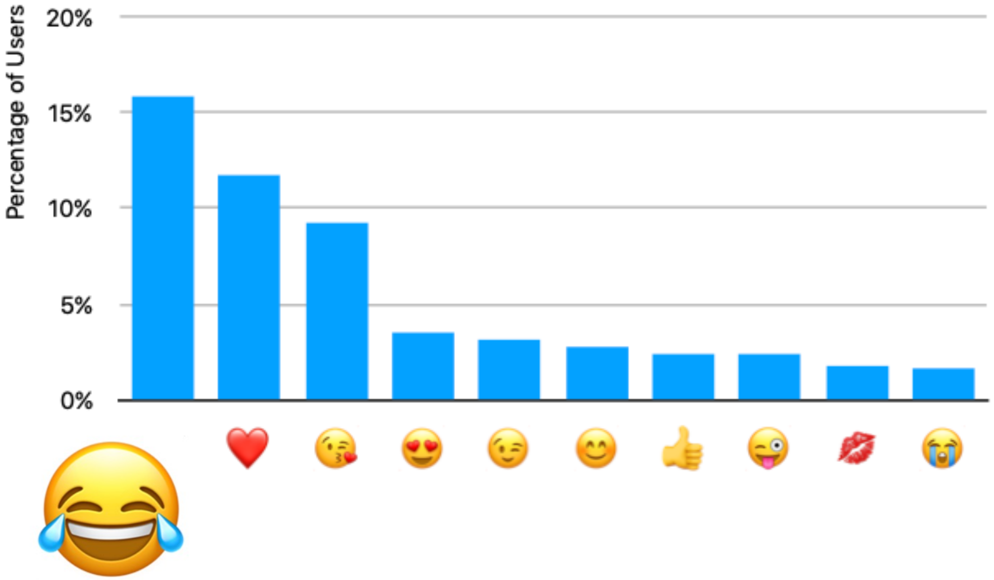
A Poisoning-Resilient LDP schema leveraging Oblivious Transfer with the Hadamard Transform

Masahiro Shimizu and Hiroaki Kikuchi

Meiji University

Backgrounds

- *What is the best Emoji used in France?*



LDP: Count Mean Sketch (CMS)[Apple 2017]

- Client side ensures that data d in D is ε-differentially private.
- Server estimates frequencies over D from sketch matrix k x m.

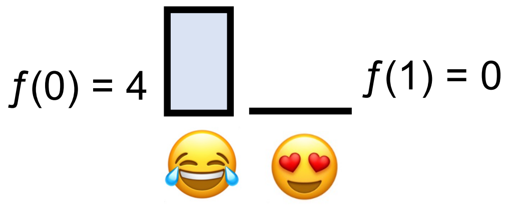
d = 0, h(0) = 2	d = 1, h(1) = 0
$\mathbf{v} = (-1, -1, 1, -1)$	$\mathbf{v} = (1, -1, -1, -1)$
$\mathbf{w} = (-1, -1, 1, 1)$	$\mathbf{w} = (-1, -1, 1, -1)$

$$\tilde{v_i} = \begin{cases} v_i & w./p.\ p = \frac{e^{\epsilon/2}}{1+e^{\varepsilon/2}}, \\ -v_i & w./p.\ q = \frac{1}{1+e^{\varepsilon/2}}. \end{cases}$$

😂	$d_1 = 0$	$\mathbf{v_1} = (-1, -1, 1, -1)$	$\mathbf{w}_1 = (-1, -1, -1, -1)$
😂	$d_2 = 0$	$\mathbf{v_2} = (-1, -1, 1, -1)$	$\mathbf{w}_2 = (-1, -1, 1, -1)$
😂	$d_3 = 0$	$\mathbf{v_3} = (-1, -1, 1, -1)$	$\mathbf{w}_3 = (-1, -1, 1, -1)$
😍	$d_4 = 1$	$\mathbf{v_4} = (1, -1, -1, -1)$	$\mathbf{w}_4 = (-1, -1, -1, -1)$

$\mathbf{M} = (-4, -4, 0, -4)$

$$\tilde{f}(d) = \left(\frac{m}{m-1}\right)\left(\frac{1}{k}\sum_{\ell=1}^{k} M_{\ell,h_\ell(d)} - \frac{n}{m}\right)$$



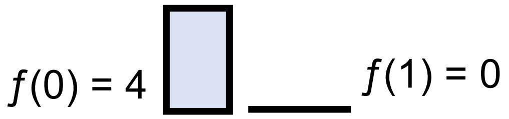
Poisoning attack [Cao 2021]

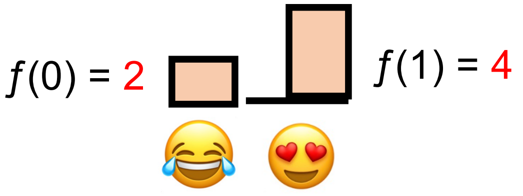
- A set of malicious users manipulate the estimated statistics by casting fake data.

😂	$d_1 = 0$	$\mathbf{v_1} = (-1, -1, 1, -1)$	$\mathbf{w}_1 = (-1, -1, -1, -1)$
😂	$d_2 = 0$	$\mathbf{v_2} = (-1, -1, 1, -1)$	$\mathbf{w}_2 = (-1, -1, 1, -1)$
😂	$d_3 = 0$	$\mathbf{v_3} = (-1, -1, 1, -1)$	$\mathbf{w}_3 = (-1, -1, 1, -1)$
😍	$d_4 = 1$	$\mathbf{v_4} = (1, -1, -1, -1)$	$\mathbf{w}_4 = (-1, -1, -1, -1)$
😍	$d_5 = 0$	$\mathbf{v_5} = (-1, -1, 1, -1)$	$\mathbf{w}_5 = (1, -1, -1, -1)$
😍	$d_6 = 0$	$\mathbf{v_6} = (-1, -1, 1, -1)$	$\mathbf{w}_5 = (1, -1, -1, -1)$

malicious

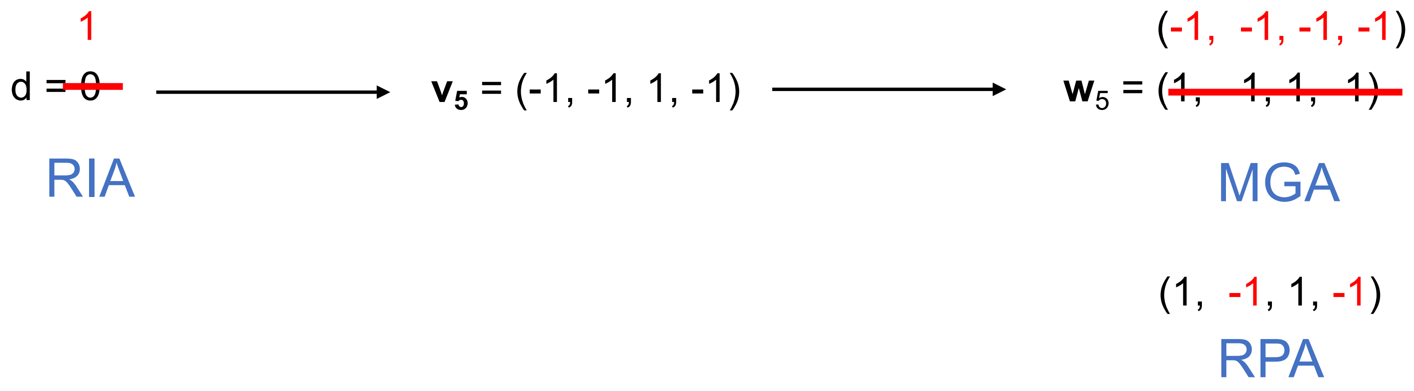
$\mathbf{M} = (-2, -6, -2, -6)$





Threat Model

- MGA (Maximum Gain Attack)
- RIA (Random Item Attack)
- RPA (Random Perturbed-value Attack)



1

d = ~~0~~

RIA

$\mathbf{v}_5 = (-1, -1, 1, -1)$

(-1, -1, -1, -1)

$\mathbf{w}_5 =$ ~~(1, 1, 1, 1)~~

MGA

(1, -1, 1, -1)

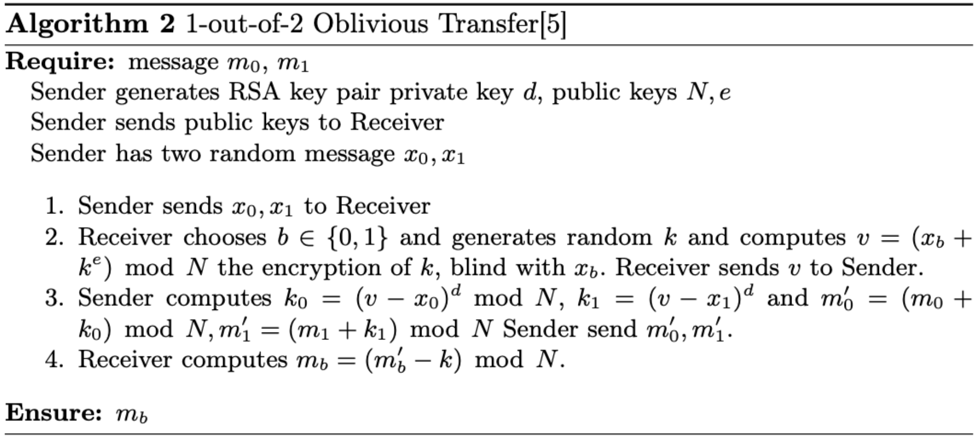
RPA

Related Works

- Countermeasures
  - Clustering [Cao 2021]
  - Outlier detection [Wu 2022]
  - Sampling and clustering [Li 2022]
  - ZKIP Verifiable LDP [Kato 2021]
  - **Oblivious Transfer** [Horigome 2023]

Oblivious Transfer

- Goal: A sender (client) sends one of some values to a receiver (server) but remains oblivious as to which has been sent.

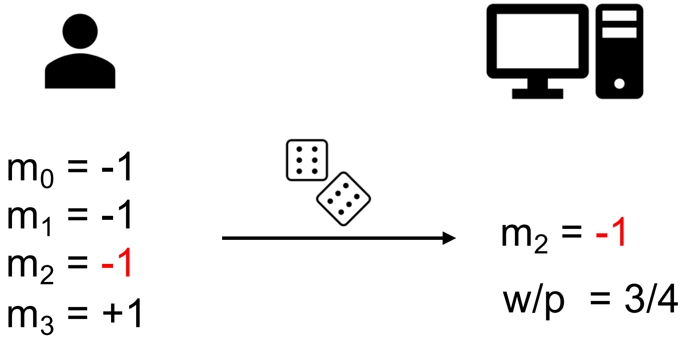


Algorithm 2 1-out-of-2 Oblivious Transfer[5]

```
Require: message $m_0$, $m_1$
  Sender generates RSA key pair private key $d$, public keys $N, e$
  Sender sends public keys to Receiver
  Sender has two random message $x_0, x_1$

   1. Sender sends $x_0, x_1$ to Receiver
   2. Receiver chooses $b \in \{0, 1\}$ and generates random $k$ and computes $v = (x_b + k^e) \mod N$ the encryption of $k$, blind with $x_b$. Receiver sends $v$ to Sender.
   3. Sender computes $k_0 = (v - x_0)^d \mod N$, $k_1 = (v - x_1)^d$ and $m_0' = (m_0 + k_0) \mod N, m_1' = (m_1 + k_1) \mod N$ Sender send $m_0', m_1'$.
   4. Receiver computes $m_b = (m_b' - k) \mod N$.

Ensure: $m_b$
```



m_0 = -1

m_1 = -1

m_2 = -1

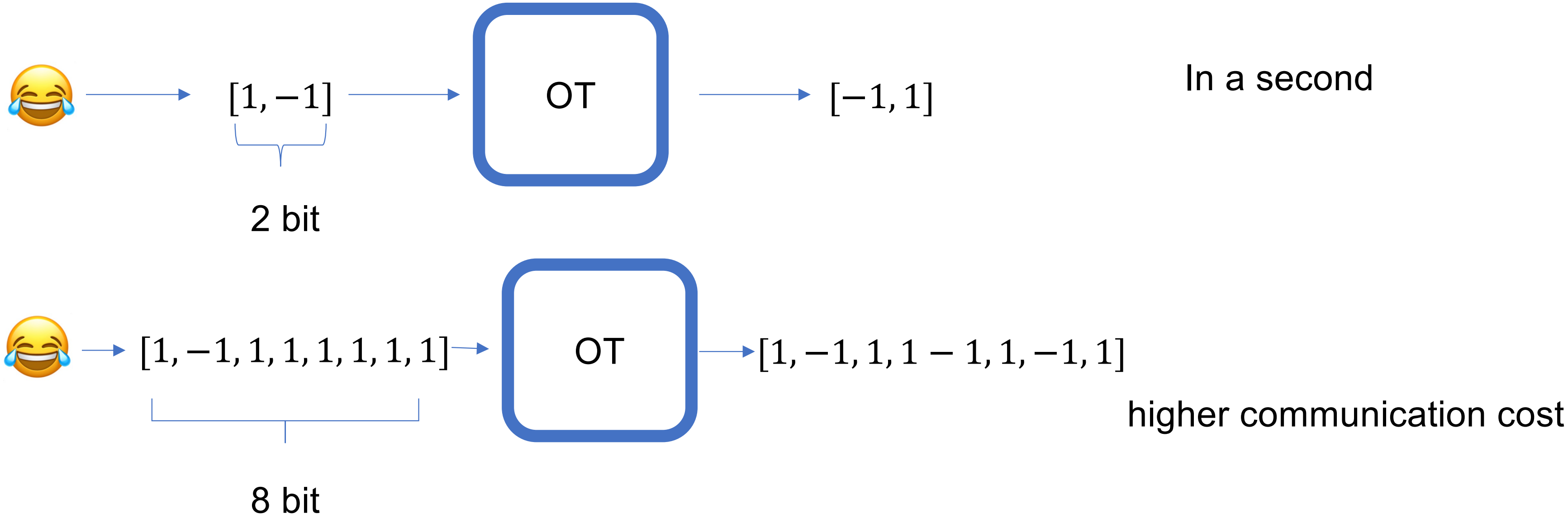
m_3 = +1

m_2 = -1

w/p = 3/4

Drawback of OT

- High communication cost. The vector size increases with domain size.



😂 → $\underbrace{[1, -1]}_{\text{2 bit}}$ → OT → $[-1, 1]$

In a second

😂 → $\underbrace{[1, -1, 1, 1, 1, 1, 1, 1]}_{\text{8 bit}}$ → OT → $[1, -1, 1, 1 - 1, 1, -1, 1]$

higher communication cost

Hadamard matrix

- **Hadamard basis** transform can be used to spread information form a sparse vector.

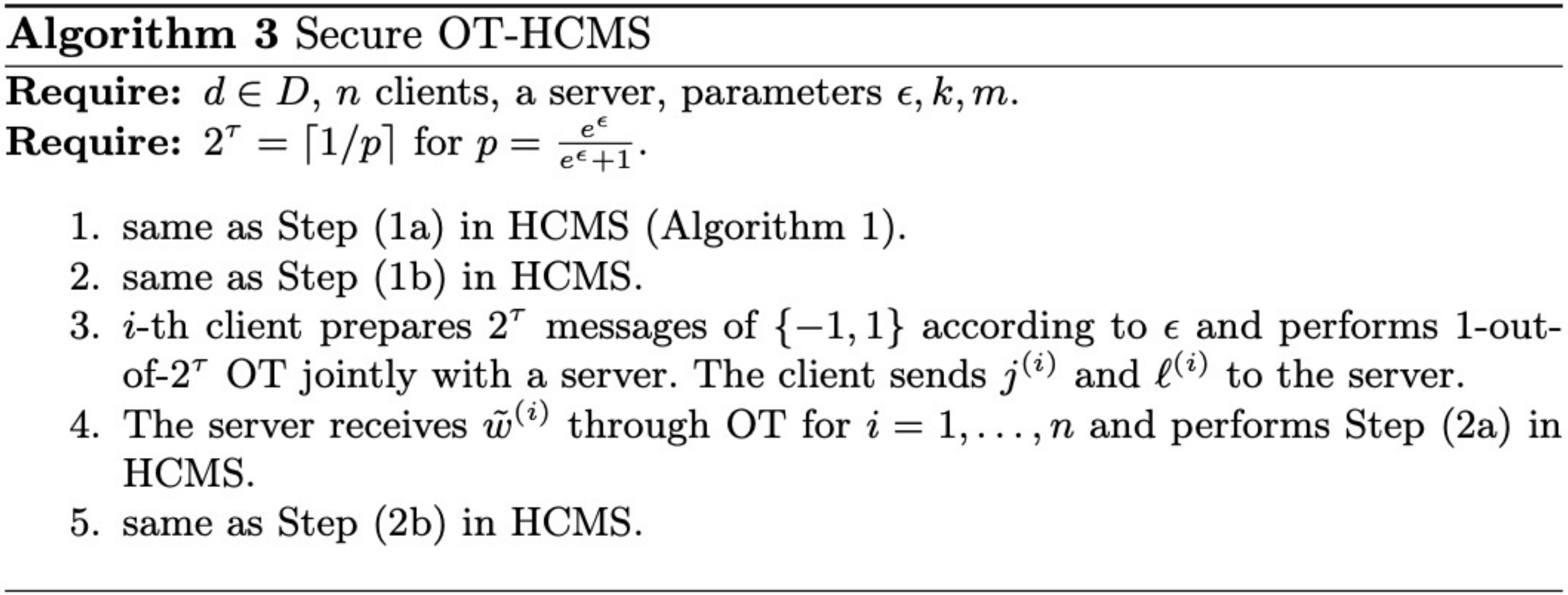
$$\boldsymbol{w} = Hm\boldsymbol{v} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} [0, 1, 0, 0]^T$$

$$= [1, -1, 1, -1]^T$$

  - After sampling uniformly from **w**, every client has one bit to be perturbed in OT protocols.

Proposed Protocol

- We combines Hadamard basis with the CMS with OT.



Algorithm 3 Secure OT-HCMS

```
Require: $d \in D$, $n$ clients, a server, parameters $\epsilon, k, m$.
Require: $2^\tau = \lceil 1/p \rceil$ for $p = \frac{e^\epsilon}{e^\epsilon+1}$.

    1. same as Step (1a) in HCMS (Algorithm 1).
    2. same as Step (1b) in HCMS.
    3. $i$-th client prepares $2^\tau$ messages of $\{-1, 1\}$ according to $\epsilon$ and performs 1-out-of-$2^\tau$ OT jointly with a server. The client sends $j^{(i)}$ and $\ell^{(i)}$ to the server.
    4. The server receives $\tilde{w}^{(i)}$ through OT for $i = 1, \ldots, n$ and performs Step (2a) in HCMS.
    5. same as Step (2b) in HCMS.
```

Research Questions

- Q1. Is our proposed OT-based LDP robust against poisoning attack?
- Q2. How much estimation accuracy is reduced with Hamdard Transform in HCMS?
- Q3. Which is more vulnerable against poisoning attack, CMS or Hamdard CMS?
- Q4. How much time does it take for poisoning countermeasures in OT-CMS and OT-HCMS?

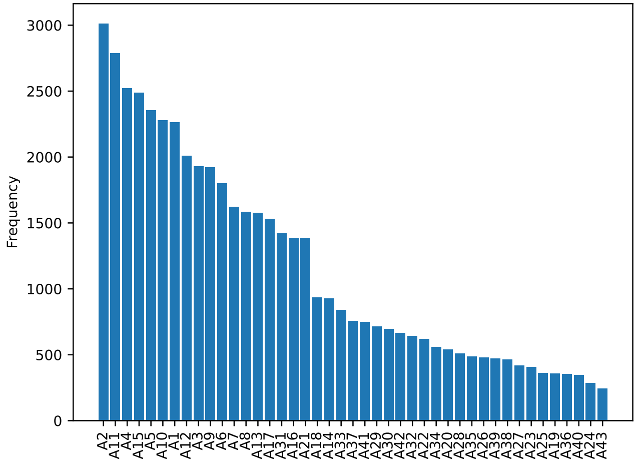
Experiments

- Evaluation metric for estimation accuracy：MSE
- Evaluation metric for safety：Frequency Gain

$$\boldsymbol{FG = \sum_{t \in T} E[\tilde{f}_t - \hat{f}_t]}$$

T: Set of target items
$\tilde{f}_t$ ：Estimated value of item t after poisoning
$\hat{f}_t$ ：Estimated value of item t before poisoning

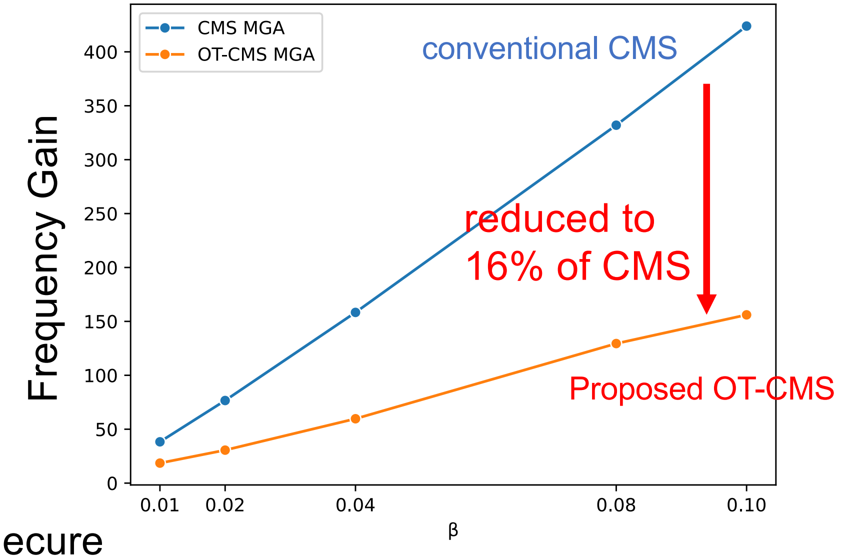
Purchase frequency data of online shopping



Record number：49742 Number of items：43

Result 1. Security of proposed schemes

Vulnerable



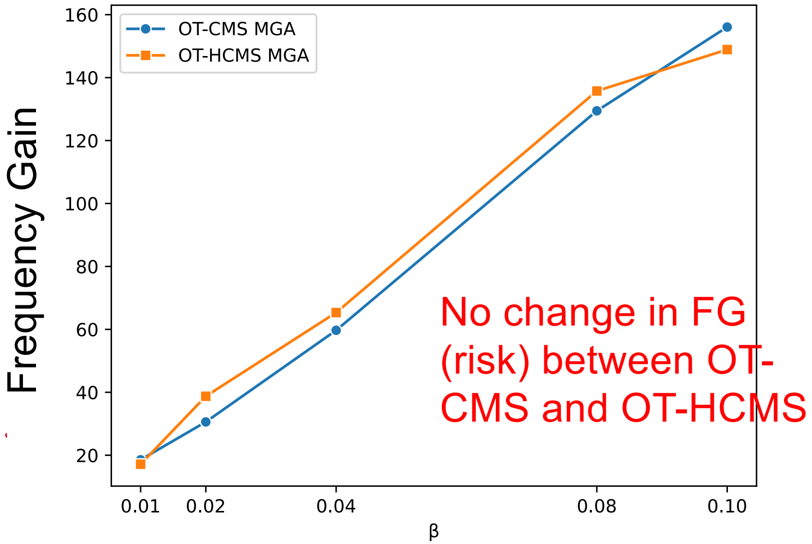
conventional CMS

reduced to 16% of CMS

Proposed OT-CMS

Secure

proportion of malicious

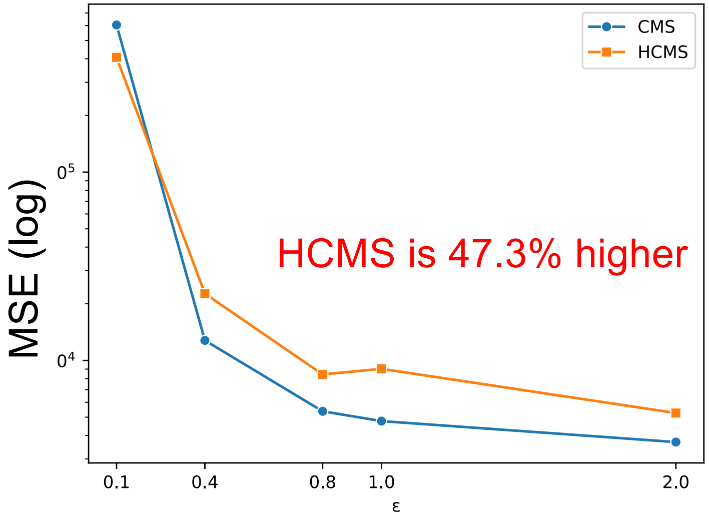


No change in FG (risk) between OT-CMS and OT-HCMS

proportion of malicious

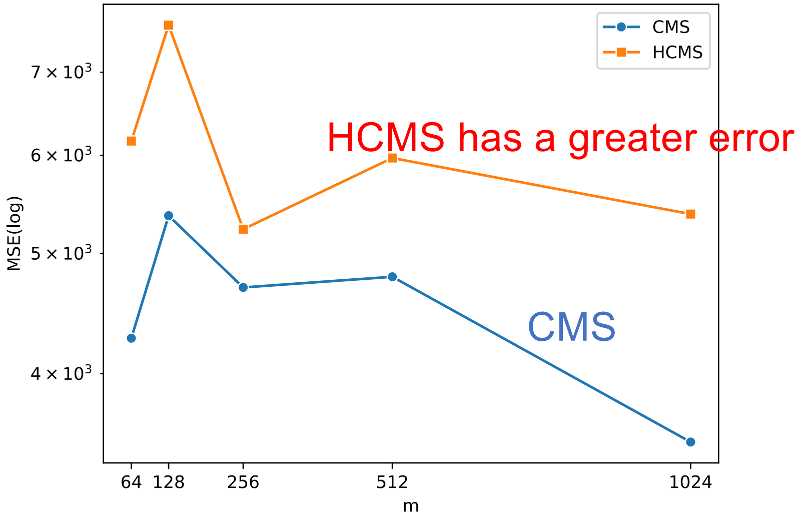
Result 2. Accuracy of CMS vs HCMS

Error



HCMS is 47.3% higher

privacy budget



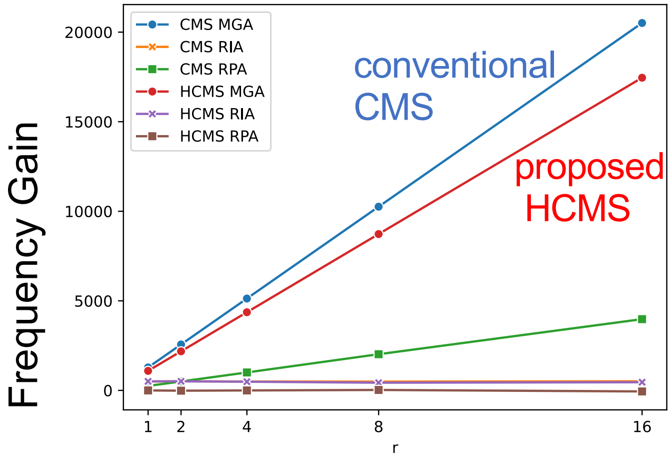
HCMS has a greater error

CMS

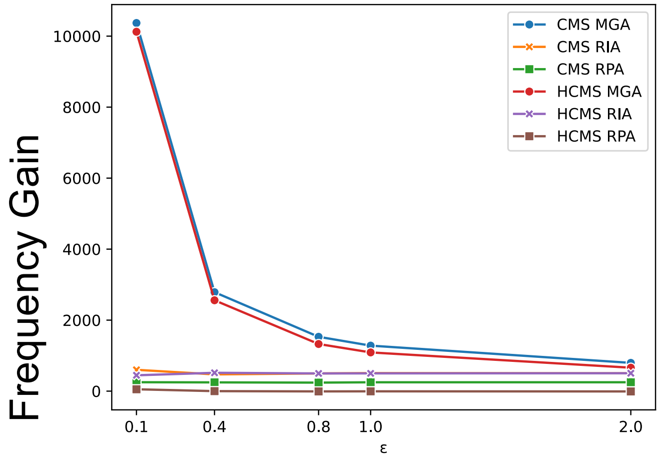
vector length

Result 3. Frequency Gains

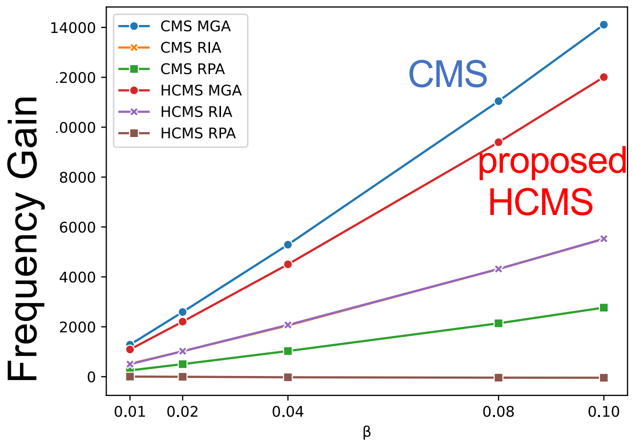
Vulnerable



The number of target items



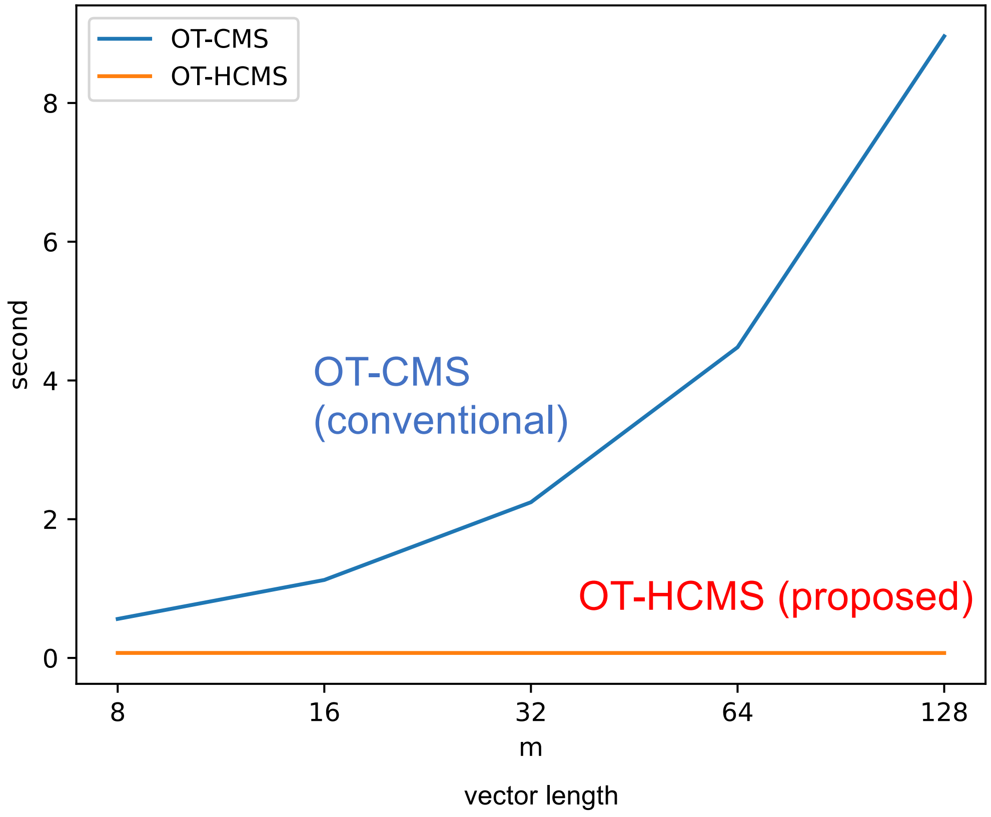
privacy budget



proportion of malicious users

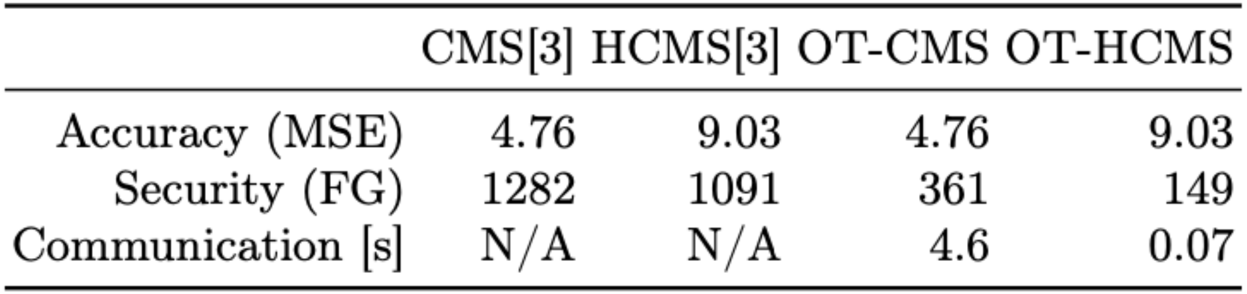
On average, CMS is 16% more vulnerable to MGA！！

Result 4. Effect of Hamdard Transfer



Limitations

- A local differential privacy scheme is a model that does not trust the data collector, but the proposed scheme requires collaboration of the server. It may sound contractional.



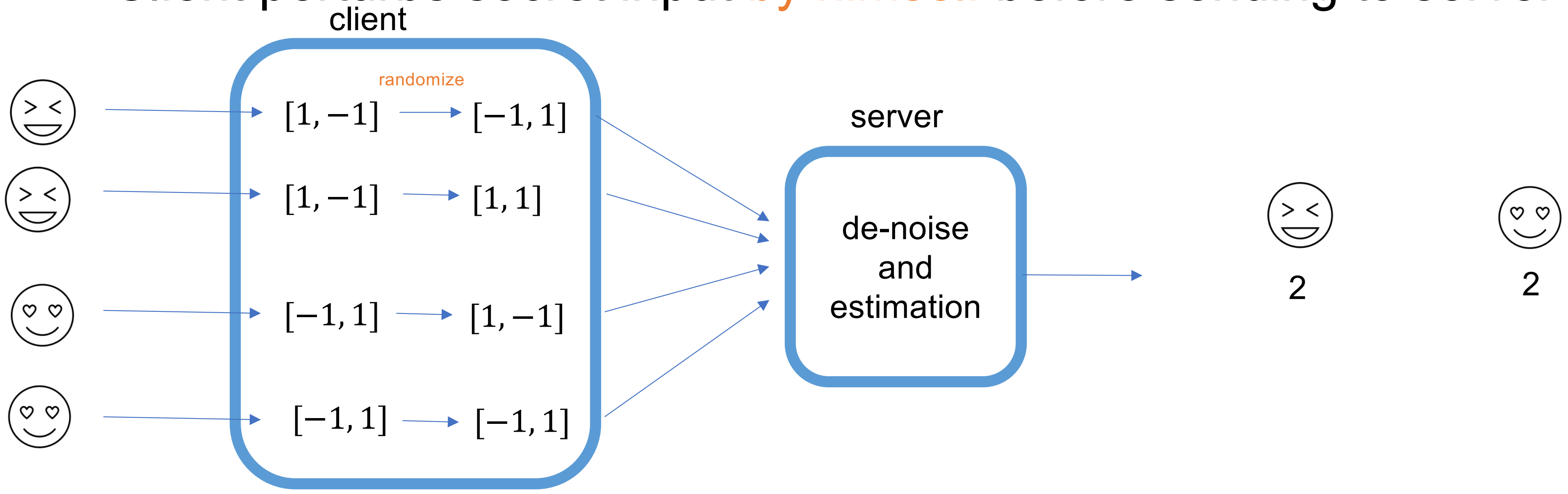
	CMS[3]	HCMS[3]	OT-CMS	OT-HCMS
Accuracy (MSE)	4.76	9.03	4.76	9.03
Security (FG)	1282	1091	361	149
Communication [s]	N/A	N/A	4.6	0.07

Conclusions

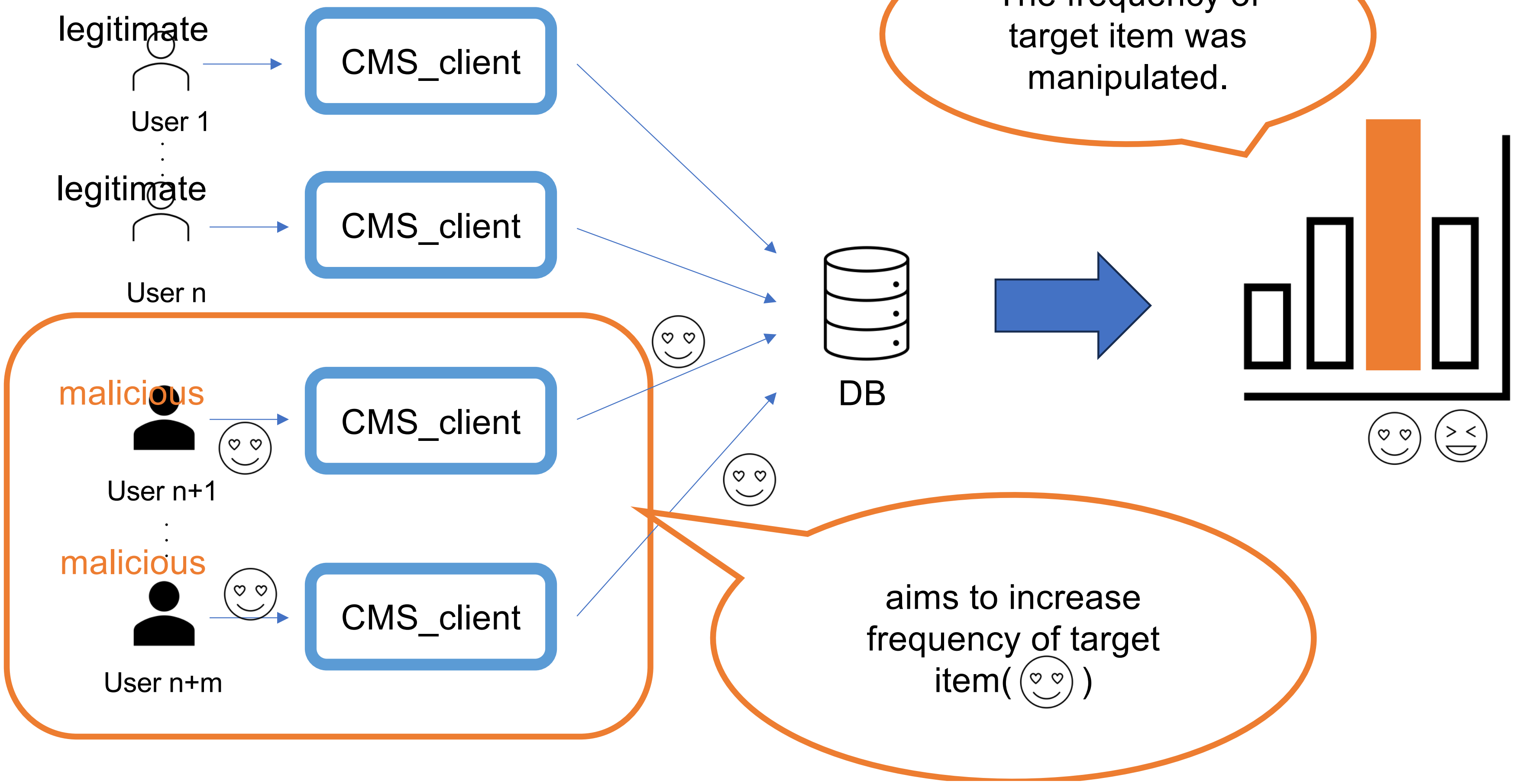
- Our study has demonstrated that the conventional LDP protocol CMS is vulnerable to poisoning attacks and we have proposed a new robust OT-CMS using Oblivious Transfer.
- We have also revised OT-HCMS, where the Hadamard matrix is used to reduce communication costs.
- Our experiment showed that the proposed schemes are effective against MGA
- We plan to address the contradiction the original concept of LDP as future study.

LDP Count Mean Sketch (CMS)[Apple 2017]

- No trust of server (true choice was hidden)
- Client perturbs secret input by himself before sending to server



Poisoning attack [Cao 2021]



legitimate

User 1

CMS_client

legitimate

User n

CMS_client

malicious

User n+1

CMS_client

malicious

User n+m

CMS_client

DB

The frequency of target item was manipulated.

aims to increase frequency of target item(😍)