SPARSI: Partitioning Sensitive Data Amongst Multiple Adversaries

Theodoros Rekatsinas, Amol Deshpande
Ashwin Machanavajjhala

University of Maryland
Duke University
Show me your data

and I shall give you useful services
Show me your data

and I shall give you useful services

but I may learn sensitive information about you.
Location services

[Map showing Alice's location in Hangzhou, China]
Location services
Location services
Location services
Location services
Location services
Location services

[Location map with markers for Alice and Bob]
Friendship is sensitive...

If the trajectories of two users are very similar they are friends with high probability. [Cho et al., KDD `11]

Alice and Bob are probably friends so start sending Bob Alice’s ads and recommendations.
<table>
<thead>
<tr>
<th>Medical transcription</th>
</tr>
</thead>
</table>

**Admitting Summary**

- **Name:** McGrath, David H
- **Address:** 56 Bowman Street
- **Birth Date:** 03/31/74
- **Admit Date:** 05/13/92
- **Time:** 1112
- **Bed Number:** 554/D
- **Social Security Number:** 99999999

**Admitting Physician:** Dr. Peter E
**Service:** Pediatrics
**Reason for Visit:** Germ Cell Brain Tumor

- **Address:** Fallon Clinic 95 E MAI
- **City:** Westboro
- **State:** MA
- **Zip:** 01581
Medical transcription

Patient’s name

Patient

No sensitive information
Medical transcription

Patient’s name: McGrath, David H

Physician’s name: R. Peter E

Disease: Germ cell brain tumor

Patient: No sensitive information

Doctor treating the patient: Some sensitive information

Doctor treating the patient and patient’s disease: Extremely sensitive information
Most techniques rely on adding controlled noise and try to preserve statistical patterns of the data (e.g., differential privacy).
Not only noise is annoying but …

For many applications:

- **no noise** to individual entries to obtain utility
- sensitive information disclosed _implicitly_ via associating data entries
How do you ensure privacy?

**Idea:** To obtain privacy, break the associations across data entries.

Fortunately there are many adversaries that have no incentive to collude (e.g., legal contracts).

**Ex.:** Multiple location service providers and multiple transcriptionists
Can we ensure privacy by “scattering” data across multiple non-colluding adversaries?
SPARSI: A framework for private data partitioning

... means "scattered or strewn"

- Problem formulation
- Algorithms for private data partitioning
- Selected experiments
SPARSI: A framework for private data partitioning

... means "scattered or strewn"

- Problem formulation
- Algorithms for private data partitioning
- Selected experiments
Problem formulation

Record field

PATIENT NAME (LAST, FIRST, M.I.)
MCGRATH, DAVID H

PATIENT ADDRESS
56 BOWMAN STREET

AGE 18
SEX M
MAR S

REASON FOR VISIT
GERM CELL BRAIN TUMOR

Transcriptionists
Problem formulation

Record field

Transcriptionists

PATIENT NAME (LAST, FIRST, M.I.)
MCGRATH, DAVID H

PATIENT ADDRESS
56 BOWMAN STREET

AGE 18
SEX M
MAR S

REASON FOR VISIT
GERM CELL BRAIN TUMOR
Problem formulation

Record field

- Age: 18
- Sex: M
- Reason for Visit: Germ Cell Brain Tumor

Transcriptionists

- Patient Name: McGrath, David H
- Patient Address: 56 Bowman Street
Problem formulation

Record field

Transcriptionists

How do you model overall utility?
Problem formulation

Record field

How do you model overall utility?

How do you model information disclosure?
Problem formulation

Record field

Transcriptionists

How do you model overall utility?
How do you model information disclosure?
How do you scatter the data?
Utility

Disclosing data to adversaries provides utility to the adversaries but also to the user.

**Location services:** Users get valuable services; providers can improve or personalize services.

**Transcription:** User completes task; transcriptionists earn money.

**Idea:** Merge adversaries’ and user’s utility into a single non-decreasing submodular function.
Implicit via sensitive properties

Dependency graph

p1: Patient’s disease

p2: Doctor’s liability

Patient’s name

Physician’s name

Disease
Information disclosure

Different families of disclosure functions for each property

Step functions:

Patient’s disease

<table>
<thead>
<tr>
<th>Patient’s name</th>
<th>Disease</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCGRATH, DAVID H</td>
<td>GERM CELL BRAIN TUMOR</td>
</tr>
</tbody>
</table>

Disclosure level

0.0 0.5 1.0
Information disclosure

Different families of disclosure functions for each property

Step functions:

Patient’s disease

Patient’s name

Disease

Disclosure level

0.0  0.5  1.0
Information disclosure

Different families of disclosure functions for each property

Step functions:

Patient’s disease

<table>
<thead>
<tr>
<th>Patient’s name</th>
<th>Disease</th>
</tr>
</thead>
<tbody>
<tr>
<td>PATIENT NAME (LAST, FIRST, M.I.)</td>
<td>GERM CELL BRAIN TUMOR</td>
</tr>
</tbody>
</table>

Disclosure level

0.0  0.5  1.0
Information disclosure

Different families of disclosure functions for each property

Superadditive functions:

Doctor’s liability

Patient’s name

Physician’s name

Disease

Disclosure level

0.0 0.5 1.0

MCGRATH, DAVID H

ADMITTING PHYSICIAN

REASON FOR VISIT

GERM CELL BRAIN TUMOR
Information disclosure

Different families of disclosure functions for each property

Superadditive functions:

Doctor’s liability

<table>
<thead>
<tr>
<th>Patient’s name</th>
<th>Physician’s name</th>
<th>Disease</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCGRATH, DAVID H</td>
<td>ADMITTING PHYSICIAN R, PETER E</td>
<td>GERM CELL BRAIN TUMOR</td>
</tr>
</tbody>
</table>

Disclosure level

0.0 0.5 1.0
Information disclosure

Different families of disclosure functions for each property

Superadditive functions:

Doctor’s liability

<table>
<thead>
<tr>
<th>Patient’s name</th>
<th>Physician’s name</th>
<th>Disease</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCGRATH, DAVID H</td>
<td>R. PETER E</td>
<td>GERM CELL BRAIN TUMOR</td>
</tr>
</tbody>
</table>

Disclosure level

0.0  0.5  1.0
Information disclosure for each adversary

\[ f_a \in F : 2^D \rightarrow [0, 1]^{|P|} \]

Overall disclosure

worst disclosure

f_\infty = \max_{a \in A} (\|f_a(S_a)\|_\infty)

average disclosure

f_{L_1} = \max_{a \in A} \left( \frac{\|f_a(S_a)\|_1}{|P|} \right)

S_a: data entry to adversary assignment
Scattering data

Assignment of data to adversaries

\[
\text{maximize } \sum_{S \in \mathcal{P}(D \times A)} S \quad \text{subject to} \quad f(S) \leq \tau_I, \\
\sum_{a=1}^{k} x_{da} \leq t, \forall d \in D, \\
x_{da} \in \{0, 1\}.
\]

Utility

\[
u(S') + \lambda(\tau_I - f(S'))
\]

Disclosure

Sensitive data partitioning is NP-hard
SPARSI: A framework for private data partitioning

... means "scattered or strewn"

- Problem formulation
- Algorithms for private data partitioning
- Selected experiments
Special disclosure functions

**Step functions:** Property is disclosed only if all the data entries connected to it are assigned to the same adversary.

**Linear functions:** Property disclosure increases linearly to the number of entries assigned to the same adversary.

**Solution:** Relax, Solve LP, Round

**Guarantees:** Submodular maximization, fair allocation
General disclosure functions

Greedy Randomized Adaptive Search Procedure (GRASP):

- **Construction:** Compute initial assignment
- **Local search:** Explore solution neighborhood for improvements
Local search variations

**Greedy:** Pick the data-adversary assignment that offers the **maximum objective improvement.**

**Randomized:** Pick the **top-k** data-adversary assignments choose one randomly.

- Randomization helps avoiding local optima
SPARSI: A framework for private data partitioning

... means "scattered or strewn"

• Problem formulation
• Algorithms for private data partitioning
• Selected experiments
Setting

Users publish check-in data using a social network and the social network discloses the check-ins to advertisers.

Data items: Check-ins

Sensitive properties: Friendship links

Information disclosure: $\Pr[\text{friends}(u_1, u_2)] \propto \cosSim(\text{trj}(u_1), \text{trj}(u_2))$

Utility: Different advertiser utilities for different locations
Check-in data from BrightKite

BK-full: 4.5 million check-ins, 58k users, 214k edges

BK-sample: 365k check-ins, 3k nodes, 2.9k edges
Algorithms

**RAND+**: Data entries partitioned at random. The probability of assigning a data entry to an adversary is proportional to the corresponding utility

**GREEDY**: Greedy local-search without randomization

**GRASP**: Greedy local-search with randomization

**GREEDYL/GRASPL**: Efficient variations with reduced local-search scope
Results: BK-sample

Clearly the proposed algorithms outperform RAND+
Results: BK-full

Clearly the proposed algorithms outperform RAND+.
Conclusions
Conclusions

For many applications exact data must be disclosed in return for services.
Conclusions

For many applications exact data must be disclosed in return for services.

In many cases we have multiple adversaries with no incentive to collude.
Conclusions

For many applications exact data must be disclosed in return for services.

In many cases we have multiple adversaries with no incentive to collude.

SPARSI: A framework that ensures privacy against multiple non-colluding adversaries.
Conclusions

For many applications exact data must be disclosed in return for services.

In many cases we have multiple adversaries with no incentive to collude.

SPARSI: A framework that ensures privacy against multiple non-colluding adversaries.

Thank you!

thodrek@cs.umd.edu