Semantic Proximity Search on Graphs with Metagraph-based Learning

Yuan Fang¹, Wenqing Lin¹, Vincent Zheng², Min Wu¹, Kevin Chang²³, Xiao-Li Li¹
ICDE 2016 @ Helsinki

¹ Institute for Infocomm Research, Singapore
² Advanced Digital Sciences Center, Singapore
³ University of Illinois at Urbana-Champaign, USA
In this talk

- Problem and Motivation
- Insights and Overall Framework
- Challenges and Solution
- Experimental Study
- Conclusion
Objects and attributes can often be organized as a heterogeneous graph.

“Typed” object graph: capturing users and their attributes on a social network.
Problem: Semantic Proximity Search

Which users are close to/related to Bob?

Family?
Classmates?
Key Criteria of Solution: Semantic differentiation + Online Search

- Social circle learning
- Relationship profiling

Online Search

Existing graph proximity (personalized PageRank, SimRank, ...)

Semantic differentiation
In this talk

- Problem and Motivation
- **Insights and Overall Framework**
- Challenges and Solution
- Experimental Study
- Conclusion
Each semantic class can often be explained by some underlying reasons:

**Family:** [Bob & Alice / same surname & address]

**Classmates:** [Kate & Jay, Bob & Tom / same school & major]

**Close friends:** [Kate & Alice / same employer & hobby]
[Kate & Jay / roommate]
Insight: common substructures, or *metagraphs*, to “explain” semantic classes

**Family**
[same surname & address]

**Classmates**
[same school & major]

**Close friends**
[same employer & hobby]
[roommate]
Overall Framework

- Mining metagraphs
- Matching metagraphs (i.e., finding instances)
- Indexing
- Training
- Testing

Offline

Online
In this talk

- Problem and Motivation
- Insights and Overall Framework
- Challenges and Solution
- Experimental Study
- Conclusion
Challenges

- Challenge #1: Metagraph-based proximity
  - Definition
  - Learning with efficiency
- Challenge #2: Metagraph matching
  - Efficiency
Challenge #1: Meta-graph based proximity (Definition of proximity)

Proximity of two nodes $x, y$ on graph

$x, y$ co-occur in many important metagraphs

$$\pi(x, y; w) \triangleq \frac{2 \cdot m_{xy} \cdot w}{m_x \cdot w + m_y \cdot w}$$

co-occurrence not by chance

$m_{xy}[i] = \# \text{ times } x, y \text{ co-occur in instances of metagraph } i$

$m_x[i] = \# \text{ times } x \text{ occurs in instances of metagraph } i$

$w[i] = \text{ weight for metagraph } i$
Challenge #1: Meta-graph based proximity (Basic learning model)

- **Pairwise learning to rank**

  \[ P(q, x, y; w) \triangleq \frac{1}{1 + e^{-\mu(\pi(q,x;w) - \pi(q,y;w))}} \]

  Each example is a triplet: for query \( q \), \( x \) is ranked before \( y \).

- **Objective function**

  \[ L(w; \Omega) = \sum_{(q,x,y) \in \Omega} \log P(q, x, y; w) \]
Challenge #1: Meta-graph based proximity (Need for efficient training)

- Expensive to process & match all metagraphs
- Yet not all metagraphs are useful

(a) LinkedIn

(b) Facebook
Challenge #1: Meta-graph based proximity (Dual-stage training)

1. Identify seed metagraphs
2. Learn with seed metagraphs
3. Based on weights of seed metagraphs and their structural relationship with other metagraphs
4. Select more metagraphs
5. Re-learn with seed + selected metagraphs
Challenge #2: Metagraph matching

- Existing method: backtracking
  - DFS search node by node until an entire matched instance is found
  - Fail to leverage symmetric components

- Symmetry-based matching
  - Many metagraphs are symmetric
  - Avoid redundant computation
In this talk

- Problem and Motivation
- Insights and Overall Framework
- Challenges and Solution
- Experimental Study
- Conclusion
Experiment setup - datasets

- LinkedIn ego networks
  - Join all into one bigger graph
  - Labelled relationships as semantic classes
    - “College” and “Coworker”
- Facebook ego networks
  - Join all into one bigger graph
  - Rules to simulate circles
    - “Classmates”: same school, and same degree or major
    - “Family”: same surname, and same location or hometown
Experiment setup - methodology

- Some restrictions on metagraphs
  - Only consider symmetric metagraphs
  - Contains at least 2 users in symmetric positions
  - Number of nodes ≤ 5
  - Ignore metagraphs with > 10^8 instances

- Training and testing
  - 20% queries as training, 80% as testing
  - Randomly repeat the split 10 times

- Ranking metrics
  - NDCG and MAP
Experiment setup – baselines

- **MGP**: metagraph-based proximity (our method)
- MPP: metapath-based proximity
- MGP-U: all metagraphs have uniform weights
- MGP-B: only use the best metagraph
- SRW: supervised random walk
Finding #1: Metagraphs are powerful representations for semantic proximity
Finding #2 & #3

- Dual-Stage training
  - reduce overall cost of metagraph matching by 83%
  - negligible compromise on accuracy

- Symmetry-based matching
  - Reduce matching time for individual metagraphs by 52%
In this talk

- Problem and Motivation
- Insights and Overall Framework
- Challenges and Solution
- Experimental Study
- Conclusion
Conclusion

- Metagraphs are powerful
  - May be extended to other tasks on graph

- Matching metagraphs are expensive
  - Improving its efficiency is crucial