AHEAD: Adaptive Hierarchical Decomposition for Range Query under Local Differential Privacy

Linkang Du, Zhikun Zhang, Shaojie Bai, Changchang Liu, Shouling Ji, Peng Cheng, Jiming Chen

CCS 2021
1. Background
Big Data Era

**Data collection**

- Browsing history, typing habit, income, work hours, location, ...

**Data analysis**

- Improving user experience, recommendation, ...

User data → Data collection → Data analysis
Privacy Accidents

2017, Yahoo breached 3 billion user data

2018, Facebook exposed 87 million user data

2018, Huazhu Hotel breached 0.5 billion user data

2019, global PACS leaked 24 million records

2020, Microsoft exposed 250 million records

2020, 6.4 million voters’ data in Israel were leaked
Deployment of Local Differential Privacy (LDP)

Google chrome browser
- Collecting browsing statistic

Apple iOS and MacOS
- Collecting typing statistic

Microsoft windows
- Collecting telemetry data

Snap (parent company of Snapchat)
- Performing modeling of user preference
2. Preliminaries
Workflow of LDP Protocol

- Data Encoding
- Perturbation Mechanism $\Psi$

- Data Encoding
- Perturbation Mechanism $\Psi$

- Data Encoding
- Perturbation Mechanism $\Psi$

- Data Encoding
- Perturbation Mechanism $\Psi$
The mechanism $\Psi$ satisfies $\epsilon$-LDP if for any input $v_1, v_2 \in D$, we have

$$\forall T \subseteq Range(\Psi): \Pr[\Psi(v_1) \in T] \leq e^{\epsilon} \Pr[\Psi(v_2) \in T]$$
The mechanism \( \Psi \) satisfies \( \epsilon \)-LDP if for any inputs \( v_1, v_2 \in D \), we have

\[
\forall T \subseteq \text{Range}(\Psi): \Pr[\Psi(v_1) \in T] \leq e^\epsilon \Pr[\Psi(v_2) \in T]
\]
Problem Setting: Answering Range Query under LDP

What is the range query problem?

- 1-dim range query: the proportion of people with $20 < \text{Age} < 40$
- 3-dim range query: the ratio of people with $20 < \text{Age} < 40$, Salary < 5000, and Loan amount < 20000

What is the challenge for answering range query under LDP?

- The perturbation noise decreases the utility of original data
- The exponential growth of multi-dimensional data domain
- The entire process needs to maintain the correlation between multi-dimensional data
Related Works For Answering Range Query under LDP

**Wavelet Transform Based Methods**

- Converting each user’s private value to a Haar wavelet coefficient vector for perturbation
  
  [Xiao et al., TKDE’ 10]

- Reintroducing the Wavelet Transforms as a useful tool in local privacy [Cormode et al., PVLDB’ 19]

**Hierarchy Based Methods**

- Hierarchically decomposing the domain based on the complete $B$-ary tree structure
  
  [Hay et al., PVLDB’ 10]

- Combining a larger branching factor with constraint inference [Qardaji et al., PVLDB’ 13]

- Multi-dimensional hierarchy of intervals to handle high-dimensional scenarios
  
  [Wang et al., SIGMOD’ 19]
Grid Based Methods

- Laying a coarse-grained grid over the dataset, and then further partitions each cell according to its noisy count [Qardaji et al., ICDE’ 13]
- Combining information from 1-dim and 2-dim grids to answer range queries [Yang et al., arXiv’ 20]

Other Methods

- Collecting low-dimensional marginals and reconstruct a high-dimensional marginal from them [Zhang et al., CCS’ 18]
- Reporting a value close to original user data with higher probability than a value farther away from original user data [Li et al., SIGMOD’ 20]
Limitations

These works are limited in practice due to at least one of the following reasons:

- The small values are highly likely to be overwhelmed by the injected noises
- The domain exponential exploding problem in multi-dimensional scenes
- Destroying the correlation between multi-dimensional attributes
- Not satisfying privacy requirements (designed for DP, not LDP)
3. AHEAD
Workflow of AHEAD

AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)
AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)
AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)
Workflow of AHEAD

AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)
AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)
AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)
AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)
Workflow of AHEAD

AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)
AHEAD contains four steps

- Step 1: User Partition (UP)
- Step 2: Noisy Frequency Construction (NFC)
- Step 3: New Decomposition Generation (NDG)
- Step 4: Post-processing (PP)

How to properly choose parameters for AHEAD?
Query error analysis
**Parameter Settings**

**Query error analysis**

- Noise error originates from the perturbation process
- Non-uniform error arises from some intervals whose values are approximated by larger intervals’ values

---

**Diagram:**

- **Noise error**
  - \( n_{11}, n_{12}, n_{13}, n_{14} \)
  - \( \tilde{f}_{11}, \tilde{f}_{12}, \tilde{f}_{6/2} \)

- **Non-uniform error**
  - \( n_{5}, n_{6} \)
  - \( \tilde{f}_{5}, \tilde{f}_{6} \)

---

**Threshold \( \theta \)**

- \( n_{11}, n_{12}, n_{13}, n_{14} \)
  - \( \tilde{f}_{11}, \tilde{f}_{12}, \tilde{f}_{6/2} \)

---

**Equations:**

- Without threshold:
  - \( n_{11} = 4 \)
  - \( n_{12} = 5 \)
  - \( n_{13} = 6 \)
  - \( n_{14} = 7 \)
  - \( 0 + \sigma^2 \)

- With threshold:
  - \( n_{11} = 4 \)
  - \( n_{12} = 5 \)
  - \( n_{13} = 6 \)
  - \( n_{14} = 7 \)
  - \( 0 + \sigma^2 \)

**Frequency:**

- \( \text{frequency}([5, 6]) = ? \)
  - \( \text{Answer: } 0.35 + 2\sigma^2 \)
  - \( \text{Query error: } 2\sigma^2 \)

---

**Query error:**

- \( 1.5\sigma^2 + 0.025 \)
Choosing threshold $\theta$

- Reducing the total errors by setting the threshold

\[ \mathbb{E} [\text{Err}_1] = \mathbb{E} \left[ (\hat{f}_1 - f_1)^2 + (\hat{f}_2 - f_2)^2 + \cdots + (\hat{f}_B - f_B)^2 \right] \]

\[ = \mathbb{E} \left[ B \cdot X^2 \right] = B \mathbb{E} [X^2] = B\sigma^2 \]

\[ \mathbb{E} [\text{Err}_2] = \mathbb{E} \left[ \left( \frac{\hat{f}}{B} - f_1 \right)^2 + \left( \frac{\hat{f}_2}{B} - f_2 \right)^2 + \cdots + \left( \frac{\hat{f}_B}{B} - f_B \right)^2 \right] \]

\[ = \mathbb{E} \left[ \eta^2 f^2 + \sum_{i=2}^{B} \left( \frac{f_i}{B} - f_i \right)^2 \right] + \mathbb{E} \left[ \frac{X^2}{B} \right] \]

Let $\mathbb{E} [\text{Err}_2] < \mathbb{E} [\text{Err}_1]$, get $\theta = \sqrt{(B + 1)\sigma^2}$
Choosing branch $B$

- $B$ is used to balance the tree height and the number of nodes required to answer the query
- Considering the non-uniform error in the worst case

$$
\mathbb{E}[\text{Err}_3] = \mathbb{E}\left[\left(\frac{X}{B} + (f - \frac{f}{B})\right)^2\right] = \frac{c\sigma^2}{B} + \left(f - \frac{f}{B}\right)^2 = \frac{c\sigma^2}{B} + \left(\frac{B - 1}{B}\right)^2(B + 1)c\sigma^2
$$

Letting the derivative of the above Equation to 0, we get $B = 0.6$ (X) and $B = 2.2$. 

\[\hat{f} = \sqrt{(B + 1)\sigma^2}\]
Extension to Multi-dimensional Settings

2-dim scenario

- Generating different granularity 2-dim grids to decompose the entire domain
- $\theta$ and $B$ settings similar to 1-dim scenes: the derivation without dimension restriction
- Four steps: User Partition (UP), Noisy Frequency Construction (NFC), New Decomposition Generation (NDG), Post-processing (PP)
Extension to Multi-dimensional Settings

High-dimensional scenario

- Direct Estimation (DE)
  - Treating the $m$-dim domain as a $m$-dim cube (direct extension of 2-dim)

- Leveraging Low-dimensional Estimation (LLE)
  - Step1: Building Block Construction
    Estimating the frequency distributions for the 2-dim attribute pairs separately
  - Step2: Consistency on Attributes
    Achieving consistency on all $m$ attributes among the related 2-dim attribute pairs
  - Step3: Maximum Entropy Optimization
    Estimating the frequency of the $m$-dim query with partial information from 2-dim queries
4. Experiment
Experiment

Dataset

- 3 real-world datasets and 5 synthetic datasets

| Dataset     | Distribution | Scale    | Field              | Type    |
|-------------|--------------|----------|--------------------|---------|
| Salaries    | -            | 148,654  | employee salary    | real    |
| BlackFriday | -            | 537,577  | shopping           | real    |
| Loan        | -            | 2,260,668| online loan        | real    |
| Financial   | -            | 6,362,620| fraud detection    | synthetic |
| Cauchy      | Cauchy       | -        | -                  | synthetic|
| Zipf        | Zipf (power-law) | -  | -                  | synthetic|
| Gaussian    | Gaussian     | -        | -                  | synthetic|
| Laplacian   | Laplacian    | -        | -                  | synthetic|

Baseline Algorithms

- 1-dim: HIO, DHT, CALM, Uni (obtaining the query answer from a uniform distribution)
- ≥2-dim: CALM, HDG

Metrics

- MSE (mean square error)
- 95% confidence interval
Performance Evaluation

Practical Deployment
Evaluation for 1-dim Range Query

- AHEAD_B2: branch $B = 2$
- AHEAD_B4: branch $B = 4$

Remarks

- **Effective**: the MSE of AHEAD is smaller than its counterparts throughout the experiment datasets.
- **Reasonable parameter setting**: the branch $B = 2$ obtains smaller MSEs compared to $B = 4$. 
Evaluation for 2-dim Range Query

- AHEAD_B2: branch $B = 2^2$
- AHEAD_B4: branch $B = 4^2$

(a) 2-dim Laplacian, $|D| = 256^2$, varv $\epsilon$  
(b) 2-dim Laplacian, $|D| = 1024^2$, varv $\epsilon$  
(c) 2-dim Laplacian, $|D| = 256^2$, vary $r$  
(d) 2-dim Laplacian, $|D| = 1024^2$, varv $r$
Evaluation for 2-dim Range Query

- AHEAD_B2: branch $B = 2^2$
- AHEAD_B4: branch $B = 4^2$

Remarks

- **Effective**: the MSE of AHEAD is smaller than its counterparts throughout the experiment datasets.
Performance Evaluation

Evaluation for 2-dim Range Query

- AHEAD_B2: branch $B = 2^2$
- AHEAD_B4: branch $B = 4^2$

Remarks

- **Effective**: the MSE of AHEAD is smaller than its counterparts throughout the experiment datasets.
- **Correlation robust**: the MSE of AHEAD almost does not change with different attribute correlations.
Performance Evaluation

Evaluation for high-dim Range Query

- DE_AHEAD_B2: branch $B = 2^m$, Direct Estimation
- LLE_AHEAD_B2: branch $B = 2^2$, Leveraging Low-dimensional Estimation

Remarks

- LLE vs. DE: AHEAD with LLE obtains lower MSEs than DE.
Evaluation for high-dim Range Query

- **DE_AHEAD_B2**: branch $B = 2^m$, Direct Estimation
- **LLE_AHEAD_B2**: branch $B = 2^2$, Leveraging Low-dimensional Estimation

Remarks

- **LLE vs. DE**: AHEAD with LLE obtains lower MSEs than DE.
- **Dimension robust**: the MSE of AHEAD is not sensitive to data dimension changes.
Performance Evaluation

Practical Deployment
### Practical Deployment

#### Impact of User Scale

- The MSE of AHEAD at increasing user scales (from $10^3$ to $10^7$)

| Scale | Privacy Budget | MSE |
|-------|----------------|-----|
| e+3   | 5  | -2.8549 | -2.7323 | -2.1946 | -1.3439 | -1.1061 | -1.0699 | -1.0078 | -0.9827 | -1.0225 |
| 5e+3  | 2  | -3.5059 | -3.0547 | -2.7126 | -2.2439 | -1.3995 | -1.0108 | -1.0380 | -0.9832 | -1.0966 |
| e+4   | 1  | -3.8405 | -3.3249 | -3.0563 | -2.6994 | -1.7333 | -1.1169 | -1.1102 | -1.0097 | -0.9892 |
| 5e+4  | 0.5 | -4.5462 | -3.8219 | -3.4670 | -3.0035 | -2.6383 | -2.0520 | -1.1286 | -1.0429 | -1.0574 |
| e+5   | 0.1 | -4.9132 | -4.1485 | -3.6929 | -3.2816 | -2.9292 | -2.2393 | -1.3518 | -1.1122 | -1.0105 |
| 5e+5  | 0.05| -5.3346 | -4.7405 | -4.2912 | -3.7869 | -3.4120 | -2.9585 | -2.4752 | -1.3787 | -1.1005 |
| e+6   | 0.025| -5.6321 | -4.9767 | -4.4729 | -4.0878 | -3.5091 | -3.0809 | -2.6335 | -1.5980 | -1.1361 |
| 5e+6  | 0.01 | -6.5165 | -5.6129 | -5.0670 | -4.6384 | -3.9769 | -3.6225 | -3.0862 | -2.4908 | -1.4455 |
| e+7   | 0.005| -6.7196 | -5.8534 | -5.3447 | -4.7592 | -4.2470 | -3.6105 | -3.3112 | -2.8546 | -2.2864 |

(a) AHEAD, 1-dim Zipf, $|D| = 256$

(b) AHEAD, 1-dim Cauchy, $|D| = 1024$

### Remarks

- **Necessity of proper user scale**: using an appropriate user scale to ensure algorithm performance.
Impact of User Scale

- The MSE of AHEAD at increasing user scales (from $10^3$ to $10^7$)

![Graph showing MSE vs user scale]

(a) AHEAD, 1-dim Zipf, $|D| = 256$

(b) AHEAD, 1-dim Cauchy, $|D| = 1024$

Remarks

- **Necessity of proper user scale**: using an appropriate user scale to ensure algorithm performance.

- **Exchangeability between scale and privacy budget**: a similar MSE when $N_2e_2^2 \approx N_2e_2^2$ is satisfied.
Impact of Domain Size

- The MSE of AHEAD at increasing domain size (from 32 to 4096)

\[
\begin{array}{cccccccc}
4096 & -4.7912 & -4.2906 & 3.8018 & 3.2829 & -2.7144 & -2.1784 & -1.0308 & -0.9586 \\
2048 & -4.6199 & -4.0956 & 3.7176 & 3.3642 & -2.8854 & -2.2808 & -1.1640 & -0.9257 \\
1024 & -4.7807 & -4.1403 & 3.8015 & 3.3106 & -2.8439 & -2.2002 & -1.1931 & -1.0182 \\
512 & -4.7666 & -4.2397 & 3.6438 & 3.2037 & -2.8629 & -2.0737 & -1.1535 & -1.0576 \\
256 & -4.7048 & -4.2147 & 3.6387 & 3.4247 & -2.7709 & -2.4497 & -1.4807 & -1.1037 \\
128 & -4.8525 & -4.1792 & 3.6994 & 3.3682 & -3.0151 & -2.4195 & -1.4778 & -1.1813 \\
64 & -4.8071 & -4.3057 & 3.7605 & 3.2468 & -2.6833 & -2.3614 & -1.6613 & -1.2401 \\
32 & -4.9424 & -4.2384 & 3.7810 & 3.1788 & -2.5742 & -2.4279 & -1.8308 & -1.2915 \\
\end{array}
\]

\[(a)\] AHEAD, 1-dim Zipf, \(N = 10^5\)

\[(b)\] AHEAD, 1-dim Cauchy, \(N = 10^5\)

Remark

- Domain size robust: AHEAD reacts robust to domain size changes.
**Impact of Domain Size**

- Ratio of the number of leaf nodes with the threshold to that without the threshold

| Origin | Loan (1-dim) | Financial (1-dim) | BlackFriday (1-dim) | Salaries (1-dim) | Laplacian $256^2$ (2-dim) | Laplacian $1024^2$ (2-dim) | Gaussian $256^2$ (2-dim) | Gaussian $1024^2$ (2-dim) |
|--------|--------------|------------------|---------------------|-----------------|---------------------|---------------------|---------------------|---------------------|
| $\epsilon = 0.1$ | 26 (10.16%) | 37 (7.23%) | 9 (0.88%) | 10 (0.49%) | 70 (1.07%) | 70 (0.07%) | 73 (1.11%) | 67 (0.06%) |
| $\epsilon = 0.3$ | 65 (25.39%) | 94 (18.36%) | 30 (2.92%) | 21 (1.03%) | 214 (3.27%) | 193 (0.18%) | 205 (3.13%) | 205 (0.20%) |
| $\epsilon = 0.5$ | 99 (38.67%) | 148 (28.91%) | 67 (6.54%) | 29 (1.42%) | 370 (5.65%) | 325 (0.31%) | 361 (5.51%) | 298 (0.28%) |
| $\epsilon = 0.7$ | 115 (44.92%) | 191 (37.30%) | 85 (8.30%) | 38 (1.86%) | 472 (7.20%) | 433 (0.41%) | 448 (6.84%) | 424 (0.40%) |
| $\epsilon = 0.9$ | 130 (50.78%) | 231 (45.12%) | 102 (9.96%) | 49 (2.39%) | 664 (10.13%) | 556 (0.53%) | 619 (9.45%) | 562 (0.54%) |
| $\epsilon = 1.1$ | 142 (55.47%) | 267 (52.15%) | 137 (13.38%) | 68 (3.32%) | 760 (11.60%) | 682 (0.65%) | 799 (12.19%) | 712 (0.68%) |
| $\epsilon = 1.3$ | 147 (57.42%) | 294 (57.42%) | 152 (14.84%) | 71 (3.47%) | 943 (14.39%) | 889 (0.85%) | 958 (14.62%) | 823 (0.78%) |
| $\epsilon = 1.5$ | 153 (59.77%) | 328 (64.06%) | 168 (16.41%) | 88 (4.30%) | 1150 (17.55%) | 1060 (1.01%) | 1147 (17.50%) | 1027 (0.98%) |

**Remarks**

- Effectively merging sparse domain: suppressing the excessive injected noises.
Impact of Domain Size

- Ratio of the number of leaf nodes with the threshold to that without the threshold

| Origin | Loan (1-dim) | Financial (1-dim) | BlackFriday (1-dim) | Salaries (1-dim) | Laplacian $256^2$ (2-dim) | Laplacian $1024^2$ (2-dim) | Gaussian $256^2$ (2-dim) | Gaussian $1024^2$ (2-dim) |
|--------|--------------|-------------------|---------------------|-----------------|--------------------------|---------------------------|------------------------|--------------------------|
| $\epsilon = 0.1$ | 26 (10.16%) | 37 (7.23%) | 9 (0.88%) | 10 (0.49%) | 70 (1.07%) | 70 (0.07%) | 73 (1.11%) | 67 (0.06%) |
| $\epsilon = 0.3$ | 65 (25.39%) | 94 (18.36%) | 30 (2.92%) | 21 (1.03%) | 214 (3.27%) | 193 (0.18%) | 205 (3.13%) | 205 (0.20%) |
| $\epsilon = 0.5$ | 99 (38.67%) | 148 (28.91%) | 67 (6.54%) | 29 (1.42%) | 370 (5.65%) | 325 (0.31%) | 361 (5.51%) | 298 (0.28%) |
| $\epsilon = 0.7$ | 115 (44.92%) | 191 (37.30%) | 85 (8.30%) | 38 (1.86%) | 472 (7.20%) | 433 (0.41%) | 448 (6.84%) | 424 (0.40%) |
| $\epsilon = 0.9$ | 130 (50.78%) | 231 (45.12%) | 102 (9.96%) | 49 (2.39%) | 664 (10.13%) | 556 (0.53%) | 619 (9.45%) | 562 (0.54%) |
| $\epsilon = 1.1$ | 142 (55.47%) | 267 (52.15%) | 137 (13.38%) | 68 (3.32%) | 760 (11.60%) | 682 (0.65%) | 799 (12.19%) | 712 (0.68%) |
| $\epsilon = 1.3$ | 147 (57.42%) | 294 (57.42%) | 152 (14.84%) | 71 (3.47%) | 943 (14.39%) | 889 (0.85%) | 958 (14.62%) | 823 (0.78%) |
| $\epsilon = 1.5$ | 153 (59.77%) | 328 (64.06%) | 168 (16.41%) | 88 (4.30%) | 1150 (17.55%) | 1060 (1.01%) | 1147 (17.50%) | 1027 (0.98%) |

Remarks

- Effectively merging sparse domain: suppressing the excessive injected noises.
- More merging intervals with domain size increasing.
Impact of Data Skewness

- The MSE of AHEAD at increasing skewness

\[(a)\] AHEAD, 1-dim *Gaussian*, \(N = 10^5\)

\[(b)\] AHEAD, 1-dim *Laplacian*, \(N = 10^5\)

**Remarks**

- When \(\epsilon \leq 0.1\), the MSEs have a tendency to decrease with the increase of data skewness.
Impact of Data Skewness

- The MSE of AHEAD at increasing skewness

Remarks

- When $\epsilon \leq 0.1$, the MSEs have a tendency to decrease with the increase of data skewness.
- When $\epsilon > 0.1$, the impact of skewness becomes insignificant on MSE of AHEAD.
We proposed AHEAD, a novel LDP protocol for range query problem

- **Effective:** It outperforms state-of-the-art methods in terms of query accuracy under both real-world and synthetic datasets.
- **Privacy:** It satisfies rigorous LDP guarantees.
- **Adaptable:** It performs well for both low-dimensional queries and high($\geq 2$)-dimensional queries.

We evaluated AHEAD on 3 real-world and 5 synthetic datasets

- **Real-world Datasets:** Salaries, BlackFriday and Loan.
- **Synthetic datasets:** Cauchy, Zipf, Gaussian, Laplacian and Financial.

We systematically analyzed AHEAD from 5 aspects and summarized 6 practical suggestions

- User Scale, Domain size, Data Skewness, Data dimension, Attribute correlation
