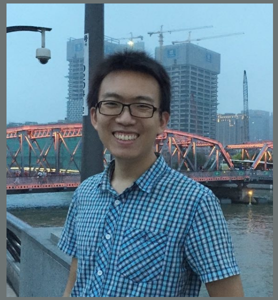
Link Prediction Based on Graph Neural Networks

Muhan Zhang and Yixin Chen, NeurIPS 2018

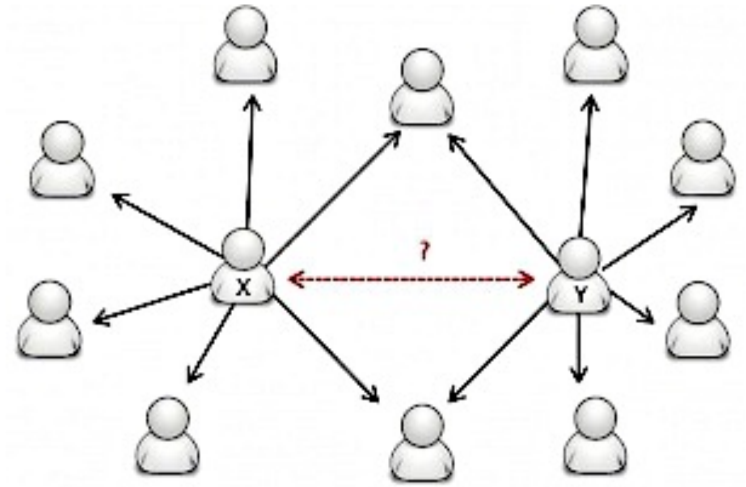


Washington University in St. Louis

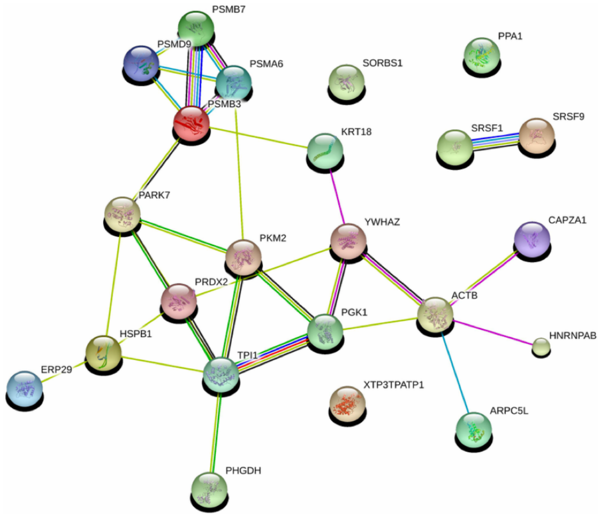
Link Prediction (LP) Problem

Given an incomplete network, predict whether two nodes are likely to have a link.

Applications:
- Friend recommendation in social networks
- Product recommendation in ecommerce
- Interaction prediction in biological networks
- Knowledge graph completion
- …



social network



Biological network

Figures from the Internet.

Heuristic Methods for LP

Calculate a proximity score for each pair of nodes.

Table 1: Popular Heuristics for Link Prediction

Name	Formula	Order
common neighbors	$\lvert\Gamma(x) \cap \Gamma(y)\rvert$	first
Jaccard	$\frac{\lvert\Gamma(x)\cap\Gamma(y)\rvert}{\lvert\Gamma(x)\cup\Gamma(y)\rvert}$	first
preferential attachment	$\lvert\Gamma(x)\rvert \cdot \lvert\Gamma(y)\rvert$	first
Adamic-Adar	$\sum_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\log \lvert\Gamma(z)\rvert}$	second
resource allocation	$\sum_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\lvert\Gamma(z)\rvert}$	second
Katz	$\sum_{l=1}^{\infty} \beta^l \lvert\text{path}(x,y) = l\rvert$	high
PageRank	$q_{xy} + q_{yx}$	high
SimRank	$\gamma \frac{\sum_{a\in\Gamma(x)}\sum_{b\in\Gamma(y)}\text{score}(a,b)}{\lvert\Gamma(x)\rvert\cdot\lvert\Gamma(y)\rvert}$	high
resistance distance	$\frac{1}{l^+_{xx}+l^+_{yy}-2l^+_{xy}}$	high

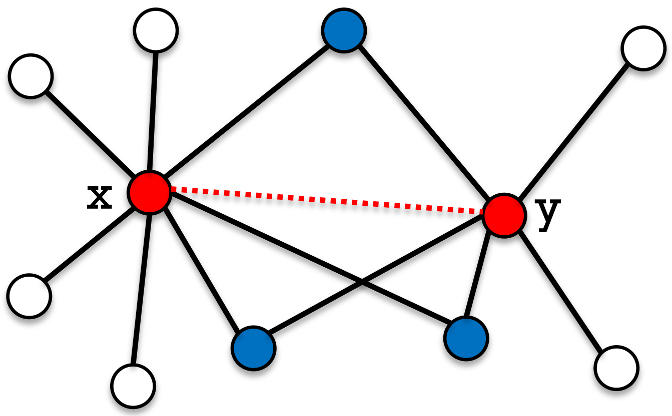
Notes: $\Gamma(x)$ denotes the neighbor set of vertex x. $\lvert\text{path}(x, y) = l\rvert$ counts the number of length-l paths between x and y. q_{xy} is the stationary distribution probability of y under the random walk from x with restart, see [10]. SimRank score is a recursive definition. l^+_{xy} is the (x, y) entry of the pseudoinverse of the graph's Laplacian matrix.

- Good performance
- Easy to calculate
- Interpretable
- No training required

First-Order Heuristics

Notations: $\Gamma(x)$ is the neighbor set of node x in the graph

- The common neighbors (CN) heuristic: $|\,\Gamma(x) \cap \Gamma(y)\,|$

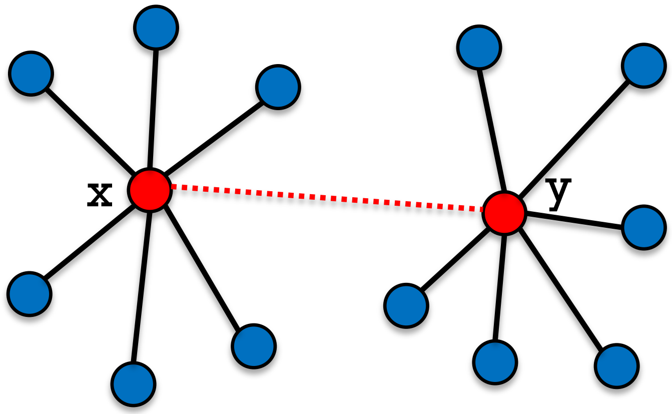


x and y are likely to have a link
if they have many common neighbors.

- First-order heuristic, need only 1-hop neighbors to compute.

First-Order Heuristics

- The preferential attachment (PA) heuristic: $|\Gamma(x)| \cdot |\Gamma(y)|$

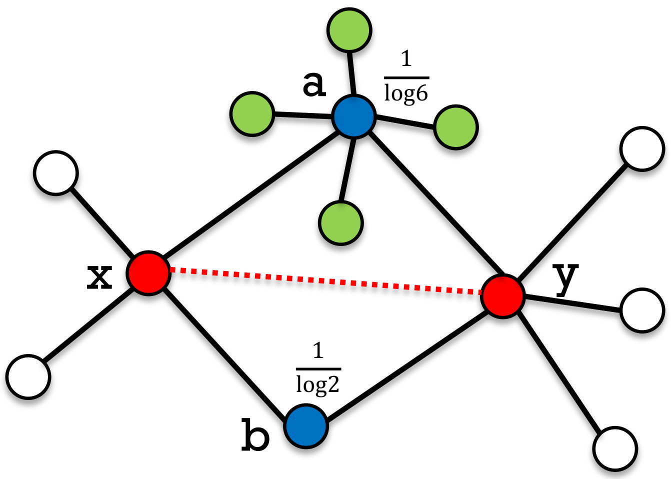


x prefers to connect to y if y is popular.

- First-order heuristic, only involves 1-hop neighbors.

Second-Order Heuristics

- The Adamic-Adar (AA) heuristic: $\sum_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\log|\Gamma(z)|}$



Weighted common neighbors;

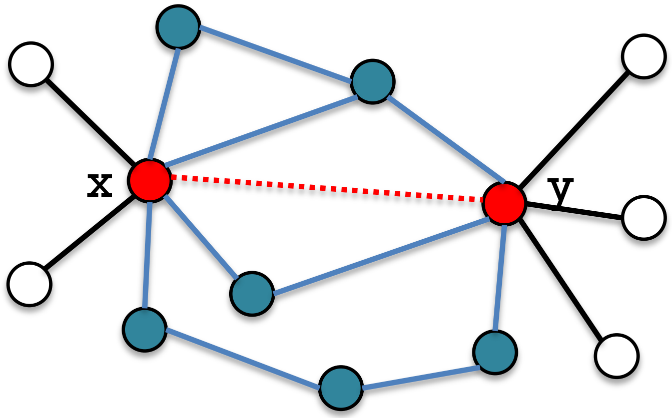
Popular common neighbors contribute less.

- Second-order heuristic. Involves 2-hop neighbors of x and y.

- First-order and second-order heuristics can be calculated from local subgraphs around links.

High-Order Heuristics

- The Katz index heuristic: $\sum_{l=1}^{\infty} \beta^l |\text{walks}(x, y) = l|$



Sum all walks between x and y; each walk discounted by β^l.

$\beta < 1$ is the discount factor

l is the length of a walk

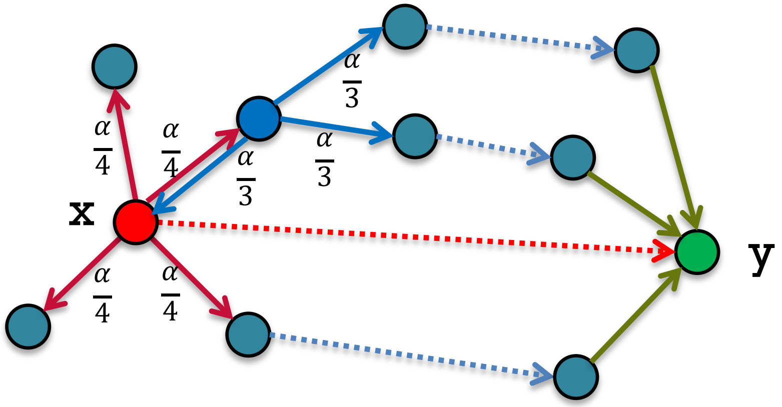
Longer walks contribute less.

- High-order heuristic
- Need to search the entire network.

High-Order Heuristics

- The Rooted PageRank heuristic:

  Let π_x be the stationary distribution of a random walker starting from x who randomly moves to one of its current neighbors with probability α or returns to x with probability $1-\alpha$.



Use $[\pi_x]_y$ as the likelihood of link (x,y).

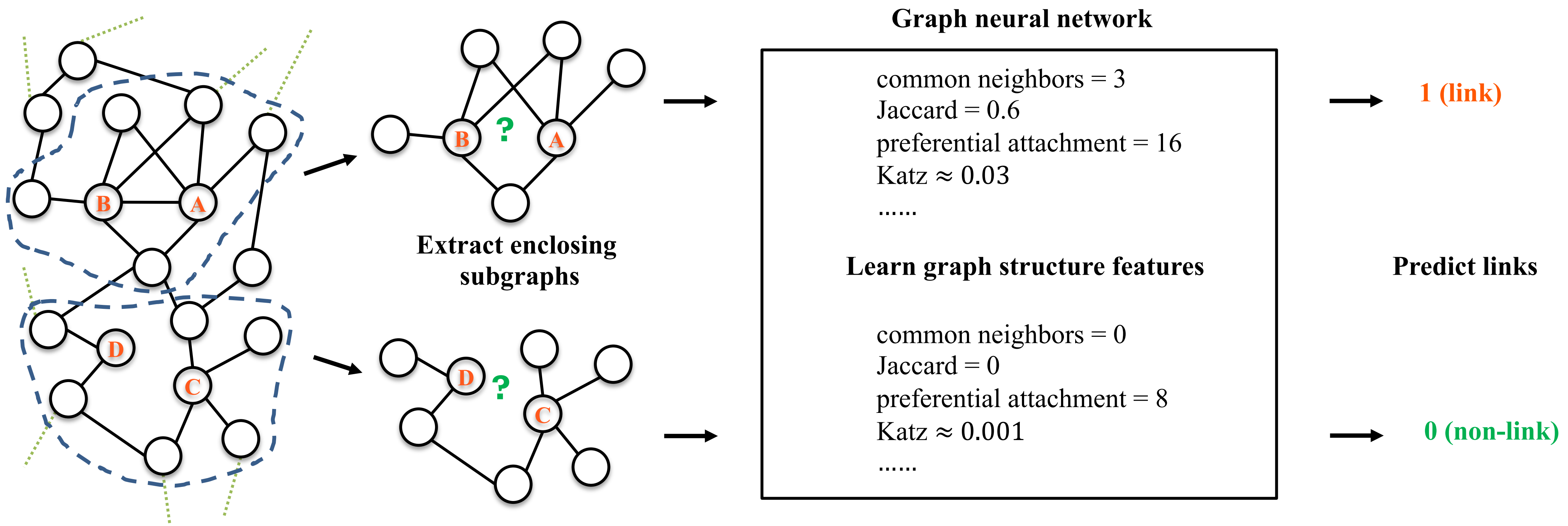
- High-order heuristic

- Need to know the entire network and iterate until convergence.

Drawbacks of Heuristic Methods

- Handcrafted graph structure features, not general.
- Have strong assumptions on link formation mechanisms.
- Only work well on certain networks.
- In our paper, we proposed **SEAL**:
    1. Automatically learn general graph structure features.
    2. No assumption on network properties at all.
    3. New state-of-the-art link prediction performance based on a graph neural network.

Proposed SEAL Framework



- Learn "heuristics" instead of using predefined ones.
- All first-order and second-order heuristics can be learned from local enclosing subgraphs.
- How about high-order heuristics?

A γ-decaying Heuristic Theory

Definition ***(γ-decaying heuristic)*** *A γ-decaying heuristic for (x, y) has the following form:*

$$\mathcal{H}(x,y) = \eta \sum_{l=1}^{\infty} \gamma^{l} f(x,y,l),$$

Main results:

1. A wide range of high-order heuristics can be unified into a γ-decaying heuristic framework, including Katz index, rooted PageRank, SimRank etc. => They intrinsically have the same form!
2. Under mild assumptions, all γ-decaying heuristics can be well approximated from local enclosing subgraphs. => We don't need the entire network to learn them!
3. The approximation error decreases exponentially with the subgraph size. => A small subgraph is enough!

Poster #121 Thurs 10:45 AM -- 12:45 PM @ Room 210 & 230 AB