

Anonymization and Uncertainty in Social Network Data

Graham Cormode

graham@research.att.com

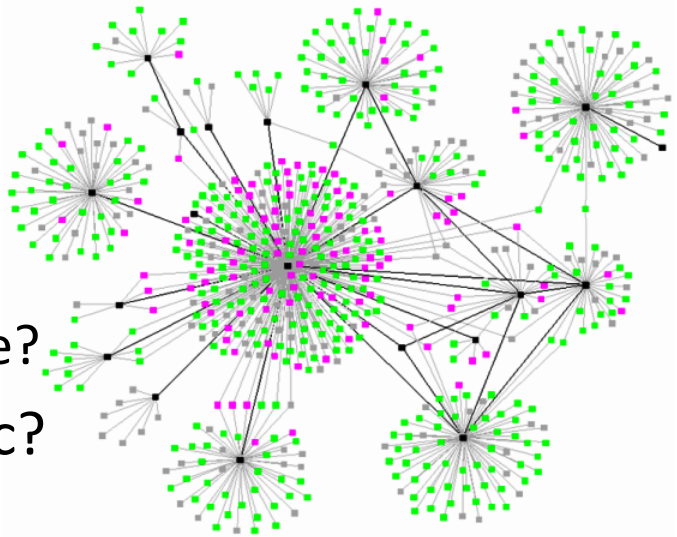
Joint work with Smriti Bhagat,
Balachander Krishnamurthy, Divesh Srivastava



at&t

Social Network Data

- Online Social Networks (OSNs) store much detailed personal data
  - Hundreds of millions use Facebook, LinkedIn, Twitter etc.
  - Demographic information about individuals, their likes, dislikes
  - Link information: friendship links, “wall posts”, comments etc.
- Many natural queries on social graphs:
  - How many users in subpopulations? (age range, location, interest groups)
  - What subpopulations are interacting?
  - How do these patterns change over time?
- …but isn’t data on social networks public?
  - No! Most OSNs have privacy controls
  - Many “private” social networks e.g. email



at&t

Woman 'sacked' on Facebook for complaining about her boss after forgetting she had added him as a friend

By JULIE MOULT
Last updated at 12:26 AM on 15th August 2009

News › Technology › Facebook

Fugitive caught after updating his status on Facebook

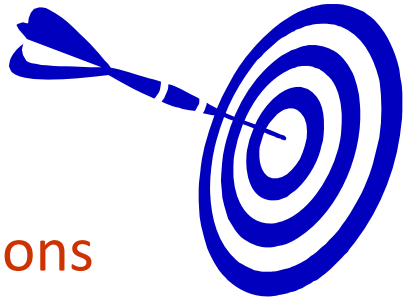
Maxi Sopo told his Facebook friends, including a former justice department official, he was living in paradise in Mexico

Although Sopo's profile was set to private, his list of friends was not. Scoville started combing through it and was surprised to see that one friend listed an affiliation with the justice department. He sent a message requesting a phone call.

Anonymization as a tool for research

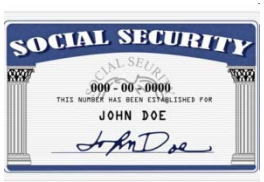
- For Data Sharing
  - Give real(istic) data to others to study without compromising privacy of individuals in the data
  - Allows third-parties to try new analysis and mining techniques not thought of by the data owner
- For Data Retention and Usage
  - Various requirements prevent companies from retaining customer information indefinitely
  - E.g. Google progressively anonymizes IP addresses in search logs
  - Internal sharing across departments (e.g. billing → marketing)

Objectives for Anonymization



- Prevent (high confidence) inference of associations
  - Prevent inference of salary for an individual in census data
  - Prevent inference of individual's activity in social network data
- Prevent inference of presence of an individual in the data set
  - Satisfying "presence" also satisfies "association" (not vice-versa)
  - Presence in a data set can violate privacy (eg STD clinic patients)
- Have to model what knowledge might be known to attacker
  - Background knowledge: facts about the data set (X has salary Y)
  - Domain knowledge: broad properties of data (illness Z rare in men)

Trivial Anonymization



- Trivial anonymization strips out identifying fields
  - Social Security Numbers (SSNs) and other unique identifiers
  - I.e. remove the keys to making "joining" impossible
- Unfortunately, this is not enough to prevent reidentification
  - Additional fields in the data can still identify individuals
  - DOB+Sex+ZIP unique for majority of US Residents [Sweeney 02]
- Trivial anonymization criticized from legal perspective [Ohm 09]

Examples of Anonymization

- US Census: information about every US household
  - Who, where; age, gender, racial, income and educational data
  - aggregated to regions (Zip code), released in full after 72 years

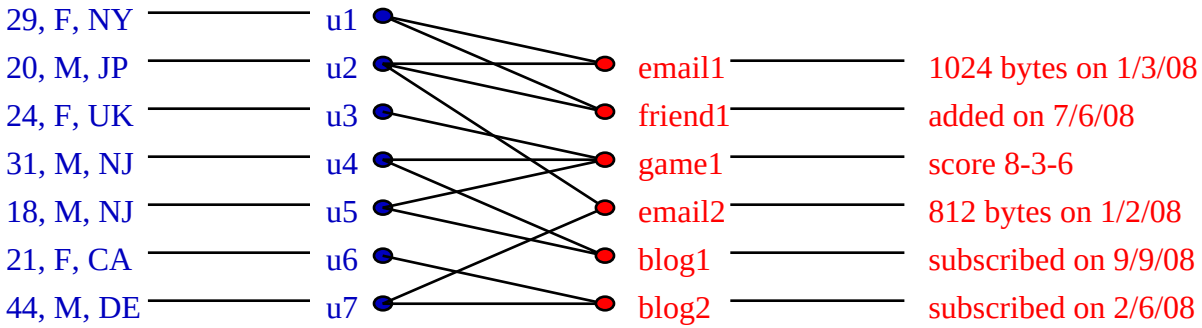
- Netflix: 100M ratings from 480K users to 18K movies
  - All direct customer information removed
  - Only subset of all data; dates modified; some ratings deleted

- AOL: 20M search queries for 650K users from 2006
  - Searches from same user linked by an arbitrary identifier
  - Many successful attacks identified individual users

Work on Network Anonymization

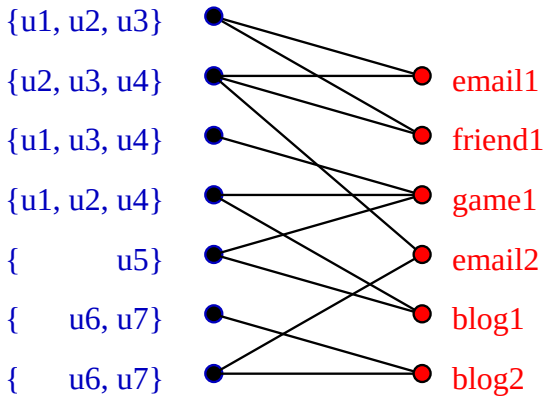
- Graph anonymization a “hot topic”
- Some negative results:
  - Powerful attacker with much knowledge can reidentify some nodes [Backstrom, Dwork, Kleinberg 07; Narayanan, Shmatikov 09]
- Modification methods: add and remove edges
  - Make neighborhoods similar [Zhou, Pei 08; Liu, Terzi 08; Zou et al 09]
- Grouping methods: mask by grouping
  - Model attacker as Machine Learning alg [Zheleva, Getoor 07]
  - Group nodes and hide mapping from nodes to entities [Hay et al 08; Cormode et al 08, 09]

Interaction Graphs



- Represent social networks with an *interaction graph*
  - Entity nodes (with demographic properties) connect to Interaction nodes (with relevant properties)
  - Can represent (directed) pairwise and group interactions
- Anonymization requirement:
  should not be able to learn of the existence of any interaction
  - Quantify how much "background knowledge" needed to break

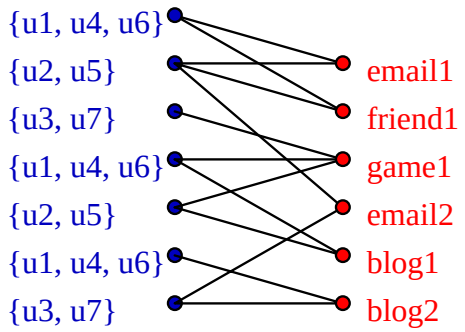
Label Lists



- Safety in numbers:
  Replace node ids with lists
- Cannot tell which is
  the true label of any node
- Aims: Preserve graph structure
  Still answer queries that touch many nodes
- Picking arbitrary lists is insufficient for security
  - As each node must appear exactly once, can eliminate options
  - In example, u1, u2, u3, u4 must be first four nodes
  - Reveals identity of some other nodes
  - Shows u6 and u7 share blog2 interaction

Uniform Lists

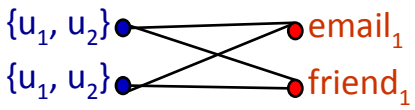
- Need more structure to the lists
  - Enforce symmetry so such deductions are not possible
- Divide nodes into classes of size *m*
- *Uniform lists* create symmetric lists in each class
  - *Full pattern*: all lists are $\{u_1, u_2 \ldots u_m\}$
  - Other symmetric patterns are possible
- Assign lists so each node's list includes true label



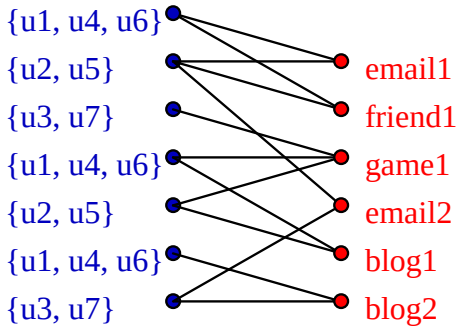
at&t

Security of published data

- Uniform lists still vulnerable if there are many interactions between nodes in same class

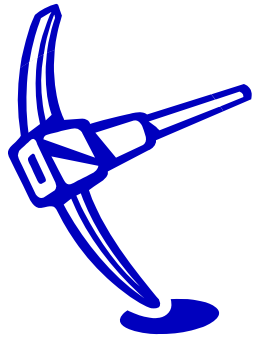


- Define a class safety condition
  - Each node v participates in interactions with at most one node in any other class
  - Keeps the inter-class interactions sparse



- Gives a provable guarantee of security
  - For classes of size ≥ m, for every way in which a node v is in an interaction, there are m-1 consistent ways where it is not
  - Based on considering structure of possible label assignments
  - Conclude attacker's belief in any possibility is at most 1/m

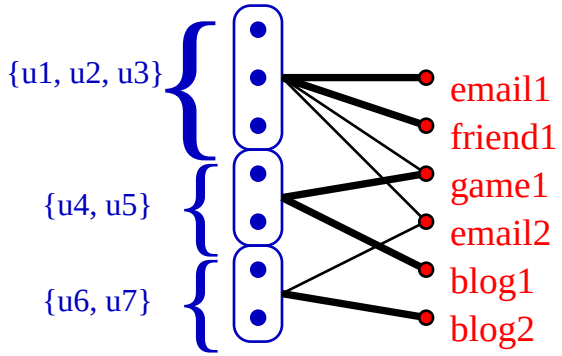
Background Knowledge Attacks



- What if an attacker knows some interactions?
  - E.g. knows Graham sent email to Tami
- May be possible to identify certain nodes with those for which some partial knowledge exists
- Can prove that other nodes are not badly affected
  - Knowledge partitions some classes into pieces of size m-1 and 1
  - Safety condition still holds on the new classes
  - So previous guarantee holds with m-1
  - For r pieces of knowledge, inductively give guarantee with ≥ m-r
- For attacker with much information this may not be enough

Partitioning Approach

- Guarantees of the label list approach may not be enough
  e.g. lots of node degree information
- Can increase the amount of security, at cost of utility
- Partition approach:
  - Divide nodes into partitions of size ≥ m
  - Only reveal the number of edges from each partition to each interaction

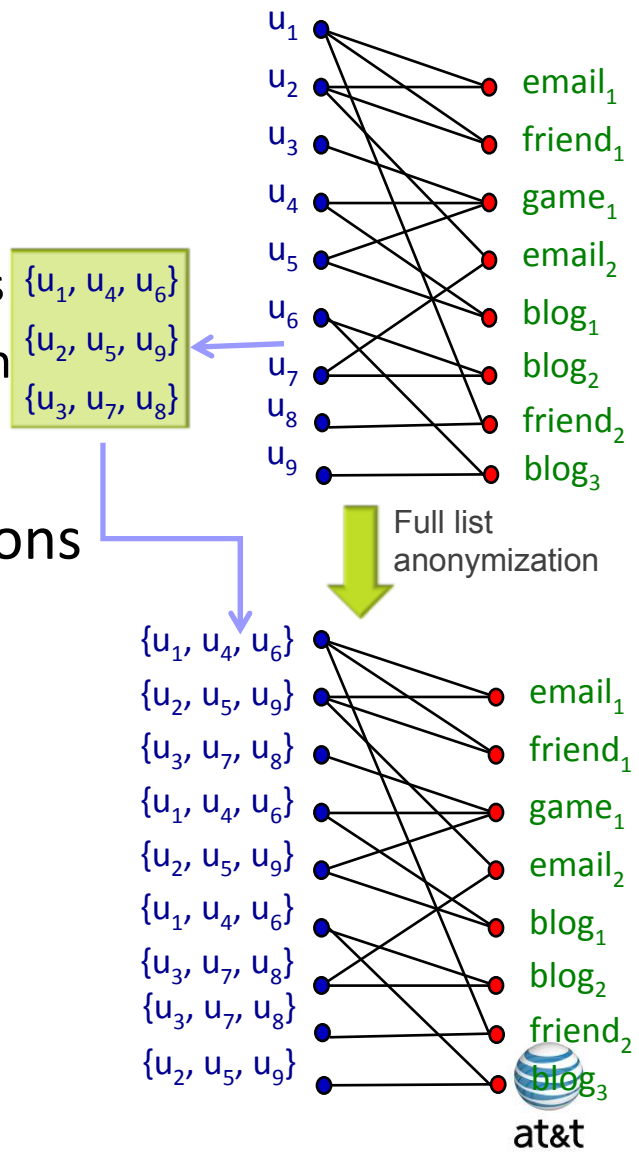


- Still need some conditions
  - Require same "safety condition" to ensure that attacker cannot use edge density to infer participation in interactions
  - Now can prove attacker with knowledge about < m entities cannot make any further inferences

Algorithm Overview

Multi-step process for label list anonymization:

1. Divide nodes into classes, respecting safety
   - Optimize: try to group similar nodes in same class
   - Sort or cluster, and try to add to each class in turn
2. Create and assign lists to nodes
3. Publish the graph using either lists or partitions
   - List: *k* labels at each node, preserving all edges
   - Partition: k labels at each node, aggregate edges



Using Anonymized Data

- A generic problem in data anonymization: how to answer queries on anonymized data?
- Output data does not have same format as input!
  - Have introduced uncertainties, replaced values with sets
- Ad hoc solutions for fixed queries are unsatisfying, unscalable
- Uncertain Databases proposed for a variety of applications:
  - Sensor readings, data integration, output of mining algorithms
- Anonymized data is an immediate application for UDBMSs
  - May require new primitives and working models

Possible Worlds Interpretation

- Uncertain anonymized data represents multiple possible worlds
  - Each possible world corresponds to a database (or graph, or…)
  - The original input data is known to be one of these worlds
  - Best approach to query answering: range over all possible worlds
- Possibilistic interpretations: is some fact possible/certain?
- Probabilistic interpretations given distribution over worlds
  - What is the probability of a fact being true?
  - What is distribution (mean, var) of answers to an aggregate query?
- Generic Monte Carlo approach:
  - uniformly sample possible worlds and evaluate query on each
  - Take the mean, max, min of multiple samples

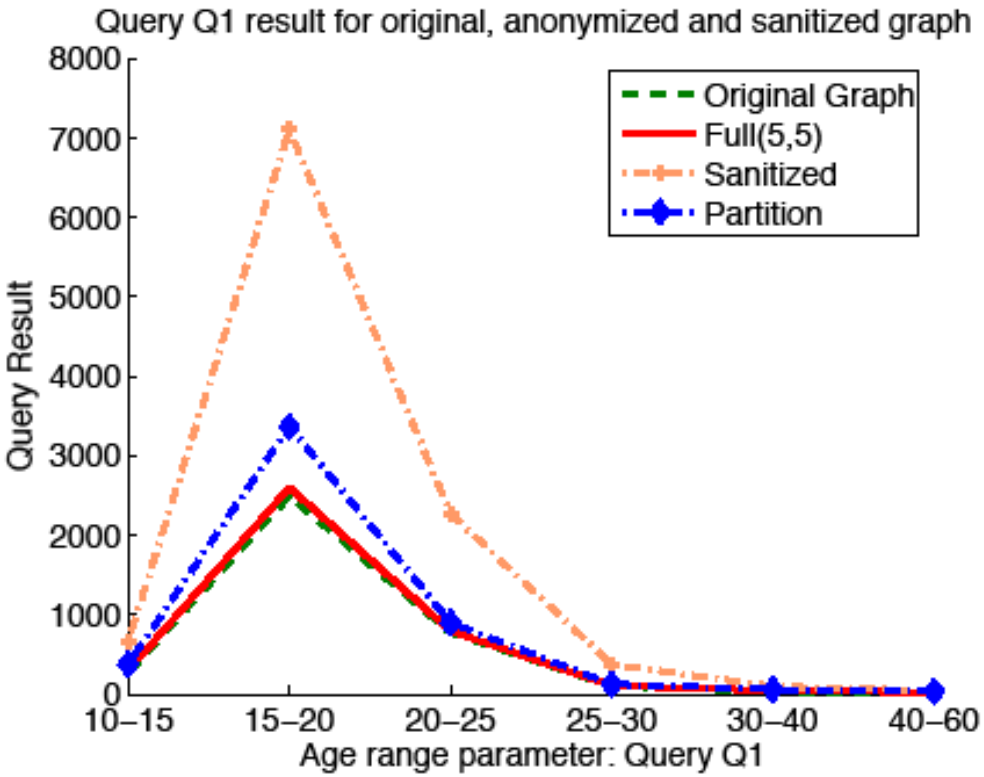
Experimental Analysis

- Data: evaluated our approaches over two social net datasets
  - Blog (780K nodes, 3M edges) drawn from Xanga network
  - Speed dating (530 nodes, 4K edges) due to Columbia study

- Utility Evaluation: on varied query workload
  - Pair: Single-hop, e.g., how many Americans befriend Germans
  - Trio: Two-hop, e.g., how many Americans are friends with Germans who have French friends
  - Triangle: Clustering co-efficient

Experimental Analysis

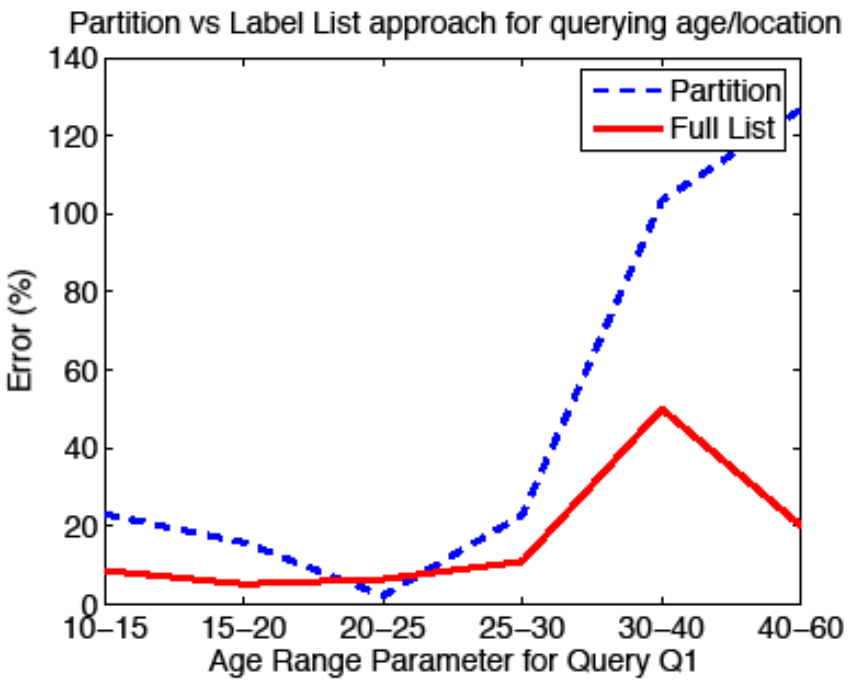
- Fixed privacy guarantee, analyzed impact on accuracy

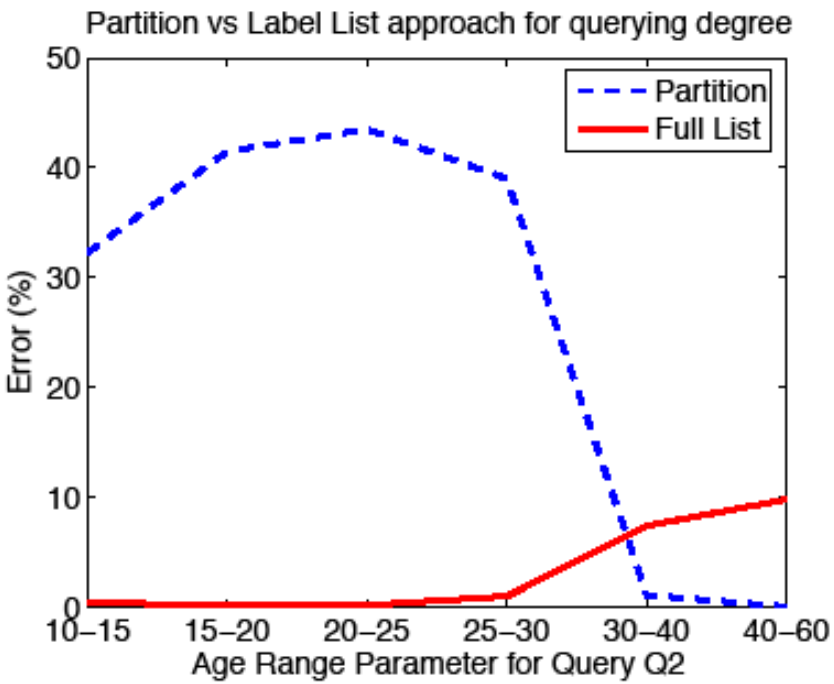


- Pair Query (Q1): How many Americans of different ages are friends with Hong Kong residents with age <20?
- Errors are typically higher for partition approach

at&t

Experimental Analysis



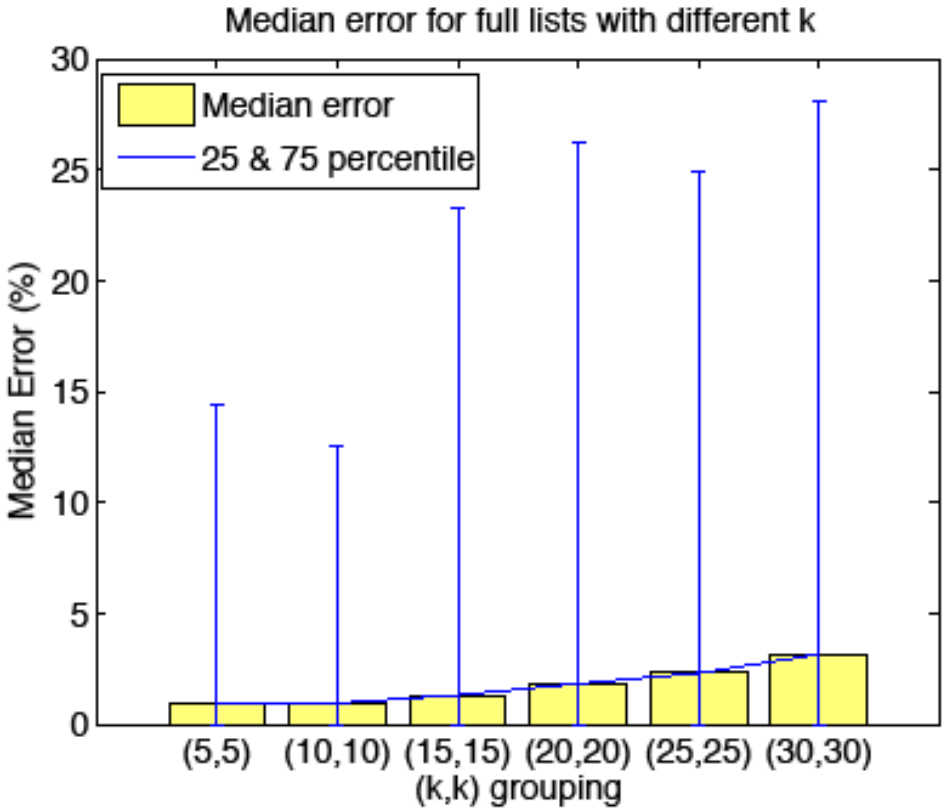


- Analysis on neighborhood size: higher accuracy on younger ages (more prevalent in data) than older
- Different group sizes vary privacy/accuracy tradeoff

at&t

Experimental Analysis

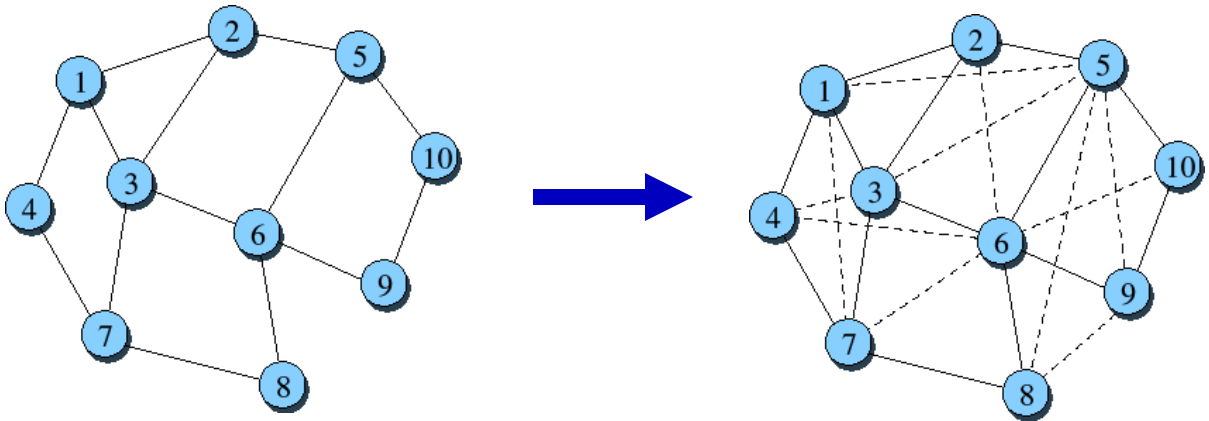
- Varied group sizes, analyzed impact on accuracy over a 100-query workload with all 3 types of queries



at&t

Dynamic Data Anonymization

- Social Network data changes over time
  - Want to allow multiple releases of same data set as it evolves
- Privacy challenge: prevent inference of associations
- Approach: use of link prediction to guess future structure
  - Combine predicted structure with observed structure
  - Find groupings that are likely to be resilient to future links
  - Pick carefully to avoid being overwhelmed with many links
- Empirical: big privacy metric improvement over naïve methods



at&t

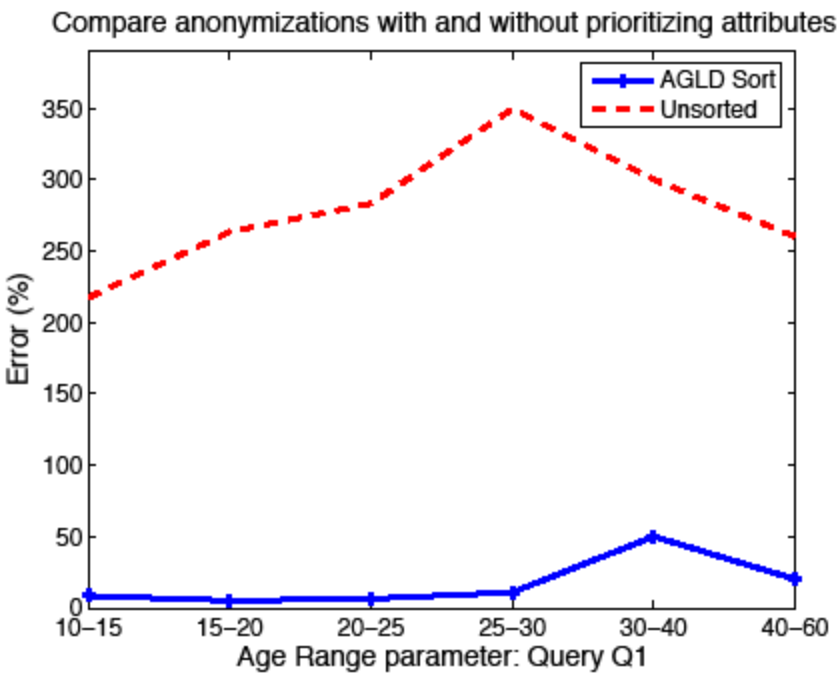
Conclusions

- Anonymization remains a challenging problem
  - Need to carefully study, what is threat model?
- Have discussed approaches for social network data
  - Offer different tradeoffs between utility and privacy
- Need to make use of the uncertain data from anonymization
  - Seems to have received only limited attention thus far
- Exact (aggregate) querying possible, but often expensive
  - Approximation needed to avoid exponential blow-ups

References: "Class-based graph anonymization for social network data" VLDB 09, "Anonymized data: Generation, models, usage" Tutorial in ICDE 10

Experimental Analysis

- Compare sort orders on Degree, Age, Gender, Location



- Using a sort order clearly improves accuracy
- A sort order that matches workload helps more

at&t