A Differentially Private Selective Aggregation Scheme for Online User Behavior Analysis

Presenter: Fudong Qiu
Dec. 9 2015

Jianwei Qian, Fudong Qiu, Fan Wu, Na Ruan, Guihai Chen, Shaojie Tang§
Shanghai Jiao Tong University, China
§University of Texas at Dallas, USA
Outline

- Motivation & Problems
- Basic Model
  - System Model
  - Main Challenges
- Protocol Design & Extensions
  - Design rationale
  - HE & DP → Protocol
  - Extensions
- Evaluation results
- Summary
Motivation & Problems

• Online user behavior analysis is widely applied
• But it results in users’ privacy disclosure

❖ Conflict between the utility of analysis results and users’ privacy
Motivation & Problems

• Data aggregation
  – Overall aggregation
    • SELECT AVG(time online) FROM db
  – Selective aggregation
    • SELECT AVG(time online) FROM db WHERE gender='female'

• Prior works do not support selection

• Our goal
  – to design a practical protocol that supports selective aggregation, while still preserving users’ privacy.
System Model

- **Clients** are installed on user side, e.g., plug-ins. *(trusted)*
- **Analyst** wants to query on user data. *(semi-honest)*
- **Intermediary**, comprised of
  - *aggregator*, in charge of data storage and manipulation
  - *authority*, in charge of key management
  - *(Both semi-honest, and do not collude)*
## System model

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Notation</th>
<th>User Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numeric attribute</td>
<td>$a_i$</td>
<td>Non-negative integer</td>
<td>“age”</td>
</tr>
<tr>
<td>Boolean attribute</td>
<td>$b_j$</td>
<td>1 or 0</td>
<td>“gender is female”</td>
</tr>
</tbody>
</table>

User attributes are stored at Aggregator as a table $T$

$$T(id, a_1, a_2, ..., a_i, b_1, b_2, ..., b_j)$$
Key idea of selective aggregation

Use Boolean attribute(s) to select numeric attribute

E.g., \texttt{SELECT AVG(age) FROM db WHERE gender='female'}

<table>
<thead>
<tr>
<th>User ID</th>
<th>“Age”</th>
<th>“Gender is female”</th>
<th>=</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>21</td>
<td>x</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>30</td>
<td>x</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>44</td>
<td>x</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>23</td>
<td>x</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>19</td>
<td>x</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>35</td>
<td>x</td>
<td>0</td>
</tr>
</tbody>
</table>

Sum = 3 \quad 63

Mean = \frac{63}{3} = 21

All computations are on ciphertexts
Challenges

• Challenges to achieve our goals
  – Compute selective aggregation obliviously/blindly
  – Achieve differential privacy in addition to confidentiality
  – Resist client churn (i.e., clients get offline frequently)
Building blocks

- Geometric distribution
- Discrete Laplace distribution
- Differential privacy
- BGN cryptosystem
- Homomorphic encryption
- Strong privacy guarantee
Differential Privacy (DP)

• DP protects any individual’s privacy by making it very hard to infer her data from the aggregate result.
• DP is usually achieved by adding noise

**Theorem 1.** Given a query $Q$, let two independent variables $Z_1, Z_2 \sim \text{Geo}(1 - p)$ and $Z = Z_1 - Z_2$, adding $Z$ to the result of $Q$ achieves $\epsilon$-differential privacy, where $p = \exp(-\epsilon/\Delta Q)$.

• $Z_1, Z_2$ as two half-noises enable the Intermediary (Aggregator & Authority) to add noise obliviously.
BGN Homomorphic Encryption

The Boneh-Goh-Nissim Cryptosystem (BGN)

Suppose \( c_1 = E(m_1), c_2 = E(m_2) \),

- addition (unlimited)
  \[ c_1 c_2 = E(m_1 + m_2) \]

- multiplication (once)
  \[ e(c_1, c_2) = E(m_1 m_2) \]

bilinear pairing
Protocol Design

Selective aggregation protocol

• 1. Setup
  – Key generation, parameter setting

• 2. Data collection
  – Clients collect and encrypt user data, then send it to Aggregator

• 3. Query evaluation
  – The key to achieve selective aggregation is counting in data items of target users by multiplying them by 1 and skipping the rest by multiplying them by 0.
  – These calculations are done in ciphertext, and thus no privacy disclosure would occur.
Protocol: query evaluation

- First, we assume the aggregation is selected by a single boolean attribute.
Extension

• What if the aggregation is selected by multiple Boolean attributes?
  – E.g., SELECT AVG(income) FROM db WHERE gender='female' and education='doctoral'
Evaluation

• Dataset
  – Implement the BGN cryptosystem using Pairing-based Cryptography library.
  – Security parameter was set to 80.
  – 1000 users’ demographics and online behaviors in 4 weeks.
  – The behaviors include webpages browsed, time spent on each webpage, browsers used.
  – The demographic profiles include gender, age, education level, income, occupation, etc.
Evaluation

• Performance
  – Relative errors
  – Computation overhead
  – Communication overhead

• Query examples
  – Ratio of male users (selective)
  – Average number of webpages viewed by users of male/female (selective)
  – Average number of times of using Internet Explorer for female and bachelor-degree users (extension)
Evaluation: client churn resistance

• Comparison with a prior work SplitX

Better accuracy esp. when most clients are offline

![Graph showing comparison between SplitX and PPSA for different values of s.](image)
Evaluation: computation overhead

- Proportional to sample size, i.e., \#users
- Run time <1s for 1000 users
Evaluation: communication overhead

• Communication overhead for one query:

<table>
<thead>
<tr>
<th>Communication overhead (Bytes)</th>
<th>Basic PPSA</th>
<th>Extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authority</td>
<td>87</td>
<td>30n+107</td>
</tr>
<tr>
<td>Aggregator</td>
<td>60</td>
<td>30n+60</td>
</tr>
</tbody>
</table>

• It is acceptable
Summary

• We designed the first system that is able to securely and selectively aggregate user data, making it practical in realistic data analytics.

• It guarantees strong privacy preservation by utilizing differential privacy mechanism and homomorphic encryption to protect users privacy.

• We implement PPSA to evaluate aggregation selected by one Boolean attribute, and extend it to multiple Boolean attributes.
Thanks