Semantic Proximity Search on Graphs with Metagraph-based Learning

Yuan Fang[1], Wenqing Lin[1], Vincent Zheng[2], Min Wu[1], Kevin Chang[23], Xiao-Li Li[1]

ICDE 2016 @ Helsinki

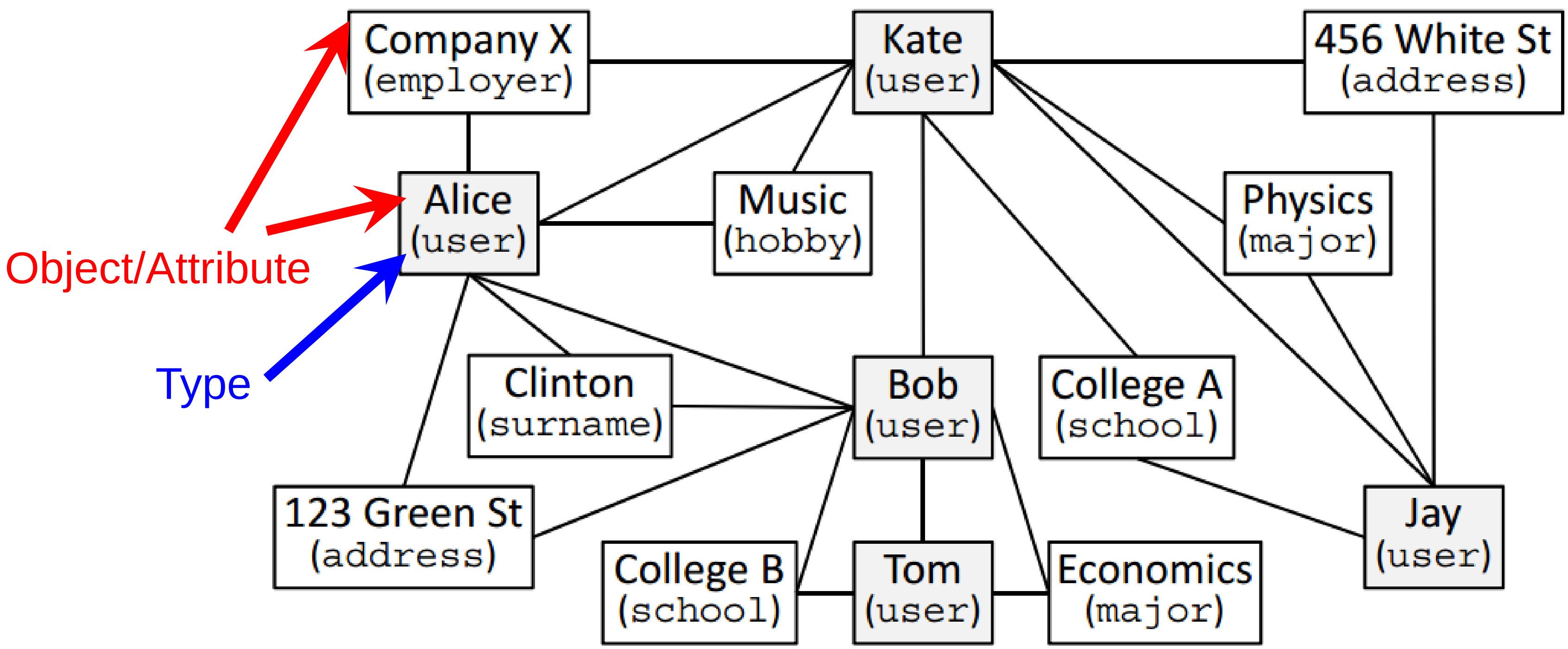
[1] Institute for Infocomm Research, Singapore
[2] Advanced Digital Sciences Center, Singapore
[3] University of Illinois at Urbana-Champaign, USA

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

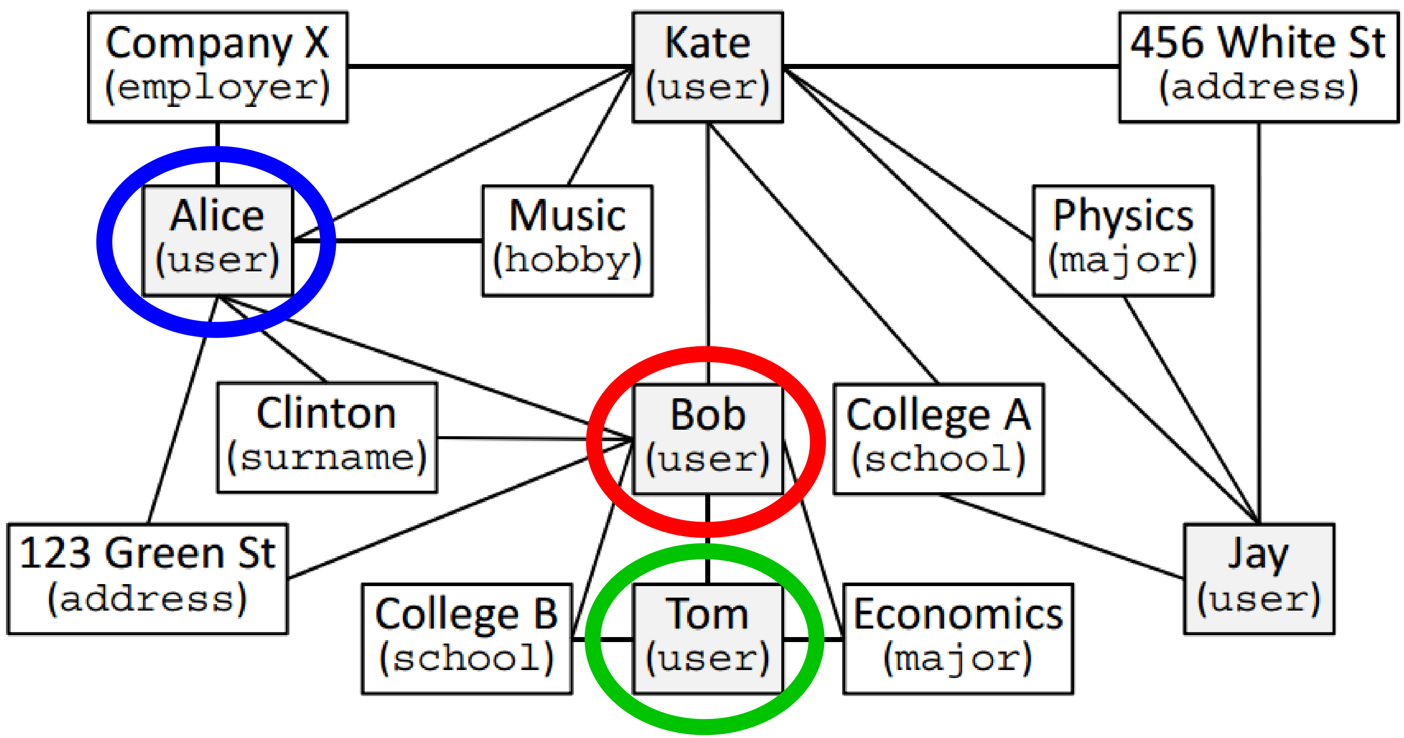


Problem: Semantic Proximity Search

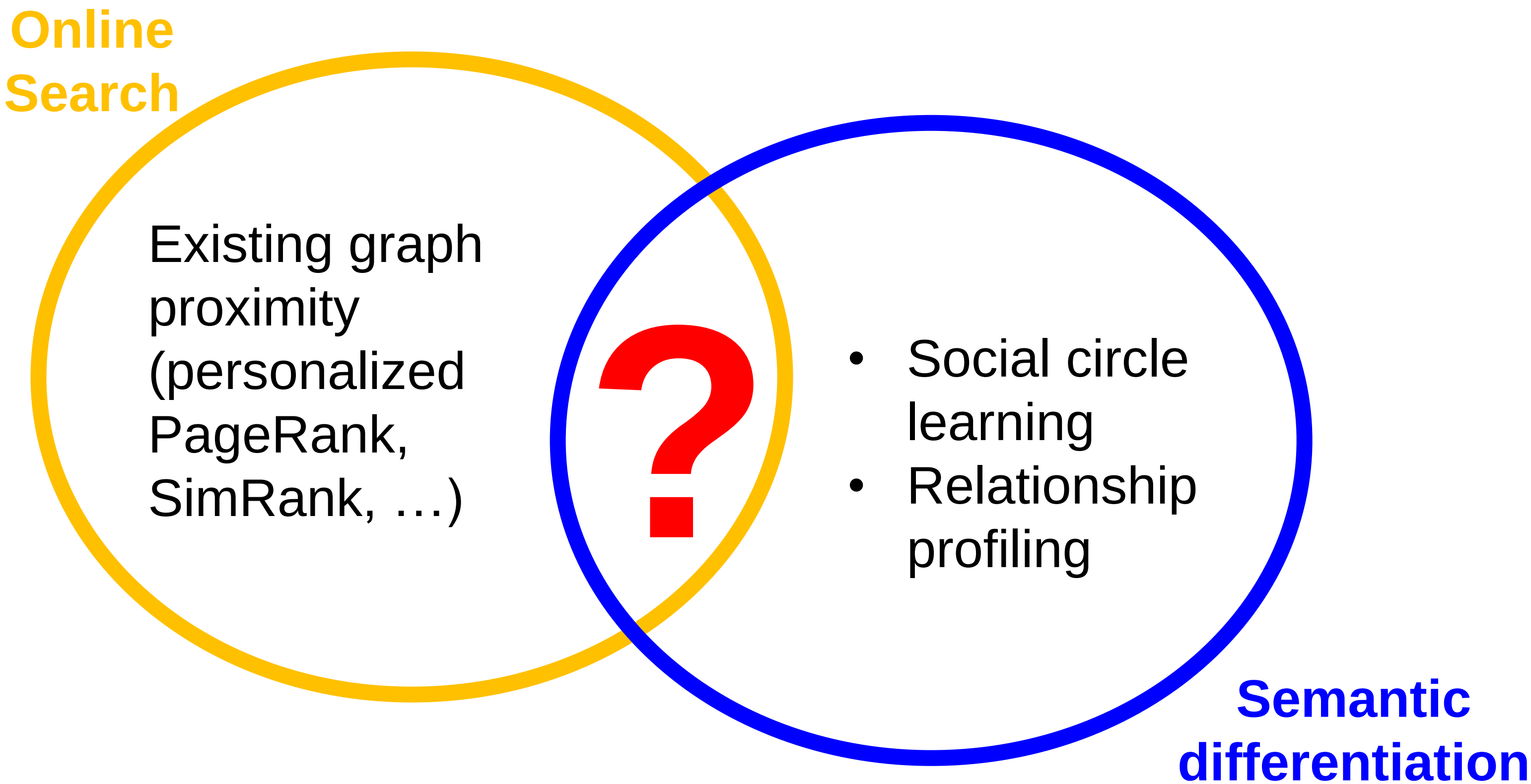
Which users are ~~close to / related~~ to Bob?

Family?

Classmates?



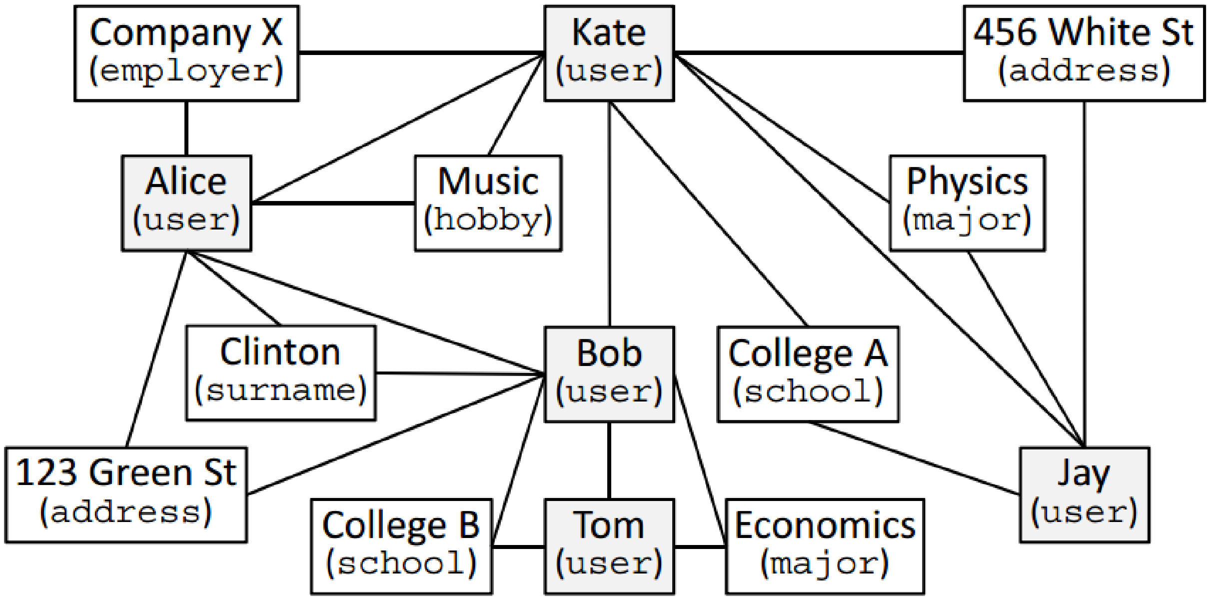
Key Criteria of Solution:
Semantic differentiation + Online Search



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



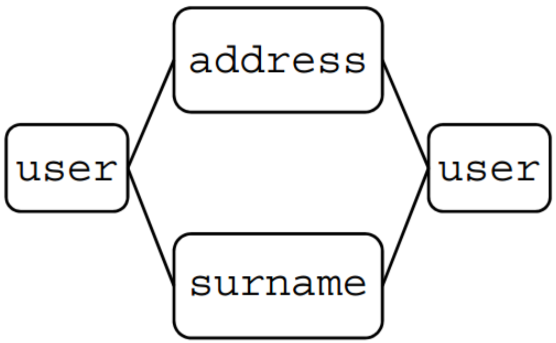
Family: [Bob & Alice / same surname & address]

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

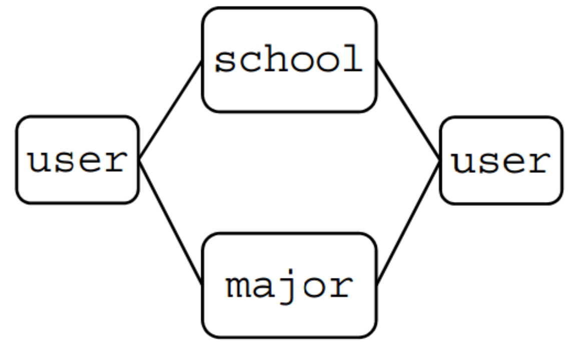
Close friends: [Kate & Alice / same employer & hobby]
[Kate & Jay / roommate]

Insight: common substructures, or *metagraphs*, to "explain" semantic classes

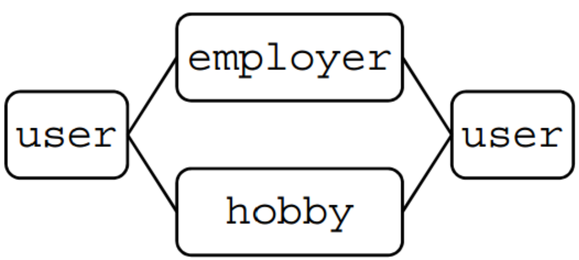
Family
[same surname & address]

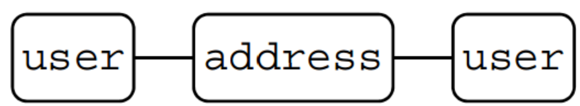


Classmates
[same school & major]

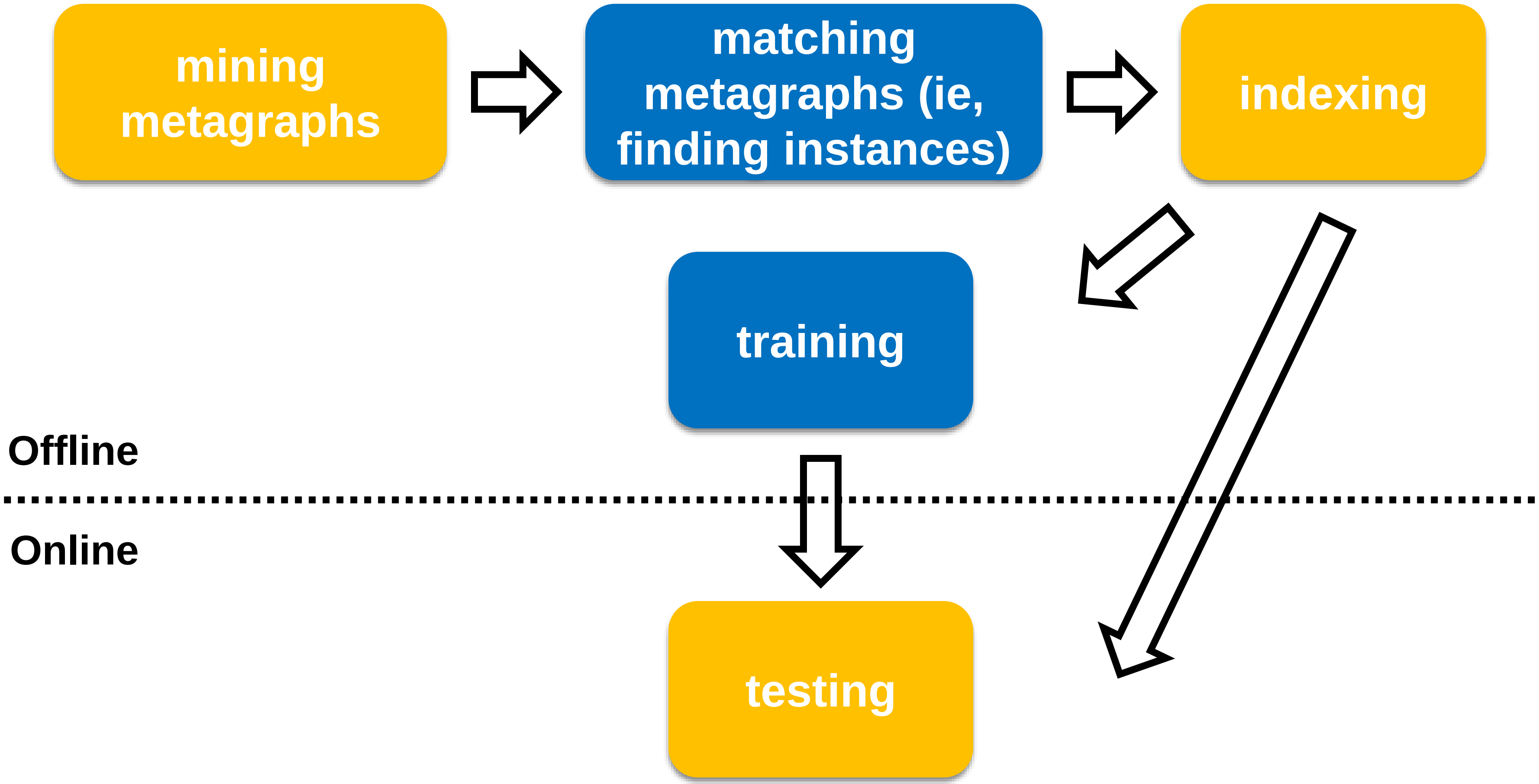


Close friends
[same employer & hobby]
[roommate]





Overall Framework



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

$$\pi(x, y; \mathbf{w}) \triangleq \frac{2\,\mathbf{m}_{xy} \cdot \mathbf{w}}{\mathbf{m}_x \cdot \mathbf{w} + \mathbf{m}_y \cdot \mathbf{w}}$$

co-occurrence not by chance

$\mathbf{m}_{xy}[\mathrm{i}]$ = # times x, y co-occur in instances of metagraph i

$\mathbf{m}_x[\mathrm{i}]$ = # times x occurs in instances of metagraph i

$\mathbf{w}[i]$ = weight for metagraph i

Challenge #1: Meta-graph based proximity (Basic learning model)

- Pairwise learning to rank

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

Each example is a triplet:
for query q, x is ranked before *y.*

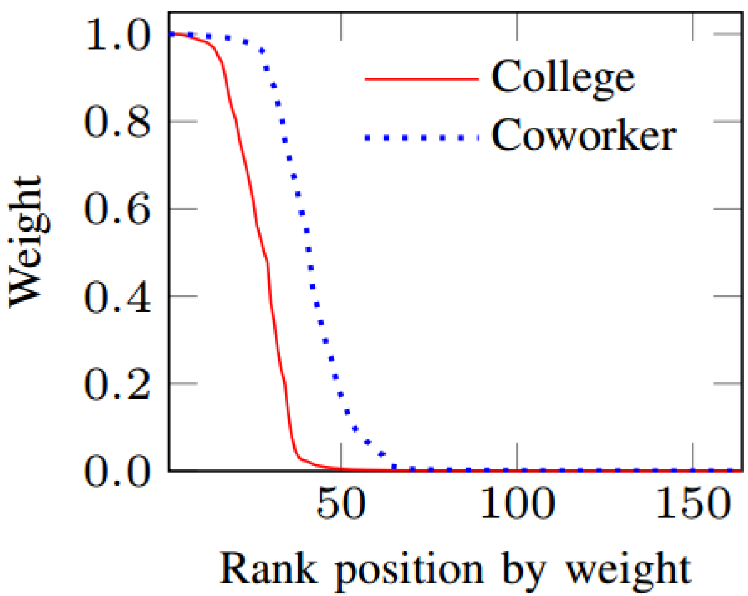
- Objective function

$$L(\mathbf{w}; \Omega) = \sum_{(q,x,y)\in\Omega} \log P(q, x, y; \mathbf{w})$$

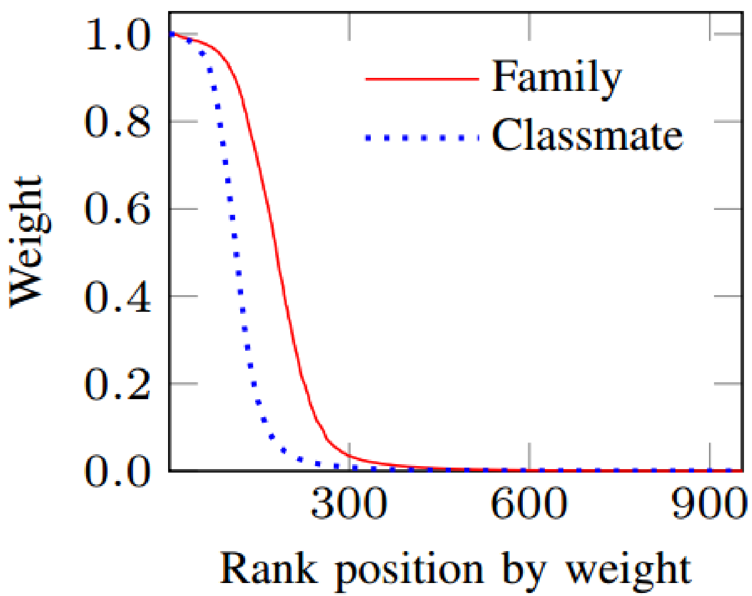
Challenge #1: Meta-graph based proximity (Need for efficient training)

- 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)

Identify seed metagraphs ⇨ **Learn with seed metagraphs**

⇩

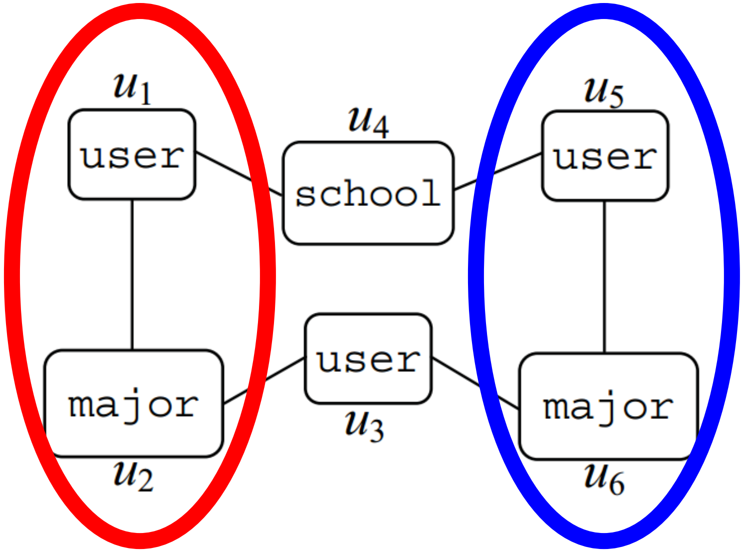
Based on weights of seed metagraphs and their structural relationship with other metagraphs

⇩

Select more metagraphs ⇨ **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

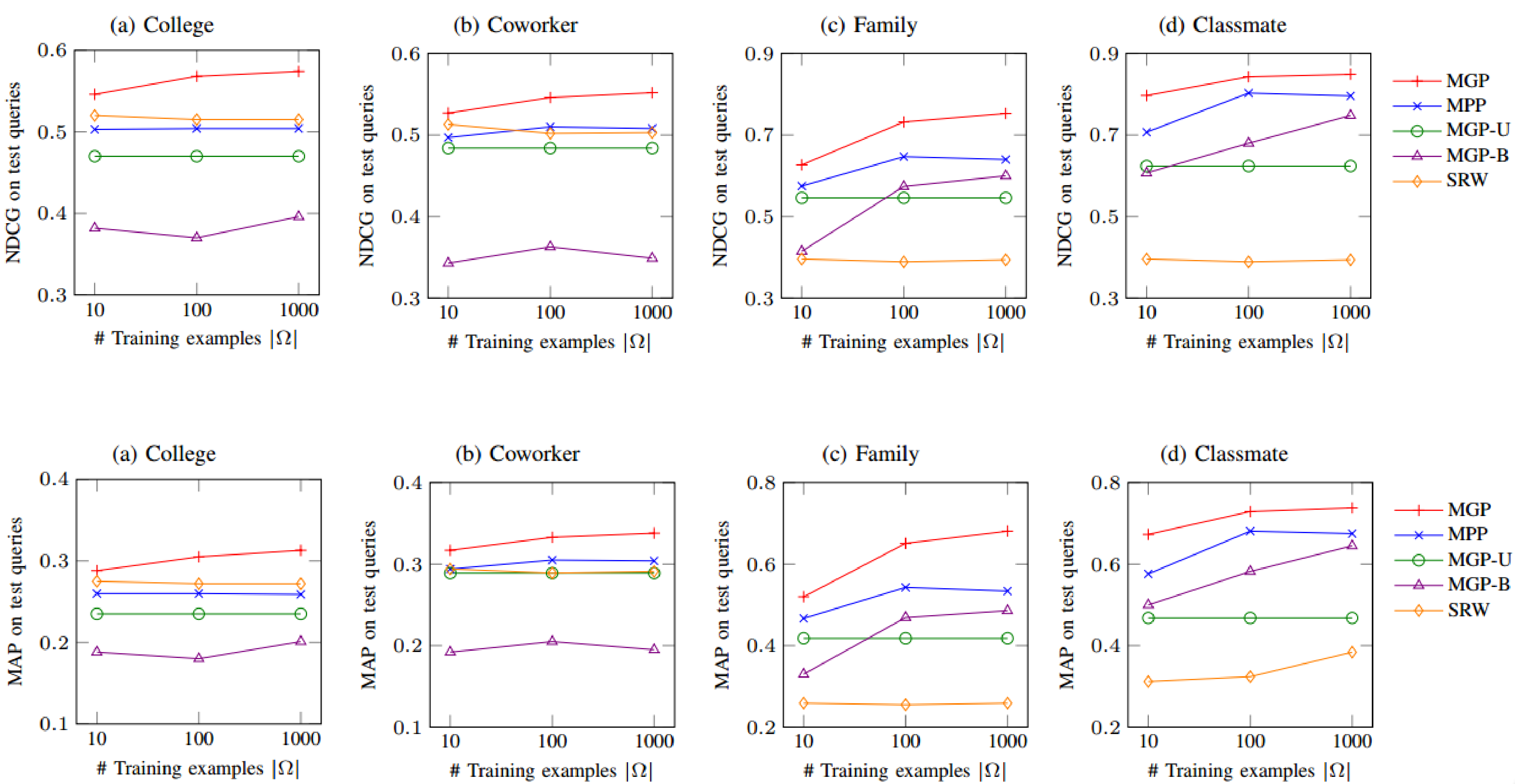
Experiment setup - methodology

- 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

Conclusion

- Metagraphs are powerful
  - May be extended to other tasks on graph
- Matching metagraphs are expensive
  - Improving its efficiency is crucial