CS910: Foundations of Data Analytics

Graham Cormode
G.Cormode@warwick.ac.uk

Social Network Analysis
Outline

♦ Introduction to social networks and social network analysis
♦ See how to model social networks as graph structures
♦ Metrics to identify important entities in social networks
♦ Classification and prediction in social networks
♦ Identifying important graph nodes for ranking
♦ Other data mining problems on social networks

♦ Core Reading:
  – Tutorial on Social Media Analytics, Jure Leskovec
    http://snap.stanford.edu/proj/socmedia-kdd/
  – “The Link Prediction Problem for Social Networks”
    J. Kleinberg and D. Liben-Nowell, CIKM 2003
“Social Networks” refers to the pattern of interactions formed between individuals
- E.g. the social network of a university, an office, a sports club...

“Social Networks” studied by sociologists for decades
- Milgram’s “six degrees of separation” experiments (1960s)
- Zachary’s karate club study (1970): links between 34 club members
- Dunbar’s number: ~150 (1990s)

Around 15 years ago, something changed...
Online Social Networks

- Online social networks started:
  - Friendster (2002-11)
  - Myspace (2003-)
  - LinkedIn (2003-)
  - Orkut (2004-14)
  - Facebook (2004-)
  - Twitter (2006-)
  - Instagram (2010-)

- Allowed users to make their social network explicit
  - Represented within computer systems
  - Quickly grew to millions of users, hundreds of millions of links
  - Social Network Analysis is now primarily a computational study
Other social networks

- Electronic social network analysis predates online social networks
  - Phone call patterns studied to detect fraud (1990s)
  - Email traffic analyzed for understanding businesses
    - The famous “enron dataset” from 2004
  - World-wide web link data analysis
    - WWW Conference since 1994
  - Blog data analysis: livejournal, blogger, wordpress...
  - Citation data from scientific papers

- Aside: social networks distinct from social media
  - Social media: text (tweets), comments, photos, videos
  - Social media is often shared via social networks
Social networks: graph model

- Much work models social networks as a graph

- Each individual **entity** (user, blog) is a **node** (vertex)
- **Links** (arcs, edges) between nodes indicate some connection
  - An explicit “friendship” link in facebook (symmetric)
  - A “follow” relation in twitter (directed)
Working with Social Network Data

- Social network data is fairly widely available to study
  - Stanford Large Datasets Collection
    http://snap.stanford.edu/data/index.html
    Curated by Jure Leskovec

- Collect your own:
  - Twitter API, but limit on what can be sampled
  - Several social networks let you pay for access to data
  - Many social networks try to detect and limit data scraping
Properties of social networks

- Concepts from graph theory help understand social networks
  - **Node degree**: the number of links that a node has
  - **In-degree (Out-degree)**: the number of directed links pointing in (out)
  - **Distance**: minimum number of hops between a given pair of nodes
  - **Diameter**: the maximum distance between any pair of nodes

Degree of node (7) is 6
Distance(1,8) is 2
Diameter is 4, realized by (1,10)
Clustering Coefficient

♦ Most social networks look different to “random graphs”
  – Erdos-Renyi random graphs: each edge present with probability $p$
♦ Much more likely to see an edge between a pair of neighbouring nodes than by chance
  – These are called triangles
  – I.e. two of my friends are also friends
♦ Clustering coefficient measures the fraction of triangles
  – $(\text{number of triangles})/(\text{number of pairs of common neighbours})$
♦ Estimated to be around 0.25 on facebook
  – Would be orders of magnitude lower if edges picked randomly
  – Computationally expensive task to count triangles
♦ Local clustering coefficient: triangles incident on a particular node
Social Network Analysis

(Online) Social Networks are a topic of much study
♦ Inherent interest: what are they, how do they work?
♦ As a surrogate for human behaviour
♦ To try to influence activity within them (advertising)
♦ To study activities transacted through networks
   – Look at influence on finance, news, ...
   – Spot new trends, topics about to “go viral”
   – Find evidence of “bad guys” in social networks
Task 1: Link Prediction on Graphs

- **A basic problem**: how to predict which new links will be formed?
- Can try to model this as classification
  - **Input**: existing graph and links
  - **Predict**: for a given pair of nodes, will a link be formed?
  - Can use features of the node pairs to train a classifier
    - E.g. in a social network, features are demographics and interests
- Link prediction is used e.g. for **recommending** friends to add
  - **Question**: how is link prediction different to recommender systems?
- Directly using a classifier to predict **ignores the graph structure**
  - People are unlikely to add friends just based on interests etc.
  - Instead, try to use proximity in the graph to predict links
Common Neighbours

- **Simplest model**: predict that common neighbours will link
  - If \((C, A)\) and \((C, B)\) are links, predict \((A, B)\)
  - Let \(N(C)\) be the neighbours of \(C\), predict \(\{(A,B) : A \in N(C), B \in N(C)\}\)
  - **Plausible**: mutual friends, similar interests...

- **Problem**: may predict a lot of links
  - In Facebook, users have on average 100 friends
  - Predicts \(~5,000\) links for each user

- **Instead**: give score to each \((A, B)\) pair. Predict highest scored pairs
Weighting Common Neighbours

- Weight the prediction by the number of common neighbours
  - If A and B have 10 mutual friends, more likely than if they have 1
  - \( \text{score}(A, B) = |N(A) \cap N(B)| \)

- Jaccard coefficient: what fraction of neighbours that are common
  - Count common neighbours of A & B, divided by total neighbours
  - \( \text{score}(A, B) = \frac{|N(A) \cap N(B)|}{|N(A) \cup N(B)|} \)

- Adamic-Adar: weight common neighbours by their degree
  - Sum over common neighbours, \( 1/\log(\text{number of neighbours}) \)
  - \( \text{score}(A, B) = \sum_{C \in N(A) \cap N(B)} \frac{1}{\log(|N(C)|)} \)

- Preferential attachment: high-degree nodes more likely to link
  - \( \text{score}(A, B) = |N(A)| \times |N(B)| \)
Experimental study

- **Experiment**: compare top-$n$ predicted edges with $n$ new edges
- **Data**: academic collaborations taken from arXiv repository

<table>
<thead>
<tr>
<th>predictor</th>
<th>astro-ph</th>
<th>cond-mat</th>
<th>gr-qc</th>
<th>hep-ph</th>
<th>hep-th</th>
</tr>
</thead>
<tbody>
<tr>
<td>probability that a random prediction is correct</td>
<td>0.475%</td>
<td>0.147%</td>
<td>0.341%</td>
<td>0.207%</td>
<td>0.153%</td>
</tr>
<tr>
<td>graph distance (all distance-two pairs)</td>
<td>9.6</td>
<td>25.3</td>
<td>21.4</td>
<td>12.2</td>
<td>29.2</td>
</tr>
<tr>
<td>common neighbors</td>
<td>18.0</td>
<td>41.1</td>
<td>27.2</td>
<td>27.0</td>
<td>47.2</td>
</tr>
<tr>
<td>preferential attachment</td>
<td>4.7</td>
<td>6.1</td>
<td>7.6</td>
<td>15.2</td>
<td>7.5</td>
</tr>
<tr>
<td>Adamic/Adar</td>
<td>16.8</td>
<td>54.8</td>
<td>30.1</td>
<td>33.3</td>
<td>50.5</td>
</tr>
<tr>
<td>Jaccard</td>
<td>16.4</td>
<td>42.3</td>
<td>19.9</td>
<td>27.7</td>
<td>41.7</td>
</tr>
</tbody>
</table>

- **Graph distance**: predict links between all pairs at distance $d$
  - $d=2$: same as predicting all pairs that have $\geq 1$ common neighbours
- No clear winners, but some trends on this data
- **Full survey**: [The Link Prediction Problem for Social Networks](#)
Task 2: Finding important nodes

- Much work on finding important or influential nodes
  - **Degree**: find node with highest degree (7)
  - Not always meaningful: can engineer nodes with high degree
    - E.g. could plant nodes which all follow/friend each other

- Other more complicated measures can be used
  - Study time cost in terms of number of nodes ($n$), edges ($m$)
  - Graphs tend to be **sparse**: average degree ~100
    - Dunbar’s number again
Graph Centrality

- The eccentricity of a node is the maximum distance to any other node in the graph
  - Eccentricity of (5) is 3 (realized by (5, 10))
  - The radius of a graph is the minimum eccentricity
- A “central node” is one that achieves the minimum eccentricity
  - (8) is the only central node here, with eccentricity 2
  - Computable in time $O(mn)$
Betweenness Centrality

- “Betweenness”: number of shortest paths passing through a node
  - For all node pairs, compute the shortest path(s) between them
  - For each node, count the number of shortest paths it is on
- Formally: for node $v$ sum over all node pairs $s \neq t \neq v$, $\sigma_{s,t}(v)/\sigma_{s,t}$
  - $\sigma_{s,t}(v)$ is the number of shortest paths from $s$ to $t$ passing through $v$
  - $\sigma_{s,t}$ is the total number of shortest paths from $s$ to $t$
  - Can normalize by dividing by number of pairs, $n(n-1)/2$
  - In example, (8) is on all shortest paths from $\{1,2,3,4,5,6,7\}$ to $\{9,10\}$
  - Betweenness of (8) is 14
  - Betweenness of (9) is 8
  - Betweenness of (10) and (3) is 0
- Compute time can be high: $O(mn)$
Measures based on Random Walks

- If I take a random walk through the network, which nodes am I more likely to end up at?
- Many measures of importance based on random walks
  - PageRank (ranking web pages) is interpreted as a random walk
- Random walks on kite graph likely to visit (7), less likely to visit (10)
PageRank algorithm

- Given a (directed) graph, want to compute importance of nodes
  - Most useful web pages on the internet
  - Most significant people in a social network
  - Most influential papers in scientific research literature

- Simple approach: find nodes with highest degree
  - Plausible? Has the most nodes pointing to it
  - On the web, highest (in)degree page used to be adobe.com
    - Get Flash/Acrobat reader
  - Easy to “game”: make lots of webpages that point to mine

- PageRank algorithm uses more complex properties of graph
  - Originally introduced to rank search results on Google
Towards PageRank (Brin & Page, 1998)

- Basic idea: each directed edge in the graph represents a “vote” from the source of the edge that the destination is “good”
  - So score a page by the sum of the votes it receives
- Why is this not the same as computing the (in)degree?
  1. Weight of the “vote” depends on the score of the page making the vote
  2. Divide the score of a page evenly by the number of links: the more votes made, the lower the weight of each vote.
Simple PageRank

- Let $E$ be an $n \times n$ matrix encoding link structure of graph
  - $E_{ij} = 1$ iff page $j$ points to page $i$ (directed)
  - Let $M = E$, with each column normalized to sum to 1
- Score of page $i = \sum_i M_{ij} \cdot (\text{score of page } j)$
- Formally, let $r = \text{vector of scores}$
  - Then PageRank satisfies $\lambda r = Mr$
  - $\implies r$ is an eigenvector of $M$

\[ \lambda \begin{bmatrix} r \end{bmatrix} = \begin{bmatrix} M \end{bmatrix} \begin{bmatrix} r \end{bmatrix} \]
Which Eigenvector?

- However, a matrix may have more than one eigenvector
  - It may have as many as $n$ orthogonal eigenvectors
- So there may be many solutions of this equation
  - PageRank picks the principal eigenvector
    (one with largest eigenvalue $\lambda=1$)
- Could find all eigenvalue/eigenvector pairs, then pick largest
  - Costly: eigenvector matrix is large, slow to compute
- A simple procedure computes it: the power method
Iterative Computation

- Start with $r_1 = [1/n, 1/n, 1/n, ... 1/n]
- Set $r_i = M r_{i-1}$, normalize (set $||r_i||_1 = 1$)
  - Iterate until $r_i$ converges ($r_i = r_{i-1}$)

♦ Under certain conditions on $M$, the algorithm will converge on the principal eigenvector.
  - But... $M$ may not obey these conditions automatically
  - Problems arise if the graph is not connected

The vectors $[1, 1, 1, 0, 0]$ and $[0, 0, 0, 1, 1]$ are both eigenvectors with eigenvalue 1
Guaranteeing Convergence

♦ Augment $M$: set $M' = \alpha M + (1-\alpha) \mathbf{1}$
  – $\mathbf{1}$ = matrix of all 1s

♦ Adds “weak” links (with weight $1-\alpha$) between all node pairs
  – Now compute PageRank by finding eigenvector of $M'$
  – Algorithm guaranteed to converge due to graph structure
    ■ For mathematicians: by the Perron-Frobenius theorem

♦ Experimentation sets $\alpha = 0.85$, convergence observed on web graph in 50 – 100 iterations.
  – Prompts questions: Is this ranking meaningful? What is meaning of $\alpha$?
Random Surfer Model

♦ Interpretation of $M'$:
- Starting from a random node, follow a random procedure
- With probability $\alpha$, pick an outgoing edge uniformly from the current node, and follow it
  - With probability $1-\alpha$, jump to a randomly chosen node
- Now the scores $r_i$ give the long term probability of visiting node $i$
  - For mathematicians: Stationary distribution of a Markov Chain $M'$ is irreducible, aperiodic and positive recurrent
PageRank issues

- PageRank is well-suited to parallel computation (e.g. MapReduce)
  - Computing product of $M$ with vector $r$
  - $M$ is sparse: only represent/transmit the non-zero entries
  - Each iteration computes $r_{i+1}$ from $r_i$

- Problem: new nodes have low PageRank, as few nodes link in
  - A “cold-start” problem: causes difficulty in rapidly changing network

- Many variations of PageRank have been proposed
  - Personalized PageRank changes the background node distribution
    - Eg. PPR for node $v$: with probability $1-\alpha$, return to node $v$
  - SimRank generalizes ideas to compute similarity of node pairs
    - Similarity of nodes $v, w$ is sum of similarity of all neighbors
    - Very costly to compute in moderate sized graphs
Task 3: Classification in Social Networks

- Often want to classify (label) nodes in social networks
  - Political persuasion, demographics, interest in a topic, ...
- Have label information about some nodes already
  - Want to propagate this labeling function to others
- Could treat as a traditional classification problem
  - Just use node information to build a standard classifier
- But we also have link information
  - Intuition: we should be able to use links to help the classifier
- A whole new area of data analytics
  - Called relational learning, or semi-supervised learning
Learning Labels on graphs

- Social media, blogs, web links, comments etc. implicitly define a (massive) (multi)graph
- Focus on problems of learning labels: properties of an account owner such as age

- As with all supervised learning, cannot always trust the training data... apparently some people lie about their age
Simple Learning on Graphs

Local: Iterative

Hypothesis: Nodes point to other nodes with similar labels ("homophily")

Label is computed from the votes by its neighbors

- Labels are computed iteratively using weighted voting by neighbors

Global: Nearest Neighbor

Hypothesis: Nodes with similar neighborhoods have similar labels ("co-citation regularity")

Label is inferred by searching for similar neighborhoods of labeled nodes
Extend Learning to Multigraphs

Iterative: Pseudo Labels

- Hypothesis: Web pages link similar communities of bloggers

- Webpages assigned a pseudo label, based on votes by its neighbors

Nearest Neighbor: Set Similarity

- Hypothesis: Distance computation is improved with additional features

- Augment distance with similarity between sets of neighboring web-nodes
Experimental Study

- Preliminary experiments guided the choice of settings:
  - Choice of similarity function for NN classifier: used correlation coefficient between vectors of adjacent labels
  - In multigraph case with additional features, extended by combining with Jaccard coefficient of set similarity of features
  - Iterative algorithm allocates label based on majority voting

- Experiments on a variety of edge combinations:
  Friends only, blog only, blog+friends, blog+web
  - Friends: explicit list of “friends” from the account owner’s profile
  - Blog: links to other blogs
  - Web: links to the wider web
Data Collection Summary

**Blogger**
- 400K profiles crawled
- 50K (12.5%) labeled
- 41K blog nodes
- 190K blog links
- 331K web nodes
- 997K web links
- Median: 4 blog links
- Median: 3 web links

Most popular weblinks
1. news.google.com
2. picasa.google.com
3. en.wikipedia.org
4. www.flickr.com
5. www.statcounter.com

**LiveJournal**
- 300K profiles crawled
- 124K (41%) labeled
- 200K blog nodes
- 404K blog links
- 289K web nodes
- 1089K web links
- Median: 2 blog links
- Median: 4 web links

Most popular weblinks
1. maps.google.com
2. www.myspace.com
3. photobucket.com
4. www.youtube.com
5. quizilla.com

**Xanga**
- 780K profiles crawled
- 500K (64%) labeled
- 535K blog nodes
- 3000K blog links
- 74K web nodes
- 895K web links
- Median: 5 blog links
- Median: 2 web links

Most popular weblinks
1. members.msn.com
2. wwp.icq.com
3. edit.yahoo.com
4. www.gottem.net
5. www.crazyarcades.com

≈50GB of data collected
Accuracy on Age Label

- Similar results on age for both methods, some data sets are “easier” than others, due to density and connectivity
- Local algorithm takes few seconds to assign labels, NN takes tens of minutes (due to exhaustive comparisons)
Multigraph Labeling for Age

- Adding web links and using pseudo labels does not significantly change accuracy, but increases coverage.
- Assigned age reflects webpage, e.g. bands Slipknot (17) vs. Radiohead (28), but also demographics of blog network.
Learning Location Labels

- Local algorithm predicts country and continent with high (80%+) accuracy over all data sets, validating hypothesis.
- Errors come from over-representing common labels: N. America has high recall, low precision, Africa vice-versa.
Other Mining in Social Networks

Many other data analysis problems studied in social networks

- **Attach strength** to links
  - Not all “friends” are equal: which are the ones that matter?
- Finding meaningful “**groups**” in social networks
  - Like clustering, but much more overlap among groups
- Picking a collection of nodes to maximize “**influence**”
  - Send free samples of products, gossip/news on movies
Further Reading/Viewing

♦ Tutorial on Social Media Analytics, Jure Leskovec
  http://snap.stanford.edu/proj/socmedia-kdd/

♦ “Applying link-based classification to label blogs”
  S. Bhagat, G. Cormode, and I. Rozenbaum
  Joint WEBKDD and SNA-KDD Workshop, 2007

♦ “The Link Prediction Problem for Social Networks”
  J. Kleinberg and D. Liben-Nowell, CIKM 2003

♦ Whole books on the topic (for the keen):
  – Social Network Data Analytics, C. C. Aggarwal (Ed)
  – Introduction to Statistical Relational Learning
Summary of Social Networks

- Introduction to social networks and social network analysis
- Saw how to model social networks as graph structures
- Metrics to identify important entities in social networks
- Classification and prediction in social networks
- Identifying important graph nodes for ranking
- Other data mining problems on social networks