Chapter 3: Anonymisation of data

Lecture PETs4DS: Privacy Enhancing Technologies for Data Science

Parts of this slide set (slides 6 – 33) are based on slides from
Vitaly Shmatikov, Cornell University.

Parts of this slide set (slides 36 - 74) are based on slides from Johannes Gehrke, Cornell University, and Ashwin Machanavajjhala, Yahoo! Research.

Dr. Benjamin Heitmann and Prof. Dr. Stefan Decker
Informatik 5
Lehrstuhl Prof. Decker

PETs4DS

RWTH AACHEN UNIVERSITY

Overview of chapter

- **Anonymisation of tabular data**
  - Release of data is non-interactive / off-line
  - k-anonymity
  - l-diversity
  - t-closeness
- **Anonymisation of graphs**
  - Relevant e.g. for social networking data
  - k-degree anonymity
  - k-neighborhood anonymity
  - k-sized grouping
- **Anonymisation of statistical databases**
  - Relevant e.g. for mobile phone usage logs
  - Release of data is interactive
  - epsilon-differential privacy

Motivation

Data sold by Web Of Trust (WOT) Plugin

- WOT Plugin collects browsing history
- Assigns userID to each user or installation
- WOT sells data with URLs grouped by userID

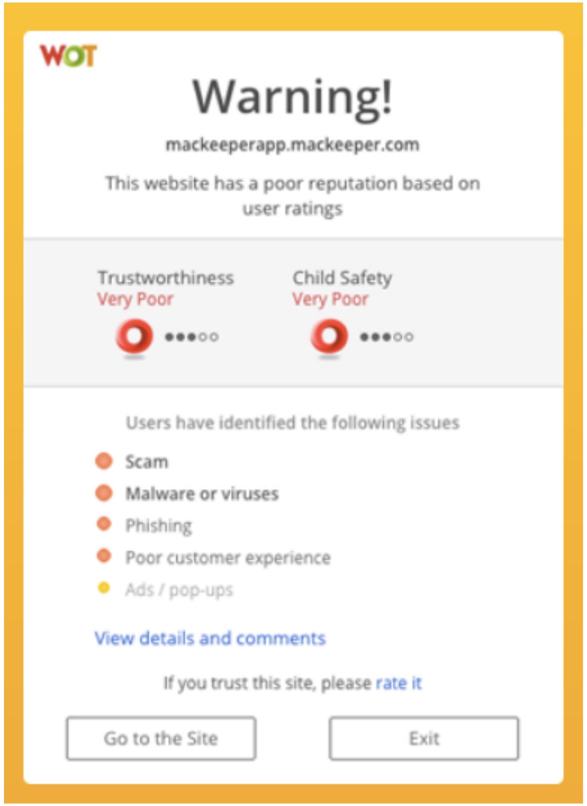
Our privacy analysis has shown:

- This data is **not** sufficiently anonymised

The **remaining privacy threats** in relation to the stored data are:

- Linkability
- Identifiability
- Non-repudiation
- Detectability
- Disclosure of information

- **How to anonymise such data correctly?**



Anonymisation of tabular data

Naïve approach to anonymisation

- **Lets just remove all "personally identifiable information" (PII)**
  - Name
  - Phone number
  - Social security number
  - Email address
  - Postal address
- **Is that enough?**
  - The WOT Plugin shows that this is NOT enough
  - They did exactly that
  - But: no PIIs or sensitive data in URLs was considered

Related literature:
Li, Li, Venkatasubramanian. "t-Closeness: Privacy Beyond k-Anonymity and l-Diversity" (ICDE 2007).

Re-identification by Linking

Microdata

ID	QID			SA
Name	Zipcode	Age	Sex	Disease
Alice	47677	29	F	Ovarian Cancer
Betty	47602	22	F	Ovarian Cancer
Charles	47678	27	M	Prostate Cancer
David	47905	43	M	Flu
Emily	47909	52	F	Heart Disease
Fred	47906	47	M	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

De-anonymisation of Hospital Data

Massachusetts hospital discharge dataset

Medical Data Released as Anonymous

SSN	Name	Ethnicity	Date Of Birth	Sex	ZIP	Marital Status	Problem
			09/27/64	female	02139	divorced	hypertension
			09/30/64	female	02139	divorced	obesity
		asian	04/18/64	male	02139	married	chest pain
		asian	04/15/64	male	02139	married	obesity
		black	03/13/63	male	02138	married	hypertension
		black	03/18/63	male	02138	married	shortness of breath
		black	09/13/64	female	02141	married	shortness of breath
		black	09/07/64	female	02141	married	obesity
		white	05/14/61	male	02138	single	chest pain
		white	05/08/61	male	02138	single	obesity
		white	09/15/61	female	02142	widow	shortness of breath

Voter List

Name	Address	City	ZIP	DOB	Sex	Party	....
....	....	....	....	....	....	....	
....	....	....	....	....	....	....	
Sue J. Carlson	1459 Main St.	Cambridge	*02142*	*9/15/61*	*female*	democrat	....
....	....	....	....	....	....	....	

Figure 1: Re-identifying anonymous data by linking to external data

Public voter dataset

From Latanya Sweeney's original k-anonymity paper (1997)

Quasi-identifiers

- Key attributes, also called personally identifiable information (PII)
  - Name, address, phone number
  - Always removed before release
- Quasi-identifiers
  - (5-digit ZIP code, birth date, gender) uniquely identify 87% of the population in the U.S.
  - Can be used for linking anonymized dataset with other datasets

Classification of Attributes

- Sensitive attributes
  - Medical records, salaries, etc.
  - These attributes is what the researchers need, so they are always released directly

Key Attribute	Quasi-identifier			Sensitive attribute
Name	DOB	Gender	Zipcode	Disease
Andre	1/21/76	Male	53715	Heart Disease
Beth	4/13/86	Female	53715	Hepatitis
Carol	2/28/76	Male	53703	Brochitis
Dan	1/21/76	Male	53703	Broken Arm
Ellen	4/13/86	Female	53706	Flu
Eric	2/28/76	Female	53706	Hang Nail

K-Anonymity: Intuition

- The information for each person contained in the released table cannot be distinguished from at least k-1 individuals whose information also appears in the release
  - Example: you try to identify a man in the released table, but the only information you have is his birth date and gender. There are k men in the table with the same birth date and gender.

- Any **quasi-identifier present in the released table must appear in at least k records**

K-Anonymity Protection Model

- Private table
- Released table: RT
- Attributes: $A_1, A_2, \ldots, A_n$
- Quasi-identifier subset: $A_i, \ldots, A_j$

Let RT($A_1,\ldots,A_n$) be a table, $QI_{RT} = (A_i, \ldots, A_j)$ be the quasi-identifier associated with RT, $A_i,\ldots,A_j \subseteq A_1,\ldots,A_n$, and RT satisfy k-anonymity. Then, each sequence of values in RT[A_x] appears with at least k occurrences in RT[QI_{RT}] for $x=i,\ldots,j$.

Lecture PETs4DS – Privacy Enhancing Technologies for Data Science
Dr. Benjamin Heitmann and Prof. Dr. Stefan Decker
Informatik 5, Lehrstuhl Prof. Decker

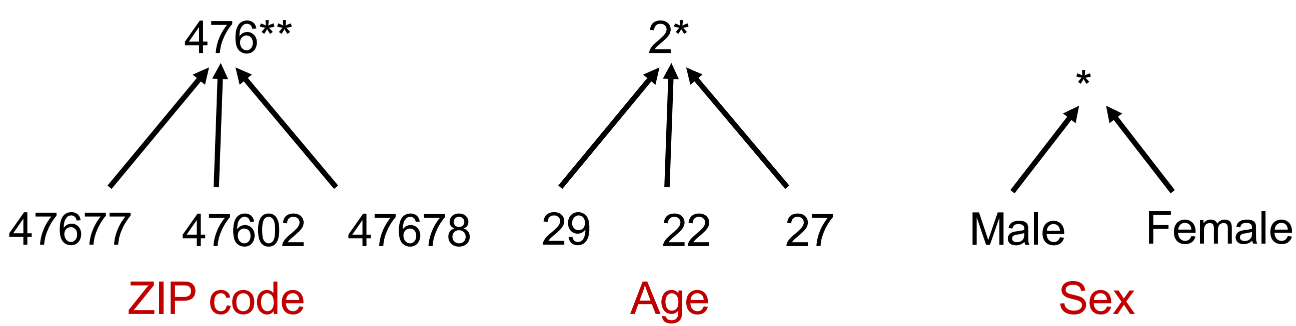
Achieving k-Anonymity

- Generalization
  - Replace specific quasi-identifiers with less specific values until get k identical values
  - Partition ordered-value domains into intervals

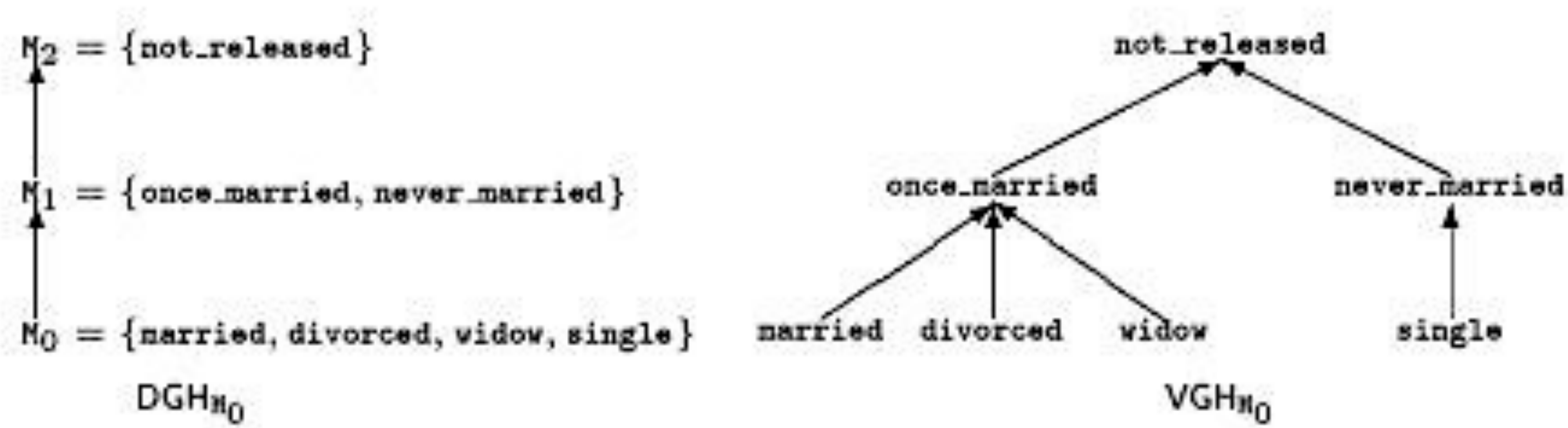
- Suppression
  - When generalization causes too much information loss
    - This is common with “outliers”

Generalisation

- Goal of k-Anonymity
  - Each record is indistinguishable from at least k-1 other records
  - These k records form an equivalence class
- **Generalization:** replace quasi-identifiers with less specific, but semantically consistent values
  - Example: instead of an age, use a range: 20 <= age <= 30
- **Suppression:** leave out or hide parts of quasi-identifiers



Example for Generalisation



Example of a k-Anonymous Table

	Race	Birth	Gender	ZIP	Problem
t1	Black	1965	m	0214*	short breath
t2	Black	1965	m	0214*	chest pain
t3	Black	1965	f	0213*	hypertension
t4	Black	1965	f	0213*	hypertension
t5	Black	1964	f	0213*	obesity
t6	Black	1964	f	0213*	chest pain
t7	White	1964	m	0213*	chest pain
t8	White	1964	m	0213*	obesity
t9	White	1964	m	0213*	short breath
t10	White	1967	m	0213*	chest pain
t11	White	1967	m	0213*	chest pain

Figure 2 Example of *k*-anonymity, where *k*=2 and QI={*Race*, *Birth*, *Gender*, *ZIP*}

Formal Definitions: Quasi-Identifier and k-anonymity

Definition: Quasi-identifier (QID)
A QID is a set of **non-sensitive attributes** of a table, if these attributes can be **linked** with external data, in order to **uniquely identify** at least one individual in the general population.

Definition: k-Anonymity
A released version of a dataset is k-anonymous, if every released combination of values for each quasi-identifier, can be indistinguishable matched to at least k records in the dataset.

Definition: Equivalence Class
The records which have the tuple of values for a QID form an equivalence class.

More intuitive description of k-anonymity:
Every tuple of values for a quasi-identifier occurs in at least k records of a dataset.

Example of a k-Anonymous Table

	Race	Birth	Gender	ZIP	Problem
t1	Black	1965	m	0214*	short breath
t2	Black	1965	m	0214*	chest pain
t3	Black	1965	f	0213*	hypertension
t4	Black	1965	f	0213*	hypertension
t5	Black	1964	f	0213*	obesity
t6	Black	1964	f	0213*	chest pain
t7	White	1964	m	0213*	chest pain
t8	White	1964	m	0213*	obesity
t9	White	1964	m	0213*	short breath
t10	White	1967	m	0213*	chest pain
t11	White	1967	m	0213*	chest pain

Figure 2 Example of *k*-anonymity, where *k*=2 and QI={*Race*, *Birth*, *Gender*, *ZIP*}

Another example

Released table

	Race	Birth	Gender	ZIP	Problem
t1	Black	1965	m	0214*	short breath
t2	Black	1965	m	0214*	chest pain
t3	Black	1965	f	0213*	hypertension
t4	Black	1965	f	0213*	hypertension
t5	Black	1964	f	0213*	obesity
t6	Black	1964	f	0213*	chest pain
t7	White	1964	m	0213*	chest pain
t8	White	1964	m	0213*	obesity
t9	White	1964	m	0213*	short breath
t10	White	1967	m	0213*	chest pain
t11	White	1967	m	0213*	chest pain

External data

Name	Birth	Gender	ZIP	Race
Andre	1964	m	02135	White
Beth	1964	f	55410	Black
Carol	1964	f	90210	White
Dan	1967	m	02174	White
Ellen	1968	f	02237	White

By linking these 2 tables, you still don't learn Andre's problem

Background knowledge attack on k-anonymous data

Microdata

QID			SA
Zipcode	Age	Sex	Disease
47677	29	F	Ovarian Cancer
47602	22	F	Ovarian Cancer
47678	27	M	Prostate Cancer
47905	43	M	Flu
47909	52	F	Heart Disease
47906	47	M	Heart Disease

Generalized table

QID			SA
Zipcode	Age	Sex	Disease
476**	2*	*	Ovarian Cancer
476**	2*	*	Ovarian Cancer
476**	2*	*	Prostate Cancer
4790*	[43,52]	*	Flu
4790*	[43,52]	*	Heart Disease
4790*	[43,52]	*	Heart Disease

!!

- Released table is 3-anonymous
- If the adversary knows Alice’s quasi-identifier (47677, 29, F), he still does not know which of the first 3 records corresponds to Alice’s record
- However, background knowledge about human anatomy allows de-anonymisation

Curse of dimensionality

- Generalization fundamentally relies on spatial locality
  - Each record must have k close neighbors
- Real-world datasets are very sparse
  - Many attributes (dimensions)
    - Netflix Prize dataset: 17,000 dimensions
    - Amazon customer records: several million dimensions
  - “Nearest neighbor” is very far
- Projection to low dimensions loses all info $\Rightarrow$ k-anonymized datasets are useless

[Aggarwal VLDB ‘05]

Attacks on k-Anonymity

- k-Anonymity does not provide privacy if
  - Sensitive values in an equivalence class lack diversity
  - The attacker has background knowledge

Homogeneity attack

Bob	
Zipcode	***Age***
47678	27

Background knowledge attack

Carl	
Zipcode	***Age***
47673	36

A 3-anonymous patient table

Zipcode	Age	Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
4790*	≥40	Flu
4790*	≥40	Heart Disease
4790*	≥40	Cancer
476**	3*	Heart Disease
476**	3*	Cancer
476**	3*	Cancer

l-Diversity builds on k-Anonymity

Caucas	787XX	Flu
Caucas	787XX	Shingles
Caucas	787XX	Acne
Caucas	787XX	Flu
Caucas	787XX	Acne
Caucas	787XX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Shingles
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Flu

[Machanavajjhala et al. ICDE '06]

Sensitive attributes must be "diverse" within each quasi-identifier equivalence class

Distinct l-Diversity

- Each equivalence class has at least l well-represented sensitive values
- Doesn't prevent probabilistic inference attacks
- Most basic definition of l-Diversity

...	**Disease**
	...
	HIV
	HIV
	...
	HIV
	pneumonia
	bronchitis
	...

10 records

8 records have HIV

2 records have other values

Other Versions of l-Diversity

- Probabilistic l-diversity
  - The frequency of the most frequent value in an equivalence class is bounded by 1/l
- Entropy l-diversity
  - The entropy of the distribution of sensitive values in each equivalence class is at least log(l)
- Recursive (c,l)-diversity
  - $r_1 < c(r_l + r_{l+1} + \ldots + r_m)$ where r_i is the frequency of the i^{th} most frequent value
  - Intuition: the most frequent value does not appear too frequently

Formal definition in Li, Li, Venkatasubramanian. "t-Closeness: Privacy Beyond k-Anonymity and l-Diversity" (ICDE 2007).

l-Diversity is Neither Necessary, Nor Sufficient to Prevent Privacy Leaks

99% have cancer

Original dataset

…	Cancer
…	Cancer
…	Cancer
…	Flu
…	Cancer
…	Cancer
…	Cancer
…	Cancer
…	Cancer
…	Cancer
…	Flu
…	Flu

Anonymization A

Q1	Flu
Q1	Flu
Q1	Cancer
Q1	Flu
Q1	Cancer
Q1	Cancer
Q2	Cancer
Q	

Anonymization B

Q1	Flu
Q1	Cancer
Q1	Cancer
Q1	Cancer
Q1	Cancer
Q1	Cancer
Q2	cer
Q2	Flu

99% cancer ⇒ quasi-identifier group is <u>not</u> “diverse”
…yet anonymized database does not leak anything

50% cancer ⇒ quasi-identifier group is “diverse”
This leaks a ton of information

l-Diversity: Skewness Attack

- Example: sensitive attribute is HIV+ (1%) or HIV- (99%)
- Consider an equivalence class that contains an equal number of HIV+ and HIV- records
  - Diverse, but potentially violates privacy!
- l-diversity does not differentiate:
  - Equivalence class 1: 49 HIV+ and 1 HIV-
  - Equivalence class 2: 1 HIV+ and 49 HIV-

l-diversity does not consider
overall distribution of sensitive values!

l-diversity: Similarity Attack

Similarity attack

Bob	
Zip	***Age***
47678	27

A 3-diverse patient table

Zipcode	Age	Salary	Disease
476**	2*	20K	Gastric Ulcer
476**	2*	30K	Gastritis
476**	2*	40K	Stomach Cancer
4790*	≥40	50K	Gastritis
4790*	≥40	100K	Flu
4790*	≥40	70K	Bronchitis
476**	3*	60K	Bronchitis
476**	3*	80K	Pneumonia
476**	3*	90K	Stomach Cancer

Conclusion

1. Bob's salary is in [20k,40k], which is relatively low
2. Bob has some stomach-related disease

l-diversity does not consider semantics of sensitive values!

t-Closeness

Caucas	787XX	Flu
Caucas	787XX	Shingles
Caucas	787XX	Acne
Caucas	787XX	Flu
Caucas	787XX	Acne
Caucas	787XX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Shingles
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Flu

[Li et al. ICDE '07]

Distribution of sensitive attributes within each quasi-identifier group should be "close" to their distribution in the entire original database

Trick question: Why publish quasi-identifiers at all??

Anonymous, “t-Close” Dataset

Caucas	787XX	HIV+	Flu
Asian/AfrAm	787XX	HIV-	Flu
Asian/AfrAm	787XX	HIV+	Shingles
Caucas	787XX	HIV-	Acne
Caucas	787XX	HIV-	Shingles
Caucas	787XX	HIV-	Acne

This is k-anonymous,
l-diverse and t-close…

…so secure, right?

Attacker might have more background knowledge

Bob is Caucasian and I heard he was admitted to hospital with flu…

Caucas	787XX	HIV+	Flu
Asian/AfrAm	787XX	HIV-	Flu
		HIV+	Shingles
Caucas	787XX	HIV-	Acne
		HIV-	Shingles
Caucas	787XX	HIV-	Acne

This is against the rules!
"flu" is not a quasi-identifier

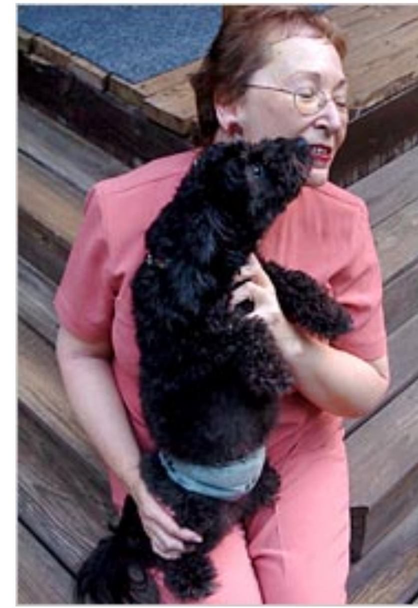
Yes… and this is yet another problem with k-anonymity

AOL Privacy Debacle

- In August 2006, AOL released anonymized search query logs
  - 657K users, 20M queries over 3 months (March-May)
- Opposing goals
  - Analyze data for research purposes, provide better services for users and advertisers
  - Protect privacy of AOL users
    - Government laws and regulations
    - Search queries may reveal income, evaluations, intentions to acquire goods and services, etc.

AOL User 4417749

- AOL query logs have the form
  <AnonID, Query, QueryTime, ItemRank, ClickURL>
  - ClickURL is the truncated URL
- NY Times re-identified AnonID 4417749
  - Sample queries: “numb fingers”, “60 single men”, “dog that urinates on everything”, “landscapers in Lilburn, GA”, several people with the last name Arnold
    - Lilburn area has only 14 citizens with the last name Arnold
  - NYT contacts the 14 citizens, finds out AOL User 4417749 is 62-year-old Thelma Arnold



Problems with anonymity measures

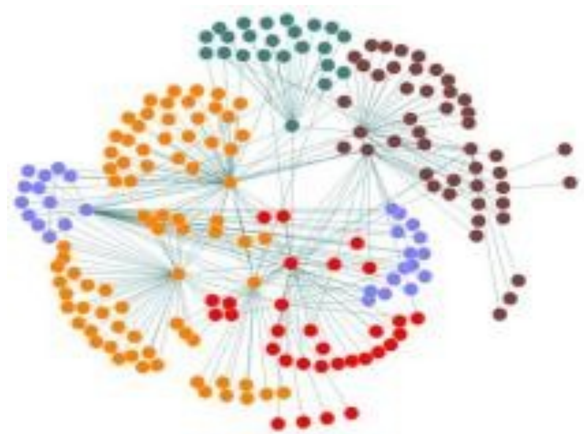
- Anonymised data sets can still enable attacker with background knowledge to re-identify individuals
- Quasi-identifiers
  - If attribute can be used as quasi-identifier depends on external background data sources
  - And on domain knowledge of the attacker
- Consider “curse of anonymity”
- Consider data minimisation
  - The less data gets published the less important background knowledge becomes.

Anonymisation of Graph Data

Overview: Anonymisation of Graph Data

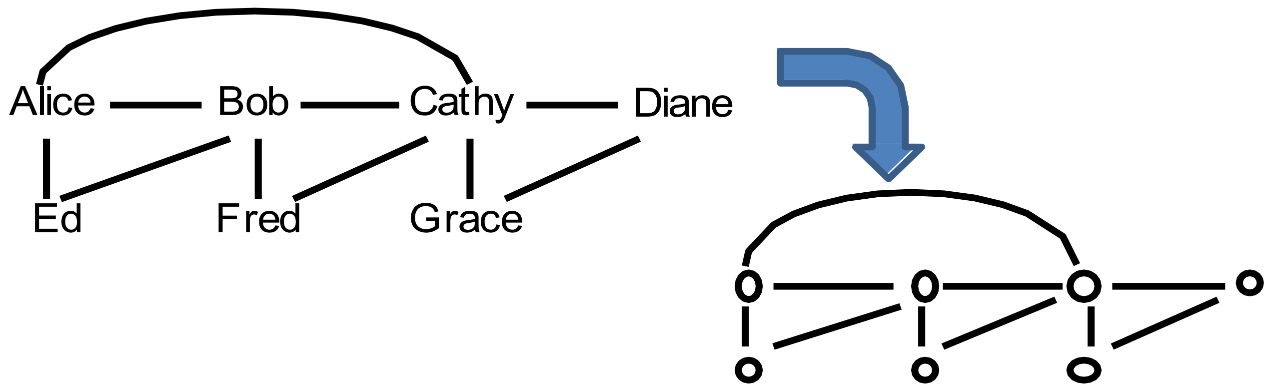
- Graph modification to preserve privacy
  - k-degree anonymity
  - k-neighbourhood anonymity
- Graph clustering to preservice privacy
  - k-sized grouping
- Example of an active attack on a social network
  - "Active" meaning that the attacker inserts fake accounts into a live service

Social Network Data



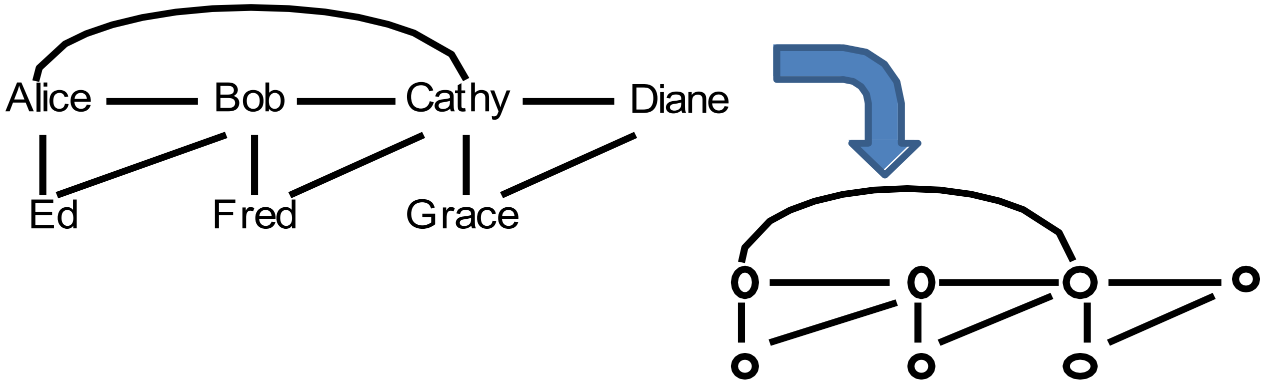
- Social networks: graphs where each node represents a social entity, and each edge represents certain relationship between two entities
- Example: email communication graphs, social interactions like in Facebook, Yahoo! Messenger, etc

Privacy in Social Networks



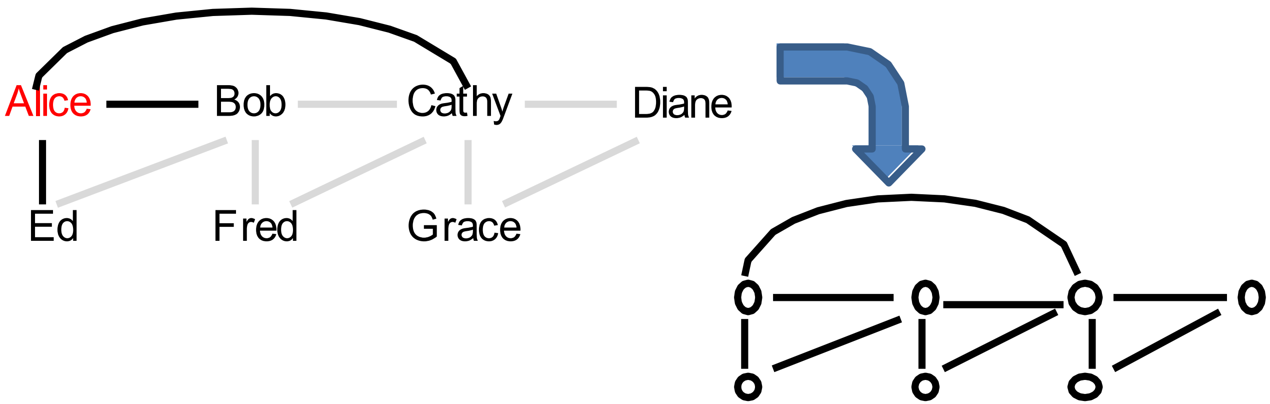
- Naïve anonymization
  - –removes the label of each node and publish only the structure of the network
- Information Leaks
  - –Nodes may still be re-identified based on network structure

Attacking an Anonymized Social Network



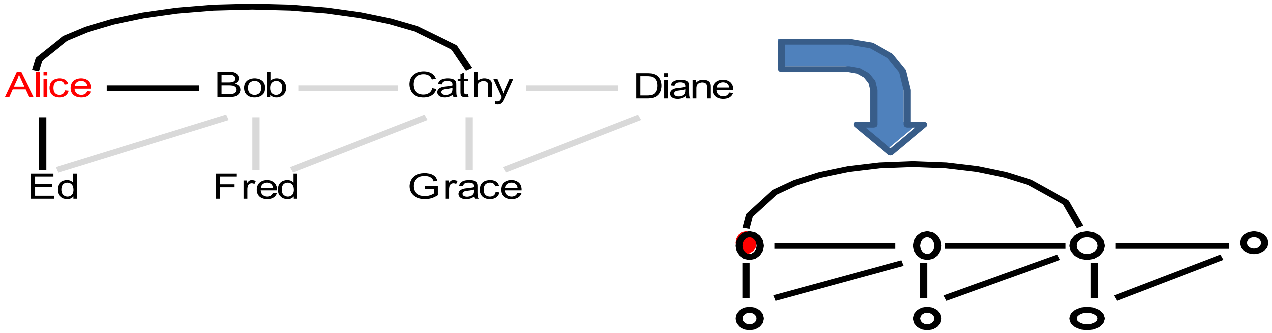
- Consider the above email communication graph
  - Each node represents an individual
  - Each edge between two individuals indicates that they have exchanged emails

Attacking an Anonymized Social Network



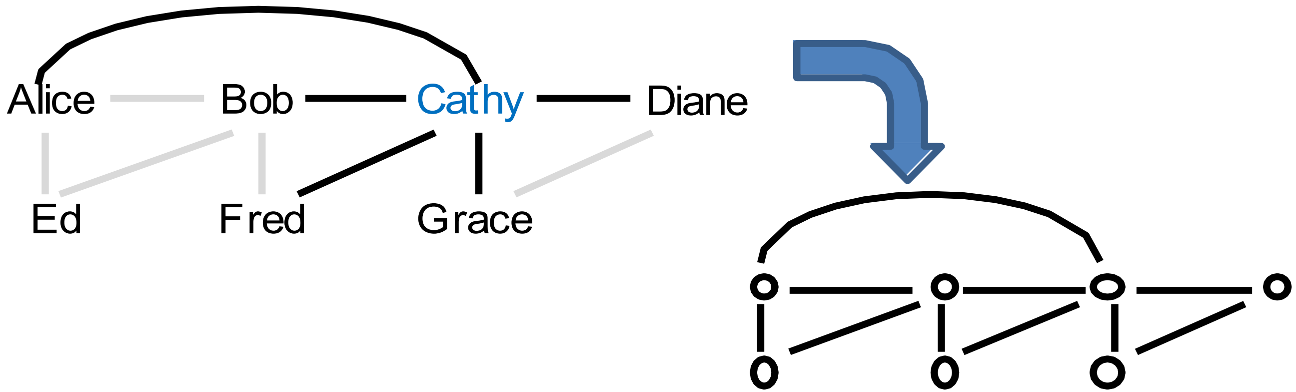
- Alice has sent emails to three individuals only

Attacking an Anonymized Social Network



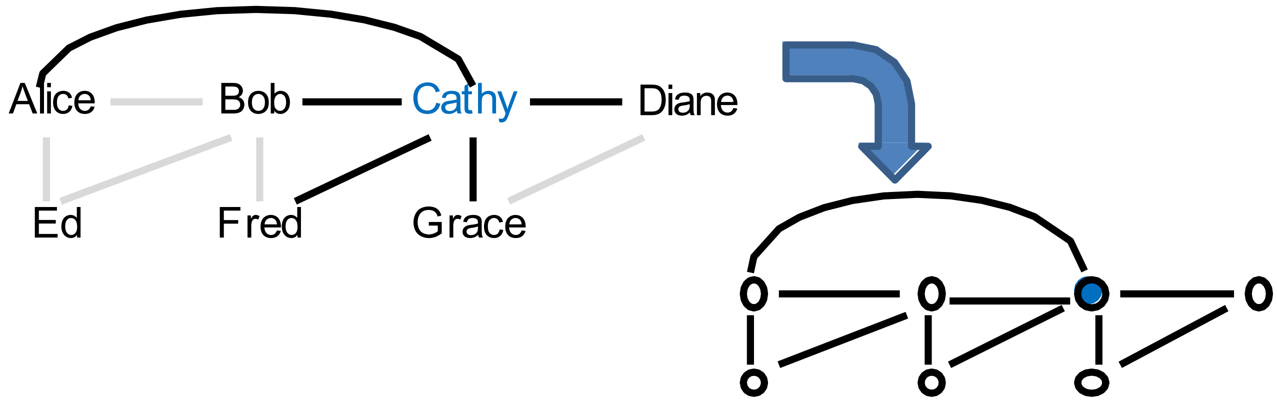
- Alice has sent emails to three individuals only
- Only one node in the anonymized network has a degree three
- Hence, Alice can re-identify herself

Attacking an Anonymized Social Network

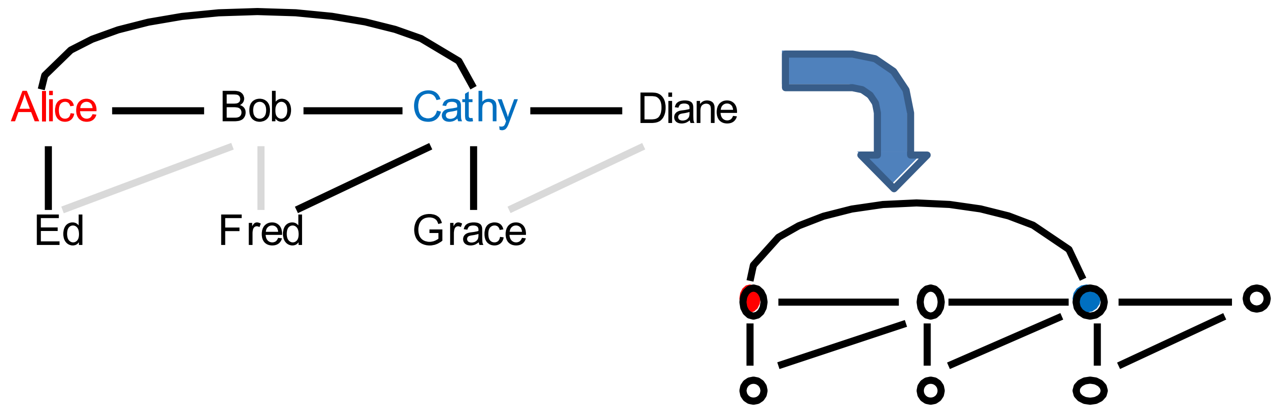


- Cathy has sent emails to five individuals

Attacking an Anonymized Social Network

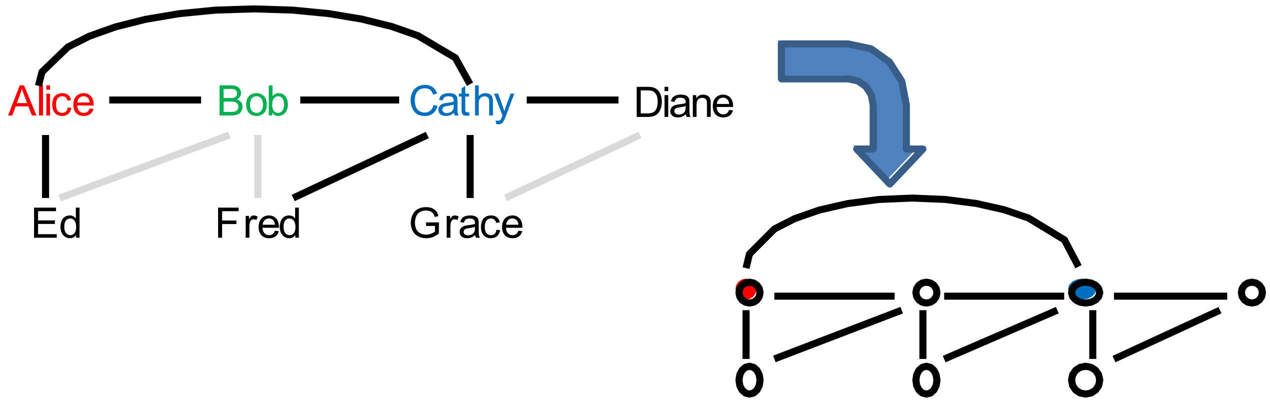


- Cathy has sent emails to five individuals
- Only one node has a degree five
- Hence, Cathy can re-identify herself



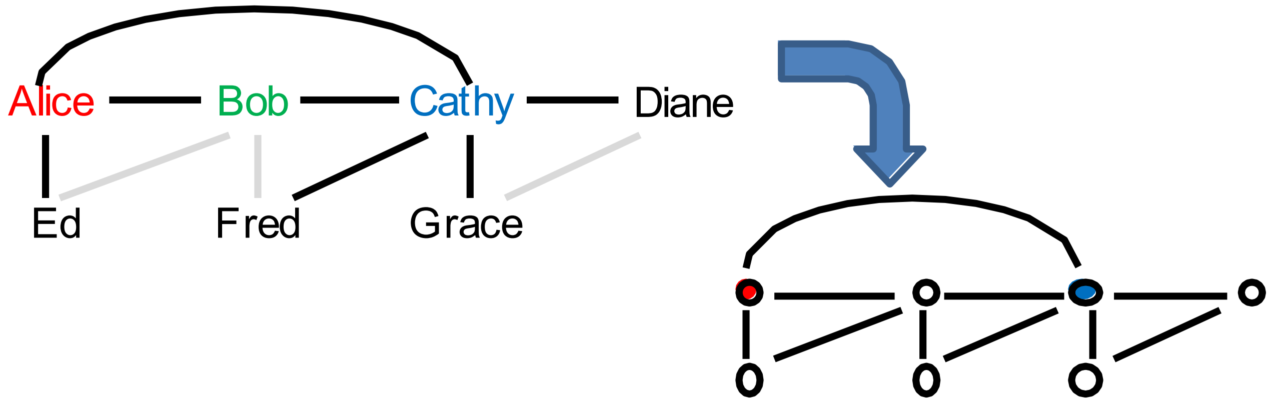
- Now consider that Alice and Cathy share their knowledge about the anonymized network
- What can they learn about the other individuals?

Attacking an Anonymized Social Network



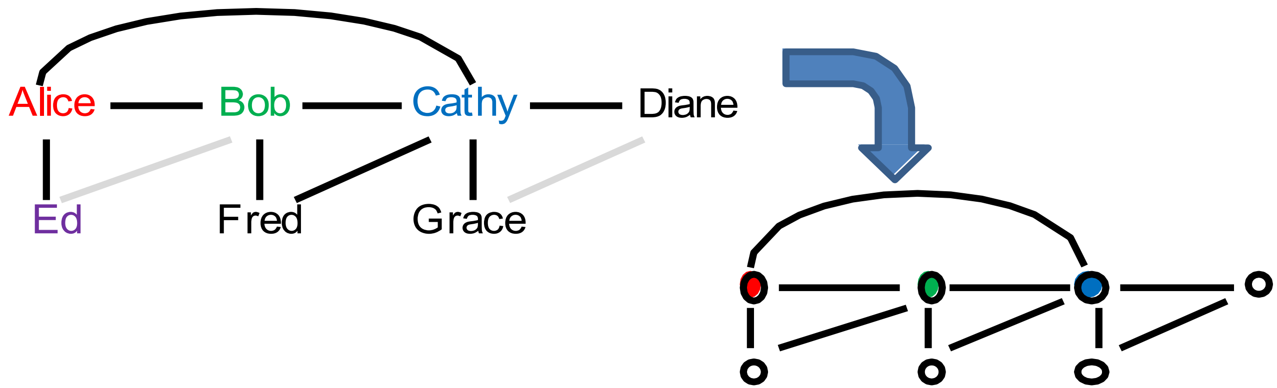
- First, Alice and Cathy know that only Bob have sent emails to both of them

Attacking an Anonymized Social Network



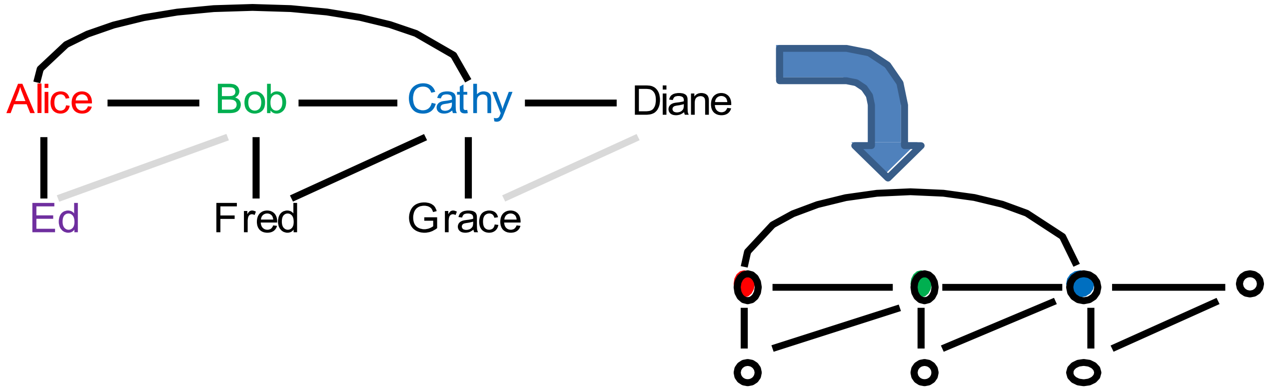
- First, Alice and Cathy know that only Bob have sent emails to both of them
- Bob can be identified

Attacking an Anonymized Social Network



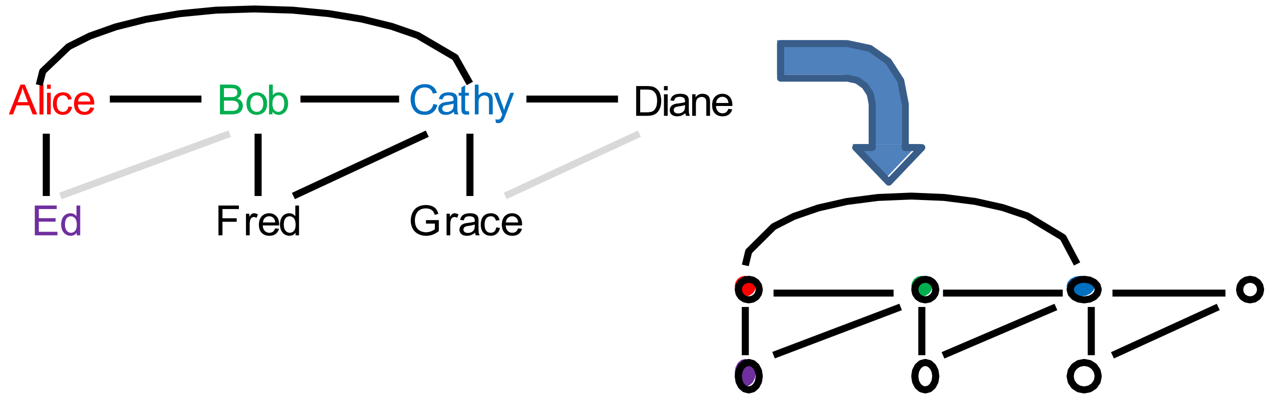
- Alice has sent emails to Bob, Cathy, and Ed only

Attacking an Anonymized Social Network



- Alice has sent emails to Bob, Cathy, and Ed only

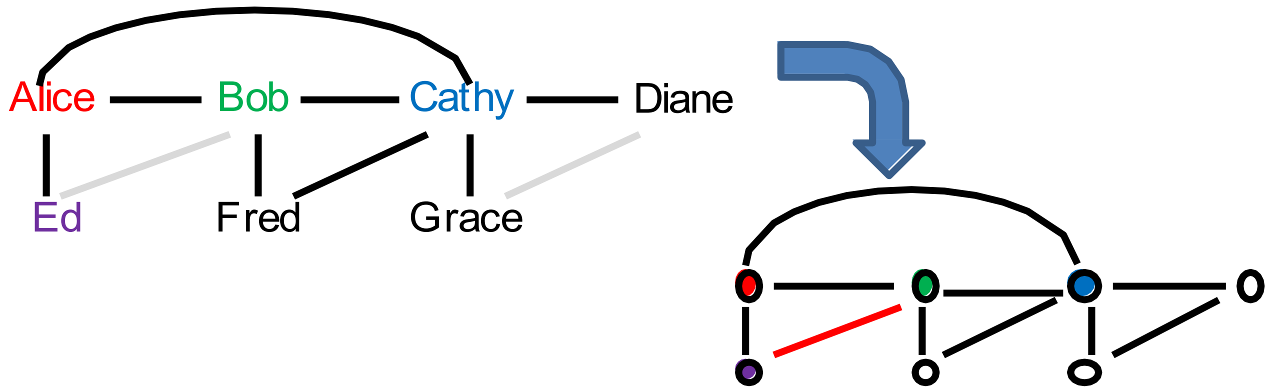
Attacking an Anonymized Social Network



- Alice has sent emails to Bob, Cathy, and Ed only
- Ed can be identified

RWTH AACHEN UNIVERSITY

Attacking an Anonymized Social Network



- Alice and Cathy can learn that Bob and Ed are connected

Attacking an Anonymized Social Network

- The above attack is based on knowledge about the degrees of the nodes

- More sophisticated attacks can be launched given additional knowledge about the network structure, e.g., a subgraph of the network.

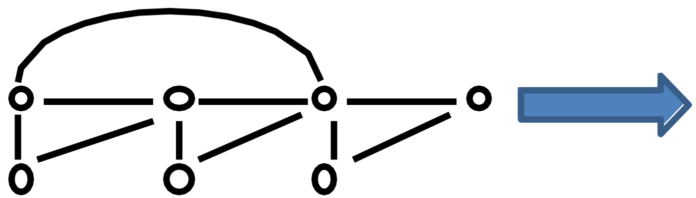
- Protecting privacy becomes even more challenging when the nodes in the anonymized network are labeled

K-degree Anonymity

[Liu and Terzi, SIGMOD 2008]

- Objective: prevent re-identification based on node degrees
- Solution: add edges into the graph, such that each node has the same degree as at least k-1 other nodes

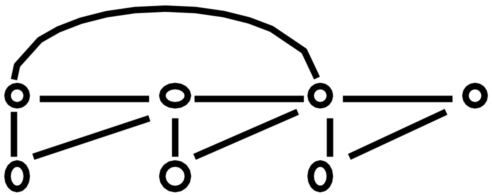
K-degree Anonymity Algorithm



[5, 4, 3, 2, 2, 2, 2]

- Given a graph, calculate the degree of each node, and stores the degrees in a vector

K-degree Anonymity Algorithm

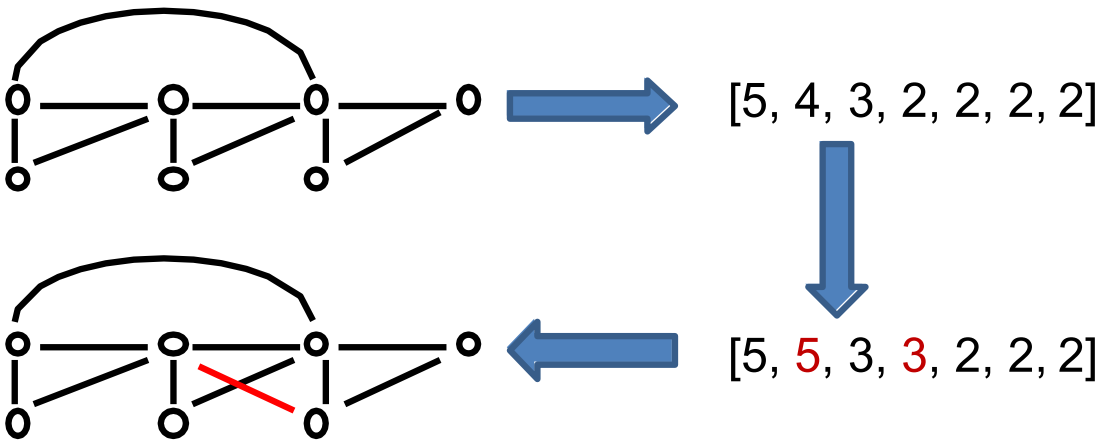


[5, 4, 3, 2, 2, 2, 2]

[5, 5, 3, 3, 2, 2, 2]

- Modify the degree vector, such that each degree appears at least k times

K-degree Anonymity Algorithm



- Add edges into the graph, such that the degrees of the nodes conform to the modified degree vector

K-degree Anonymity Algorithm

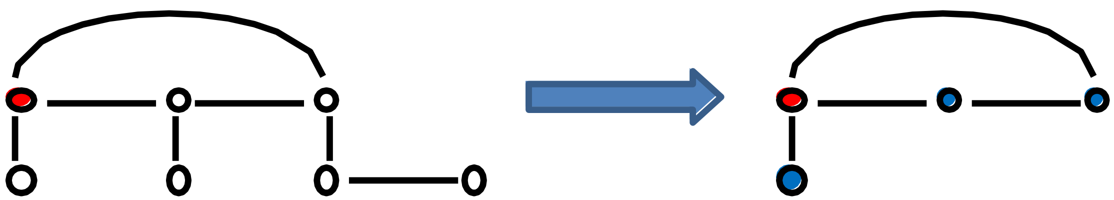
- How do we modify the degree vector?
  - A dynamic programming algorithm can be used to minimize the L1 distance between the original and modified vectors

- How do we modify the graph according to the degree vector?
  - Greedily add edges into the graph to make the node degrees closer to the given vector

K-neighborhood Anonymity

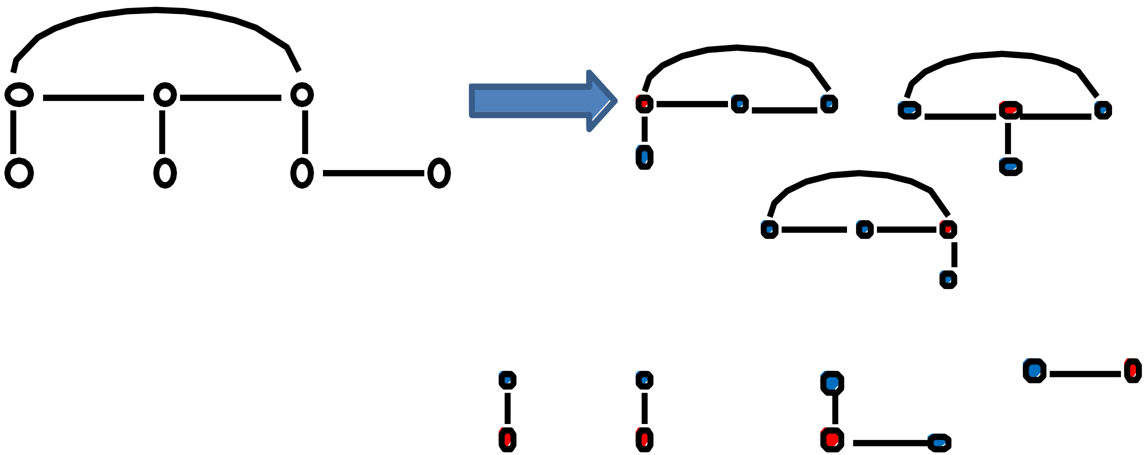
[Zhou and Pei, ICDE 2008]

- Neighborhood: sub-graph induced by one-hop neighbors



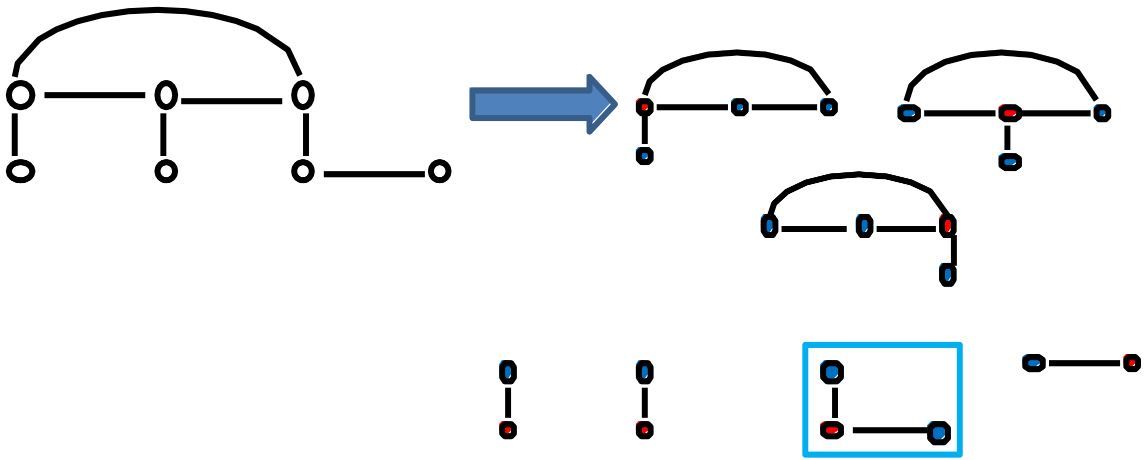
- Objective: prevent re-identification based on neighborhood structure
- Solution: add edges into the graph, such that each node has the same *neighborhood* as at least k-1 other nodes

K-neighborhood Anonymity Algorithm



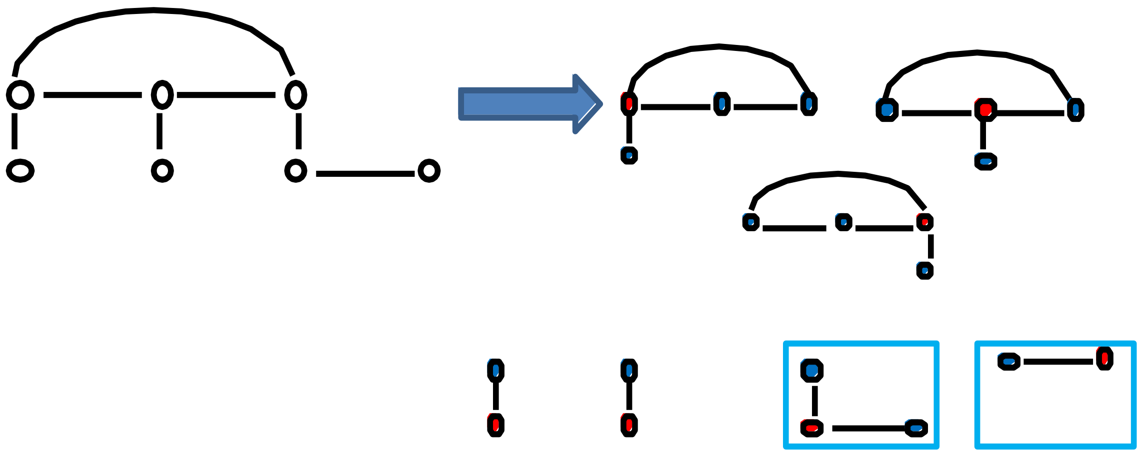
- Compute the neighborhood of each node

K-neighborhood Anonymity



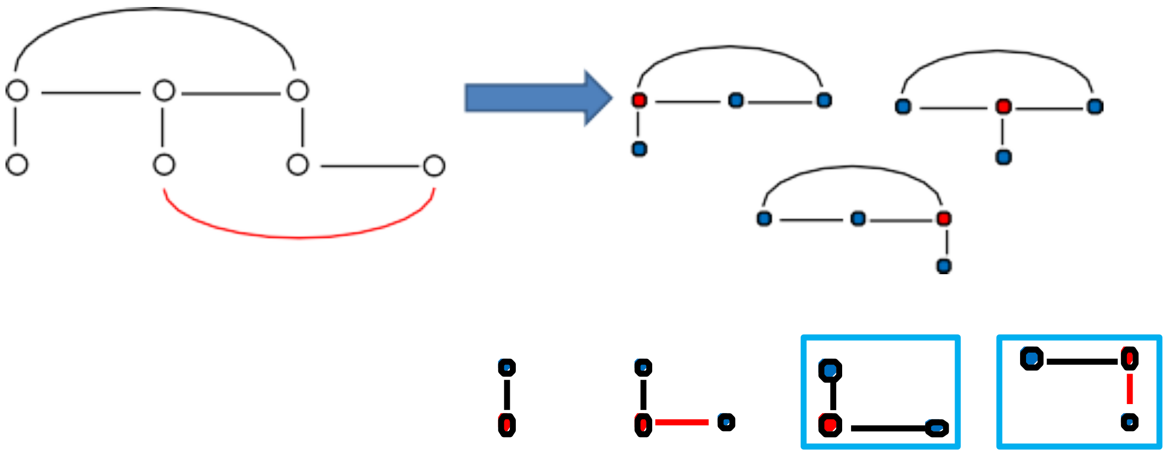
- While there is a node N whose neighborhood is not k-anonymous
  - Find a node N’ whose neighborhood is similar to that of N
  - Greedily add edges in the graph to make the neighborhoods of N and N’ isomorphic

K-neighborhood Anonymity



- While there is a node N whose neighborhood is not k-anonymous
  - –Find a node N’ whose neighborhood is similar to that of N
  - –Greedily add edges in the graph to make the neighborhoods of N and N’ isomorphic

K-neighborhood Anonymity

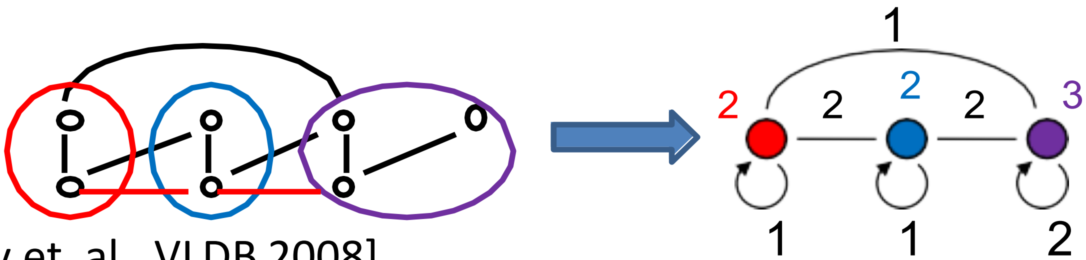


- While there is a node N whose neighborhood is not k-anonymous
  - Find a node N’ whose neighborhood is similar to that of N
  - Greedily add edges in the graph to make the neighborhoods of N and N’ isomorphic

K-neighborhood Anonymity Algorithm

- The algorithm always terminates: in the worst case it returns a complete graph
- How do we check whether two neighborhood structures are the same?
  - Graph isomorphism is NP-hard in general
  - But neighborhoods are usually small, in which case a brute- force checking is feasible
  - Some pre-processing can be done to reduce computation cost

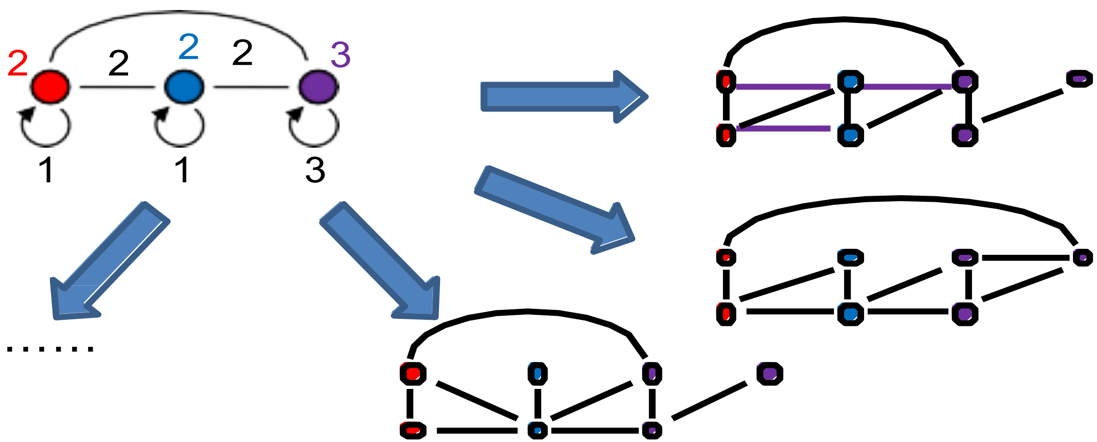
K-Sized Grouping



[Hay et al., VLDB 2008]

- Objective: prevent re-identification based on network structure
- Solution:
  - Partition the nodes into groups with sizes at least k
  - Coalesce the nodes in each group into a *super-node*
  - Each super-node has a weight that denotes its size
  - Super-nodes are connected by *super-edges* with weights

Quality of K-Sized Grouping



- A k-sized grouping represents a number of possible worlds
- The smaller number of possible worlds, the more accurate the anonymized network

Summary of Social Networking Publishing

- Structural information of a social network can be exploited to infer sensitive information
- Edge insertion and node grouping reduce the risk of re-identification
- Limitations
  - k-degree anonymity, k-neighborhood anonymity, and k-sized grouping only achieve k-anonymity

Active Attacks on Social Networks

What can go wrong if an unlabeled graph is published?
[Backstrom et al., WWW 2007]

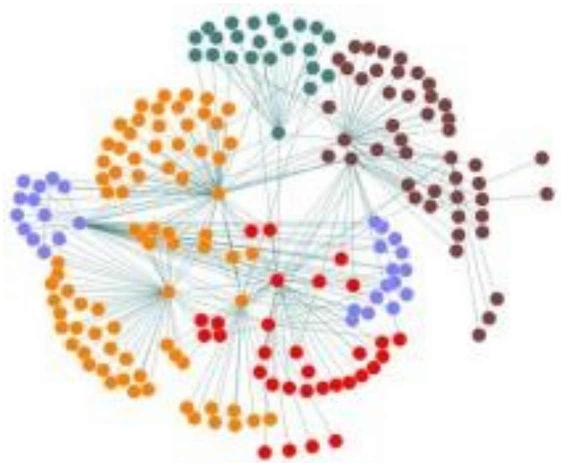
- Attacker may create a few nodes in the graph
  - Creates a few 'fake' Facebook user accounts.

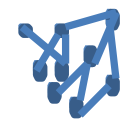
- Attacker may add edges from the new nodes.
  - Create friends using 'fake' accounts.

- Goal: Discover an edge between two legitimate users

High Level View of Attack

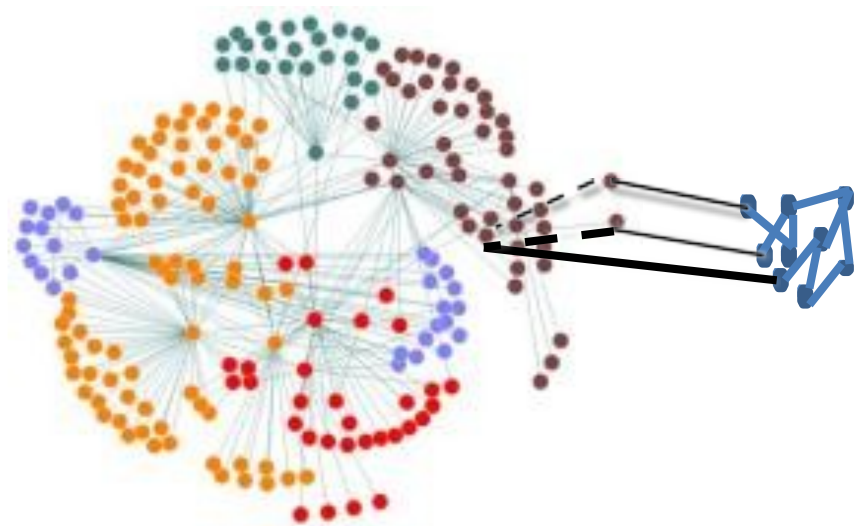
- Step 1: Create a graph structure with the 'fake' nodes such that it can be identified in the anonymous data.





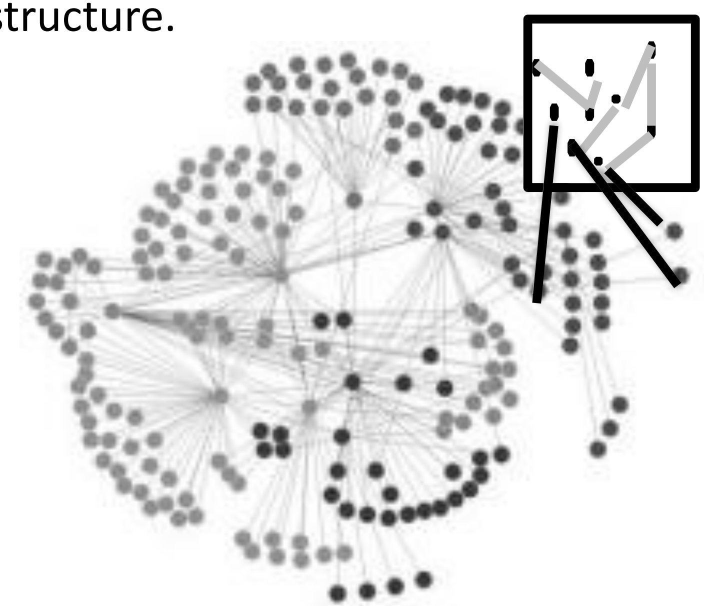
High Level View of Attack

- Step 2: Add edges from the ‘fake’ nodes to real nodes.



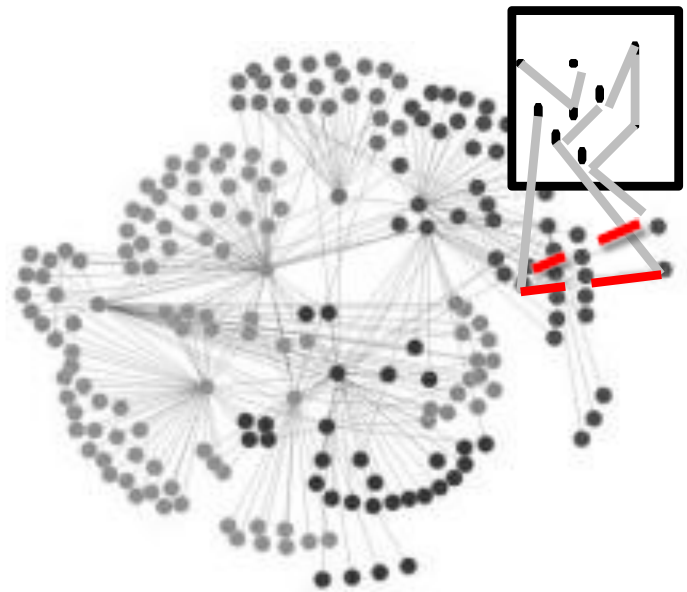
High Level View of Attack

- Step 3: From the anonymized data, identify fake graph due to its special graph structure.



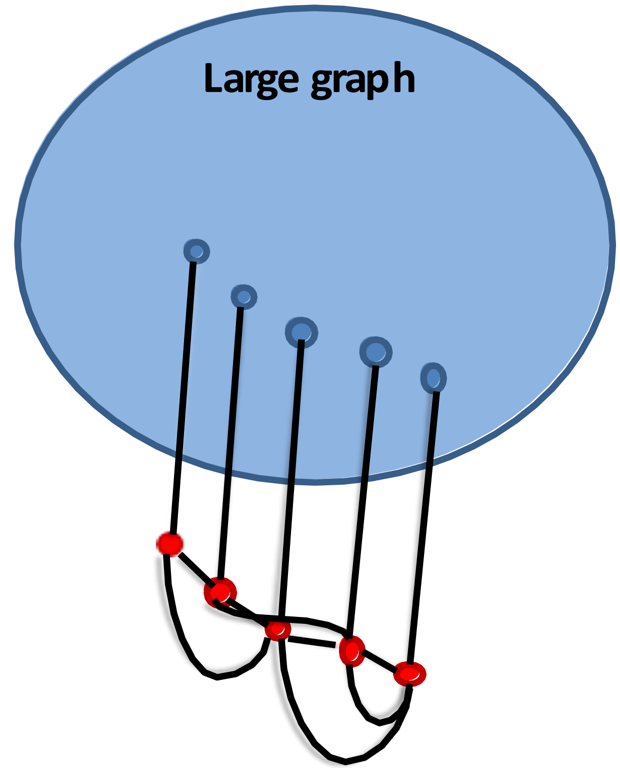
High Level View of Attack

- Step 4: Deduce edges by following links



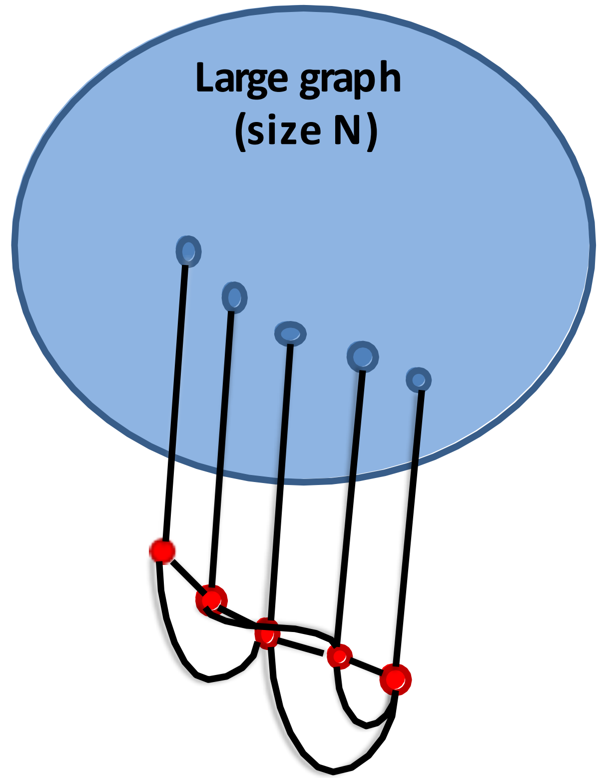
Details of Attack

- Choose k real users
  $W = \{w_1, \ldots, w_k\}$
- Create k fake users
  $X = \{x_1, \ldots, x_k\}$
- Creates edges (x_i, w_i)
- Create edges (x_i, x_{i+1})
- Create all other edges in X with probability 0.5.



Uniqueness

X is guaranteed to be unique
when k is 2+δ log N, for small δ

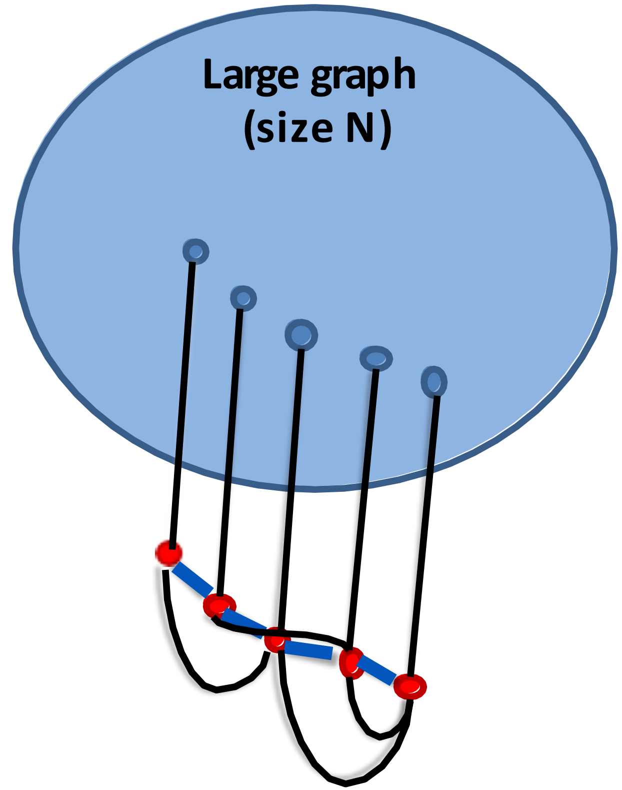


Recovery

Subgraph isomorphism is NP-hard.

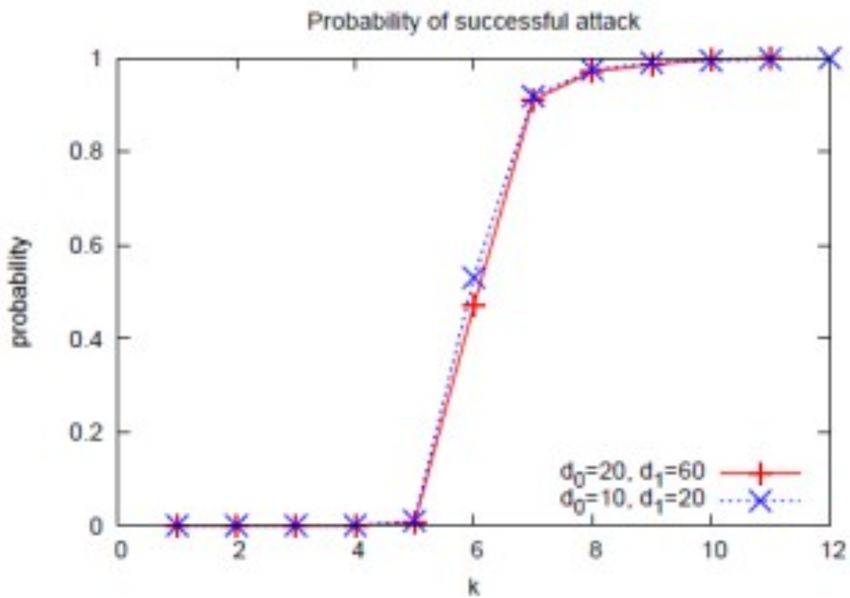
But since we have a path, with random edges, there is a simple brute force search with pruning algorithm.

Run Time: $\mathbf{O(N\ 2^{O(\log \log N)})}$



Works in Real Life!

- LiveJournal – 4.4 million nodes, 77 million edges
- Success all but guaranteed by adding 10 nodes.
- Recovery typically takes a second.



Summary of Attacks on Social Networks

- Several simple algorithms proposed for variants of k-anonymity.
- Active attacks that add nodes and edges are shown to be very successful.
  - Reminiscent of Sybil attacks.
- Guarding against active attacks is an open area of research !