- Towards identity-anonymization on graphs

  K. Liu & E. Terzi, SIGMOD 2008

- A framework for computing the privacy score of users in social networks

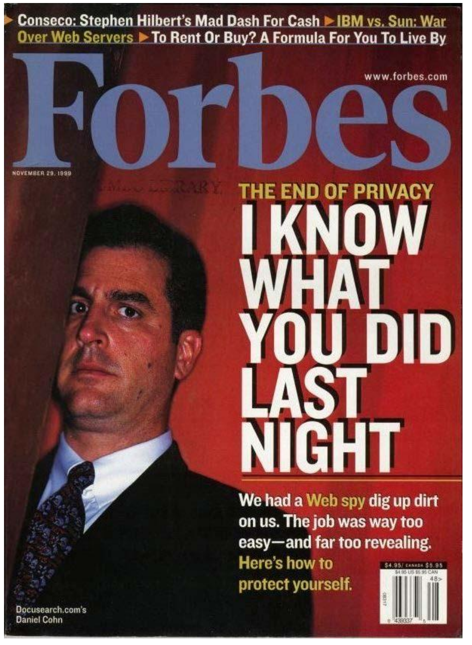
  K. Liu, E. Terzi, ICDM 2009

Growing Privacy Concerns

- Person specific information is being routinely collected.

"Detailed information on an individual's credit, health, and financial status, on characteristic purchasing patterns, and on other personal preferences is routinely recorded and analyzed by a variety of governmental and commercial organizations."

- M. J. Cronin, "e-Privacy?" Hoover Digest, 2000.



Proliferation of Graph Data

http://www.touchgraph.com/

Privacy breaches on graph data

- Identity disclosure
  - Identity of individuals associated with nodes is disclosed
- Link disclosure
  - Relationships between individuals are disclosed
- Content disclosure
  - Attribute data associated with a node is disclosed

Identity anonymization on graphs

- Question
  - How to share a network in a manner that permits useful analysis without disclosing the identity of the individuals involved?
- Observations
  - Simply removing the identifying information of the nodes before publishing the actual graph does not guarantee identity anonymization.

    L. Backstrom, C. Dwork, and J. Kleinberg, "Wherefore art thou R3579X?: Anonymized social netwoks, hidden patterns, and structural steganography," In WWW 2007.

    J. Kleinberg, "Challenges in Social Network Data: Processes, Privacy and Paradoxes, " KDD 2007 Keynote Talk.
- Can we borrow ideas from ***k***-anonymity?

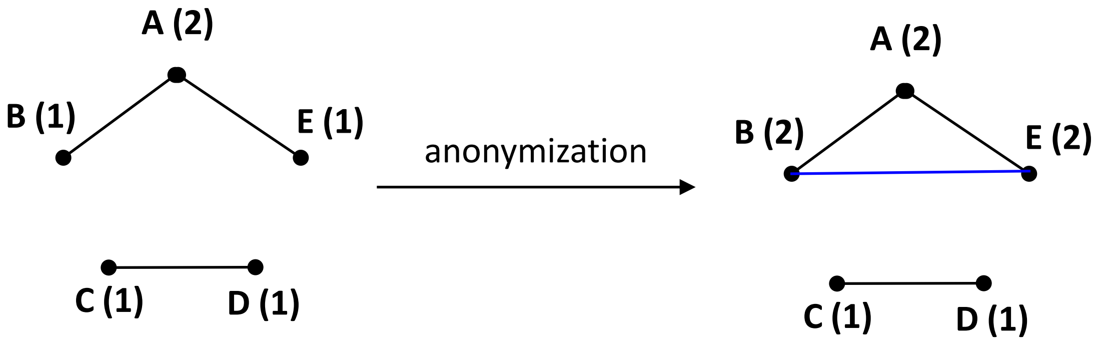
What if you want to prevent the following from happening

- Assume that adversary **A** knows that **B** has 327 connections in a social network!

- If the graph is released by removing the identity of the nodes
  - **A** can find all nodes that have degree 327
  - If there is only one node with degree 327, **A** can identify this node as being **B**.

Privacy model

[k-degree anonymity] A graph ***G(V, E)*** is ***k****-degree anonymous* if every node in ***V*** has the same degree as ***k-1*** other nodes in ***V***.



[Properties] It prevents the re-identification of individuals by adversaries with *a priori* knowledge of the degree of certain nodes.

Problem Definition

Given a graph ***G(V, E)*** and an integer ***k***, modify ***G*** via a **minimal** set of edge addition or deletion operations to construct a new graph ***G'(V', E')*** such that

1) ***G'*** is ***k***-degree anonymous;
2) ***V' = V***;
3) The ***symmetric difference*** of ***G*** and ***G'*** is as small as possible

- Symmetric difference between graphs ***G(V,E)*** and ***G'(V,E')*** :

$$\mathrm{SymDiff}(G', G) = (E' \setminus E) \cup (E \setminus E')$$

GraphAnonymization algorithm

Input: Graph ***G*** with degree sequence ***d***, integer ***k***

Output: ***k***-degree anonymous graph ***G'***

[**Degree Sequence Anonymization**]:

- Contruct an anonymized degree sequence ***d'*** from the original degree sequence ***d***

[**Graph Construction**]:

[**Construct**]: Given degree sequence ***d'***, construct a new graph $G^0(V, E^0)$ such that the degree sequence of G^0 is ***d'***

[**Transform**]: Transform $G^0(V, E^0)$ to $G'(V, E')$ so that ***SymDiff****(G',G)* is minimized.

Degree-sequence anonymization

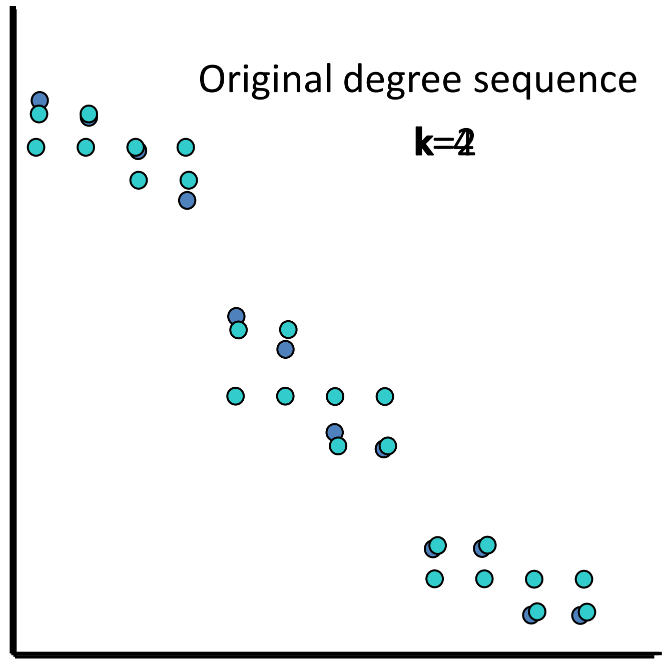
[k-anonymous sequence] A sequence of integers ***d*** is ***k****-anonymous* if every distinct element value in ***d*** appears at least ***k*** times.

[100,100, 100, 98, 98,15,15,15]

[degree-sequence anonymization] Given degree sequence ***d***, and integer ***k***, construct ***k***-anonymous sequence ***d'*** such that ***||d'-d||*** **is minimized**

Increase/decrease of degrees correspond to additions/deletions of edges

Algorithm for degree-sequence anonymization



DP for degree-sequence anonymization

- ***d*** *(1)* ≥ ***d*** *(2)* ≥… ≥ ***d*** *(i)* ≥… ≥ ***d*** *(n)* : original degree sequence.
- ***d'*** *(1)* ≥ ***d'*** *(2)* ≥…≥ ***d'*** *(i)* ≥…≥ ***d'*** *(n)* : k-anonymized degree sequence.
- ***C(i, j)*** : anonymization cost when all nodes *i, i+1, …, j* are put in the same anonymized group, i.e.,

$$C(i,j) = \sum_{\ell=i}^{j} \left(d(i) - d^{*} \right)$$

- ***DA(1, n)*** : the optimal degree-sequence anonymization cost
- Dynamic Programming with **O(n²)**

$$DA(1,i) = \min_{k \le t \le i-k} \left\{ DA(1,t) + C(t+1,i) \right\}$$

- Dynamic Programming with **O(nk)**

$$DA(1,i) = \min_{\max\{k,\, i-2k+1\} \le t \le i-k} \left\{ DA(1,t) + C(t+1,i) \right\}$$

- Dynamic Programming can be done in **O(n)** with some additional bookkeeping

GraphAnonymization algorithm

Input: Graph ***G*** with degree sequence ***d***, integer ***k***

Output: ***k***-degree anonymous graph ***G'***

[**Degree Sequence Anonymization**]:

- Contruct an anonymized degree sequence ***d'*** from the original degree sequence ***d***

[**Graph Construction**]:

[**Construct**]: Given degree sequence ***d'***, construct a new graph $G^0(V, E^0)$ such that the degree sequence of G^0 is ***d'***

[**Transform**]: Transform $G^0(V, E^0)$ to $G'(V, E')$ so that *SymDiff(G',G)* is minimized.

Are all degree sequences realizable?

- A degree sequence ***d*** is **realizable** if there exists a simple undirected graph with nodes having degree sequence ***d.***
- Not all vectors of integers are realizable degree sequences
  - d = {4,2,2,2,1} ?
- How can we decide?

Realizability of degree sequences

[**Erdös and Gallai**] A degree sequence $\boldsymbol{d}$ with $\boldsymbol{d}(1) \geq \boldsymbol{d}(2) \geq \ldots \geq \boldsymbol{d}(i) \geq \ldots \geq \boldsymbol{d}(n)$ and $\Sigma \boldsymbol{d}(i)$ even, is realizable if and only if

$$\sum_{i=1}^{l} \boldsymbol{d}(i) \leq l(l-1) + \sum_{i=l+1}^{n} \min\{l, \boldsymbol{d}(i)\}, \text{ for every } 1 \leq l \leq n-1.$$

Input: Degree sequence ***d'***
Output: Graph $G^0(V, E^0)$ with degree sequence ***d'*** or ***NO!***

→If the degree sequence ***d'*** is NOT realizable?

- Convert it into a realizable and ***k***-anonymous degree sequence

GraphAnonymization algorithm

Input: Graph ***G*** with degree sequence ***d***, integer ***k***

Output: ***k***-degree anonymous graph ***G'***

[**Degree Sequence Anonymization**]:

- Contruct an anonymized degree sequence ***d'*** from the original degree sequence ***d***

[**Graph Construction**]:

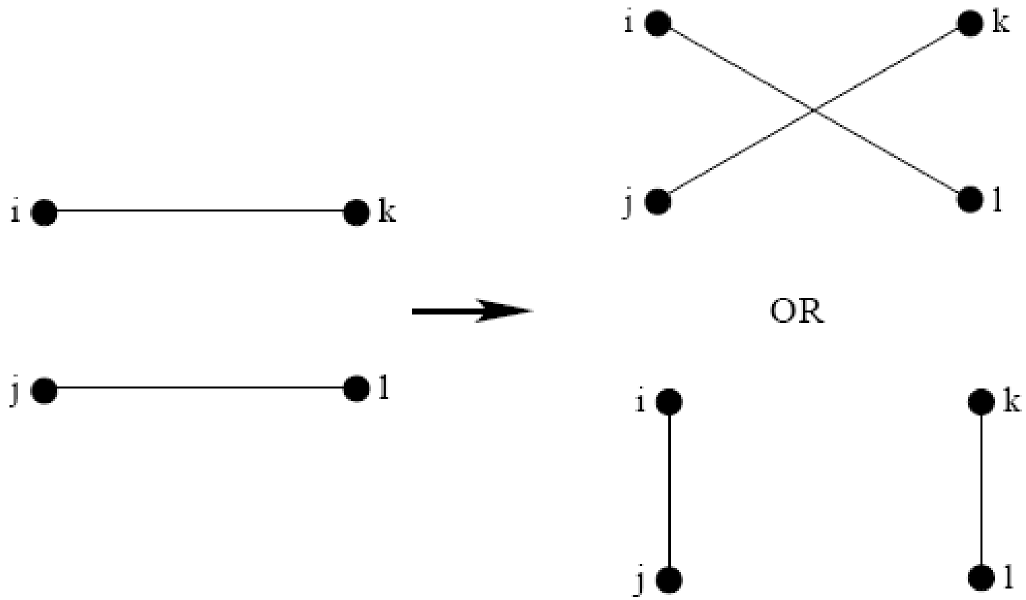
[**Construct**]: Given degree sequence ***d'***, construct a new graph $G^0(V, E^0)$ such that the degree sequence of G^0 is ***d'***

[**Transform**]: Transform $G^0(V, E^0)$ to $G'(V, E')$ so that ***SymDiff***(G',G) is minimized.

Graph-transformation algorithm

- GreedySwap transforms $G^0 = (V, E^0)$ into $G'(V, E')$ with the same degree sequence d', and min symmetric difference $SymDiff(G',G)$.
- GreedySwap is a greedy heuristic with several iterations.
- At each step, GreedySwap swaps a pair of edges to make the graph more similar to the original graph G, while leaving the nodes' degrees intact.

Valid swappable pairs of edges



A swap is ***valid*** if the resulting graph is simple

GreedySwap algorithm

Input: A pliable graph $G^0(V, E^0)$, fixed graph $G(V,E)$

Output: Graph $G'(V, E')$ with the same degree sequence as $G^0(V,E^0)$

$i=0$

Repeat

find the valid swap in G^i that most reduces its symmetric difference with G , and form graph G^{i+1}

i++

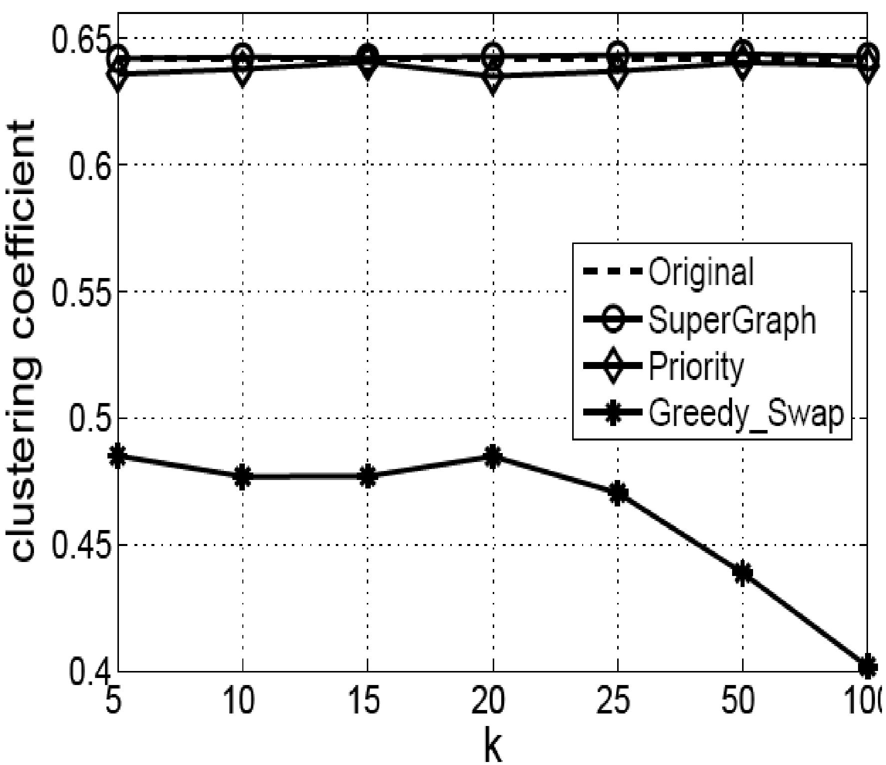
Experiments

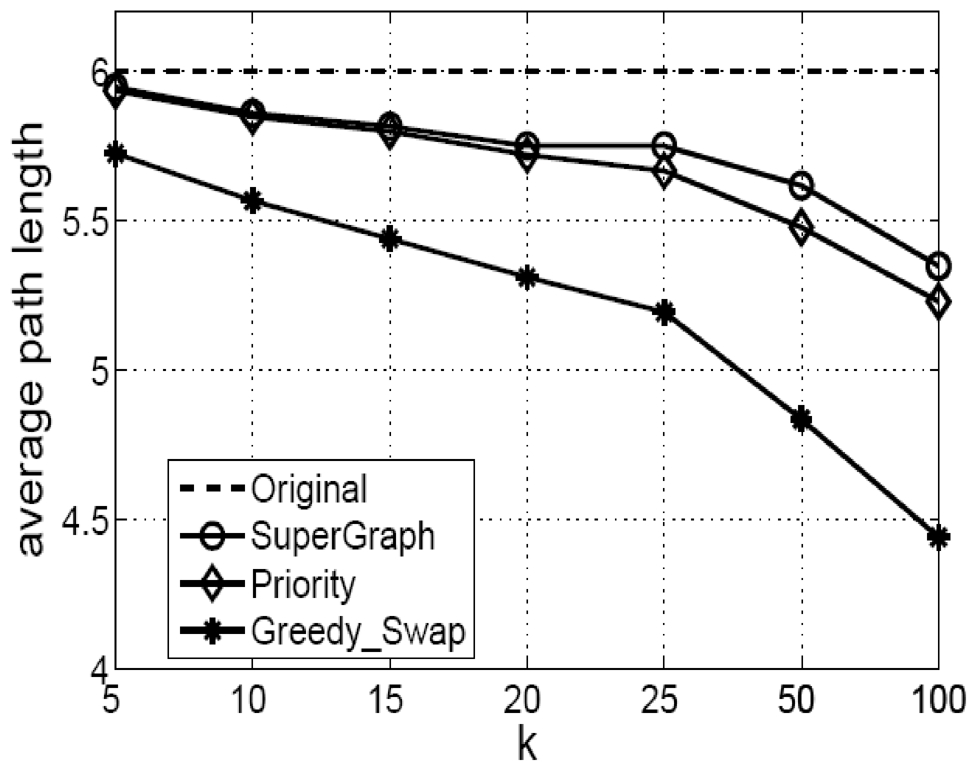
- **Datasets:** Co-authors, Enron emails, powergrid, Erdos-Renyi, small-world and power-law graphs

- **Goal:** degree-anonymization does not destroy the structure of the graph
  - Average path length
  - Clustering coefficient
  - Exponent of power-law distribution

Experiments: Clustering coefficient and Avg Path Length

- **Co-author** dataset
- APL and CC do not change dramatically even for large values of ***k***





Experiments: Edge intersections

Edge intersection achieved by the GreedySwap algorithm for different datasets.

Parenthesis value indicates the original value of edge intersection

Synthetic datasets	
Small world graphs*	0.99 (0.01)
Random graphs	0.99 (0.01)
Power law graphs**	0.93 (0.04)
Real datasets	
Enron	0.95 (0.16)
Powergrid	0.97 (0.01)
Co-authors	0.91(0.01)

(*) L. Barabasi and R. Albert: Emergence of scaling in random networks. *Science 1999.*

(**) Watts, D. J. Networks, dynamics, and the small-world phenomenon. *American Journal of Sociology* 1999

Experiments: Exponent of power law distributions

Original	2.07
k=10	2.45
k=15	2.33
k=20	2.28
k=25	2.25
k=50	2.05
k=100	1.92

Co-author dataset

Exponent of the power-law distribution as a function of ***k***

- Towards identity-anonymization on graphs

  K. Liu & E. Terzi, SIGMOD 2008

- A framework for computing the privacy score of users in social networks

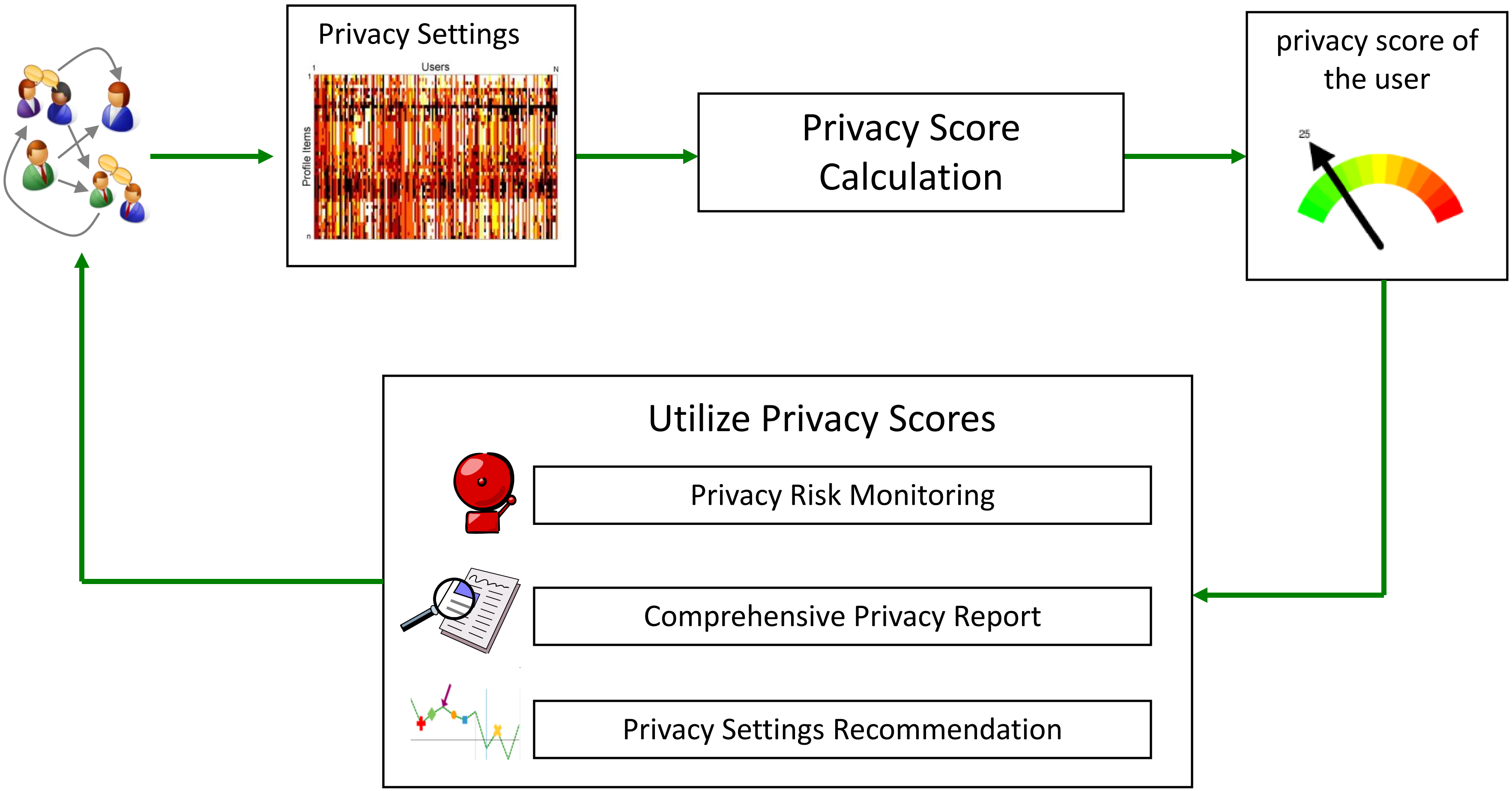
  K. Liu, E. Terzi, ICDM 2009

What is privacy risk score and why is it useful?

- What?
  - It is a credit-score-like indicator to measure the potential privacy risks of online social-networking users.
- Why?
  - It aims to boost public awareness of privacy, and to reduce the cognitive burden on end-users in managing their privacy settings.
    - privacy risk monitoring & early alarm
    - comparison with the rest of population
    - help sociologists to study online behaviors, information propagation

Privacy Score Overview

Privacy Score measures the potential privacy risks of online social-networking users.



How is Privacy Score Calculated? – Basic Premises

- **Sensitivity**: The more sensitive the

mother's maiden name is more sensitive than ***mobile-phone number***

privacy risk.

home address known ***by everyone*** poses higher risks than ***by friends only***

- **Visibility**: The wider the information about a user spreads, the higher his privacy risk.

Privacy Score Calculation

name, or gender, birthday, address, phone number, degree, job, etc.

Privacy Score of User *j* due to Profile Item *i*

$$PR(i, j) = \beta_i \times V(i, j).$$

sensitivity of profile item *i*

visibility of profile item *i*

Privacy Score Calculation

name, or gender, birthday, address, phone number, degree, job, etc.

Privacy Score of User *j* due to Profile Item *i*

$$\mathrm{PR}(i, j) = \beta_i \times V(i, j).$$

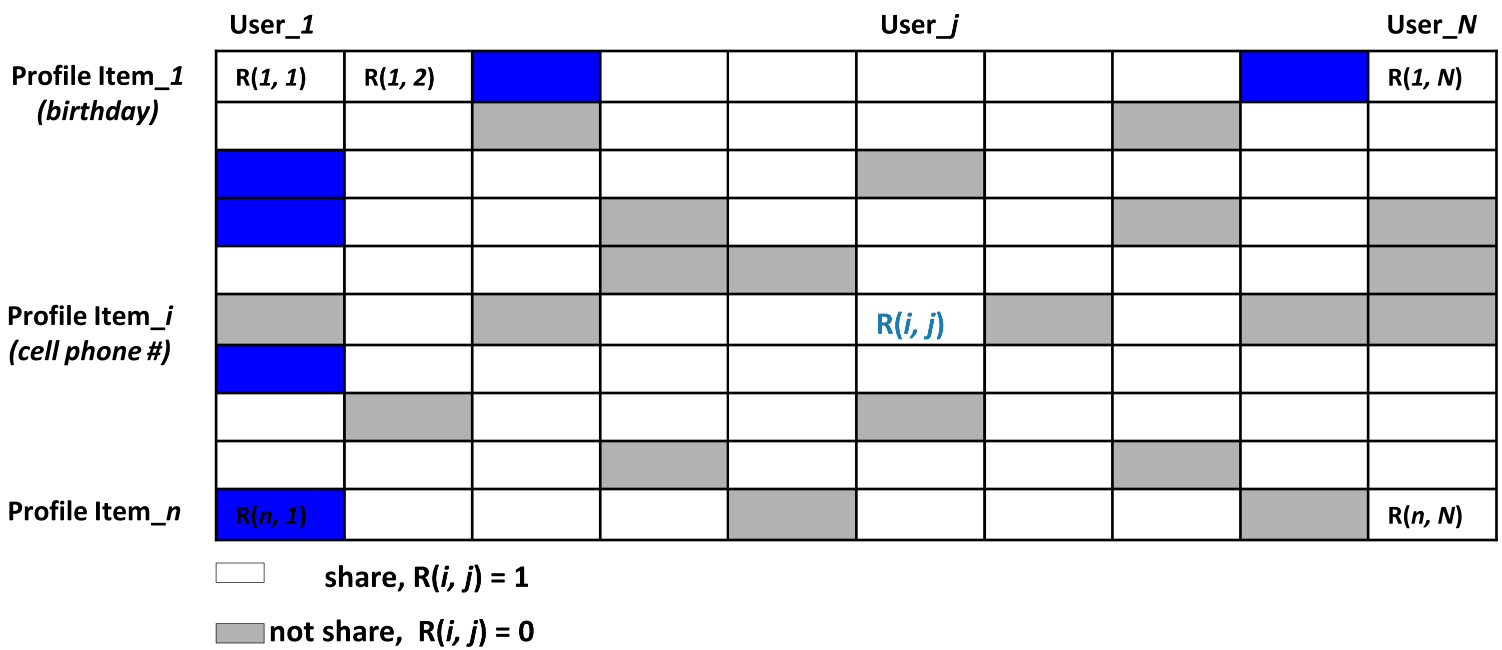
sensitivity of profile item *i*

visibility of profile item *i*

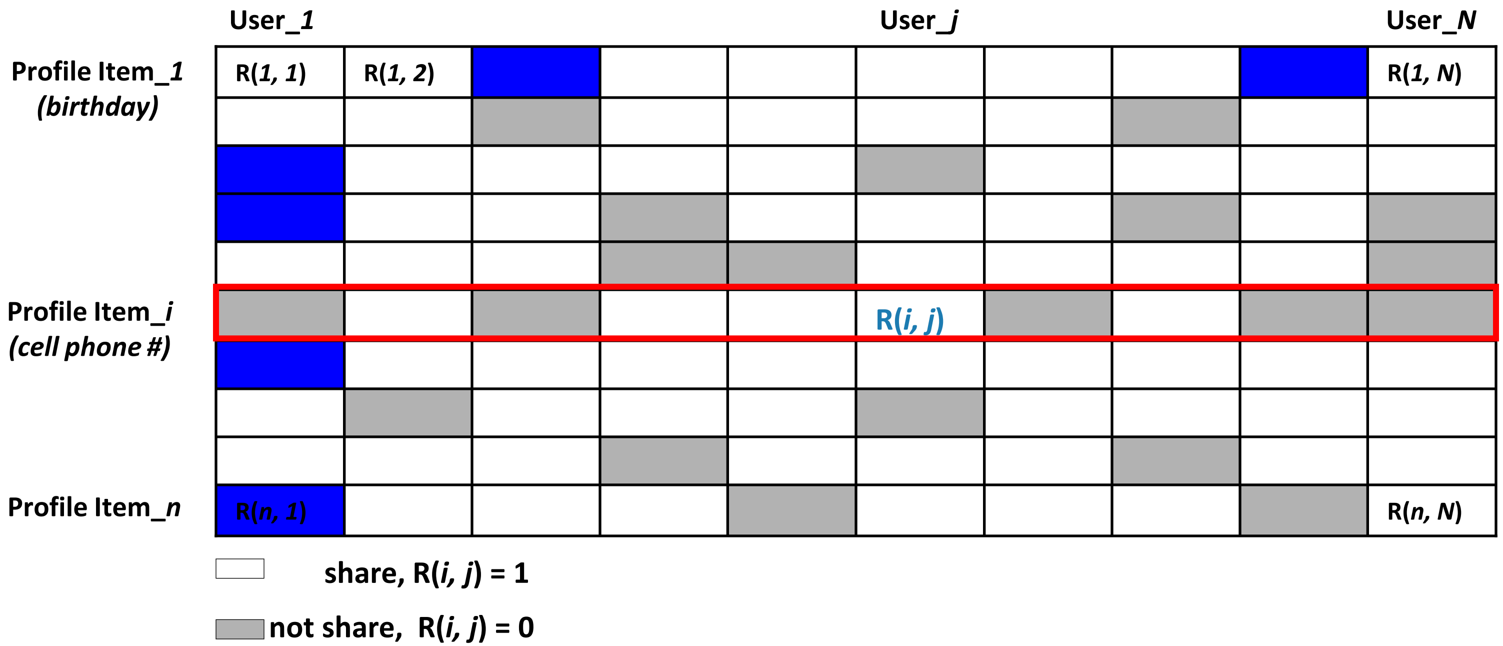
Overall Privacy Score of User *j*

$$\mathrm{PR}(j) = \sum_i \mathrm{PR}(i, j) = \sum_i \beta_i \times V(i, j).$$

The Naïve Approach



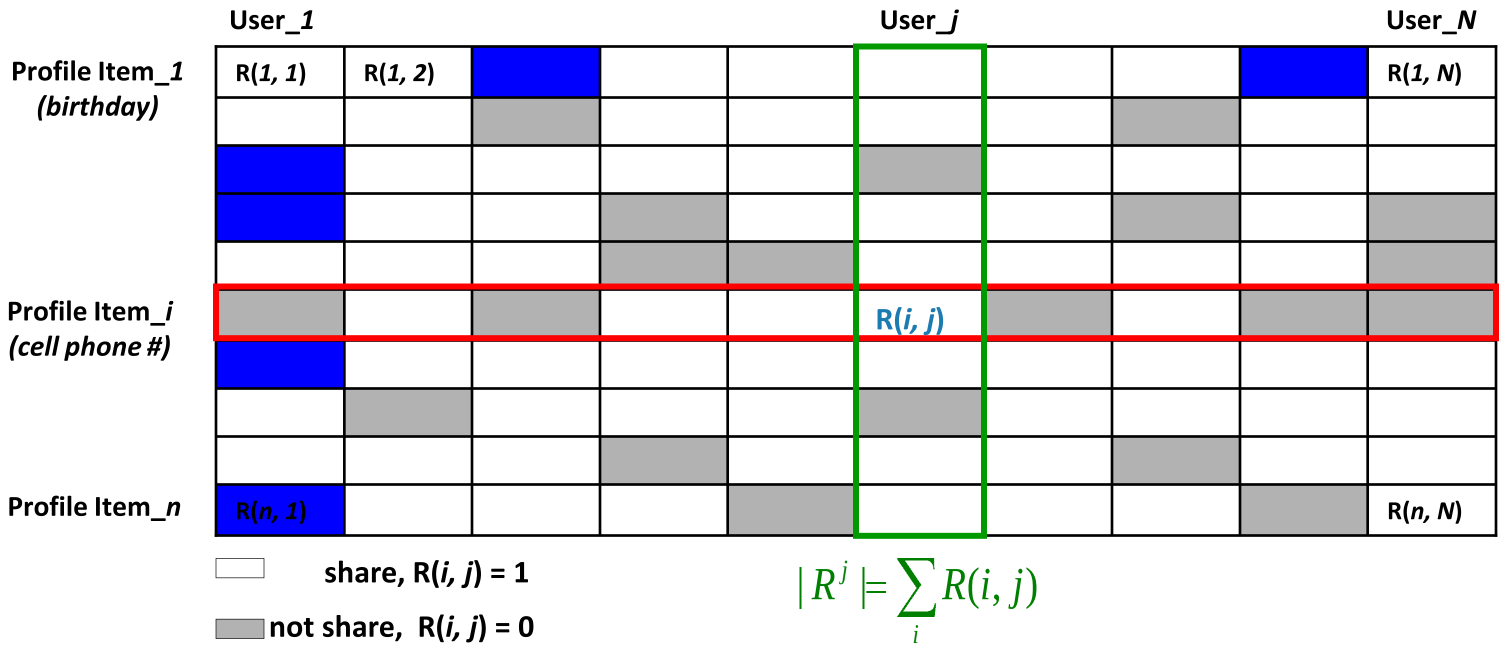
The Naïve Approach



$$|R_i| = \sum_j R(i, j)$$

Sensitivity: $$\beta_i = \frac{N - |R_i|}{N}$$

The Naïve Approach



$$|R_i| = \sum_j R(i,j)$$

$$|R^j| = \sum_i R(i,j)$$

Sensitivity: $$\beta_i = \frac{N - |R_i|}{N}$$

Visibility: $$V(i,j) = \Pr\{R(i,j) = 1\}$$

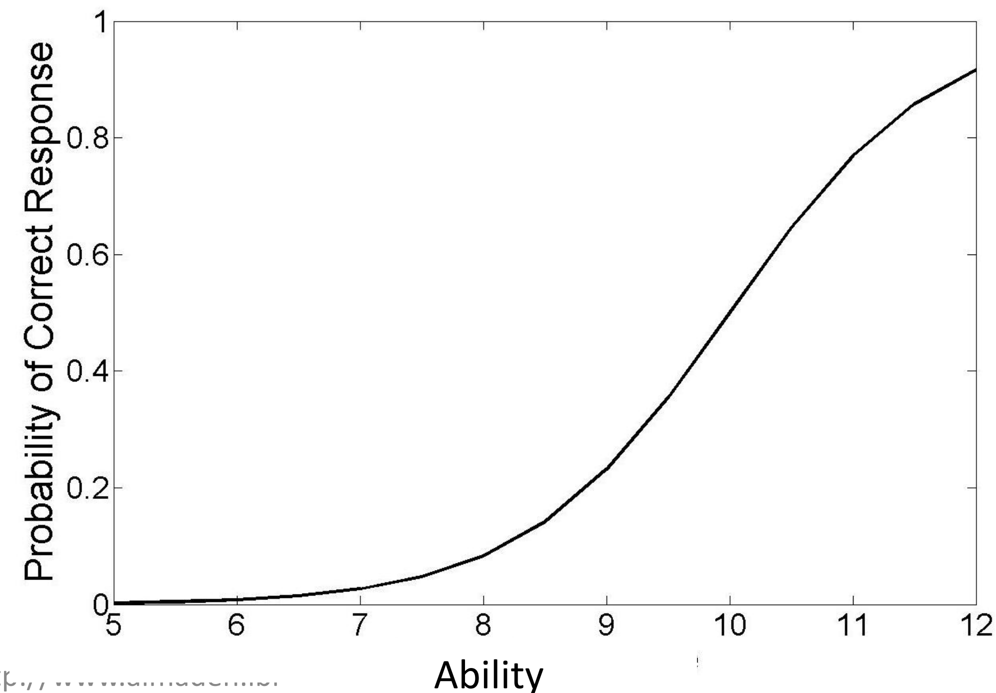
$$P_{ij} = \Pr\{R(i,j) = 1\} = \frac{|R_i|}{N} \times \frac{|R^j|}{n} = (1 - \beta_i) \times \frac{|R^j|}{n}$$

Advantages and Disadvantages of Naïve

- Computational Complexity O (Nn) – best one can hope
- Scores are sample dependent
  - Studies show that Facebook users reveal more identifying information than MySpace users
  - Sensitivity of the same information estimated from Facebook and from MySpace are different
- What properties do we really want?
  - Group Invariance: scores calculated from different social networks and/or user base are comparable.
  - Goodness-of-Fit: mathematical models fit the observed user data well.

Item Response Theory (IRT)

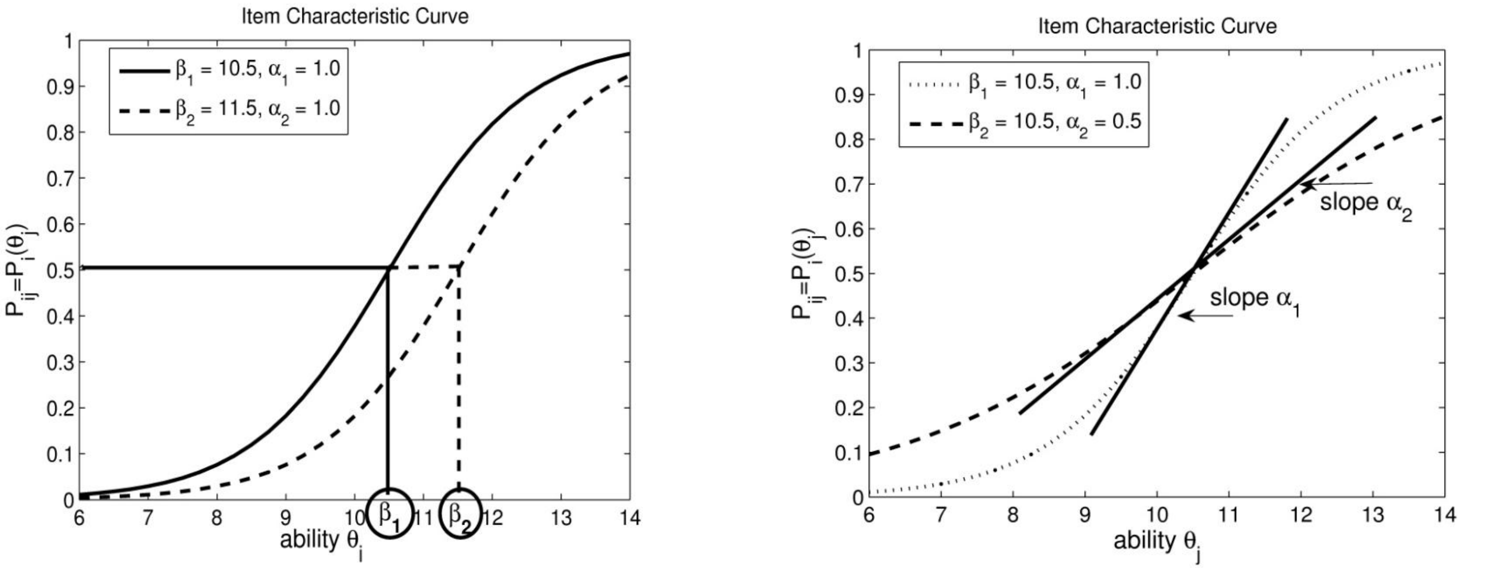
- IRT (Lawley,1943 and Lord,1952) has its origin in psychometrics.
- It is used to analyze data from questionnaires and tests.
- It is the foundation of Computerized Adaptive Test like GRE, GMAT



Item Characteristic Curve (ICC)

$$\text{ICC: } P_{ij} = \Pr\{R(i, j) = 1\} = \frac{1}{1 + e^{-\alpha_i(\theta_j - \beta_i)}}$$

- θ_j (ability): is an unobserved hypothetical variable such as intelligence, scholastic ability, cognitive capabilities, physical skills, etc.
- β_i (difficulty): is the location parameter, indicates the point on the ability scale at which the probability of correct response is .50
- α_i (discrimination): is the scale parameter that indexes the discriminating power of an item



Mapping from PRS to IRT

	Student_1 / User_1					Student_j / User_j				Student_N / User_N
Question Item_1 / Profile Item_1	R(*1, 1*)	R(*1, 2*)	……	……	……	……	……	……	……	R(*1, N*)
Question Item_i / Profile Item_i						R(*i, j*)				
Question Item_n / Profile Item_n	R(*n, 1*)	……	……	……	……	……	……	……	……	R(*n, N*)

correct answer, R(*i, j*) = 1 / share, R(*i, j*) = 1

wrong answer, R(*i, j*) = 0 / not share, R(*i, j*) = 0

$$P_{ij} = \Pr\{R(i, j) = 1\} = \frac{1}{1 + e^{-\alpha_i(\theta_j - \beta_i)}}$$

discrimination → discrimination

ability → attitude/privacy concerns

difficulty → sensitivity

Prob of correct answer → Prob of share the profile

Computing PRS using IRT

Overall Privacy Risk Score of User *j*

$$\mathrm{PR}(j) = \sum_i \beta_i \times V(i, j)$$

Sensitivity: β_i

Visibility: $V(i, j) = P_{ij} \times 1 + (1 - P_{ij}) \times 0 = P_{ij}$, where $P_{ij} = \Pr\{R(i, j) = 1\}$

$$P_{ij} = \Pr\{R(i, j) = 1\} = \frac{1}{1 + e^{-\alpha_i(\theta_j - \beta_i)}}$$

Byproduct: profile item's discrimination and user's attitude

Calculating Privacy Score using IRT

Overall Privacy Score of User *j*

$$\mathrm{PR}(j) = \sum_i \beta_i \times V(i, j)$$

Sensitivity: β_i

Visibility: $V(i, j) = \Pr\{R(i, j) = 1\}$

$$P_{ij} = \Pr\{R(i, j) = 1\} = \frac{1}{1 + e^{-\alpha_i(\theta_j - \beta_i)}}$$

byproducts: profile item's discrimination and user's attitude

All the parameters can be estimated using Maximum Likelihood Estimation and EM.

Advantages of the IRT Model

- The mathematical model fits the observed data well

- The quantities IRT computes (*i.e.*, sensitivity, attitude and visibility) have intuitive interpretations

- Computation is parallelizable using e.g. MapReduce

Estimating the Parameters of a Profile Item

Known Input:

Attitude level	Share	Not share	# of users at this attitude level
θ_1	*r_i1*	*f_1 – r_i1*	*f_1*
θ_2	*r_i2*	*f_2 – r_i2*	*f_2*
θ_3	*r_i3*	*f_3 – r_i3*	*f_3*
……	……	……	……
θ_g	*r_ig*	*f_g – r_ig*	*f_g*
……	……	……	……
……	……	……	……
θ_K	*r_iK*	*f_g – r_iK*	*f_K*

$$\sum_{g=1}^{K} f_g = N$$

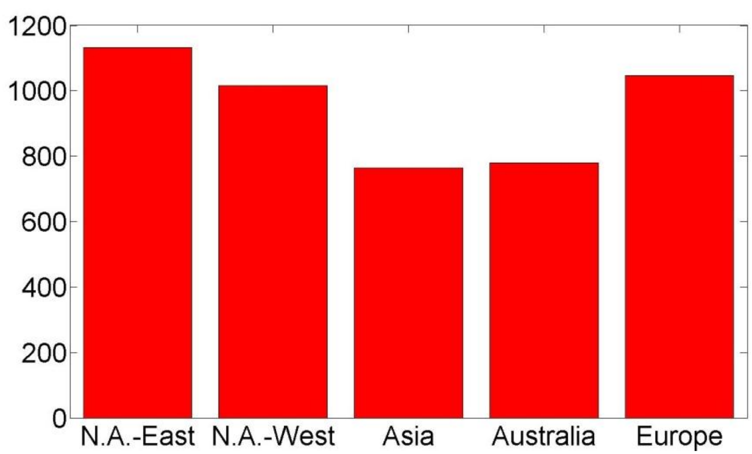
Log likelihood: $$L = \log\left[\prod_{g=1}^{K}\binom{f_g}{r_{ig}} P_{ig}^{\;r_{ig}}\; 1 - P_{ig}^{\;f_g - r_{ig}}\right],$$ where $P_{ig} = \dfrac{1}{1+e^{-\alpha_i(\theta_g-\beta_i)}}$

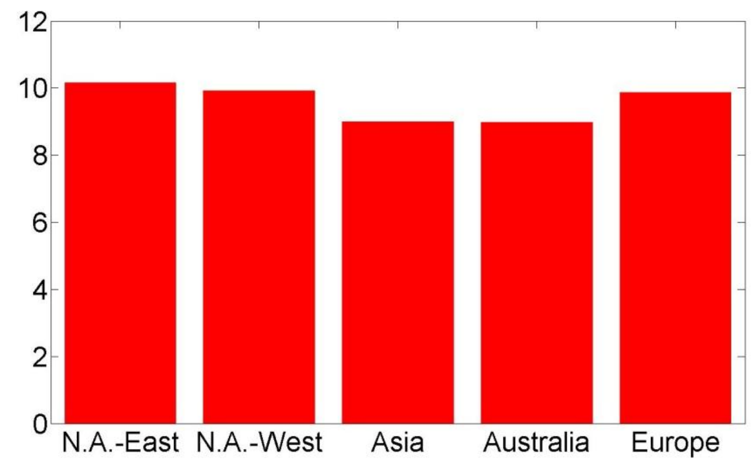
Newton-Raphson: $$\begin{bmatrix}\alpha_i\\ \beta_i\end{bmatrix}_{t+1} = \begin{bmatrix}\alpha_i\\ \beta_i\end{bmatrix}_{t} - \begin{bmatrix}L_{11} & L_{12}\\ L_{21} & L_{22}\end{bmatrix}_t^{-1} \times \begin{bmatrix}L_1\\ L_2\end{bmatrix}_t,$$

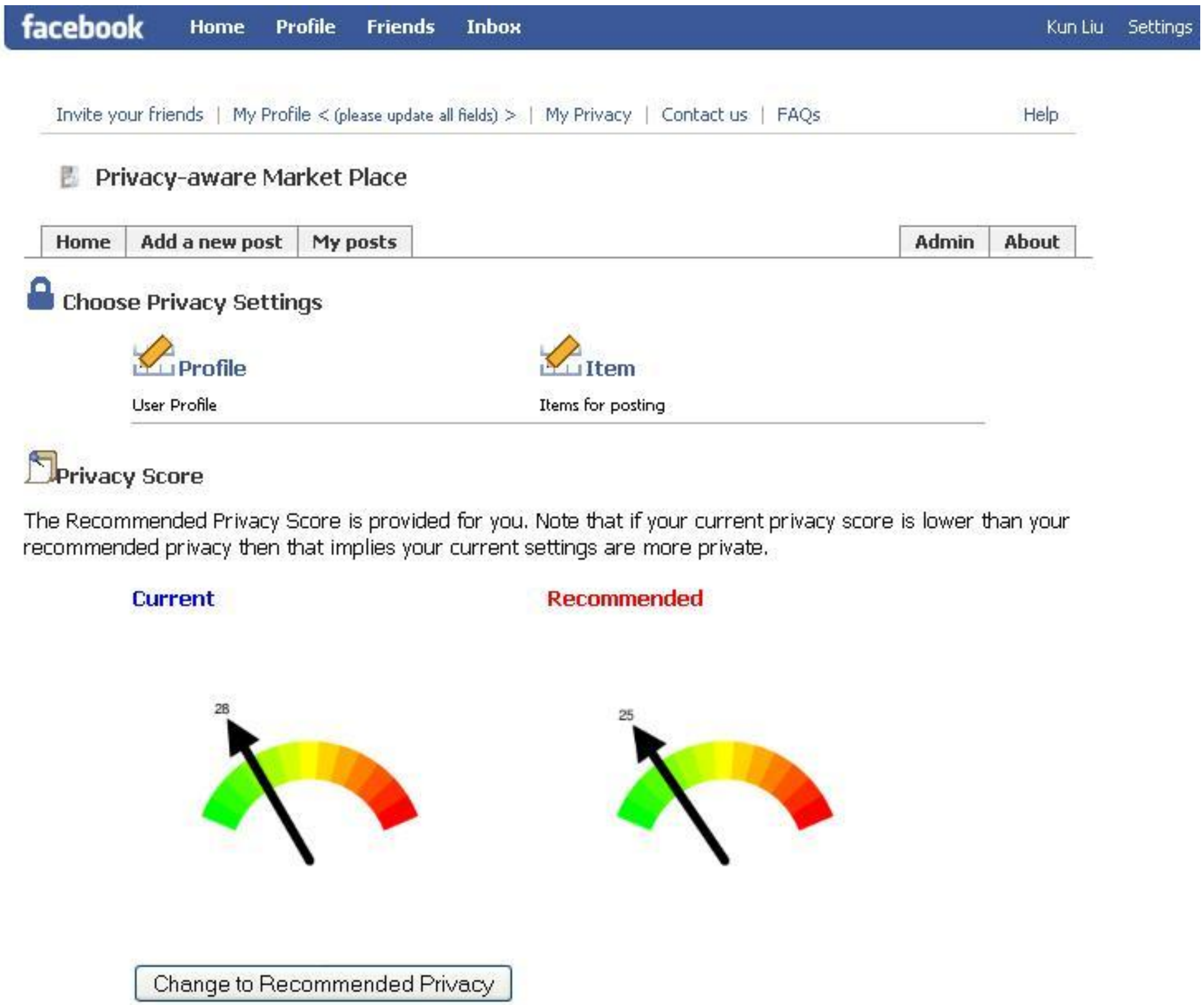
where $L_1 = \dfrac{\delta L}{\delta \alpha_i}$, $L_2 = \dfrac{\delta L}{\delta \beta_i}$,

$L_{11} = \dfrac{\partial^2 L}{\partial \alpha_i^{\;2}}$, $L_{22} = \dfrac{\partial^2 L}{\partial \beta_i^{\;2}}$, $L_{12} = L_{21} = \dfrac{\partial^2 L}{\partial \alpha_i \beta_i}$.

Experiments – Interesting Results






facebook
Home
Profile
Friends
Inbox
Kun Liu
Settings
Invite your friends | My Profile < (please update all fields) > | My Privacy | Contact us | FAQs
Help
Privacy-aware Market Place
Home
Add a new post
My posts
Admin
About
Choose Privacy Settings
Profile
User Profile
Item
Items for posting
Privacy Score
The Recommended Privacy Score is provided for you. Note that if your current privacy score is lower than your recommended privacy then that implies your current settings are more private.
Current
Recommended
28
25
Change to Recommended Privacy