ILLINOIS INSTITUTE OF TECHNOLOGY

De-anonymizing Social Networks and Inferring Private Attributes Using Knowledge Graphs

Jianwei Qian

Illinois Tech

Xiang-Yang Li

USTC, Illinois Tech

Chunhong Zhang

BUPT

Linlin Chen

Illinois Tech

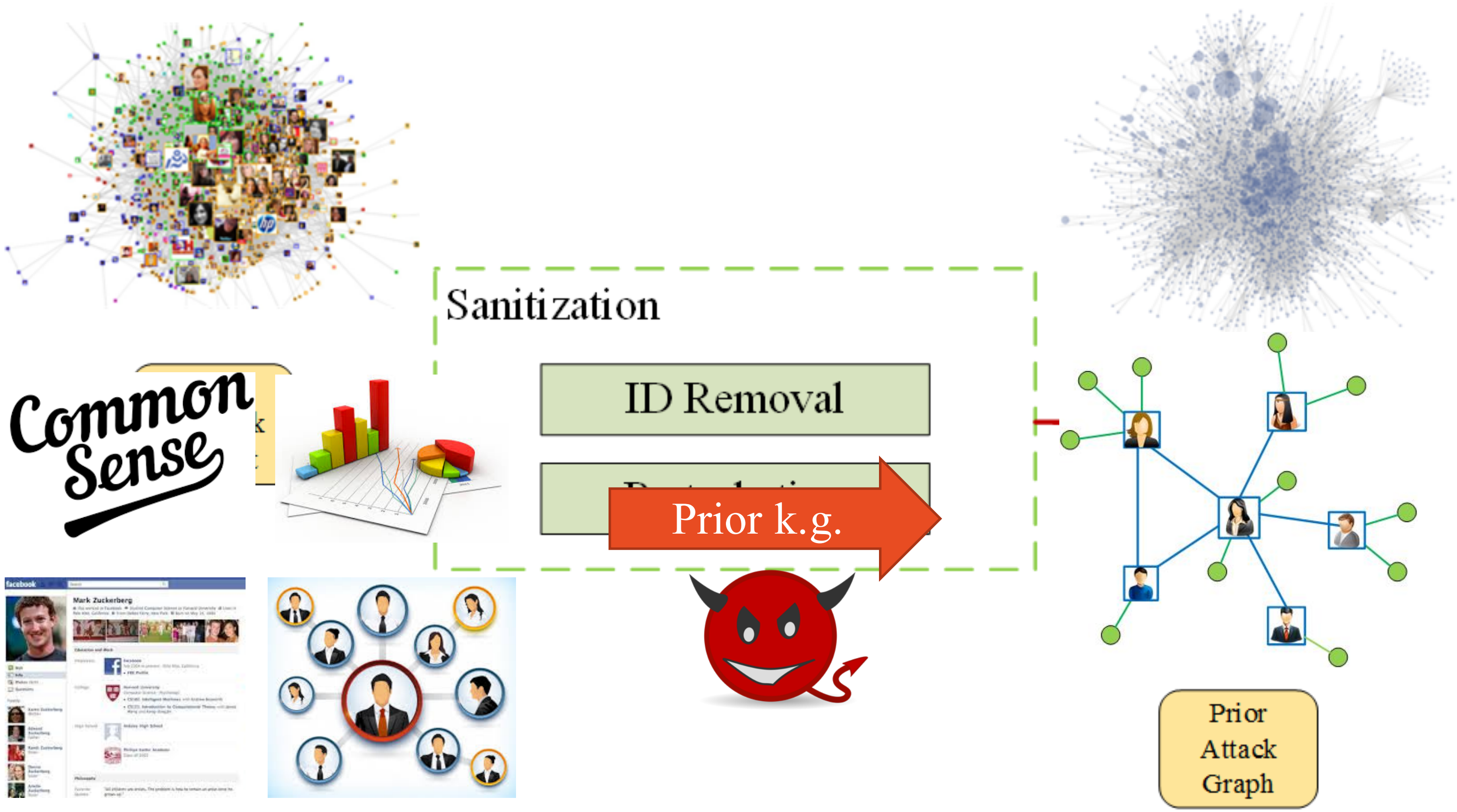
Outline

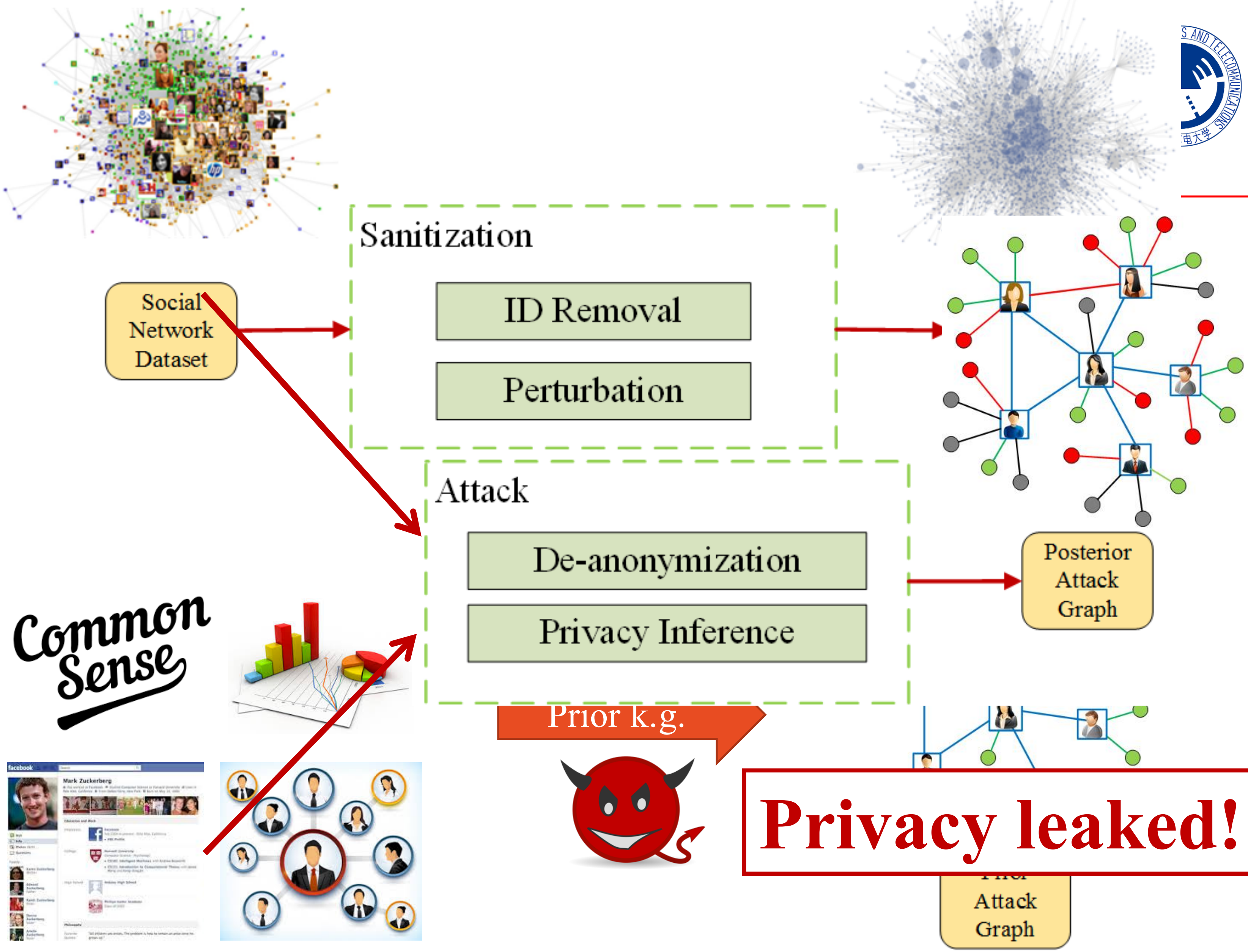
- Background
- Prior Work
- Our Work
- Conclusion

Background

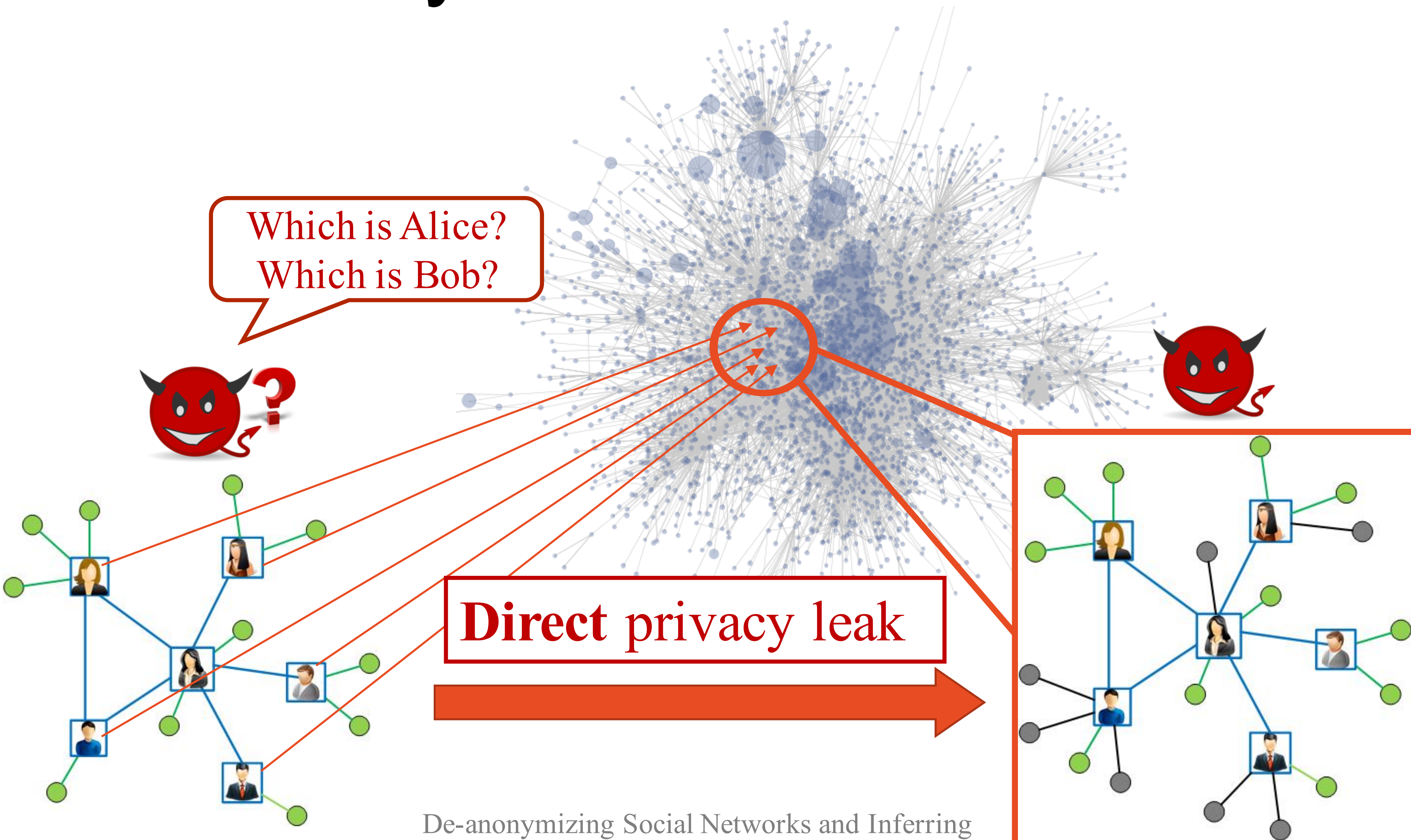
- Tons of social network data
- Released to third-parties for research and business
- Though user IDs removed, attackers with prior knowledge can de-anonymize them. → privacy leak

Attacking Process




Social Network Dataset
Sanitization
ID Removal
Perturbation
Attack
De-anonymization
Privacy Inference
Posterior Attack Graph
Common Sense
Prior k.g.
Privacy leaked!
Attack Graph

Attack Stage 1
De-Anonymization

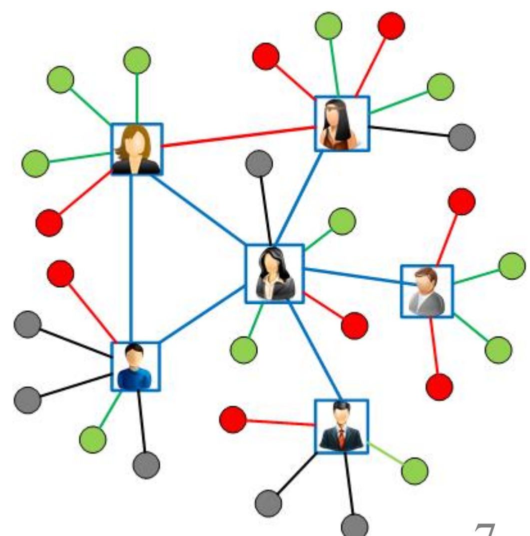


Attack Stage 2
Privacy Inference

- Correlations between attributes/users
  - Higher education => higher salary
  - Colleagues=> same company
  - Common hobbies => friends

- Infer new info that is not published

Indirect privacy leak



What Do We Want to Do?

To understand

How privacy is leaked to the attacker

Outline

- Background
- **Prior Work**
- Our Work
- Conclusion

Prior Work

De-anonymize one user

Never ending!

- Degree attack [SIGM
- 1-neighborhood attack [INFOCOM'13]

e anonymity

- 1-neighborhood anonymity

Assume specific prior knowledge!

- Community re-identification [SDM'11]
- *k*-structural diversity

Prior Work

De-anonymize all the users

– Graph mapping based de-anonymization

[WWW’07, S&P’09, CCS’12, COSN’13, CCS’14, NDSS’15]

Attacker holds an auxiliary SN that overlaps with the published SN

Twitter

Flickr

Limitations

- Assume attacker has specific prior knowledge
  - We assume diverse and probabilistic knowledge

- Focus on de-anonymization only. How attacker infers privacy afterwards is barely discussed
  - We consider it as 2^{nd} attacking step!

Outline

Background

Prior Work

Our Work

Conclusion

Goals

- To construct a comprehensive and realistic model of the attacker's knowledge
- To use this model to depict how privacy is leaked.

Challenges

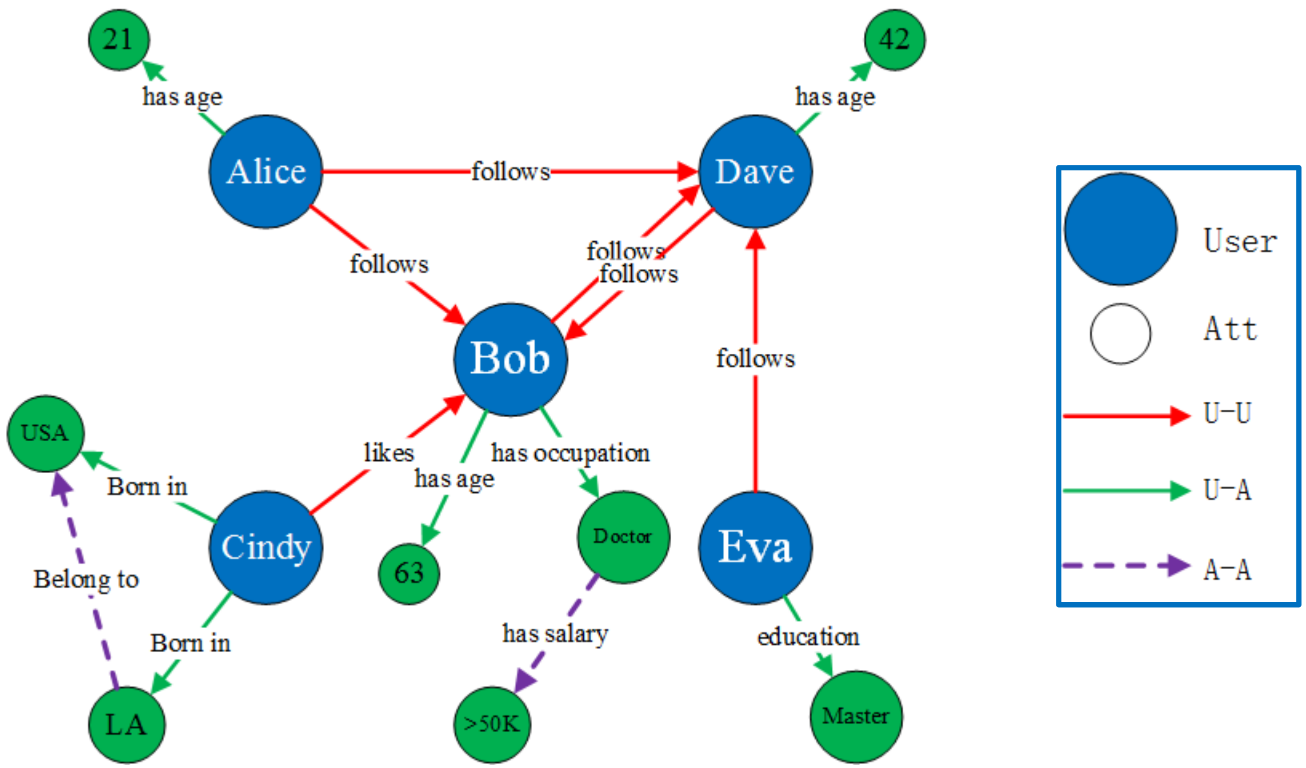
- Hard to build such an expressive model, given that the attacker has various prior knowledge
- Hard to simulate attacking process, since the attacker has various techniques

Solution

Use knowledge graph to model attacker's knowledge

Knowledge Graph

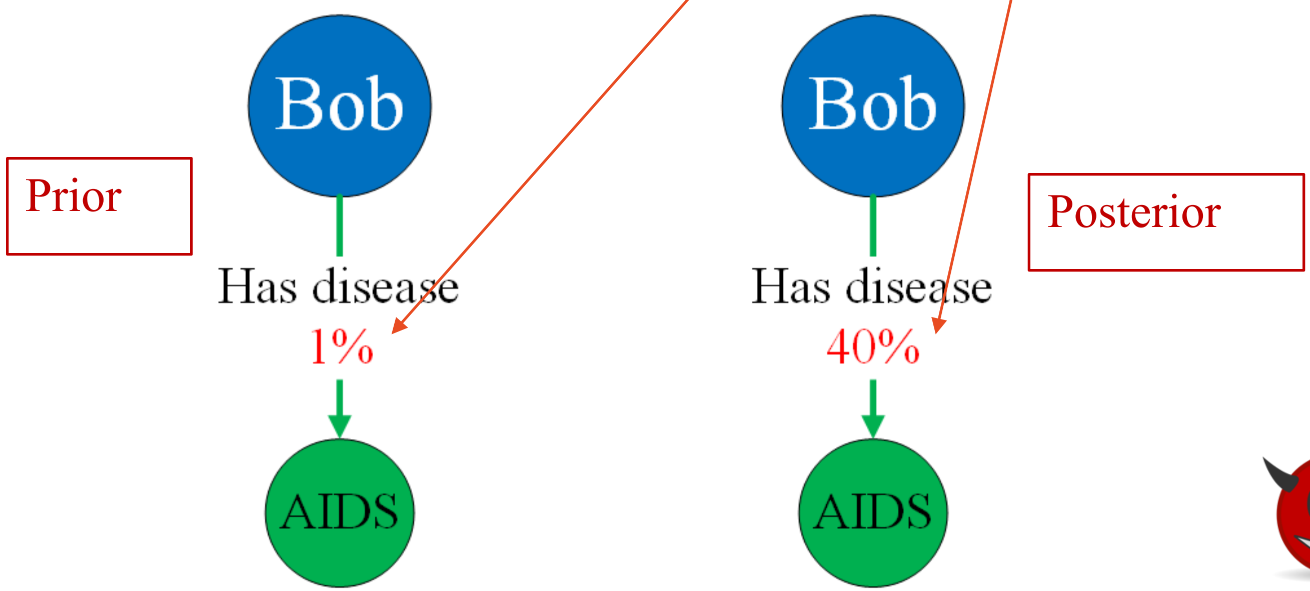
- Knowledge => directed edge
- Each edge has a confidence score



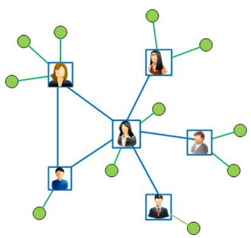
What's Privacy?

- Every edge is privacy
- Privacy is leaked when $\left|c_p(\mathrm{e}) - c_q(e)\right| > \theta(e)$

Say 30%



De-Anonymization



Prior knowledge
G_p

Mapping π

Anonymized graph
G_a

$$\operatorname{argmax} Sim_{\pi}(G_p, G_a)$$

$$Sim_{\pi}(G_p, G_a) = \sum_{(i,j)\in\pi} S(i,j),$$

S is node similarity function

Node Similarity

- Attribute Similarity
  - Use Jaccard index to compare attribute sets
- Relation similarity
  - Inbound neighborhood
  - outbound neighborhood
  - *l*-hop neighborhood

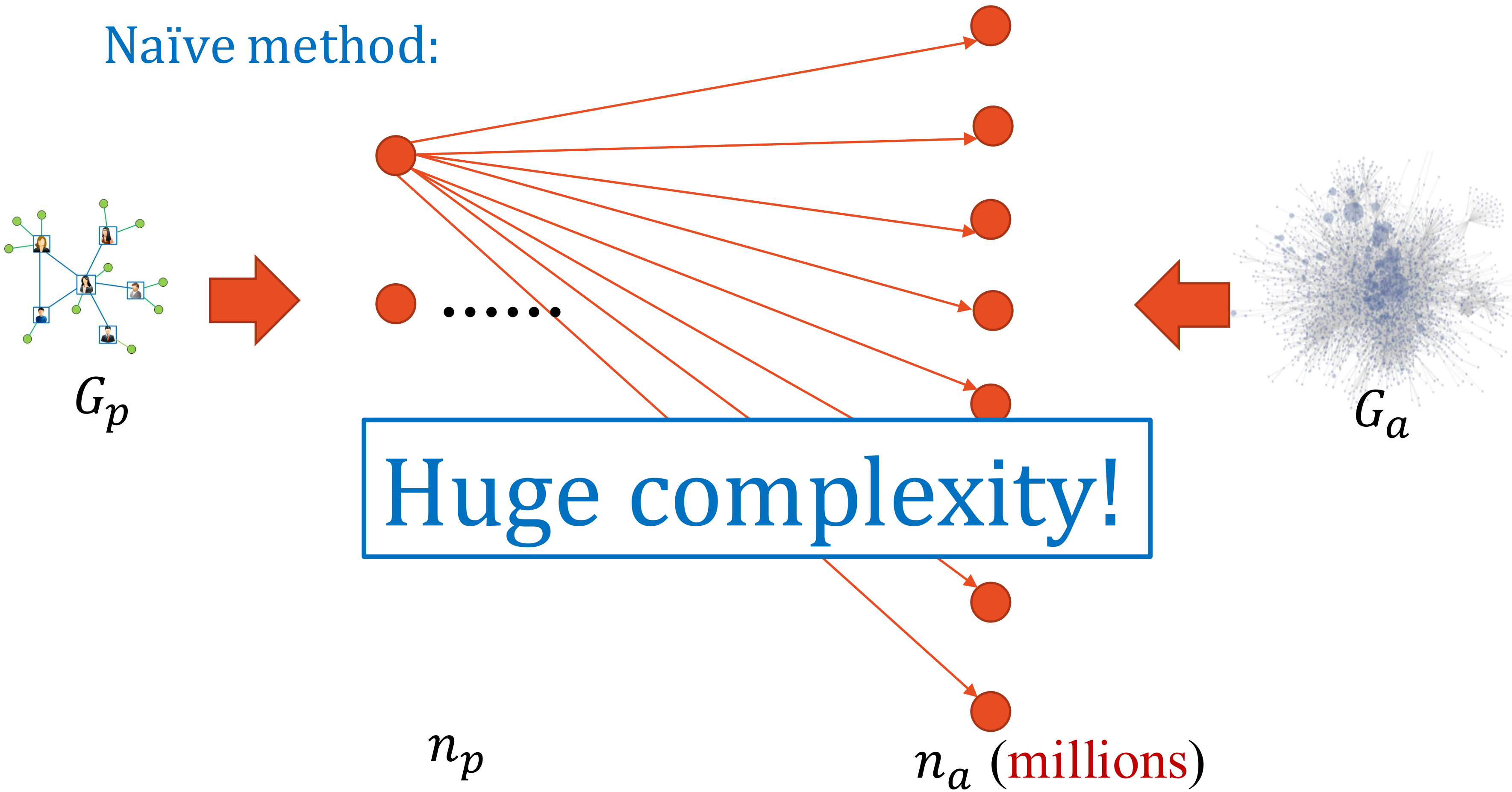
$$S_R(i,j) = w_i S_i(i,j) + w_o S_o(i,j) + w_l S_l(i,j)$$

$$S(i,j) = w_A S_A(i,j) + (1 - w_A) S_R(i,j)$$

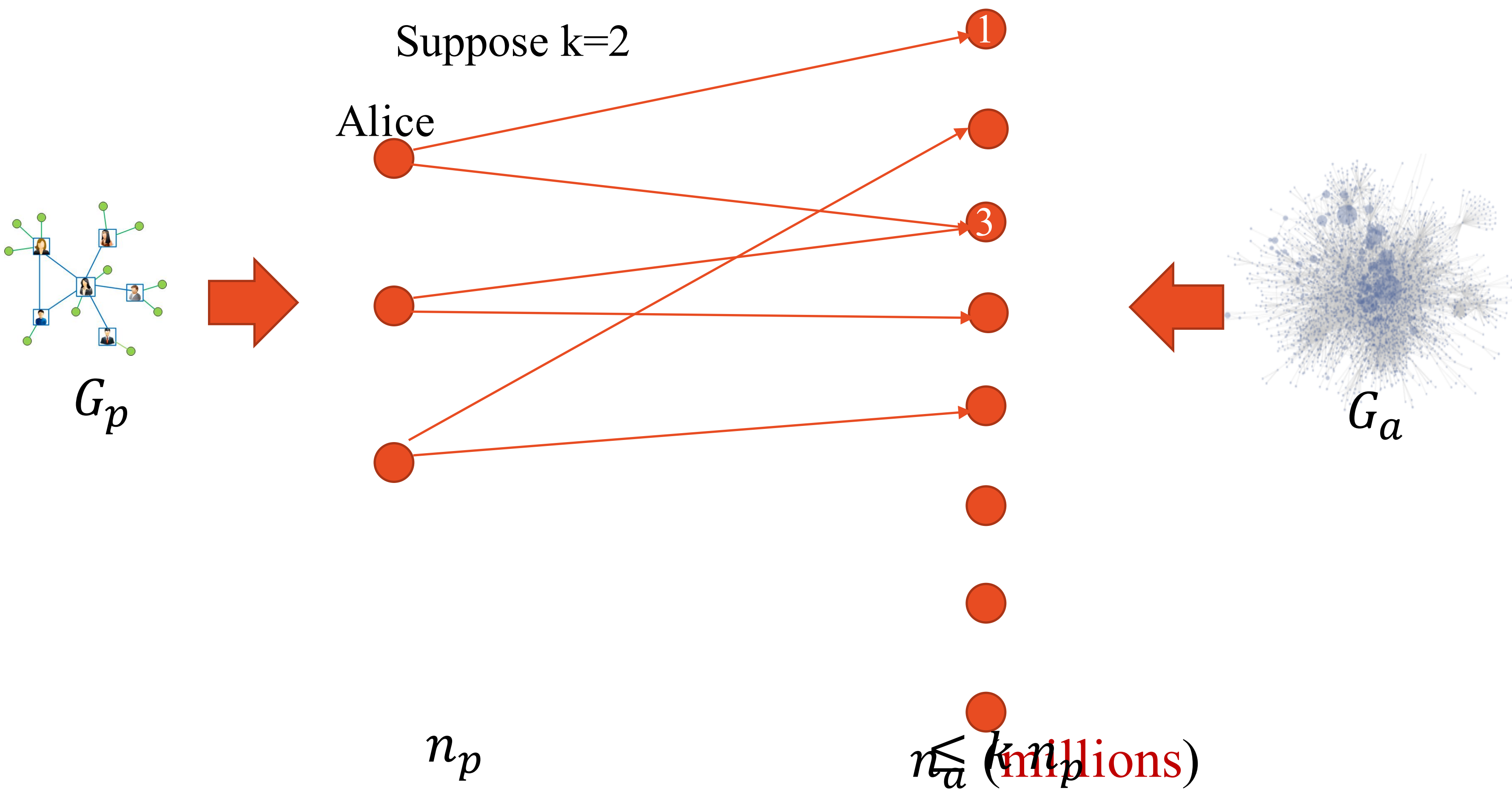
Problem Transformation

Mapping => Max weighted bipartite matching

Naïve method:

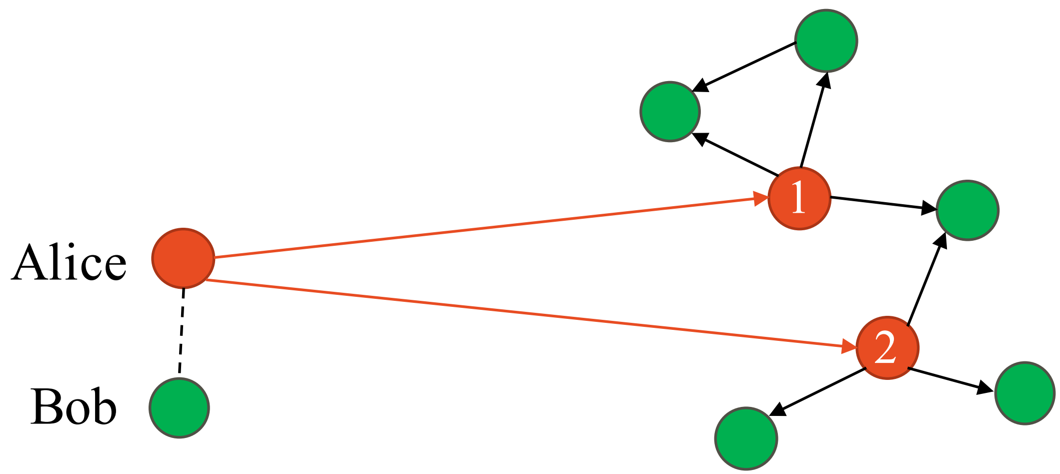


Top-*k* Strategy



How to Choose Top-k Candidates?

- Intuition
  - If two nodes match, their neighbors are also very likely to match.



- Perform BFS on G_p

Complexity Analysis

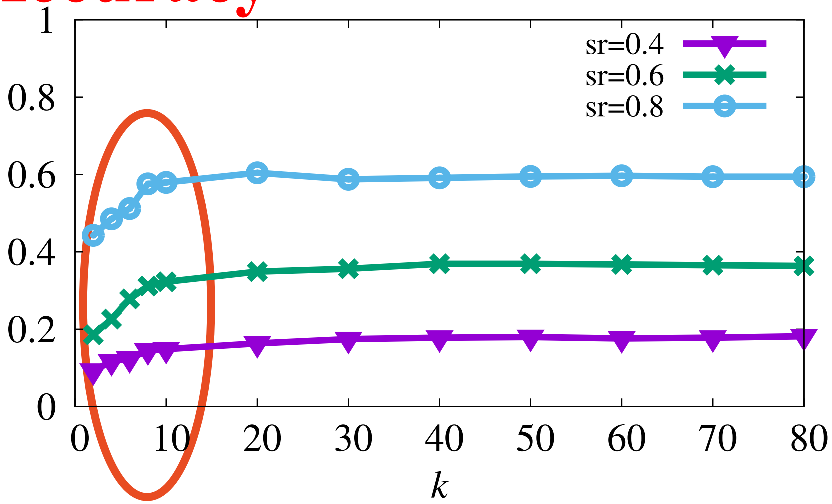
	Time		Space
	Building Bipartite	Finding Matching	
Naïve method	$n_p n_a$	$O\left((n_p + n_a)n_p^2 n_a\right)$	$O\left((n_p + n_a)^2\right)$
Top-k strategy	$\ll n_p n_a$	$O(k^2 n_p^3)$	$O(k^2 n_p^2)$

Complexity greatly reduced!

Tradeoff

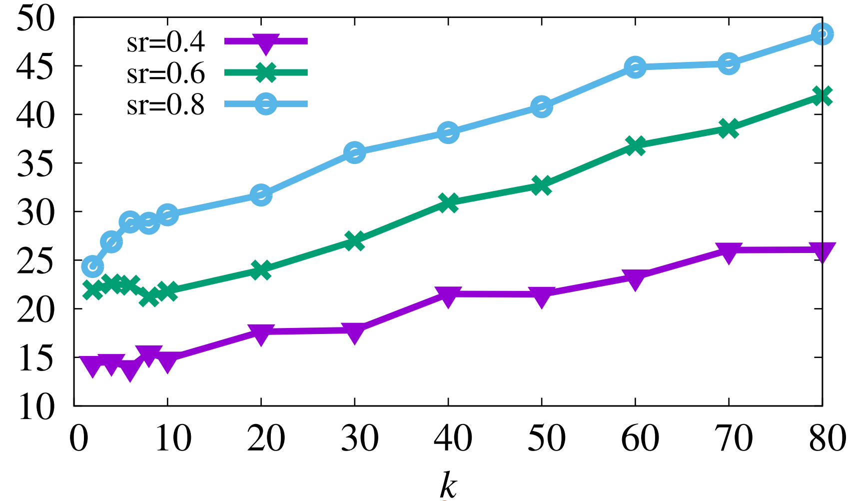
- k balances accuracy and complexity
- $k = 10$ is enough to achieve high accuracy

Accuracy



k

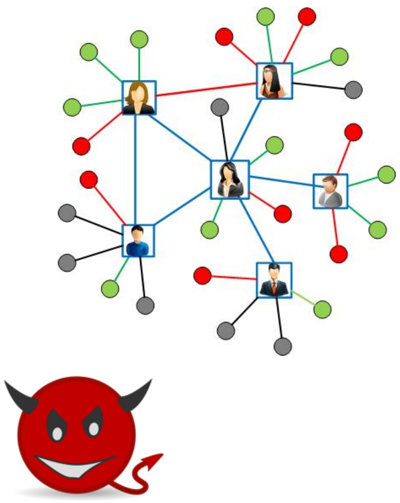
Time

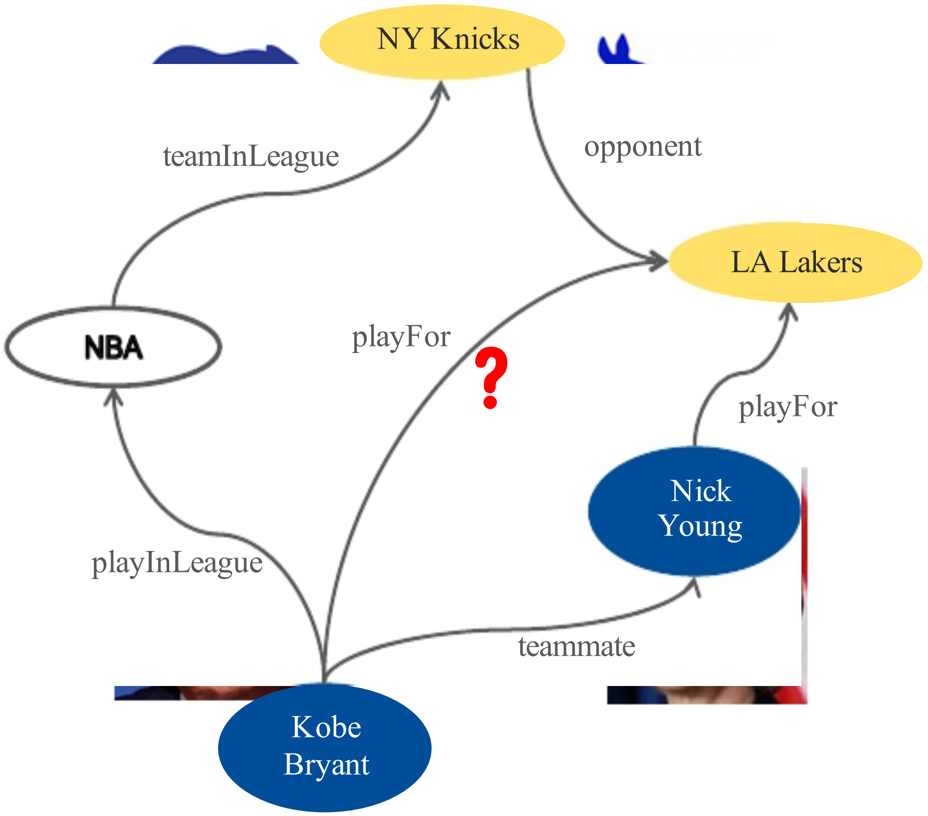


k

Privacy inference

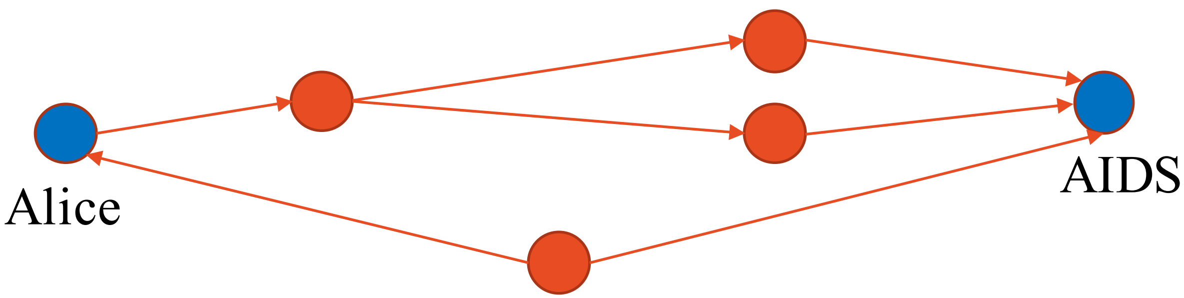
Predict new edges in knowledge graph





Path Ranking Algorithm

- Proposed by Ni Lao *et al.* in 2011 for a different topic



- Correlations => "rules" => paths
- Logistic regression

Experiments

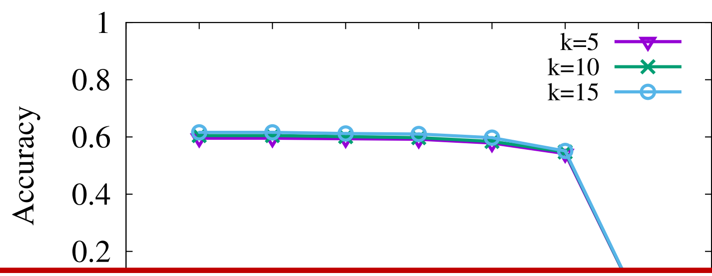
- Datasets
  - Google+, Pokec

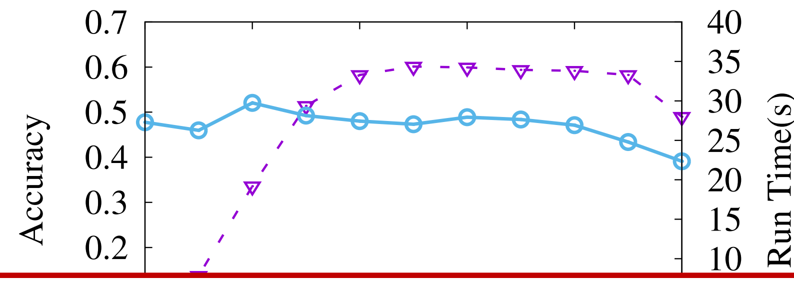
Dataset	$\lvert\mathcal{V}^U\rvert$	$\lvert\mathcal{V}^A\rvert$	$\lvert\mathcal{E}^{UU}\rvert$	$\lvert\mathcal{E}^{UA}\rvert$	$\lvert\mathcal{E}_p^{AA}\rvert$
Google+	107,614	15,691	13,673,453	378,880	2,262
Pokec	306,568	576	2,822,492	1,532,840	38

- Steps
  - Generate G_a
  - Generate G_p
  - Run the algorithms

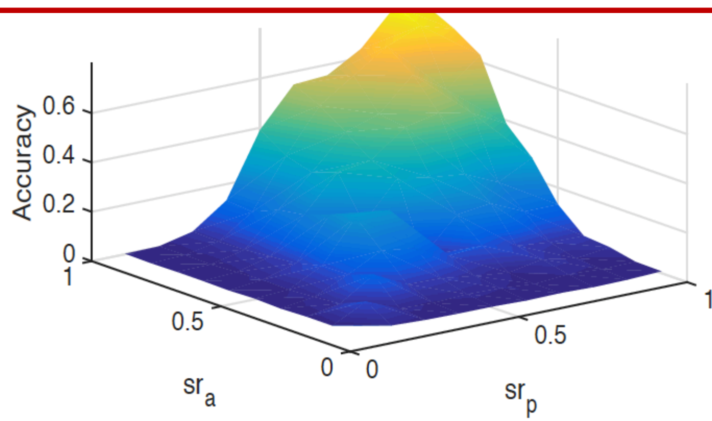
De-Anonymization Results

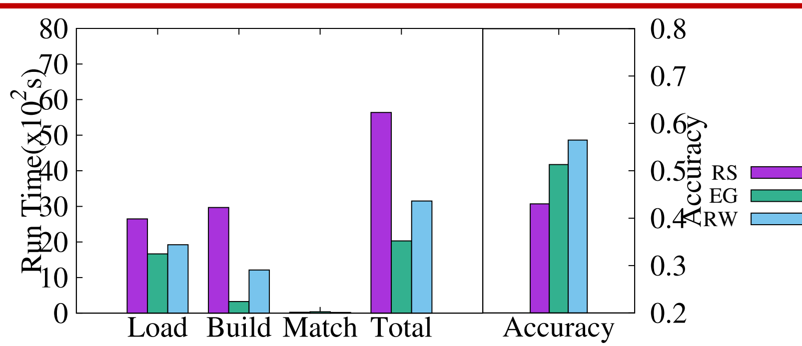
Metrics: accuracy, run time





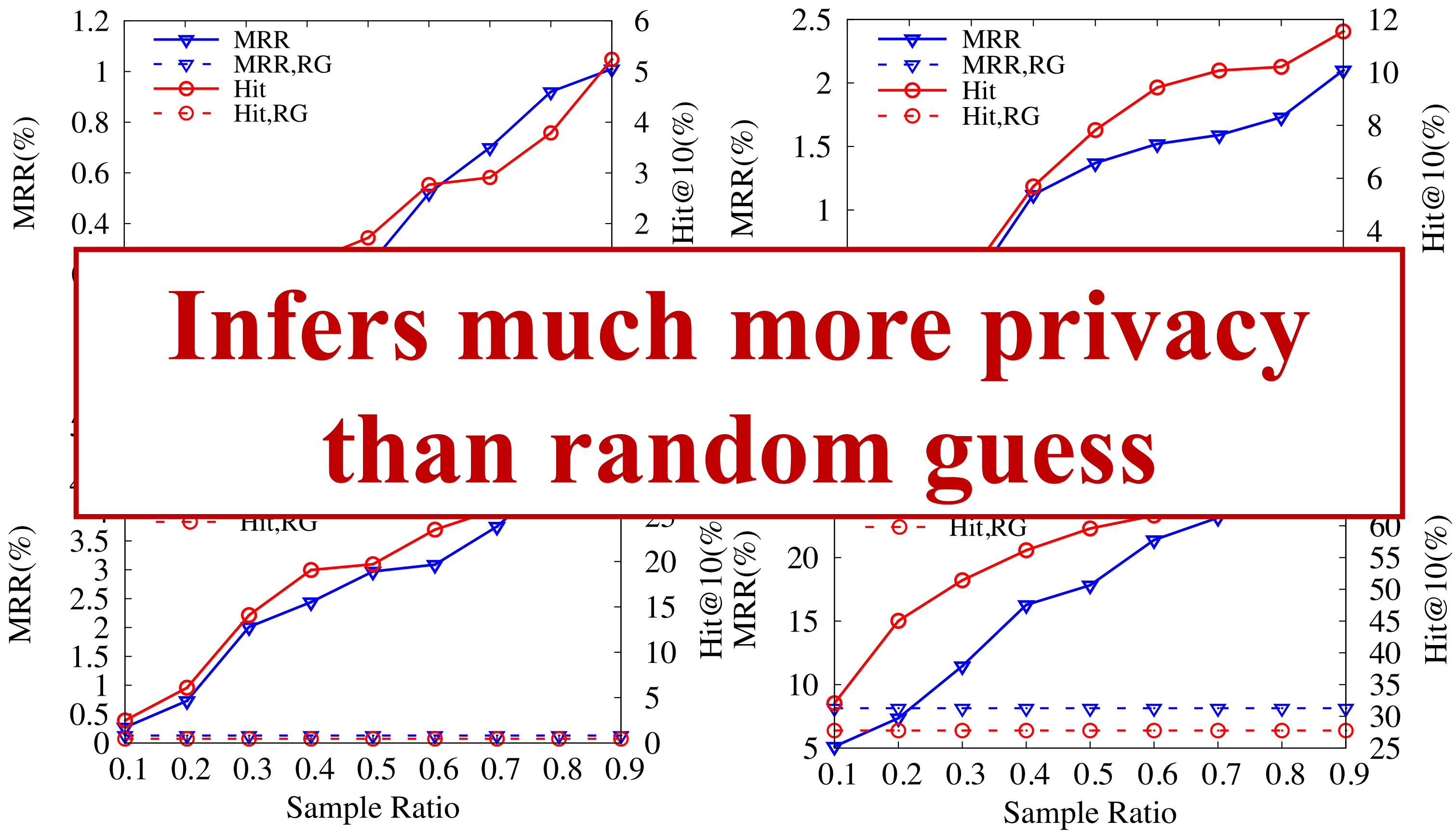
De-anonymize about 60% of users





Privacy Inference Results

Metrics: hit@k, MRR (*Mean reciprocal rank*)



Infers much more privacy than random guess

Outline

- Background
- Prior Work
- Our Work
- **Conclusion**

Conclusion

We have

- Applied knowledge graphs to model the attacker's prior knowledge
- Studied the attack process: de-anonymization & privacy inference
- Designed methods to perform attack
- Done simulations and evaluations on two real world social networks

Future work

- Effective construction of the bipartite for large scale social networks
- Impact of adversarial knowledge on de-anonymizability
- Fine-grained privacy inference on the knowledge graph

Thank you!

Jianwei Qian
jqian15@hawk.iit.edu
https://sites.google.com/site/jianweiqianshomepage