Networks: Modeling Interactions

Jure Leskovec (@jure)
Stanford University

KDD Summer School, Beijing, August 2012
Models for networks

- **Graph:**
  - Kronecker graphs

- **Graph + Node attributes:**
  - MAG model

- **Graph + Edge attributes:**
  - Signed networks

- **Link Prediction/Recommendation:**
  - Supervised Random Walks
Many networks come with:

- The graph (wiring diagram)
- Node/edge metadata (attributes/features)

How to generate realistic looking graphs?

- 1: Kronecker Graphs

How to model networks with node attributes?

- 2: Multiplicative Attributes Graph (MAG) model

How to model networks with edge attributes?

- 3: Networks of Positive and Negative Edges

How to predict/recommend new edges?

- 4: Supervised Random Walks
Stanford Large Network Dataset Collection

- http://snap.stanford.edu
- 60+ large networks:
  - Social network, Geo-location networks, Information networks, Evolving networks, Citation networks, Internet networks, Amazon, Twitter, ...

Stanford Network Analysis Platform (SNAP):

- http://snap.stanford.edu
- C++ Library for massive networks
- Has no problem working with 1B nodes, 10B edges
Want to learn more? (2)

- Stanford CS224W: Social and Information Networks Analysis
  - http://cs224w.stanford.edu
  - Graduate course on topics discusses today
  - Slides, homeworks, readings, data, ...

- My webpage
  - http://cs.stanford.edu/~jure/
  - Videos of talks and tutorials

- Twitter: @jure
Kronecker Graphs Model

Reliably models the global network structure using only 4 parameters!
The Setting

- Want to have a model that can generate a realistic networks with realistic growth:
  - **Static Patterns**
    - Power Law Degree Distribution
    - Small Diameter
    - Power Law Eigenvalue and Eigenvector Distribution
  - **Temporal Patterns**
    - Densification Power Law
    - Shrinking/Constant Diameter
- **For Kronecker graphs:**
  1) *analytically tractable* (i.e., prove power-laws, etc.)
  2) *statistically interesting* (i.e., fit it to real data)
Classical example:
Heavy-tailed degree distributions

\[ p_k \sim k^{-\alpha} \]

Flickr social network
\[ n = 584,207, m = 3,555,115 \]

Scale free networks
many hub nodes
Scaling of Network Properties

- How do network properties scale with the size of the network?

\[ E(t) \propto N(t)^\alpha \]

\( a = 1.6 \)

- Densification
  - Average degree increases

- Shrinking diameter
  - Path lengths get shorter

[Citations]

[Leskovec et al. KDD 05]
How can we think of network structure recursively?

\[
K_1 = \begin{pmatrix}
a & b \\
c & d \\
\end{pmatrix}
\]
Kronecker graphs:

- A recursive model of network structure

Initiator

\[
K_1 = \begin{pmatrix}
1 & 1 & 0 \\
1 & 1 & 1 \\
0 & 1 & 1
\end{pmatrix}
\]

\[
K_2 = K_1 \otimes K_1
\]

\[
K_1 = \begin{pmatrix}
K_1 & K_1 & 0 \\
K_1 & K_1 & K_1 \\
0 & K_1 & K_1
\end{pmatrix}
\]

(3x3) \rightarrow (9x9) \rightarrow (27x27)
Kronecker Graphs

- **Kronecker product** of matrices $A$ and $B$ is given by

$$C = A \otimes B = \begin{pmatrix}
    a_{1,1}B & a_{1,2}B & \ldots & a_{1,m}B \\
    a_{2,1}B & a_{2,2}B & \ldots & a_{2,m}B \\
    \vdots & \vdots & \ddots & \vdots \\
    a_{n,1}B & a_{n,2}B & \ldots & a_{n,m}B
\end{pmatrix}_{N*K \times M*L}$$

- **Define**: Kronecker product of two graphs is a Kronecker product of their adjacency matrices

- **Kronecker graph**: a growing sequence of graphs by iterating the Kronecker product

$$K_1^{[k]} = K_k = \underbrace{K_1 \otimes K_1 \otimes \ldots \otimes K_1}_{k \text{ times}} = K_{k-1} \otimes K_1$$
Kronecker Initiator Matrices

Initiator $K_1$

$K_1$ adjacency matrix

$K_3$ adjacency matrix
Properties of deterministic Kronecker graphs (can be proved!)

- Properties of static networks:
  - Power-Law like Degree Distribution
  - Power-Law eigenvalue and eigenvector distribution
  - Constant Diameter

- Properties of evolving networks:
  - Densification Power Law
  - Shrinking/Stabilizing Diameter

- Can we make the model stochastic?
Create $N_1 \times N_1$ probability matrix $\Theta_1$

Compute the $i^{th}$ Kronecker power $\Theta_i$

For each entry $p_{uv}$ of $\Theta_k$ include an edge $(u, v)$ with probability $p_{uv}$

$\Theta_1 = \begin{bmatrix} 0.5 & 0.2 \\ 0.1 & 0.3 \end{bmatrix}$

$\Theta_2 = \Theta_1 \otimes \Theta_1 = \begin{bmatrix} 0.25 & 0.10 & 0.10 & 0.04 \\ 0.05 & 0.15 & 0.02 & 0.06 \\ 0.05 & 0.02 & 0.15 & 0.06 \\ 0.01 & 0.03 & 0.03 & 0.09 \end{bmatrix}$

Instance matrix $K_2$

Probability of edge $p_{ij}$

Kronecker multiplication

flip biased coins
Given a graph $G$

What is the parameter matrix $\Theta$?

Find $\Theta$ that maximizes $P(G|\Theta)$

\[
P(G|\Theta) = \prod_{(u,v) \in G} \Theta_k[u,v] \prod_{(u,v) \notin G} (1 - \Theta_k[u,v])
\]
Maximum likelihood estimation

\[ \arg \max_{\Theta_1} P(\Theta_1) \]

Naïve estimation takes \( O(N!N^2) \):
- \( N! \) for different node labelings:
- \( N^2 \) for traversing graph adjacency matrix

Do gradient descent

We estimate the model in \( O(E) \)
Epinions (n=76k, m=510k)

- **Real** and Kronecker are very close:

\[
\Theta_1 = \begin{pmatrix} 0.99 & 0.54 \\ 0.49 & 0.13 \end{pmatrix}
\]

(a) In-Degree

(b) Out-degree

(c) Triangle participation

(d) Hop plot

(e) Scree plot

(f) “Network” value
The MAG Model

- For networks with node attributes
- Can do power-law and log-normal degrees
When modeling networks, what would we like to know?

- How to model the links in the network
- How to model the interaction of node attributes/properties and the network structure

Goal:

- A family of models of networks with node attributes
- The models are:
  1) Analytically tractable (prove network properties)
  2) Statistically interesting (can be fit to real data)
Our Approach: Node attributes

- Each node has a set of categorical attributes
  - Gender: Male, Female
  - Home country: US, Canada, Russia, etc.

- How do node attributes influence link formation?
  - Example: MSN Instant Messenger [Leskovec & Horvitz ’08]

```
<table>
<thead>
<tr>
<th>u's gender</th>
<th>v's gender</th>
<th>Link probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>FEMALE</td>
<td>FEMALE</td>
<td>0.3</td>
</tr>
<tr>
<td></td>
<td>MALE</td>
<td>0.7</td>
</tr>
<tr>
<td>MALE</td>
<td>FEMALE</td>
<td>0.7</td>
</tr>
<tr>
<td></td>
<td>MALE</td>
<td>0.3</td>
</tr>
</tbody>
</table>
```
Let the values of the \textit{i-th attribute} for node \(u\) and \(v\) be \(a_i(u)\) and \(a_i(v)\)

- \(a_i(u)\) and \(a_i(v)\) can take values \(\{0, \ldots, d_i - 1\}\)

**Question:** How can we capture the influence of the attributes on link formation?

**Key:** \textit{Attribute link-affinity matrix} \(\Theta\)

\[
\begin{array}{ccc}
\theta[0, 0] & \theta[0, 1] \\
\theta[1, 0] & \theta[1, 1] \\
\end{array}
\]

\[
P(u, v) = \Theta[a_i(u), a_i(v)]
\]

Each entry captures the \textit{affinity of a link} between two nodes associated with the attributes of them.
Link-Affinity Matrices offer *flexibility* in modeling the network structure:

- **Homophily**: love of the *same*
  - e.g., political views, hobbies

- **Heterophily**: love of the *opposite*
  - e.g., genders

- **Core-periphery**: love of the *core*
  - e.g. extrovert personalities
From Attributes to Links

- **How do we combine the effects of multiple attributes?**
  - We *multiply the probabilities* from all attributes

\[
\alpha(u) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \alpha(v) = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}
\]

\[
\Theta_i = \begin{bmatrix} \alpha_1 & \beta_1 \\ \beta_1 & \gamma_1 \end{bmatrix} \begin{bmatrix} \alpha_2 & \beta_2 \\ \beta_2 & \gamma_2 \end{bmatrix} \begin{bmatrix} \alpha_3 & \beta_3 \\ \beta_3 & \gamma_3 \end{bmatrix} \begin{bmatrix} \alpha_4 & \beta_4 \\ \beta_4 & \gamma_4 \end{bmatrix}
\]

\[
P(u, v) = \alpha_1 \times \beta_2 \times \gamma_3 \times \alpha_4
\]
Multiplcative Attribute Graph

- **MAG model** \( M(n, l, A, \Theta) \):
  - A network contains \( n \) nodes
  - Each node has \( l \) categorical attributes
  - \( A = [a_i(u)] \) represents the \( i \)-th attribute of node \( u \)
  - Each attribute can take \( d_i \) different values
  - Each attribute has a \( d_i \times d_i \) link-affinity matrix \( \Theta_i \)
  - Edge probability between nodes \( u \) and \( v \)

\[
P(u, v) = \prod_{i=1}^{l} \Theta_i[a_i(u), a_i(v)]
\]
MAG can model global network structure!

MAG generates networks with similar properties as found in real-world networks:
- Unique giant connected component
- Densification Power Law
- Small diameter
- Heavy-tailed degree distribution
  - Either log-normal or power-law
Theorem 1: A unique giant connected component of size $\theta(n)$ exists in $M(n, l, \mu, \Theta)$ w.h.p. as $n \to \infty$ if

$$P(a_i(u) = 1) = \mu$$

$$\left[ (\mu \alpha + (1 - \mu) \beta)^\mu (\mu \beta + (1 - \mu) \gamma)^{1 - \mu} \right]^\rho \geq \frac{1}{2}$$

Simulation:
Theorem 3: $M(n, l, \mu, \Theta)$ follows a log-normal degree distribution as $n \to \infty$ for some constant $R$

$$\ln p_k \sim \mathcal{N}\left(\ln(n(\mu \beta + (1-\mu)\gamma)^l) + l\mu \ln R + \frac{1}{2}l\mu(1-\mu)(\ln R)^2, \ l\mu(1-\mu)(\ln R)^2\right)$$

if the network has a giant connected component.
Theorem 4: MAG follows a power-law degree distribution

\[ p_k \propto k^{-\delta-0.5} \]

for some \( \delta > 0 \)

when we set \( \frac{\mu_i}{1-\mu_i} = \left( \frac{\mu_i \alpha_i + (1-\mu_i) \beta_i}{\mu_i \beta_i + (1-\mu_i) \gamma_i} \right)^{-\delta} \)

Simulation:
Fitting the MAG model

- MAG model is also statistically “interesting”
- Estimate model parameters from the data
  - **Given:**
    - Links of the network
  - **Estimate:**
    - Node attributes
    - Link-affinity matrices
- Formulate as a maximum likelihood problem
- Solve it using variational EM

\[
\alpha(u) = [...]
\]

\[
\Theta_i = \begin{bmatrix}
0.9 & 0.1 \\
0.1 & 0.8
\end{bmatrix}
\]
Fitting the MAG model

- **Edge probability:**
  \[ P(u, v) = \prod_{i=1}^{l} \Theta_i[a_i(u), a_i(v)] \]

- **Network likelihood:**
  \[ P(G|A, \Theta) = \prod_{G_{uv}=1} P(u, v) \cdot \prod_{G_{uv}=0} 1 - P(u, v) \]
  - G ... graph adjacency matrix
  - A ... matrix of node attributes
  - \( \Theta \) ... link-affinity matrices

- **Want to solve:**
  \[ \arg \max_{A, \Theta} P(G|A, \Theta) \]
Variational EM

Node attribute estimation

\[ P(A|G, \Theta) \]

Model parameter estimation

\[ \Theta \]

E-step: Variational inference

M-step: Gradient method

[UAI. '11]
Experiments: Global Structure

- LinkedIn network
  - When it was super-young (4k nodes, 10k edges)
- Fit using 11 latent binary attributes per node
Experiments: AddHealth

- **Case study: AddHealth**
  - School friendship network
  - Largest network: 457 nodes, 2259 edges
  - Over 70 school-related attributes for each student
  - Real features are selected in the greedy way to maximize the likelihood of MAG model
    - We fit only $\Theta$ (since $A$ is given): $\arg\max_{\Theta} P(G, A|\Theta)$
    - 7 features

- **Model accurately fits the network structure**
Experiments: AddHealth

- **Most important features for tie creation**

<table>
<thead>
<tr>
<th>Affinity matrix</th>
<th>Attribute description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[0.572 0.146; 0.146 0.999]</td>
<td>School year (0 if ≥ 2)</td>
</tr>
<tr>
<td>[0.845 0.332; 0.332 0.816]</td>
<td>Highest level math (0 if ≥ 6)</td>
</tr>
<tr>
<td>[0.788 0.377; 0.377 0.784]</td>
<td>Cumulative GPA (0 if ≥ 2.65)</td>
</tr>
<tr>
<td>[0.999 0.246; 0.246 0.352]</td>
<td>AP/IB English (0 if taken)</td>
</tr>
<tr>
<td>[0.794 0.407; 0.407 0.717]</td>
<td>Foreign language (0 if taken)</td>
</tr>
</tbody>
</table>
Models of Networks with Signed Edges

- How people determine friends and foes?
- Predict friend vs. foe with 90% accuracy
So far we viewed links as positive but links can also be negative

**Question:**
- How do edge signs and network interact?
- How to model and predict edge signs?

**Applications:**
- Friend recommendation
  - Not just whether you know someone but what do you think of them
Networks with Explicit Signs

- Each link $A \rightarrow B$ is explicitly tagged with a sign:
  - **Epinions**: Trust/Distrust
    - Does A trust B’s product reviews? (only positive links are visible)
  - **Wikipedia**: Support/Oppose
    - Does A support B to become Wikipedia administrator?
  - **Slashdot**: Friend/Foe
    - Does A like B’s comments?
  - Other examples:
    - Sentiment analysis of the communication
Start with intuition [Heider ’46]:
- Friend of my friend is my friend
- Enemy of enemy is my friend
- Enemy of friend is my enemy

Look at connected triples of nodes:

Consistent with “friend of a friend” or “enemy of the enemy” intuition

Inconsistent with the “friend of a friend” or “enemy of the enemy” intuition
Theory of Status

- **Status theory** [Davis-Leinhardt ‘68, Leskovec et al. ‘10]
  - Link $A \rightarrow B$ means: $B$ has higher status than $A$
  - Link $A \rightarrow B$ means: $B$ has lower status than $A$
  - Signs/directions of links to $X$ make a prediction

- **Status and balance make different predictions:**

  ![Diagram](image)

  - Balance: +
  - Status: –
  - Balance: +
  - Status: –
  - Balance: –
  - Status: –
Consider networks as undirected

Compare frequencies of signed triads in real and shuffled data

- 4 triad types $t$:

Real data

Shuffled data

Surprise value for triad type $t$:

- Number of std. deviations by which number of occurrences of triad $t$ differs from the expected number in shuffled data
Surprise values: 

*i.e.*, z-score
(deivation from random measured in the number of std. devs.)

<table>
<thead>
<tr>
<th></th>
<th>Triad</th>
<th>Epin</th>
<th>Wiki</th>
<th>Slashdot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Balanced</td>
<td>+ +</td>
<td>1,881</td>
<td>380</td>
<td>927</td>
</tr>
<tr>
<td>Unbalanced</td>
<td>- -</td>
<td>249</td>
<td>289</td>
<td>-175</td>
</tr>
<tr>
<td>Unbalanced</td>
<td>+ +</td>
<td>-2,105</td>
<td>-573</td>
<td>-824</td>
</tr>
<tr>
<td>Unbalanced</td>
<td>- -</td>
<td>288</td>
<td>11</td>
<td>-9</td>
</tr>
</tbody>
</table>

Observations:

- Strong signal for balance
- Epinions and Wikipedia agree on all types
- Consistency with Davis’s [‘67] weak balance
Evolving Directed Networks

- Links are directed and created over time
- To compare balance and status we need to formalize two issues:
  - Links are embedded in triads which provide contexts for signs
  - Users are heterogeneous in their linking behavior
- **Link contexts:**
  - A contextualized link is a triple \((A,B;X)\) such that directed \(A-B\) link forms after there is a two-step semi-path \(A-X-B\)
  - \(A-X\) and \(B-X\) links can have either direction and either sign: 16 possible types
Different users make signs differently:

- **Generative baseline**: (frac. of + given by A)
- **Receptive baseline**: (frac. of + received by B)

How do different link contexts cause users to deviate from baselines?

**Surprise**: How much behavior of A/B deviates from baseline when they are in context.

Vs.
Status: Two Examples

- Two basic examples:

More **negative** than gen. baseline of A
More **negative** than rec. baseline of B
Out of 16 triad contexts

- **Generative surprise:**
  - Balance-consistent: 8
  - Status-consistent: 14
  - Both mistakes of status happen when A and B have low status

- **Receptive surprise:**
  - Status-consistent: 13
  - Balance-consistent: 7
Edge sign prediction problem
- Given a network and signs on all but one edge, predict the missing sign

Machine Learning formulation:
- Predict sign of edge \((u,v)\)
- Class label:
  - +1: positive edge
  - -1: negative edge
- Learning method:
  - Logistic regression

\[
P(+|x) = \frac{1}{1 + e^{-(b_0 + \sum_i^n b_i x_i)}}
\]

Dataset:
- Original: 80% +edges
- Balanced: 50% +edges

Evaluation:
- Accuracy and ROC curves

Features for learning:
- Next slide
For each edge \((u,v)\) create features:

- **Triad counts (16):**
  - Counts of signed triads edge \(u \rightarrow v\) takes part in

- **Degree (7 features):**
  - Signed degree:
    - \(d^+_{\text{out}}(u), d^-_{\text{out}}(u), d^+_{\text{in}}(v), d^-_{\text{in}}(v)\)
  - Total degree:
    - \(d_{\text{out}}(u), d_{\text{in}}(v)\)
  - Embeddedness of edge \((u,v)\)
Edge Sign Prediction

- **Error rates:**
  - Epinions: 6.5%
  - Slashdot: 6.6%
  - Wikipedia: 19%

- Signs can be modeled from network structure alone

- Performance degrades for less embedded edges

- Wikipedia is harder:
  - Votes are publicly visible
Generalization

- Do people use these very different linking systems by obeying the same principles?
  - Generalization of results across the datasets?
    - Train on row “dataset”, predict on “column”

<table>
<thead>
<tr>
<th>All23</th>
<th>Epinions</th>
<th>Slashdot</th>
<th>Wikipedia</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epinions</td>
<td>0.9342</td>
<td>0.9289</td>
<td>0.7722</td>
</tr>
<tr>
<td>Slashdot</td>
<td>0.9249</td>
<td>0.9351</td>
<td>0.7717</td>
</tr>
<tr>
<td>Wikipedia</td>
<td>0.9272</td>
<td>0.9260</td>
<td>0.8021</td>
</tr>
</tbody>
</table>

- Nearly **perfect generalization** of the models even though networks come from very different applications
Signed networks provide insight into how social computing systems are used:
- Status vs. Balance

Sign of relationship can be reliably predicted from the local network context
- ~90% accuracy sign of the edge
More evidence that networks are globally organized based on status

People use signed edges consistently regardless of particular application
  - Near perfect generalization of models across datasets

Many further directions:
  - Status difference [ICWSM ‘10]
Final Remarks: Status

- Status difference on Wikipedia:

![Graph showing status difference on Wikipedia]
Supervised Random Walks

- Learning to rank nodes on a graph
- For recommending people you may know
How to learn to predict/recommend new friends in networks?

- Facebook People You May Know
- Let’s look at the data:
  - 92% of new friendships on FB are friend-of-a-friend
  - More common friends helps
Link Prediction: Challenges

- How to learn models that combine:
  - Network connectivity structure
  - Node/edge metadata

- Class imbalance:
  - You only have 1,000 (out of 800M possible) friends on Facebook
  - Even if we limit prediction to friends-of-friends a typical Facebook person has 20,000 FoFs
Want to predict new Facebook friends!

Combining link information and metadata:
- PageRank is great with network structure
- Logistic regression is great for classification

Let's combine the two!

Class imbalance:
- Formulate prediction task a ranking problem

Supervised Random Walks
- Supervised learning to rank nodes on a graph using PageRank
Recommend a list of possible friends

Supervised machine learning setting:

- **Training example:**
  - For every node $s$ have a list of nodes she will create links to $\{v_1, \ldots, v_k\}$
  - *E.g.*, use FB network from May 2011 and $\{v_1, \ldots, v_k\}$ are the new friendships you created since then

- **Problem:**
  - For a given node $s$ learn to rank nodes $\{v_1, \ldots, v_k\}$ higher than other nodes in the network

- **Supervised Random Walks** based on work by Agarwal & Chakrabarti
How to combine node/edge attributes and the network structure?

- Learn a strength of each edge based on:
  - Profile of user $u$, profile of user $v$
  - Interaction history of $u$ and $v$
- Do a PageRank-like random walk from $s$ to measure the “proximity” between $s$ and other nodes
- Rank nodes by their “proximity” (i.e., visiting prob.)
Let $s$ be the center node

Let $f_w(u,v)$ be a function that assigns a strength to each edge:

$$a_{uv} = f_w(u,v) = \exp(-w^T \Psi_{uv})$$

- $\Psi_{uv}$ is a feature vector
  - Features of nodes $u$ and $v$
  - Features of edge $(u,v)$

- $w$ is the parameter vector we want to learn

Do Random Walk with Restarts from $s$ where transitions are according to edge strengths

How to learn $f_w(u,v)$?
Personalized PageRank

- Random walk transition matrix:

\[ Q'_{uv} = \begin{cases} \frac{a_{uv}}{\sum_w a_{uw}} & \text{if } (u, v) \in E, \\ 0 & \text{otherwise} \end{cases} \]

- PageRank transition matrix:

\[ Q_{ij} = (1 - \alpha)Q'_{ij} + \alpha \mathbf{1}(j = s) \]

  - with prob. \( \alpha \) jump back to \( s \)

- Compute PageRank vector: \( p = p^T Q \)

- Rank nodes by \( p_u \)
The Optimization Problem

- Each node $u$ has a score $p_u$
- Destination nodes $D = \{v_1, \ldots, v_k\}$
- No-link nodes $L = \{\text{the rest}\}$
- What do we want?
  - Want to find $w$ such that $p_l < p_d$

\[
\min_w F(w) = \|w\|^2
\]

such that
\[
\forall d \in D, l \in L : p_l < p_d
\]

- Hard constraints, make them soft
Want to minimize:

\[
\min_w F(w) = ||w||^2 + \lambda \sum_{ld} h(p_l - p_d)
\]

- **Loss:** \( h(x) = 0 \) if \( x < 0 \), \( x^2 \) else
How to minimize $F$?

$$\min_w F(w) = ||w||^2 + \lambda \sum_{l,d} h(p_l - p_d)$$

- $p_l$ and $p_d$ depend on $w$
  - Given $w$ assign edge weights $a_{uv} = f_w(u,v)$
  - Using transition matrix $Q = [a_{uv}]$
    compute PageRank scores $p_u$
  - Rank nodes by the PageRank score

- Want to find $w$ such that $p_l < p_d$
### Gradient Descent

- **How to minimize F?**
  \[
  \min_w F(w) = \|w\|^2 + \lambda \sum_{l,d} h(p_l - p_d)
  \]
- **Take the derivative!**
  \[
  \frac{\partial F}{\partial w} = 2w + \sum_{l,d} \frac{\partial h(p_l - p_d)}{\partial w}
  \]
  \[
  = 2w + \sum_{l,d} \frac{\partial h(\delta_{ld})}{\partial \delta_{ld}} \left( \frac{\partial p_l}{\partial w} - \frac{\partial p_d}{\partial w} \right)
  \]

- **We know:**
  \[
  p = p^T Q \quad \text{i.e.} \quad p_u = \sum_j p_j Q_{ju}
  \]

- **So:**
  \[
  \frac{\partial p_u}{\partial w} = \sum_j Q_{ju} \frac{\partial p_j}{\partial w} + p_j \frac{\partial Q_{ju}}{\partial w}
  \]

Solve using power iteration!
To optimize $F$, use gradient based method:

- Pick a random starting point $w_0$
- Compute the personalized PageRank vector $p$
- Compute gradient with respect to weight vector $w$
- Update $w$
  - Optimize using quasi-Newton method
Facebook Iceland network
- 174,000 nodes (55% of population)
- Avg. degree 168
- Avg. person added 26 new friends/month

For every node $s$:
- Positive examples:
  - $D = \{ \text{new friendships of } s \text{ created in Nov '09} \}$
- Negative examples:
  - $L = \{ \text{other nodes } s \text{ did not create new links to} \}$
- Limit to friends of friends
  - on avg. there are 20k FoFs (max 2M)!
Node and edge features:
- Node:
  - Age, Gender, Degree
- Edge:
  - Edge age, Communication, Profile visits, Co-tagged photos

Baselines:
- Decision trees and logistic regression:
  - Above features + 10 network features (PageRank, common friends, ...)

Evaluation:
- AUC and Precision at Top20
### Results: Facebook Iceland

- **Facebook:** predict future friends
  - Adamic-Adar already works great
  - Logistic regression also strong
  - SRW gives slight improvement

<table>
<thead>
<tr>
<th>Learning Method</th>
<th>AUC</th>
<th>Prec@20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Walk with Restart</td>
<td>0.81725</td>
<td>6.80</td>
</tr>
<tr>
<td>Adamic-Adar</td>
<td>0.81586</td>
<td>7.35</td>
</tr>
<tr>
<td>Common Friends</td>
<td>0.80054</td>
<td>7.35</td>
</tr>
<tr>
<td>Degree</td>
<td>0.58535</td>
<td>3.25</td>
</tr>
<tr>
<td>DT: Node features</td>
<td>0.59248</td>
<td>2.38</td>
</tr>
<tr>
<td>DT: Network features</td>
<td>0.76979</td>
<td>5.38</td>
</tr>
<tr>
<td>DT: Node+Network</td>
<td>0.76217</td>
<td>5.86</td>
</tr>
<tr>
<td>DT: Path features</td>
<td>0.62836</td>
<td>2.46</td>
</tr>
<tr>
<td>DT: All features</td>
<td>0.72986</td>
<td>5.34</td>
</tr>
<tr>
<td>LR: Node features</td>
<td>0.54134</td>
<td>1.38</td>
</tr>
<tr>
<td>LR: Network features</td>
<td>0.80560</td>
<td>7.56</td>
</tr>
<tr>
<td>LR: Node+Network</td>
<td>0.80280</td>
<td>7.56</td>
</tr>
<tr>
<td>LR: Path features</td>
<td>0.51418</td>
<td>0.74</td>
</tr>
<tr>
<td>LR: All features</td>
<td>0.81681</td>
<td>7.52</td>
</tr>
<tr>
<td>SRW: one edge type</td>
<td>0.82502</td>
<td>6.87</td>
</tr>
<tr>
<td>SRW: multiple edge types</td>
<td>0.82799</td>
<td>7.57</td>
</tr>
</tbody>
</table>
Results: Co-authorship

- Arxiv Hep-Ph collaboration network:
  - Poor performance of unsupervised methods
  - Logistic regression and decision trees don’t work to well
  - SRW gives 10% boost in Prec@20

<table>
<thead>
<tr>
<th>Learning Method</th>
<th>AUC</th>
<th>Prec@20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Walk with Restart</td>
<td>0.63831</td>
<td>3.41</td>
</tr>
<tr>
<td>Adamic-Adar</td>
<td>0.60570</td>
<td>3.13</td>
</tr>
<tr>
<td>Common Friends</td>
<td>0.59370</td>
<td>3.11</td>
</tr>
<tr>
<td>Degree</td>
<td>0.56522</td>
<td>3.05</td>
</tr>
<tr>
<td>DT: Node features</td>
<td>0.60961</td>
<td>3.54</td>
</tr>
<tr>
<td>DT: Network features</td>
<td>0.59302</td>
<td>3.69</td>
</tr>
<tr>
<td>DT: Node+Network</td>
<td>0.63711</td>
<td>3.95</td>
</tr>
<tr>
<td>DT: Path features</td>
<td>0.56213</td>
<td>1.72</td>
</tr>
<tr>
<td>DT: All features</td>
<td>0.61820</td>
<td>3.77</td>
</tr>
<tr>
<td>LR: Node features</td>
<td>0.64754</td>
<td>3.19</td>
</tr>
<tr>
<td>LR: Network features</td>
<td>0.58732</td>
<td>3.27</td>
</tr>
<tr>
<td>LR: Node+Network</td>
<td>0.64644</td>
<td>3.81</td>
</tr>
<tr>
<td>LR: Path features</td>
<td>0.67237</td>
<td>2.78</td>
</tr>
<tr>
<td>LR: All features</td>
<td>0.67426</td>
<td>3.82</td>
</tr>
<tr>
<td>SRW: one edge type</td>
<td>0.69996</td>
<td>4.24</td>
</tr>
<tr>
<td>SRW: multiple edge types</td>
<td>0.71238</td>
<td>4.25</td>
</tr>
</tbody>
</table>
References


- **Multiplicative Attribute Graph Model of Real-World Networks** by M. Kim, J. Leskovec. *Internet Mathematics* 8(1-2) 113--160, 2012.

- **Modeling Social Networks with Node Attributes using the Multiplicative Attribute Graph Model** by M. Kim, J. Leskovec. *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2011.


References


THANKS!
http://snap.stanford.edu