Graphs

• Fundamental abstractions for many real-world situations
  – E.g. Internet topology, social networks, biological networks
Studying Real Graphs

• Research community: critical for progress in many areas
  – Understand underlying structure and process
  – Validate models and theories
  – Reveal hidden or unknown properties

• Application designers
  – Validate new algorithms and features
  – Provide guidance on future design
Challenges of Obtaining Real Graphs

• Most complete graphs are only available by data owners
  – e.g. Facebook, AS networks

• Crawling: far from ideal solutions
  – Many data are not crawlable
  – Requires significant resources
    • Time, machines
  – Partial information leads to inaccurate conclusions

• Graph data sharing is an alternative
  – SNAP library: graph datasets shared online by SNAP group[1]

Risks of Graph Data Sharing

- Graph data contains sensitive information
  - User identities (nodes), interactions (edges)
  - Private meta data information
    - e.g. Religion, age

- Protect graph privacy by naïve anonymization
  - Remove all meta data, leaving only graph structure
  - Not enough: examples of privacy leaks
    - AOL dataset, 2006: 650K user search histories[1]
    - Netflix dataset, 2008: 480K user movie viewing histories (led to lawsuits)[2]

Why Naïve Anonymization Failed?

• Easily broken by using *external* data

• Kleignberg’s active attacks\(^1\)
  – Embed adversary information (a fake subgraph)
  – Identify individuals by locating the fake subgraph

• Narayanan’s de-anonymization\(^2\)[3]
  – Use public data to cross compare

**Need to protect graph privacy!**

---

\(^1\)Backstrom, Dwork and Kleinberg. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. WWW 2007

\(^2\)Narayanan and Shmatikov. Robust de-anonymization of large sparse datasets. Security and Privacy 2008

\(^3\)Narayanan and Shmatikov. De-anonymizing social networks. Security and Privacy 2009
Outline

• Motivation
• Background
• Research Directions
  – k-anonymity Based Modification
  – Graph Randomization
  – Cluster-based Generalization
  – Cryptographic Approaches
• Open questions
Targeted Scenarios

- Conventional graph sharing
  - Public sharing: graphs are public available
    - SNAP library
  - Private sharing: graphs shared with collaborators
    - Facebook share data with third-party applications

- Graph query answering
  - Graph not shared
  - Only queries allowed on the graph through a controlled interface
Definitions

• **Graph privacy breaches**\[1\]\[2\]
  – Identity
  – Link
  – Meta data

• **Graph utility**
  – Structural loss
    • # of changes
    • Probability of error when reconstructing
    • Property-based measurement
  – Meta data loss

Preserve Graph Privacy

• Goals
  – Prevent certain privacy breaches
  – Minimize graph utility loss

• Challenges
  – Graphs contain complex details
  – Graphs are often highly dynamic
    • Growing and changing networks
  – Scalability
    • Facebook: 1.2B active users per month
Existing Solutions

• Public graph sharing: focus on operations on the graph
  – Add noise to the graph
    • k-anonymity based modification
    • Graph randomization
  – Summarize the graph
    • Clustering-based generalization

• Private graph sharing: focus on data controlling mechanism
  – Cryptographic approaches
k-anonymity Based Modification
What is k-anonymity?

- A privacy definition: each individual cannot be distinguished from at least k-1 others
  - Prevent individuals from being uniquely identified

k-anonymity to Protect Degree Distribution

• Assumption
  – Adversaries know *degree* of a target node
  – Use degree to narrow down targets to disclose identity

• Targeted output
  – Every node has the same degree with at least k-1 other nodes

• Algorithm
  – Extract degree sequence
  – Make up k-anonymous degree sequence
  – Modify the original graph accordingly

Other k-anonymity Methods

• Differ in adversary assumptions
  – Applied to: direct neighborhood\textsuperscript{[1]}, any surrounding subgraph\textsuperscript{[2][3]}
  – Usually aim to prevent identity disclosure

Pros
- Intuitive
- Easy to verify

Cons
- Specific assumptions on adversaries
- High graph utility loss
- High computational complexity to minimize modification

\textsuperscript{[1]} Zhou and Pei. Preserving privacy in social networks against neighborhood attacks. ICDE 2008
\textsuperscript{[2]} Zou and Chen. K-automorphism: A general framework for privacy preserving network publication. VLDB 2009
\textsuperscript{[3]} Cheng, Fu and Liu. K-isomorphism: privacy preserving network publication against structural attacks. SIGMOD 2010
Graph Randomization
Graph Randomization

• More general: little assumption on adversary

• Randomization techniques
  – Random edge modification
  – Produce synthetic graphs similar to original graphs

Basic Randomization

• Strategies
  – Rand Add/Del
    • Add and delete edges
    • Preserve total # of edges
  – Rand Switch
    • Switch endpoints of a pair of edges
    • Also preserve degree

• Discussion
  – Computationally efficient
  – Significant modification in graph structure

Structure-Preserving Randomization

• Motivation: improve graph utility by remaining certain features

• An example: spectrum-preserving randomization
  – Aim to protect eigenvalues
  – Apply basic randomization only to edges that preserve the target eigenvalues

Pro
  Simple and usually efficient

Con
  Probabilistic manner, not guarantee individual privacy


MAE 12/09/14  Qingyun Liu
Measurement-Calibrated Graph Model

- Motivation: provide synthetic graphs that remain features

- Framework

[1] Sala, Cao, Wilson, Zablit, Zheng and Zhao. Measurement-calibrated graph models for social network experiments. WWW 2010
Calibrating Models

• Determine optimal model parameters
  – Generate synthetic graphs
  – Measure graph similarity
  – Search parameter space

• Privacy concerns for generated synthetic graphs
  – What if they are too similar to the original graph?
  – How to guarantee privacy?
  – How to generate private synthetic graphs?
Generate Private Synthetic Graph

• Solution: differential privacy
  – A numerical metric to quantify privacy[1]
  – Provide privacy guarantees on the synthetic graphs[2][3]

• Framework

Pros
  Do not reveal real data, with privacy guarantee
  Much easier for large-scale data sharing

Con
  Difficult to capture all graph properties in synthetic graphs

Clustering-based Generalization
Method Framework

• Motivation: use graph summarization
  – Hide privacy details by publishing only *aggregated* information
  – Still useful to study macro-properties

• Framework
  – Construct a super graph
  – Publish information about the super graph
  – If needed, generate random sample instances from the generalization for detailed analysis

Hay’s Clustering Method

• Scenario
  – Consider only graph structure
  – Prevent identity disclosure

• Method
  – Choose a partition of nodes
  – Build the super-graph

• Published data
  – Super graph structure
  – Some statistic information


MAE 12/09/14
Campan’s Clustering Method

• Scenario
  – Nodes have meta data, edges don’t
  – Prevent identity and meta data disclosure

• Method
  – Structural generalization: similar to Hay’s
  – Meta data generalization
    • Categorical, numerical

Pros
  Widely applied, especially multiple disclosures may happen
  Low risk for individual privacy

Con
  Maintain only macro-properties, graph utility is doubted

MAE 12/09/14 Qingyun Liu
Cryptographic Approaches
Cryptographic Approaches

• Private sharing privacy breaches: also aggregated information
  – Graph structure, statistics

• Emphasis on controlling mechanism
  – Data access: access control
  – Data exchange: secure computation

Access Control Protocols

• Motivation
  - manage a resource so that it is accessible only to authorized users

• Framework

1. Resource request

2. Access rules

3. Provide proof

4. Ensure the proof and give access

MAE 12/09/14  Qingyun Liu
Secure Computation

• Motivation
  – Multi parties with their own graphs
  – Want to jointly compute a function, while keeping their graphs private

• Graph function
  – Calculate geometric problems
    • Convex hull
  – Graph algorithm

Pros
Graph data often unchanged
Orthogonal direction to previous methods

Cons
Never foolproof, e.g. social engineering
Often high computation complexity

Blanton, Steele and Alisagari. Data-oblivious graph algorithms for secure computation and outsourcing. ASIACCS 2013
Outline

• Motivation
• Background
• Research Directions
• Open Questions
Thank you!

Any Questions?
Backup Slides
Beyond K-anonymity

- Limitation of k-anonymity: can still cause meta data disclosure

- L-diversity: at least L different values of meta data

- T-closeness: distance between local meta data distribution and overall distribution <= t

[2] Li, Li and Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. ICDE 2007