
- Start at $u$
- Move to $v$ chosen from neighbors
- Can repeat for any number of steps
- Interested in limiting behavior
Random walk stationary distribution

Probability that random walk is at $u$?

As time goes to infinity:
- Constant probability on each node
- Proportional to degree distribution
Random walks are not always useful

Next: **Personalized PageRank**
Another diffusion model
Personalized PageRank

Another model for diffusion on a graph $G$

At each time step:

Probability $(1 - \alpha)$: take a random walk step

Probability $\alpha$: restart the random walk

to a vertex uniformly at random (traditional)

to a distribution $s$: personalized
A vector with $n$ components

Each component: a vertex $u$

2 parameters:

- Jumping constant $\alpha$
- Starting distribution $s$

Can be a single vertex

Denoted by $pr(\alpha, s)$
Personalized PageRank distributions

$\alpha = 0.1$  
$\alpha = 0.01$
Personalized PageRank distributions

$\alpha = 0.1$  \hspace{1cm}  $\alpha = 0.01$
Computing PageRank vectors

**Method 1:** Solve matrix equation:

\[ \text{pr}(\alpha, s) = \alpha s + (1 - \alpha)\text{pr}(\alpha, s)W \]

- \( W \) is the *random walk matrix*
- Intractable for large graphs

**Method 2:** Iterate diffusion model

- Fast convergence, but still intractable for large graphs

**Method 3:** Use \( \varepsilon \)-approximate PageRank vector

- Uses only local computations
- Running time: \( O(1/\varepsilon \alpha) \) independent of \( n \)
- [Andersen, Chung, Lang ’06; Chung, Zhao ’10]
Diffusion: recap

Two models of diffusion

Next: graph clustering and relation to diffusion
Graph clustering

Dividing a graph into **clusters**

- Well-connected internally
- Well-separated
Applications of graph clustering

Product recommendations
Image segmentation
Finding communities
etc.
Diffusion, clustering, and bottlenecks

Hard to diffuse through bottlenecks
Bottlenecks are good cluster boundaries
Back to network epidemics: how to distribute antidote?

Find a local **cluster** around starting points
Give each vertex in that cluster antidote proportional to its **personalized PageRank**
  – $\alpha$ depends on the infection rate
  – $s$ is the infection’s starting point
Proposal outline

Background on diffusion and clustering
  Introductory example: network epidemics

Preliminary work
  Global graph partitioning using PageRank
  Graph drawing using PageRank

Ongoing work
  Planted partition model
  Spectral analysis of truncated communication networks

Future work
  Hierarchical clustering
Global graph partitioning

Given a graph $G = (V, E)$, find $k$ centers and $k$ corresponding clusters

Want two properties:

- Well-connected internally
- Well-separated
Global graph partitioning: algorithms

\textbf{$k$-means}  
Attempts to find $k$ centers that optimize a sum of squared Euclidean distance  
Optimization problem is NP-complete

\textbf{Spectral clustering, Markov clustering, ...}  
Require an expensive matrix computation
Finding graph clusters using PageRank

**Result:** A new graph clustering algorithm
Finds $k$ clusters with the desired properties
Does not require matrix computations
   – Relies on more efficient PageRank computations
Pairwise distances using PageRank

Graph setting: not Euclidean space

Graph distance (hops) is not very helpful

Real-world graphs have the small-world phenomenon

PageRank distance:

$$\text{dist}_\alpha(u, v) = \|\text{pr}(\alpha, u)D^{-1/2} - \text{pr}(\alpha, v)D^{-1/2}\|_2$$

where $D$ is the diagonal degree matrix

Closely related vertices: close in PageRank distance
Evaluating clusters

We will develop 2 metrics for $k$ centers $C$:

$\mu(C)$ : measures internal cluster connectivity

$\Psi_\alpha(C)$ : measures separation

**Goal**: find clusters with small $\mu(C)$ and large $\Psi_\alpha(C)$
Internal cluster connectivity

\[ \mu(C) = \sum_{v \in V} d_v \text{dist}_\alpha(c_v, v)^2 \]

(similar to k-means)

\( c_v \) is the center closest to \( v \)

Small \( \mu(C) \): clusters are well-connected internally

Small distances to purple, large distances to orange
Cluster separation

\[ \Psi_\alpha(C') = \sum_{c \in C} \text{vol}(R_c) \text{dist}_\alpha(c, \pi)^2 \]

\( R_c \) is the cluster containing \( c \)
If \( \Psi_\alpha(C) \) is large, then the clusters are well-separated
Finding clusters using $\mu(C)$ and $\Psi_\alpha(C)$

Well-connected internally: $\mu(C)$ is small
Well-separated: $\Psi_\alpha(C)$ is large
Optimizing these metrics: computationally hard
Observation: components of $\text{pr}(\alpha, \nu)$ give a ranking of potential cluster centers for $\nu$. 

Good centers

Bad centers
New method to find clusters

Form $C$ randomly

**Step 1**: choose a set $C'$ of $k$ vertices sampled from $\pi$

**Step 2**: for each $v \in C'$, choose a center $c_v$ from $\text{pr}(\alpha, v)$

Assign each non-center $u$ to closest center using PageRank distance
Clusters should be well-connected internally

Before: want small $\mu(C)$

$$\mu(C) = \sum_{v \in V} d_v \text{dist}_\alpha(v, c_v)^2$$

Now: replace $c_v$ with a sample from $\text{pr}(\alpha, v)$. Expectation of $\mu(C)$: (no dependence on $C$)

$$\Phi(\alpha) = \sum_{v \in V} d_v \text{dist}_\alpha(v, \text{pr}(\alpha, v))^2$$
Clusters should be well-separated

Before: want large $\Psi_\alpha(C)$

$$\Psi_\alpha(C) = \sum_{c \in C} \text{vol}(R_c) \text{dist}_\alpha(c, \pi)^2$$

Now: replace $c$ with a sample from $\text{pr}(\alpha, v)$, for each vertex $v$

Expectation of $\Psi_\alpha(C)$: (no dependence on $C$)

$$\Psi(\alpha) = \sum_{v \in V} d_v \text{dist}_\alpha(\text{pr}(\alpha, v), \pi)^2$$
New metrics are tractably optimizable

Choose $C$ randomly and optimize expectations:
- Want small $\Phi(\alpha)$, large $\Psi(\alpha)$
- Only depends on $\alpha$
- No such $\alpha$? $G$ isn’t clusterable.
To find centers and clusters

**Step 1**: find an $\alpha$ with small $\Phi(\alpha)$, large $\Psi(\alpha)$

**Step 2**: randomly select $C$ using PageRank with this chosen $\alpha$

**Result**: $E[\mu(C)]$ is small, $E[\Psi_\alpha(C)]$ large

Choose enough sets $C$ so that metrics are close to expectation
The complete algorithm

**PageRank-Clustering**($G,k,\varepsilon$):

1. For each vertex $v$, compute $\text{pr}(\alpha,v)$

2. For each root of $\Phi'(\alpha)$:
   
   If $\Phi(\alpha) \leq \varepsilon$ and $k \geq \Psi(\alpha) - 2 - \varepsilon$:
   
   1. Randomly select $c \log n$ sets of $k$ vertices from $\pi$
   2. For each set $S = \{v_1, \ldots, v_k\}$, randomly select $c_i$ according to $\text{pr}(\alpha,v_i)$
   3. If $|\mu(C) - \Phi(\alpha)| \leq \varepsilon$ and $|\psi_\alpha(C) - \Psi(\alpha)| \leq \varepsilon$ return $C = \{c_1, \ldots, c_k\}$ and assign nodes to nearest center

3. If clusters haven’t been found, return nothing
Proposal outline

Background on diffusion and clustering
  Introductory example: network epidemics

Preliminary work
  Global graph partitioning using PageRank
  Graph drawing using PageRank

Ongoing work
  Planted partition model
  Spectral analysis of truncated communication networks

Future work
  Hierarchical clustering
Visualizing graph clusters using PageRank

Previously: algorithm to find graph clusters
Next: an algorithm to draw clustered graphs

Hard to show local structure
Graph drawing algorithm

1. Compute personalized PageRank for centers
2. Ignore edges of the original $G$
3. Simulate springs between node pairs
   - For a center and non-center, force inversely proportional to personalized PageRank
   - For two centers, a strong repelling force
4. Lay out nodes in 2D space by optimizing the potential energy [Kamada, Kawai]
5. Draw original edges of $G$
Graph drawing example

Social network of dolphins [Newman, Girvan ‘04]
2 clusters and outliers
More complex example

• NCAA Division I football [Girvan, Newman ‘02]
• Teams are organized into *conferences*
  – Drawing highlights several of them
Proposal outline

Background on diffusion and clustering
   Introductory example: network epidemics

Preliminary work
   Global graph partitioning using PageRank
   Graph drawing using PageRank

Ongoing work
   Planted partition model
   Spectral analysis of truncated communication networks

Future work
   Hierarchical clustering
Planted partition model

\( n \) nodes, \( k \) clusters

Edge probabilities:
- \( p \) for same cluster
- \( q \) for different clusters

\( G = (V, E) \) is an instantiation

Q: Can we recover clusters from \( G \)?
Extended planted partition model

Assign a vertex weights $d_u$

New edge probabilities:

$\begin{align*}
    d_u p d_v & \text{ for same cluster} \\
    d_u q d_v & \text{ for different clusters}
\end{align*}$

Can we still recover the clusters?
Recovering clusters for “vanilla” model

Each vertex: a vector in a matrix of data
Take SVD
Project vertices into the subspace spanned by the top $k$ eigenvectors
Conditions on $(p - q)$, cluster sizes, graph size
Recovering clusters for extended model

If weights are known: SVD still works
What if weights are not given?
How to normalize data matrix?
Proposal outline

Background on diffusion and clustering
  Introductory example: network epidemics

Preliminary work
  Global graph partitioning using PageRank
  Graph drawing using PageRank

Ongoing work
  Planted partition model
  Spectral analysis of truncated communication networks

Future work
  Hierarchical clustering
Communication network datasets

Subsets of Internet communication networks
  121-10152 nodes
  456-28638 edges

Created using freely available tools

Drawback: cannot see topology past a certain point
  Creates an artificial boundary of degree-1 nodes
Bottlenecks in network datasets

**Goal**: find bottlenecks

**Problem**: “Bag of whiskers”  
[Leskovec, Lang, Dasgupta, Mahoney ‘09]  
Best cut of a given size often cuts small segments near the boundary

**Solution**: use Dirichlet boundary conditions
Cuts in communication networks

Dirichlet cut

Traditional cut
Cuts in communication networks

Dirichlet cut

Traditional cut
Proposal outline

Background on diffusion and clustering
  Introductory example: network epidemics

Preliminary work
  Global graph partitioning using PageRank
  Graph drawing using PageRank

Ongoing work
  Planted partition model
  Spectral analysis of truncated communication networks

Future work
  Hierarchical clustering
Hierarchical clustering

Find a hierarchy of clusters in a graph

Alex’s social network

UCSD

UCSD CSE

Room 4232

UCSD Music

Cornell
Some questions

Can we use previous clustering algorithm to find hierarchical clusters?

Use the jumping constant $\alpha$ to find sub-clusters

Can we propose and analyze a hierarchical planted partition model?
Acknowledgments

Network epidemics
  Joint work with Fan Chung and Paul Horn
Global graph partitioning and visualization
  Joint work with Fan Chung
Trust-based ranking
  Joint work with Fan Chung and Wensong Xu
Planted partitions
  Joint work with Kamalika Chaudhuri and Fan Chung
Spectral analysis of communication networks
  Joint work with Iraj Saniee, Onuttom Narayan, and Matthew Andrews
Hypergraph coloring games and voter models
  Joint work with Fan Chung
Questions?