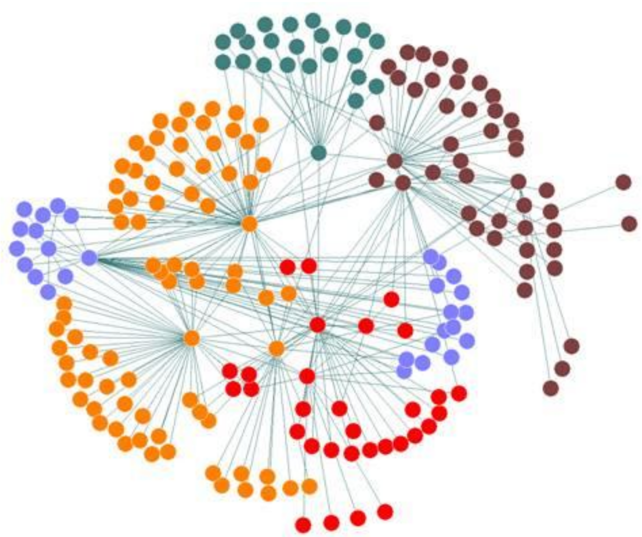
TEL AVIV UNIVERSITY

Algorithms and Applications in Social Networks



2023/2024, Semester A

Slava Novgorodov

Lesson #9

- Link Prediction
- Prediction Heuristics
- Evaluation Methods
- Experimental Results
- (Bonus) More Riddles

Link Prediction

Link Prediction

- The task of link prediction is to compute the chance of each two non-connected nodes to form a connection

- Another point of view – rank all pairs by the chance and take the top-k

- Static mode – taking a snapshot of the graph

Three formulations of the problem

- **Link prediction**: A network is changing over time. Given a snapshot of a network at time t0, predict edges added in the next time interval

- **Link completion** (missing links identification): Given a network, infer links that are consistent with the structure, but missing

- **Link reliability**: Estimate the reliability of given links in the graph.

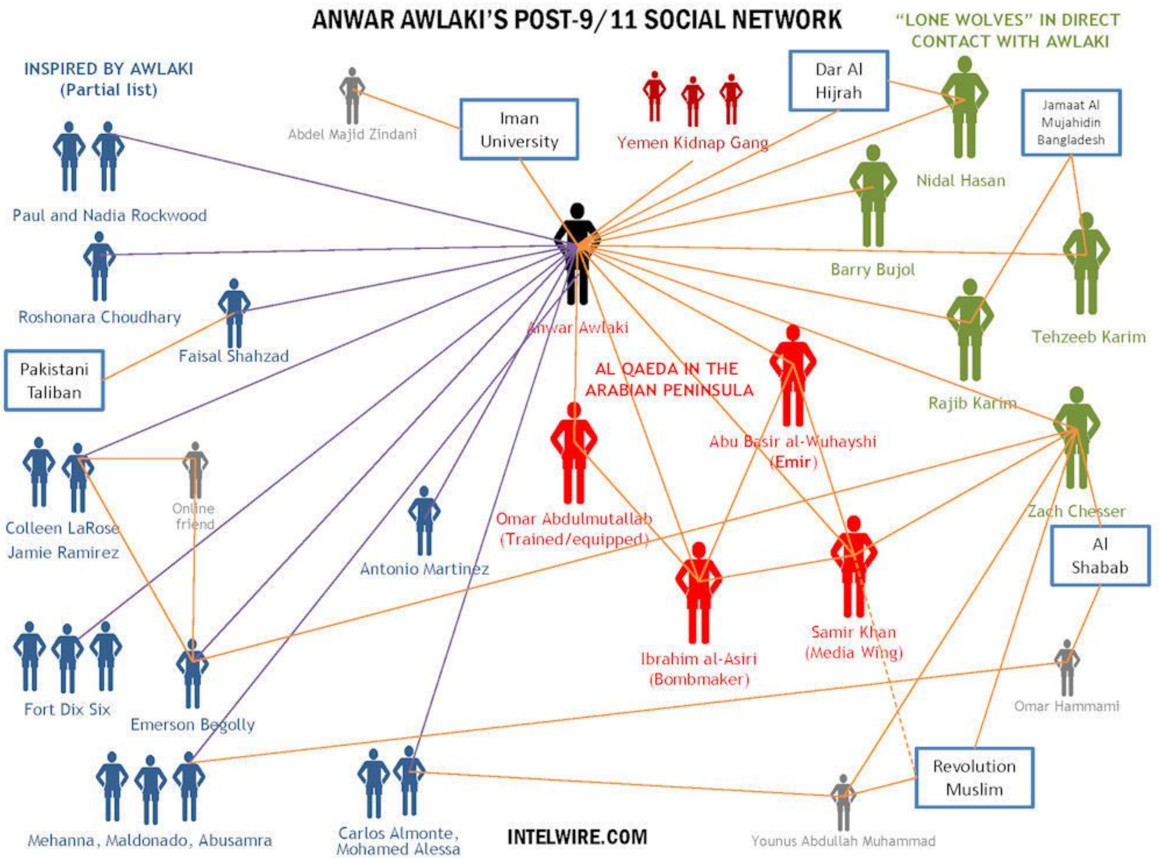
Outcome of the Prediction

While working on Link Prediction problem, possible outcomes can be:

- link existence
- link weight
- link type
- link sign
- …

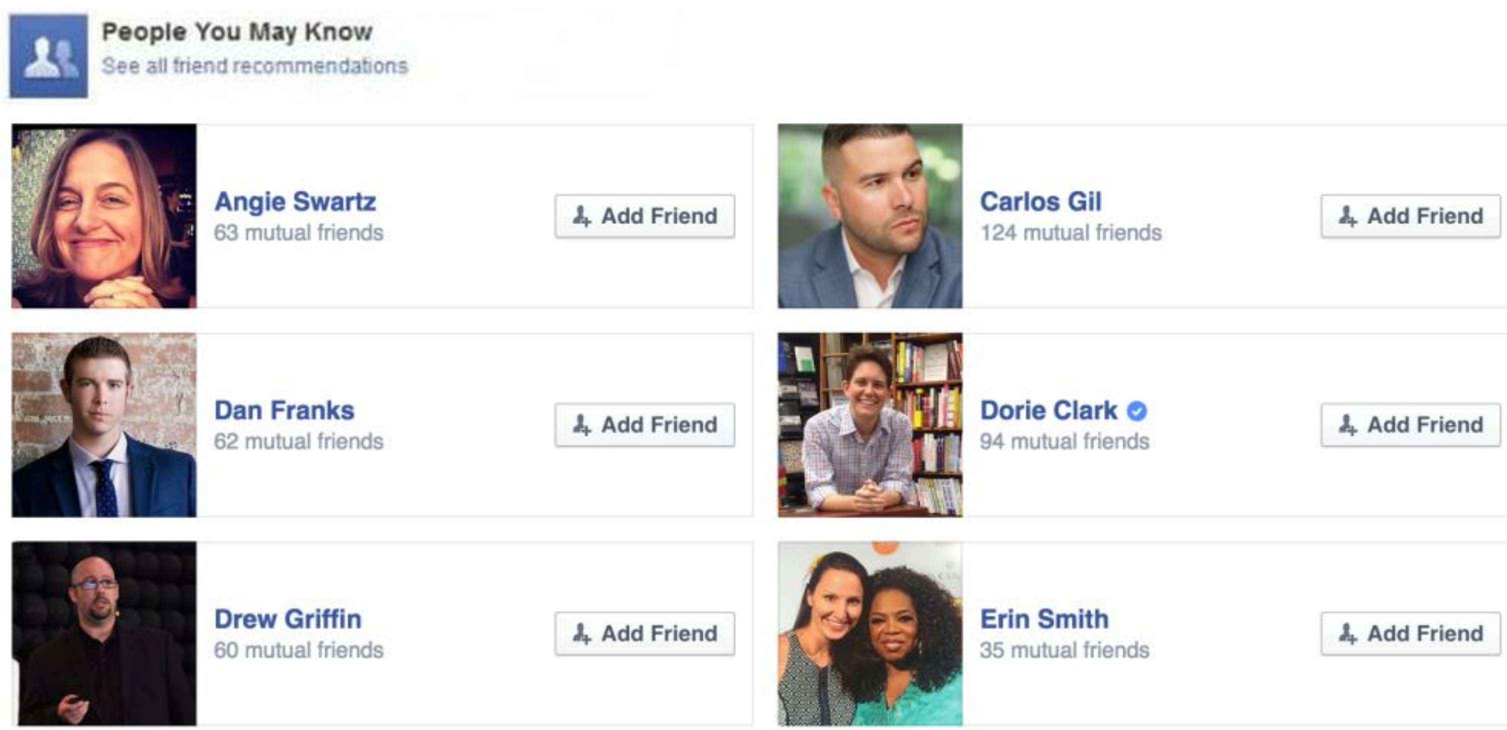
Use cases

- Crime/terrorists networks, who are going to interact with whom?



Use cases

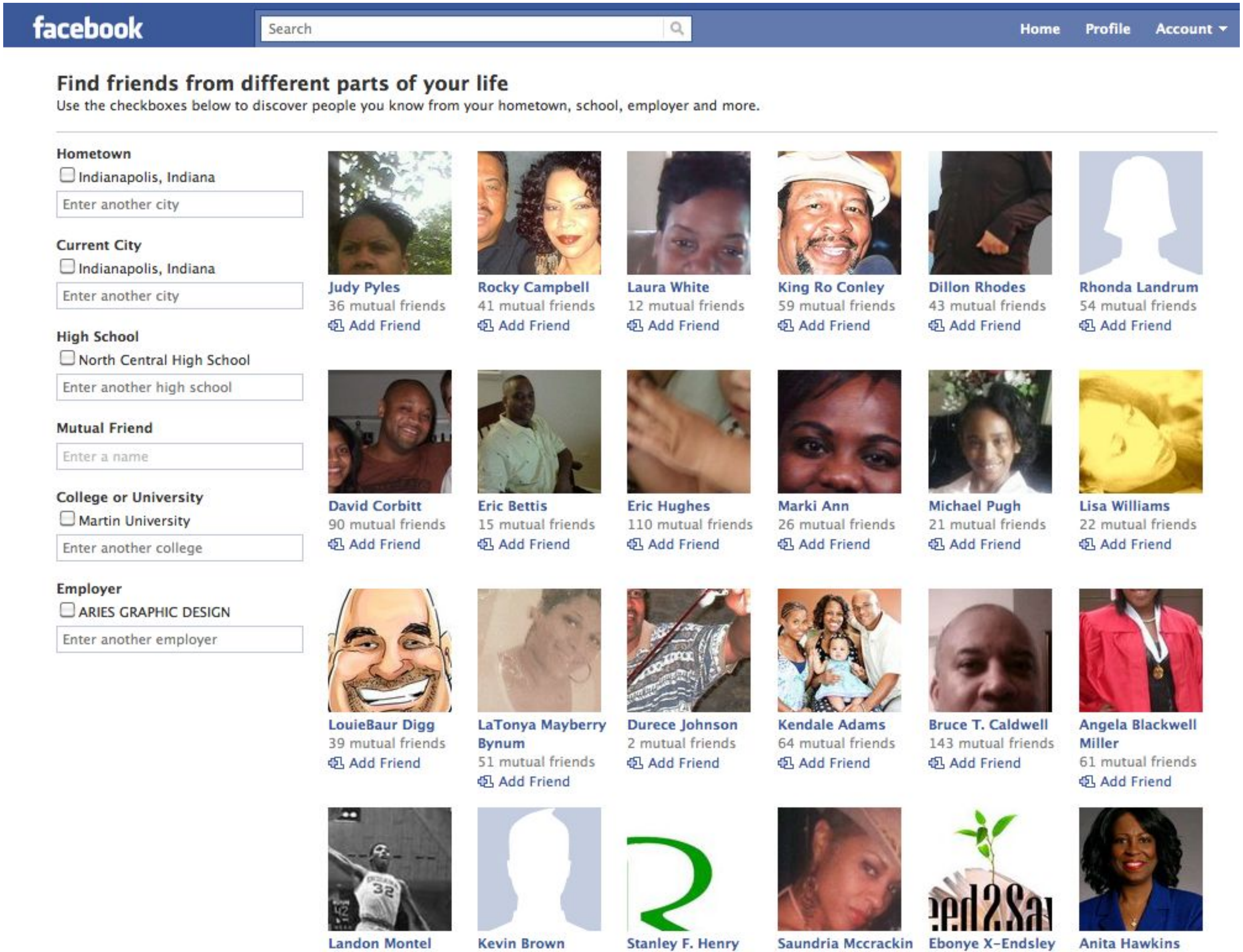
- Facebook's suggested friends



Use cases

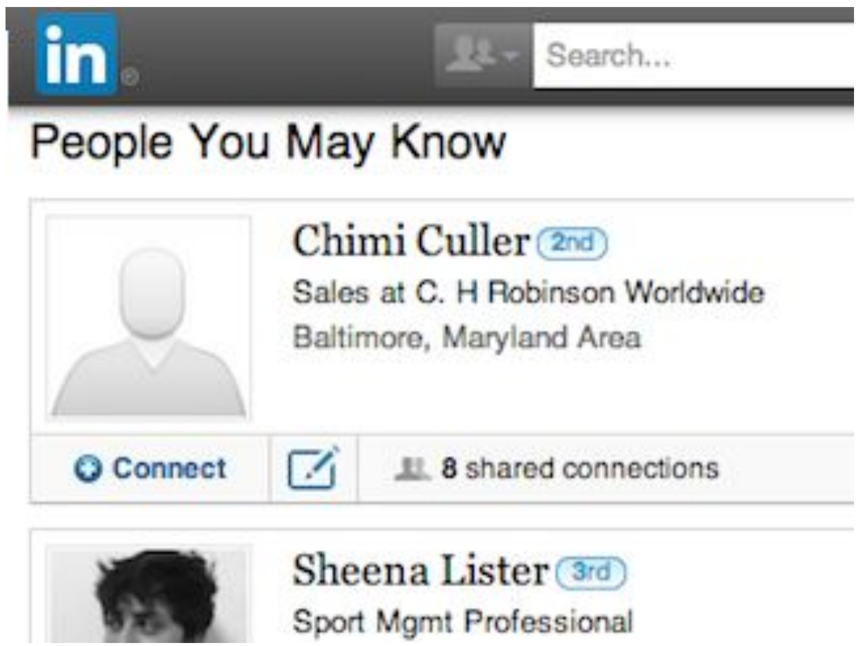
- Facebook’s suggested friends

Use cases


facebook
Search
Home
Profile
Account
Find friends from different parts of your life
Use the checkboxes below to discover people you know from your hometown, school, employer and more.
Hometown
Indianapolis, Indiana
Enter another city
Current City
Indianapolis, Indiana
Enter another city
High School
North Central High School
Enter another high school
Mutual Friend
Enter a name
College or University
Martin University
Enter another college
Employer
ARIES GRAPHIC DESIGN
Enter another employer
Judy Pyles
36 mutual friends
Add Friend
Rocky Campbell
41 mutual friends
Add Friend
Laura White
12 mutual friends
Add Friend
King Ro Conley
59 mutual friends
Add Friend
Dillon Rhodes
43 mutual friends
Add Friend
Rhonda Landrum
54 mutual friends
Add Friend
David Corbitt
90 mutual friends
Add Friend
Eric Bettis
15 mutual friends
Add Friend
Eric Hughes
110 mutual friends
Add Friend
Marki Ann
26 mutual friends
Add Friend
Michael Pugh
21 mutual friends
Add Friend
Lisa Williams
22 mutual friends
Add Friend
LouieBaur Digg
39 mutual friends
Add Friend
LaTonya Mayberry Bynum
51 mutual friends
Add Friend
Durece Johnson
2 mutual friends
Add Friend
Kendale Adams
64 mutual friends
Add Friend
Bruce T. Caldwell
143 mutual friends
Add Friend
Angela Blackwell Miller
61 mutual friends
Add Friend
Landon Montel
Kevin Brown
Stanley F. Henry
Saundria Mccrackin
Ebonye X-Endsley
Anita Hawkins

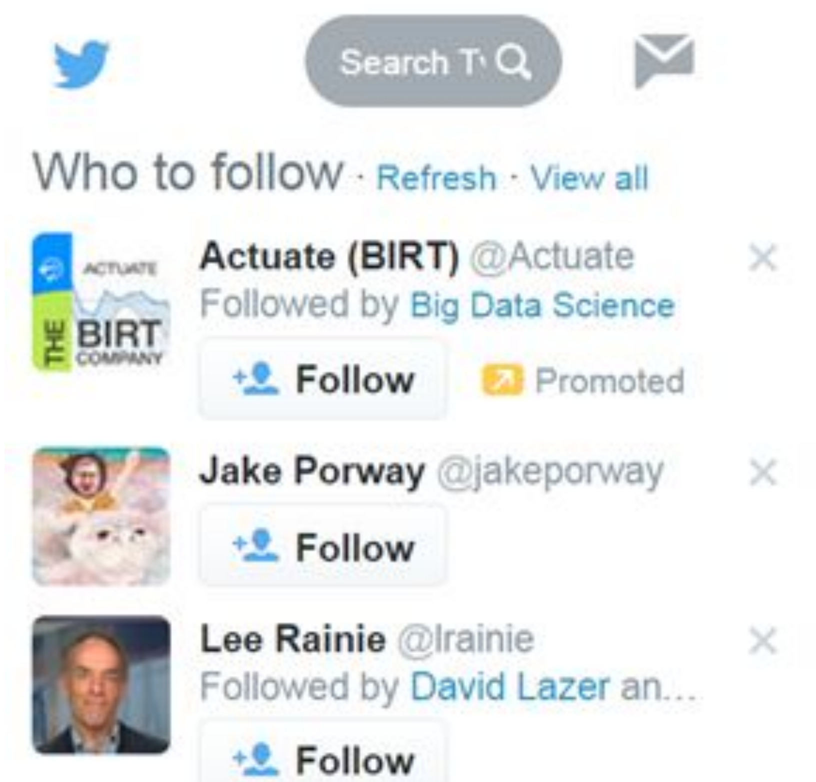
Use cases

- LinkedIn – similar, indicating the distance, not just the number of common friends

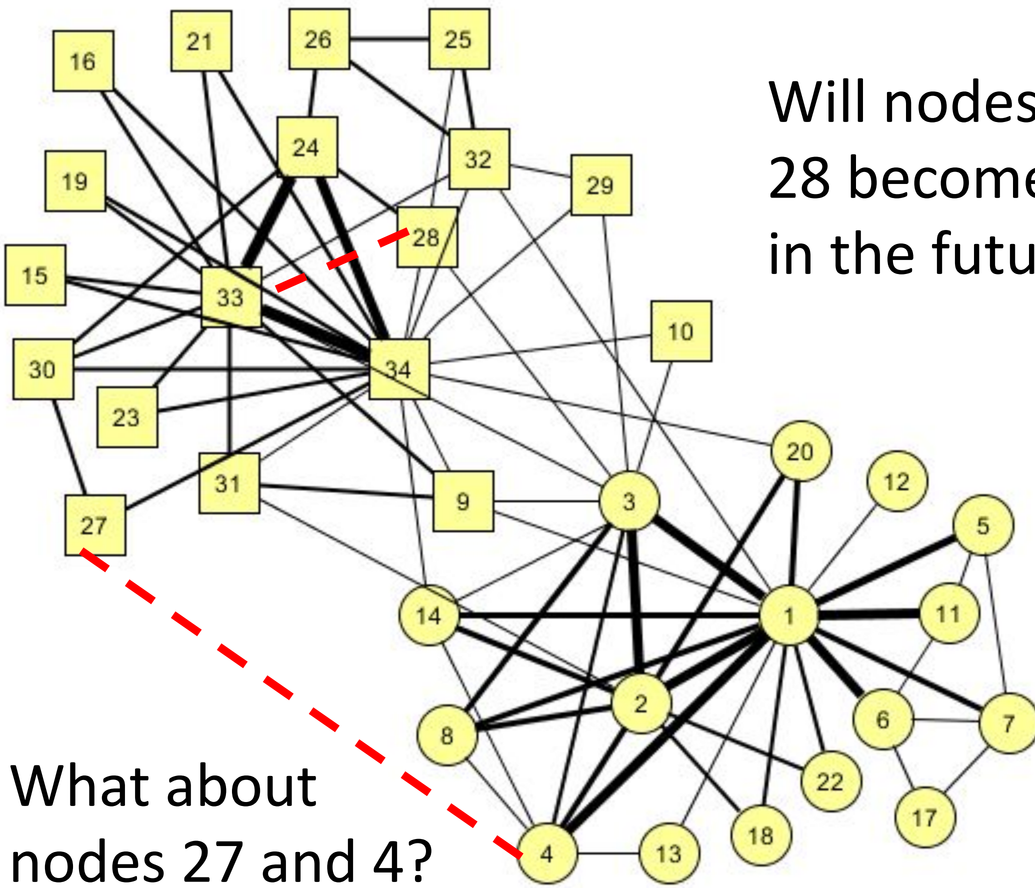


Use cases

- Twitter – suggestions whom to follow (indicates who also follows it)



Summary



Will nodes 33 and 28 become friends in the future?

Does network structure contain enough information to predict what new links will form in the future?

What about nodes 27 and 4?

Prediction Heuristics

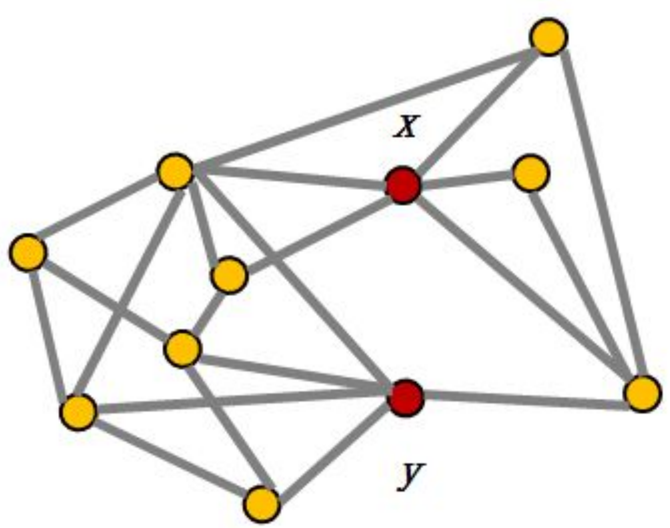
The Link-Prediction Problem

1. Formalize the problem
2. Propose link prediction heuristics based on measures for analyzing the "proximity" of
3. Evaluate different heuristics on different datasets

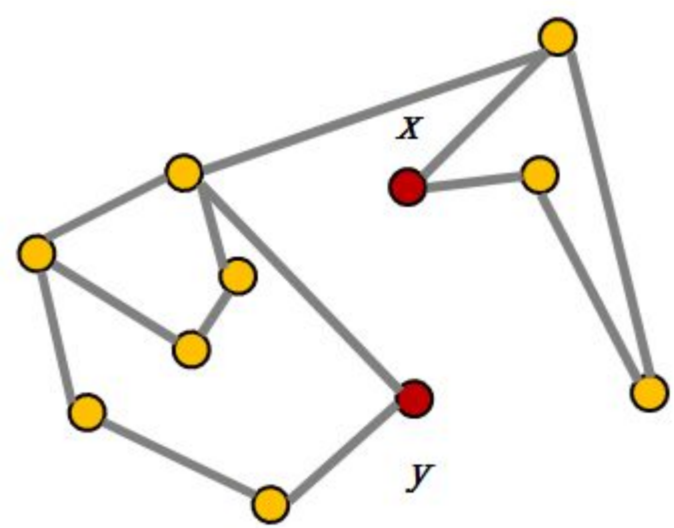
"The Link-Prediction Problem for Social Networks" by Liben-Nowel and Kleinberg
https://www.cs.cornell.edu/home/kleinber/link-pred.pdf

Intuition

- In many networks, people who are "close" belong to the same social circles and will inevitably encounter one another and become linked themselves.
- Link prediction heuristics measure how "close" people are

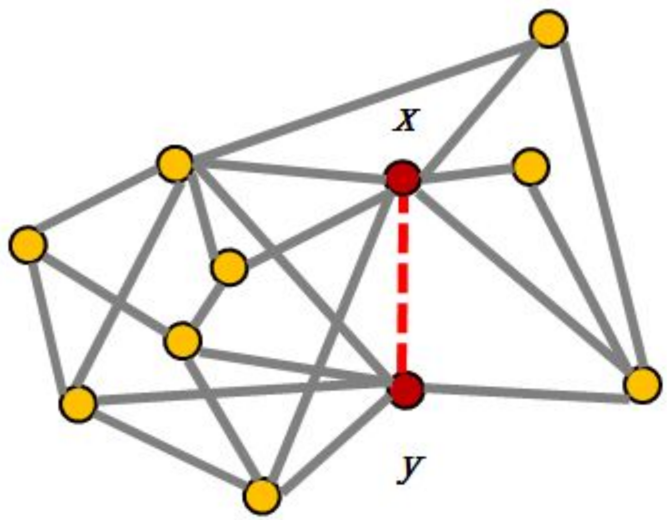


Red nodes are close to each other



Red nodes are more distant

Types of heuristics



- **Local**
  - (negated) Shortest path (SP)
  - Common neighbors (CN)
  - Jaccard (JC)
  - Adamic-Adar (AA)
  - Preferential attachment (PA)
  - …
- **Global**
  - Katz score
  - Hitting time
  - PageRank
  - …

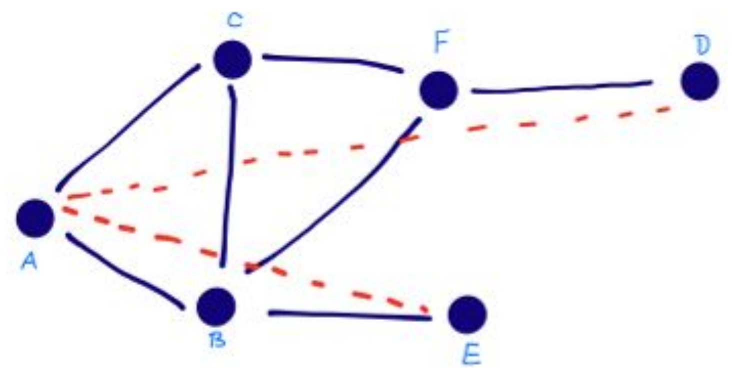
Notation: Neighbors of x: N(x) = $\Gamma(x)$

Degree of x: d_x = |N(x)| = $|\Gamma(x)|$

(negated) Shortest Path (SP)

negated

Score(x,y) = Lenght of Shortest Path Between x and y

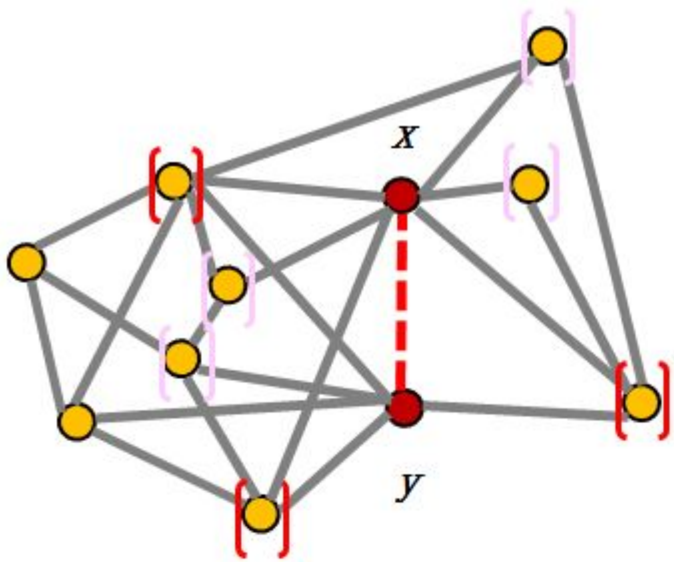


Score(A,E) = -2 ✓

Score(A,D) = -3

desc order

Common Neighbors (CN)



$$CN = 3$$

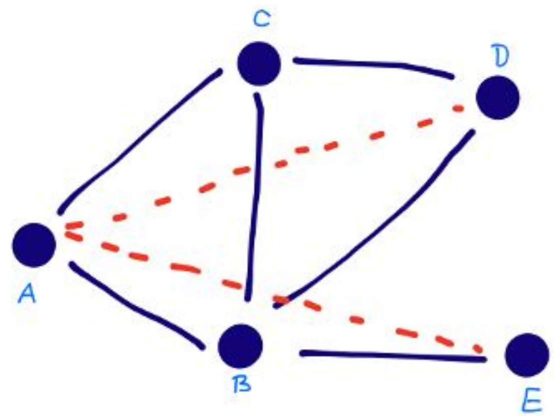
Number of common neighbors between x and y

$$|\Gamma(x) \cap \Gamma(y)|$$

Common Neighbors (CN)

$$\mathrm{Score}(x,y) = |\underline{\Gamma(x)} \cap \Gamma(y)|$$

Neighbors of x



$$|\Gamma(A) \cap \Gamma(D)|$$

B, C   B, C

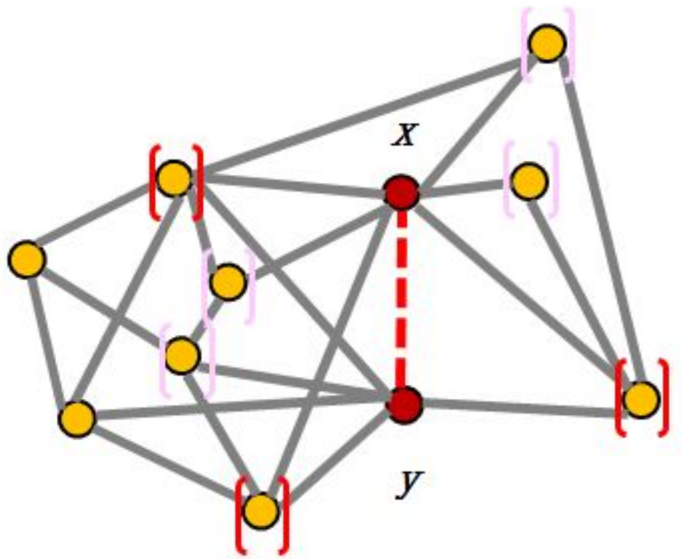
S = 2 ✓

$$|\Gamma(A) \cap \Gamma(E)|$$

B, C   B

S = 1

Jaccard (JC)



The fraction of common nodes

$$\frac{|\Gamma(x)\cap\Gamma(y)|}{|\Gamma(x)\cup\Gamma(y)|}$$

$$JC = \frac{CN}{d_x + d_y - CN}$$

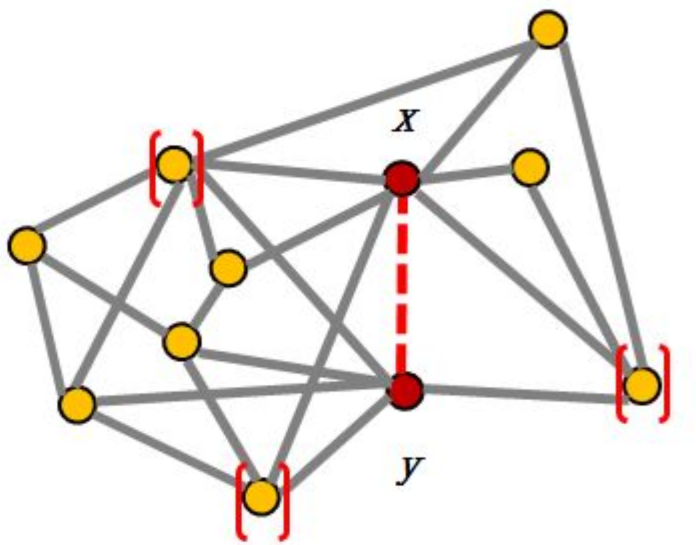
Jaccard (JC)

Common friends

$$\mathrm{Score}(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

total friends

Adamic/Adar (AA)



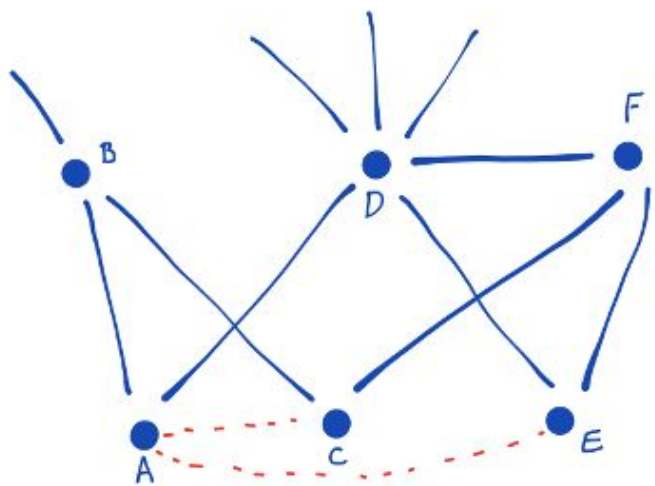
Number of common neighbors normalized by neighbors degrees

$$\sum_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\log|\Gamma(z)|}$$

$$AA = \sum_{z\in CN} \frac{1}{\log d_z}$$

Adamic/Adar (AA)

$$Score(x,y) = \sum_{z \in \Gamma(x)\cap\Gamma(y)} \frac{1}{\log|\Gamma(z)|}$$



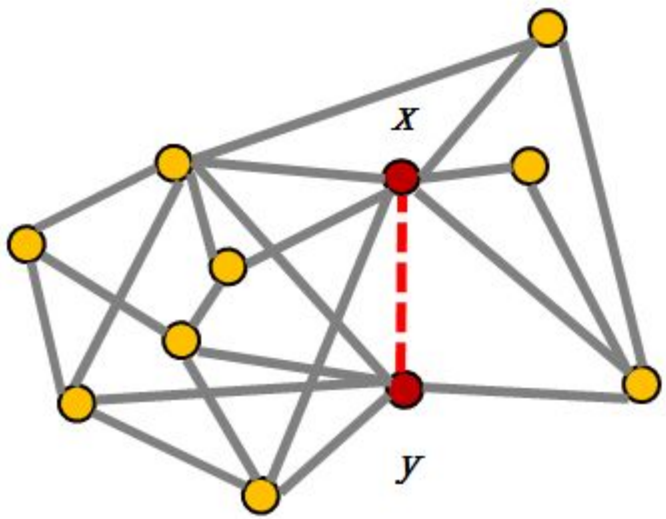
$\Gamma(A)\cap\Gamma(C) = B$

$\frac{1}{\log(\Gamma(B))} = \frac{1}{\log 3} = \underline{\underline{2.09}}$

$\Gamma(A)\cap\Gamma(E) = D$

$\frac{1}{\log(\Gamma(D))} = \frac{1}{\log 6} = 1.2$

Preferential Attachment (PA)



Better connected nodes are most likely to connect

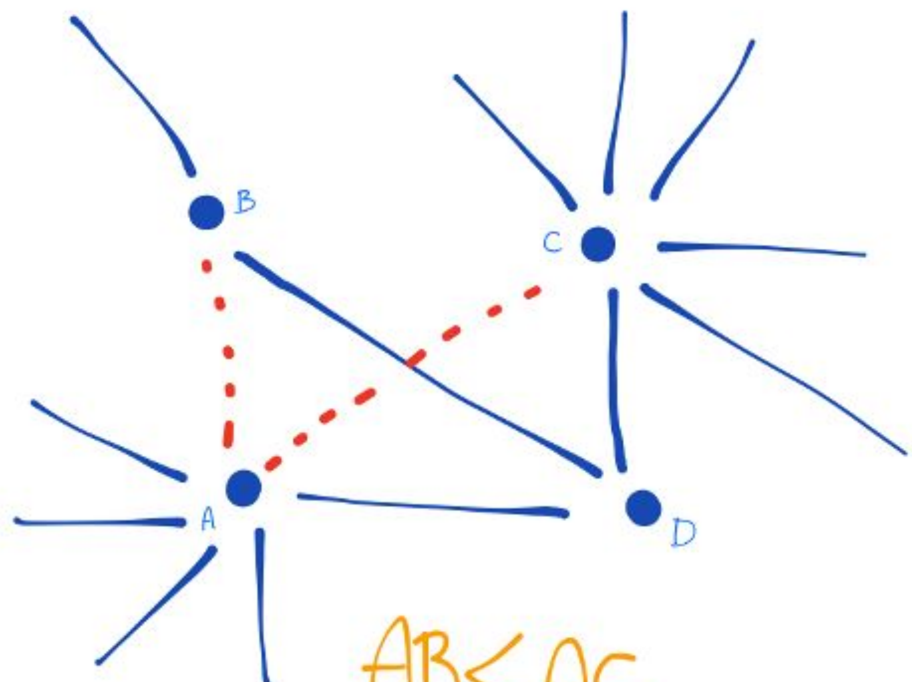
("Rich get richer")

$$|\Gamma(x)| \cdot |\Gamma(y)|$$

$$PA = d_x d_y$$

Preferential Attachment (PA)

$$\mathrm{Score}(x,y) = |\Gamma(x)| \cdot |\Gamma(y)|$$



AB < AC

Katz score

- Sum of number of paths of length l

$$\sum_{\ell=1}^{\infty} \beta^{\ell} \cdot |\mathsf{paths}_{x,y}^{\langle\ell\rangle}|$$

Where $\mathsf{paths}_{x,y}^{\langle\ell\rangle} := \{\text{paths of length exactly } \ell \text{ from } x \text{ to } y\}$

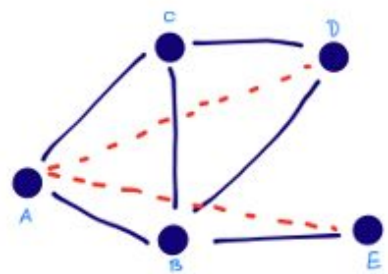
Betta – dumping factor

Katz score

$$Score(x,y) = \sum_{l=1}^{\infty} \beta^{l} \cdot \left|Path^{l}_{x,y}\right|$$

exponentially damped by lenght

Set of all lenght l Paths from x to y



# of Hops

$Path^{2}_{A,D} = 2$ $Path^{3}_{A,D} = 2$

$S = \frac{1}{2} \cdot 2 + \frac{1}{4} \cdot 2 + \cdots$

Damping Factor

$Path^{2}_{A,E} = 1$ $Path^{3}_{A,E} = 1$

$S = \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 1 + \cdots$

Other scoring functions

- Hitting time – expected number of steps from x to y

- SimRank – state of the art similarity measure

$$\text{score(x, y)} = \begin{cases} 1 & \text{if } x = y \\ \gamma \cdot \frac{\sum_{a \in \Gamma(x)} \sum_{b \in \Gamma(y)} \text{score}(a,b)}{|\Gamma(x)| \cdot |\Gamma(y)|} & \text{otherwise} \end{cases}$$

- Clustering coefficient:

  CC(x) * CC(y) or CC(x) + CC(y)

Summary

- Pick a favorite heuristic method
- Compute over all pairs of nodes
- Sort
- Take the top-k

- How to chose best heuristics?
  - Need to evaluate!

Evaluation Methods

Evaluation

Undirected network $G = (V, E)$, universal set $|U| = |V|(|V|-1)/2$

Task: Find out missing links in $U – E$.

Evaluation:

Randomly split E into two sets: training set E^T , validation set E^V

k-fold cross validation

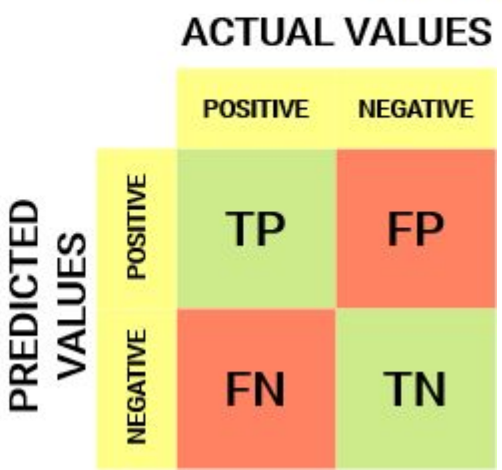
– Randomly partition into k subsets
– Each time one subset is selected as probe set, the others as training set
– Repeat k times, each with a different probe set

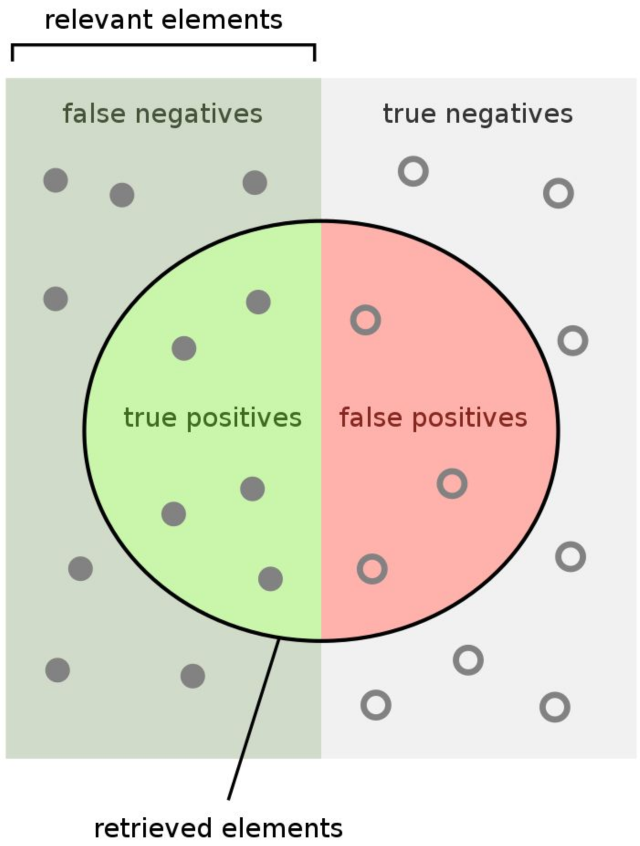
Metrics

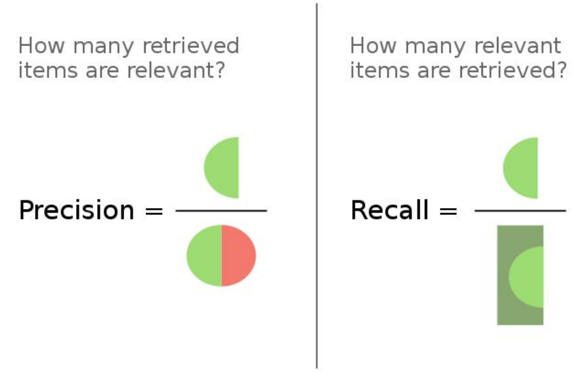
- False positive – we predicted, but doesn't exists in the ground truth (full network)
- False negative – we missed the prediction

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}$$

$$F = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$







Experimental Results

Datasets

Real-world networks

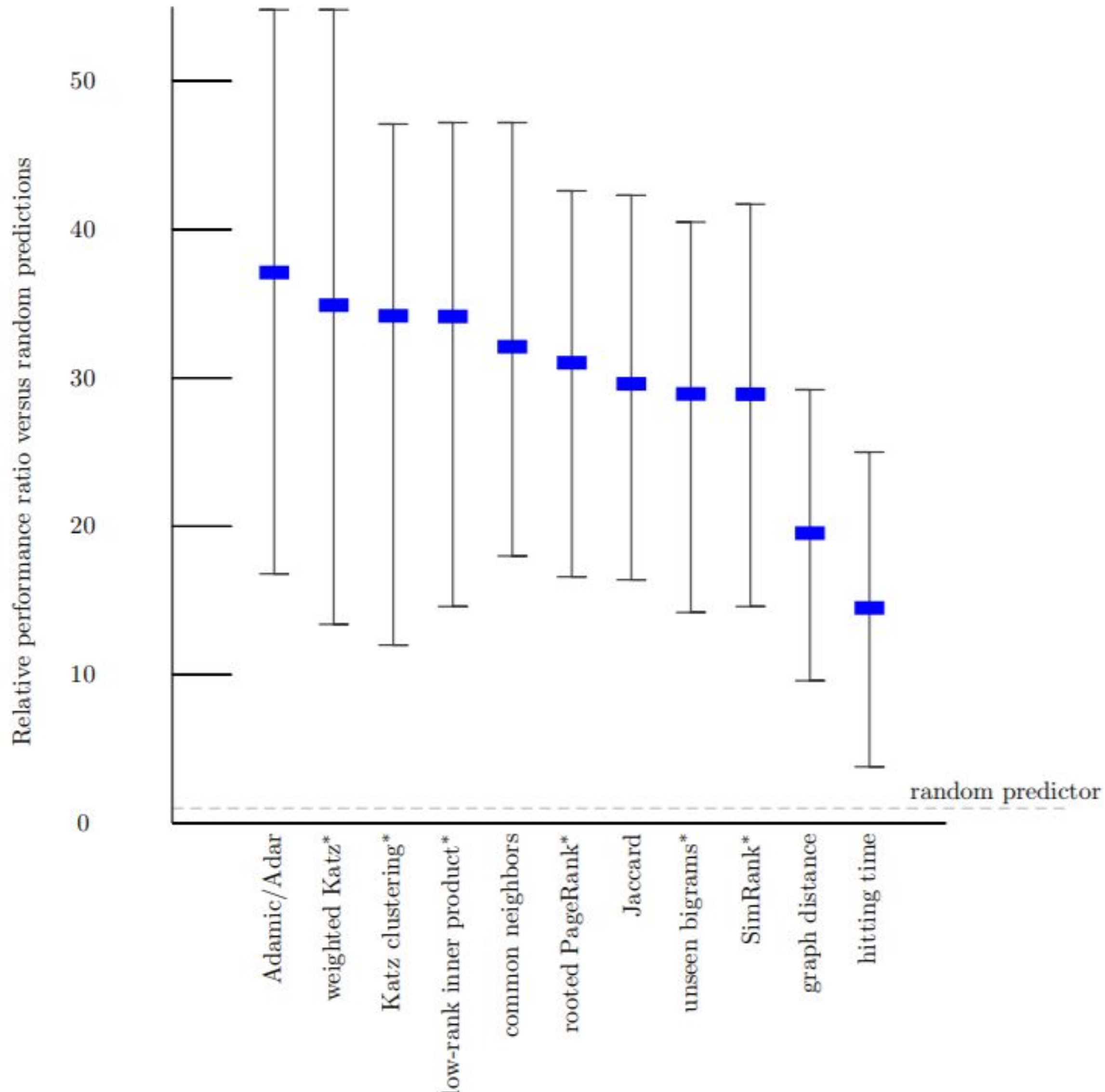
- PPI: protein-protein interaction
- NS: co-authorship
- Grid: electrical power-grid
- PB: US political blogs
- INT: router-level Internet
- USAir: US air transportation

Results

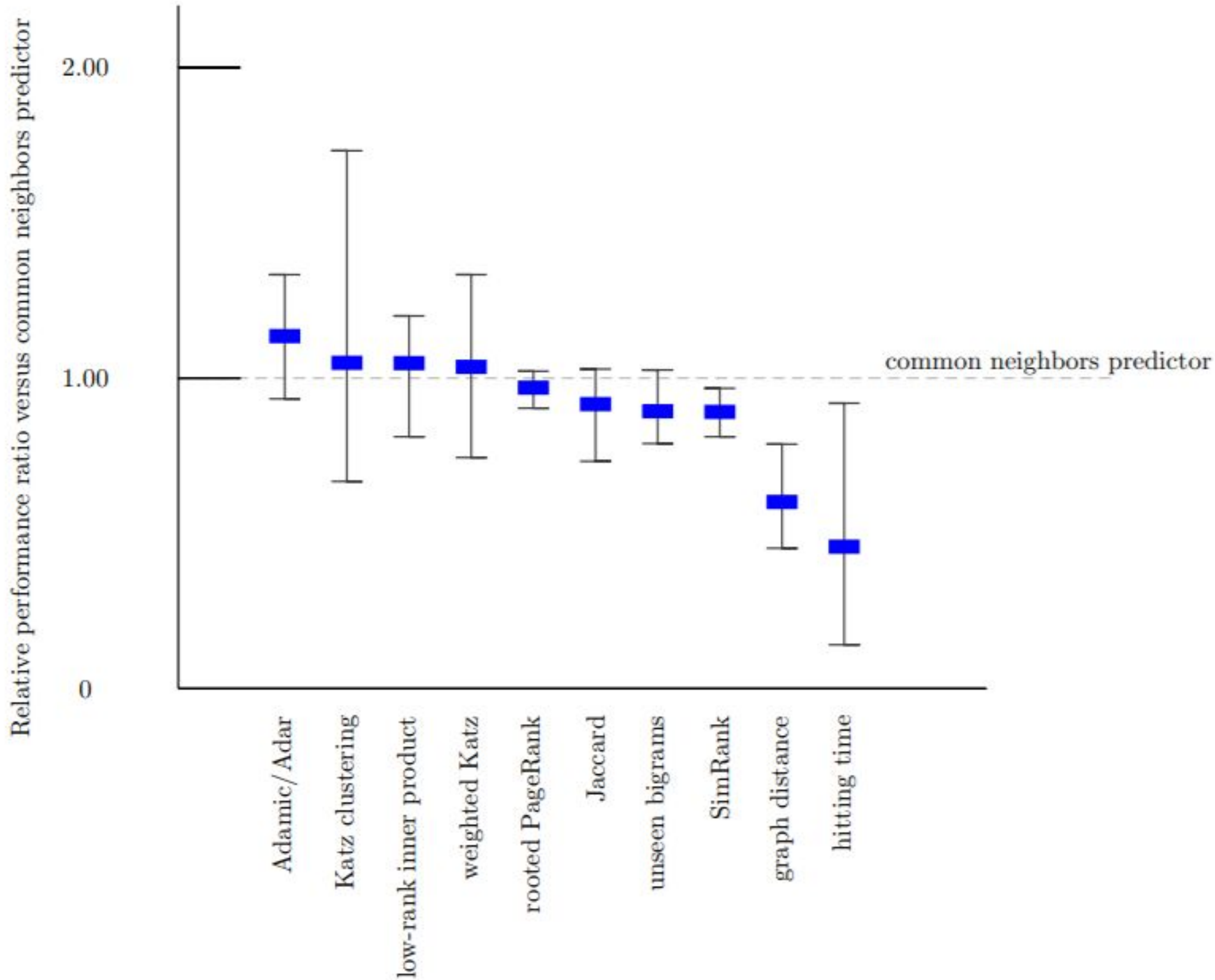
Indices	PPI	NS	Grid	PB	INT	USAir
CN	0.889	**0.933**	**0.590**	0.925	**0.559**	0.937
Jaccard	0.888	**0.933**	**0.590**	0.882	**0.559**	0.901
PA	0.828	0.623	0.446	0.907	0.464	0.886
AA	0.888	0.932	**0.590**	0.922	**0.559**	0.925

* AUC results

Results


Relative performance ratio versus random predictions
50
40
30
20
10
0
random predictor
Adamic/Adar
weighted Katz*
Katz clustering*
low-rank inner product*
common neighbors
rooted PageRank*
Jaccard
unseen bigrams*
SimRank*
graph distance
hitting time

Results



Relative performance ratio versus common neighbors predictor
2.00
1.00
0
common neighbors predictor
Adamic/Adar
Katz clustering
low-rank inner product
weighted Katz
rooted PageRank
Jaccard
unseen bigrams
SimRank
graph distance
hitting time

Results

predictor		astro-ph	cond-mat	gr-qc	hep-ph	hep-th
probability that a random prediction is correct		0.475%	0.147%	0.341%	0.207%	0.153%
graph distance (all distance-two pairs)		*9.6*	*25.3*	*21.4*	*12.2*	*29.2*
common neighbors		***18.0***	***41.1***	***27.2***	***27.0***	***47.2***
preferential attachment		4.7	6.1	7.6	*15.2*	7.5
Adamic/Adar		*16.8*	***54.8***	***30.1***	***33.3***	***50.5***
Jaccard		*16.4*	***42.3***	19.9	***27.7***	*41.7*
SimRank	$\gamma = 0.8$	*14.6*	*39.3*	*22.8*	*26.1*	*41.7*
hitting time		6.5	23.8	*25.0*	3.8	13.4
hitting time, stationary-distribution normed		5.3	23.8	11.0	11.3	21.3
commute time		5.2	15.5	***33.1***	*17.1*	23.4
commute time, stationary-distribution normed		5.3	16.1	11.0	11.3	16.3
rooted PageRank	$\alpha = 0.01$	*10.8*	*28.0*	***33.1***	*18.7*	*29.2*
	$\alpha = 0.05$	*13.8*	*39.9*	***35.3***	*24.6*	*41.3*
	$\alpha = 0.15$	*16.6*	***41.1***	***27.2***	***27.6***	*42.6*
	$\alpha = 0.30$	*17.1*	***42.3***	*25.0*	***29.9***	*46.8*
	$\alpha = 0.50$	*16.8*	***41.1***	*24.3*	***30.7***	*46.8*
Katz (weighted)	$\beta = 0.05$	3.0	21.4	19.9	2.4	12.9
	$\beta = 0.005$	*13.4*	***54.8***	***30.1***	*24.0*	***52.2***
	$\beta = 0.0005$	*14.5*	***54.2***	***30.1***	***32.6***	***51.8***
Katz (unweighted)	$\beta = 0.05$	*10.9*	***41.7***	***37.5***	*18.7*	***48.0***
	$\beta = 0.005$	*16.8*	***41.7***	***37.5***	*24.2*	***49.7***
	$\beta = 0.0005$	*16.8*	***41.7***	***37.5***	*24.9*	***49.7***

Figure 3-3: *Performance of the basic predictors on the link-prediction task defined in Section 3.2. See Sections 3.3.1, 3.3.2, and 3.3.3 for definitions of these predictors. For each predictor and each arXiv section, the displayed number specifies the factor improvement over random prediction. Two predictors in particular are used as baselines for comparison: graph distance and common neighbors. Italicized entries have performance at least as good as the graph-distance predictor; bold entries are at least as good as the common-neighbors predictor. See also Figure 3-4.*

Related reading

http://be.amazd.com/link-prediction/

https://www.cs.cornell.edu/home/kleinber/link-pred.pdf

The Link Prediction Problem for Social Networks*

David Liben-Nowell†
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
dln@theory.lcs.mit.edu

Jon Kleinberg‡
Department of Computer Science
Cornell University
Ithaca, NY 14853 USA
kleinber@cs.cornell.edu

January 8, 2004

Abstract

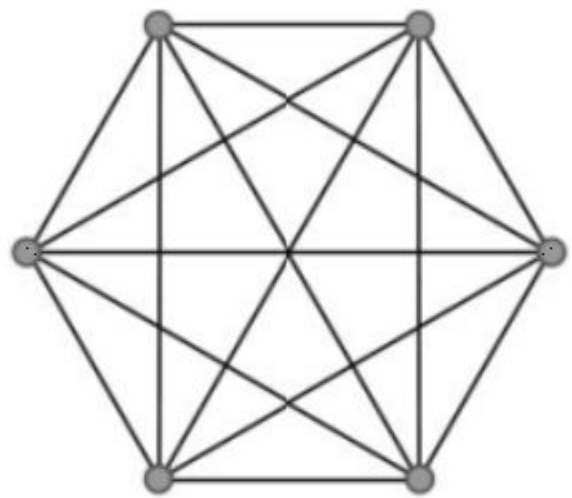
Given a snapshot of a social network, can we infer which new interactions among its members are likely to occur in the near future? We formalize this question as the *link prediction problem*, and develop approaches to link prediction based on measures for analyzing the "proximity" of nodes in a network. Experiments on large co-authorship networks suggest that information about future interactions can be extracted from network topology alone, and that fairly subtle measures for detecting node proximity can outperform more direct measures.

More Riddles

Riddle #1

In group of N people, every two know each other and communicate in one-way direction.

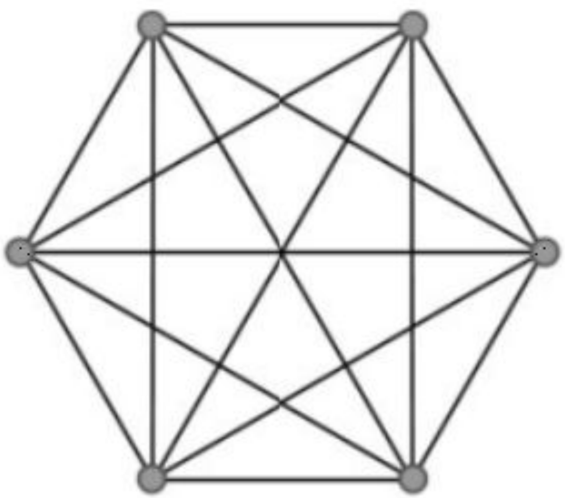
Prove that there is a node (called "main node") that can reach any other node in 2 steps



Riddle #1 - hint

In group of N people, every two know each other and communicate in one-way direction.

Prove that there is a node (called “main node”) that can reach any other node in 2 steps

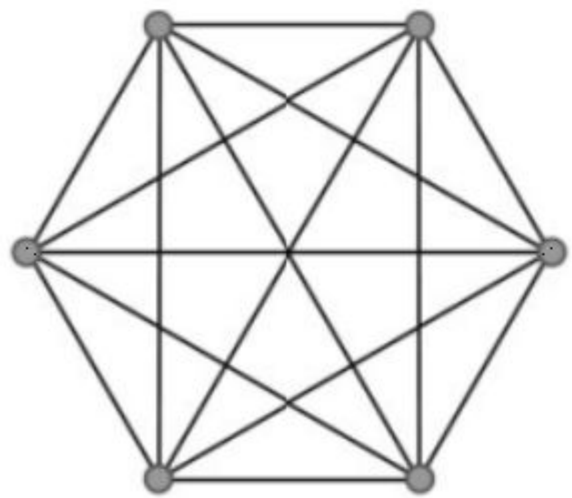


Hint: look at the node with highest degree!

Riddle #1 - solution

In group of N people, every two know each other and communicate in one-way direction.

Prove that there is a node (called “main node”) that can reach any other node in 2 steps

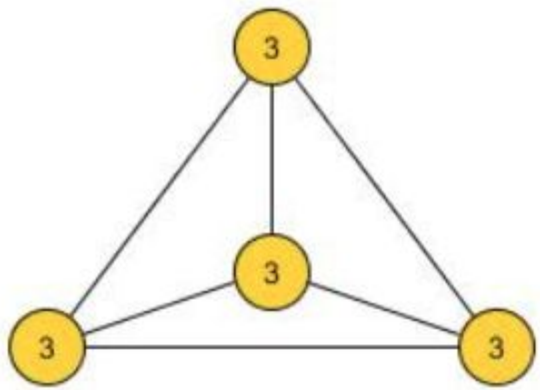


Solution in class!

Riddle #2

In a group of 20 people, each knows exactly 14 others.

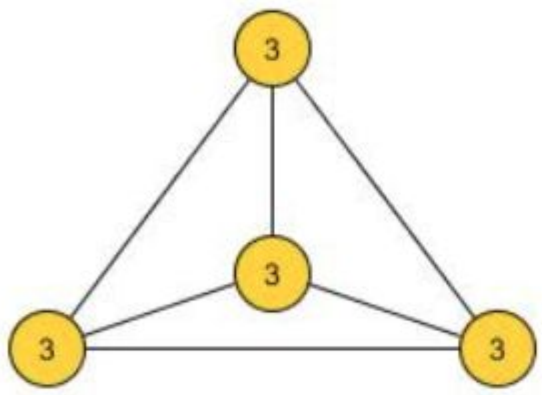
Prove that there is a group of 4 people that know each other



Riddle #2 - hint

In a group of 20 people, each knows exactly 14 others.

Prove that there is a group of 4 people that know each other

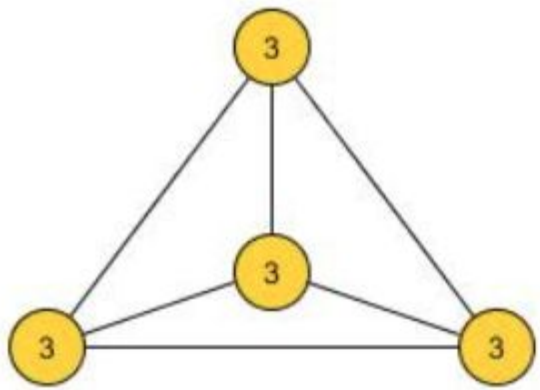


Hint: go in the negative direction, removing all people that doesn't know X

Riddle #2 - solution

In a group of 20 people, each knows exactly 14 others.

Prove that there is a group of 4 people that know each other



Solution in class!

Riddle #3

In group of 100 people that all know each other, we removed 98 connection.

Prove that the graph is still connected!

Riddle #3 - solution

In group of 100 people that all know each other, we removed 98 connection.

Prove that the graph is still connected!

Solution:
Number of removed edges – n * (100-n)

$$n(100-n) = 50^2 - (n-50)^2 \geq 50^2 - 49^2 = 1 \cdot 99 > 98.$$

Thank you!

Questions?