Local, Private, Efficient Protocols for Succinct Histograms

Raef Bassily　　Adam Smith

Pennsylvania State University

STOC 2015
Portland, OR
June 15, 2015

A conundrum

Fashion.com

Finance.com

.
.
.

WeirdStuff.com

How many users like Google.com?

Google server

How can the server compute aggregate statistics about users without storing user-specific information?

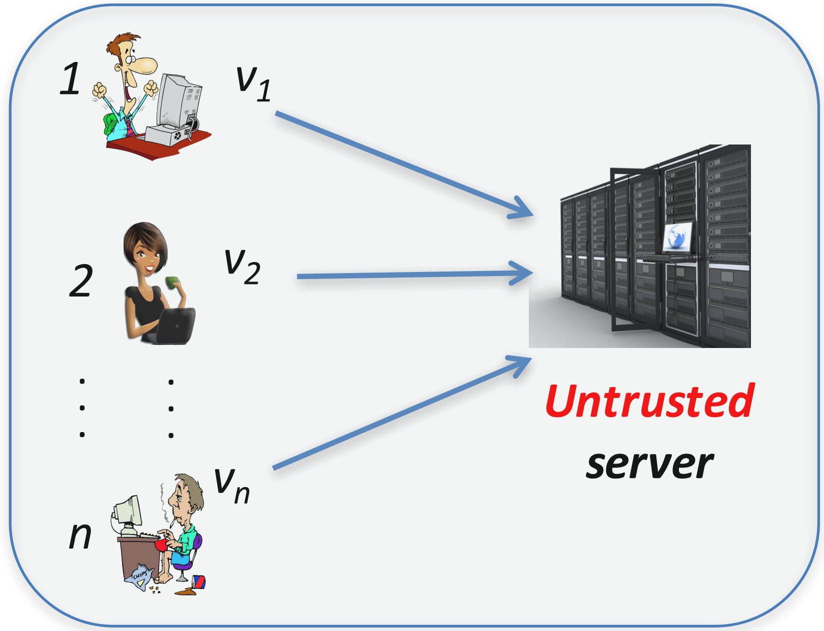
Succinct histograms

Set of users = [n].

A set of items (e.g. websites) = [d] = {1, …, d}.

Frequency of an item a is:

$$f(a) = (\#\ \text{users holding } a)/n$$

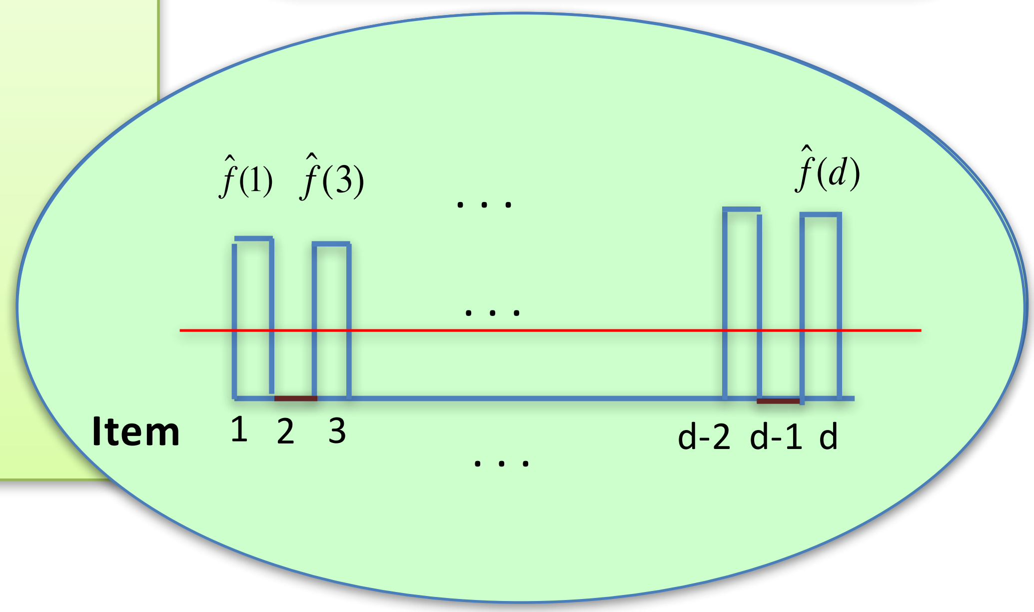


Succinct histogram:

- *subset* $\mathcal{S} \subseteq [d]$ *of items* (think "heavy hitters")
- *estimates of their frequencies*

$$\left\{\left(v, \hat{f}(v)\right) : v \in \mathcal{S}\right\}$$

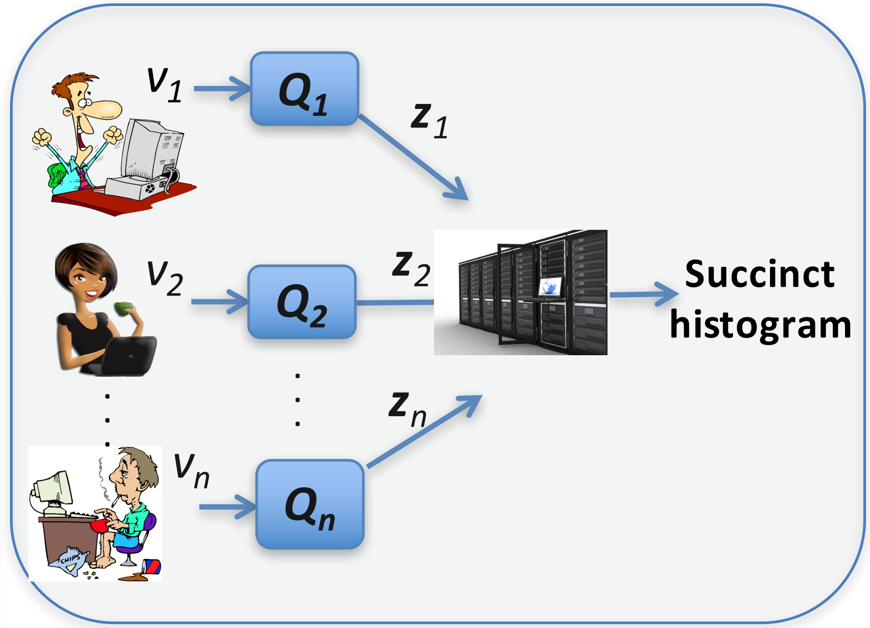
- *Implicitly,* $\hat{f}(v) = 0$ *for* $v \notin \mathcal{S}$



Local model of Differential Privacy

$v_i \in [d]$ is item of user $i \in [n]$

$\boldsymbol{z}_i$ is the *differentially-private report* of user i



<u>Definition</u>: Randomized algorithm ***Q*** is -local differentially private (LDP) *if*

for any pair *v, v'* [d],

for all events *S*, $\in$

$$\Pr[Q(v) \in S] \leq e^{\epsilon} \Pr[Q(v') \in S]$$

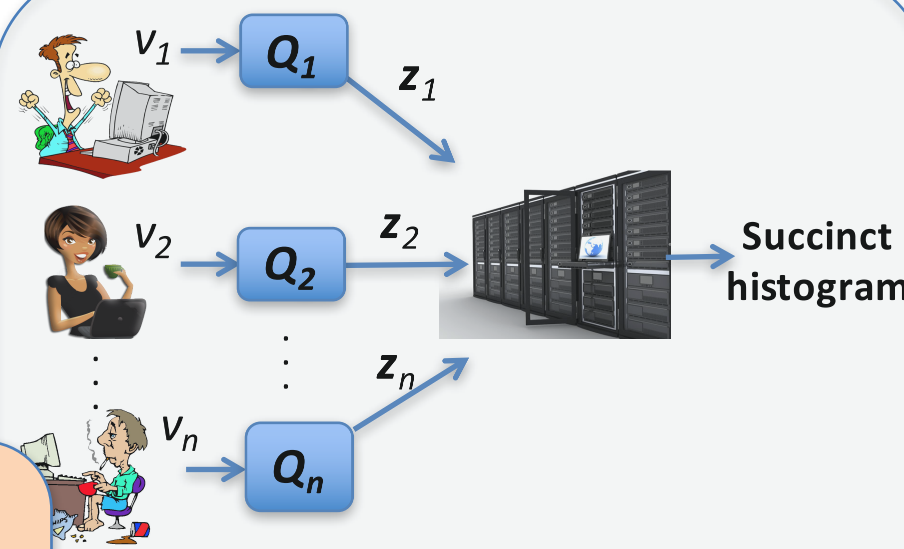
LDP for succinct histograms

- Studied under various names [Mishra-Sandler'06, Hsu et al.'12, Erlingsson et al.'14, Fanti et al.'15, Duchi et al.'13].
- Deployed in Google's Chrome (RAPPOR) [Erlingsson et al.'14].

System requirements

Privacy

A protocol that satisfies ϵ-LDP



$$v_i \in [d],\ i \in [n]$$

Accuracy

Small worst-case estimation error:

$$\max_{v_1,\ \ldots,\ v_n} \left\| \hat{\mathbf{f}} - \mathbf{f} \right\|_\infty = \max_{v_1,\ldots,v_n} \max_{j \in [d]} \left| \hat{f}(j) - f(j) \right|$$

with high probability over coins of $\boldsymbol{Q_i}$

Computational efficiency

A protocol is efficient if it runs in time *poly(log(d), n)*

~~*poly(d, n)*~~

***log*(d)** = # of bits to describe single item

Contributions [B, Smith '15]

1. Efficient ϵLDP protocol with optimal error:
   - run in time poly(log(d), n).
   - Estimate all frequencies up to error $O\left(\sqrt{\frac{\log(d)}{\epsilon^2 n}}\right)$
2. Matching lower bound on the error.
3. Efficient transformation reducing report length to 1 bit/user in public-coin model.

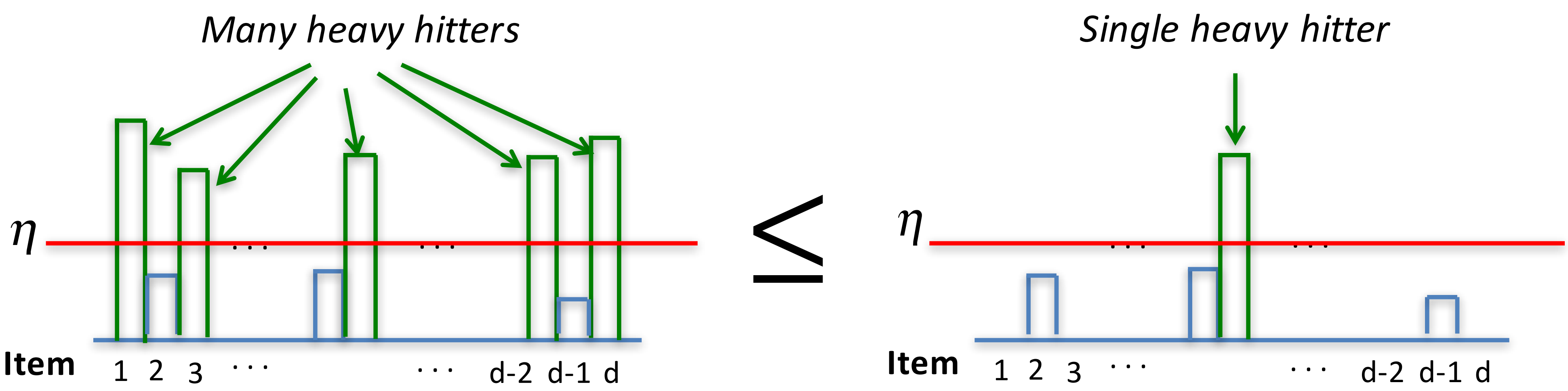
- Previous protocols either
  - ran in time $\Omega(d)$ [Mishra-Sandler'06, Hsu et al.'12, Erlingsson et al.'14]
  - or, had worse error $\sim\left(\frac{\log(d)}{\epsilon^2 n}\right)^{\frac{1}{6}}$ [Hsu et al.'12]

Exp. Time

Larger error

- Best previous lower bound was $\sim\frac{1}{\sqrt{n}}$

Construction approach



General case ***Simpler problem***

Single Heavy Hitter (SHH) problem: at least η fraction of users have the same item, say $v^* \in [d]$, while the rest have (i.e., “no item”)

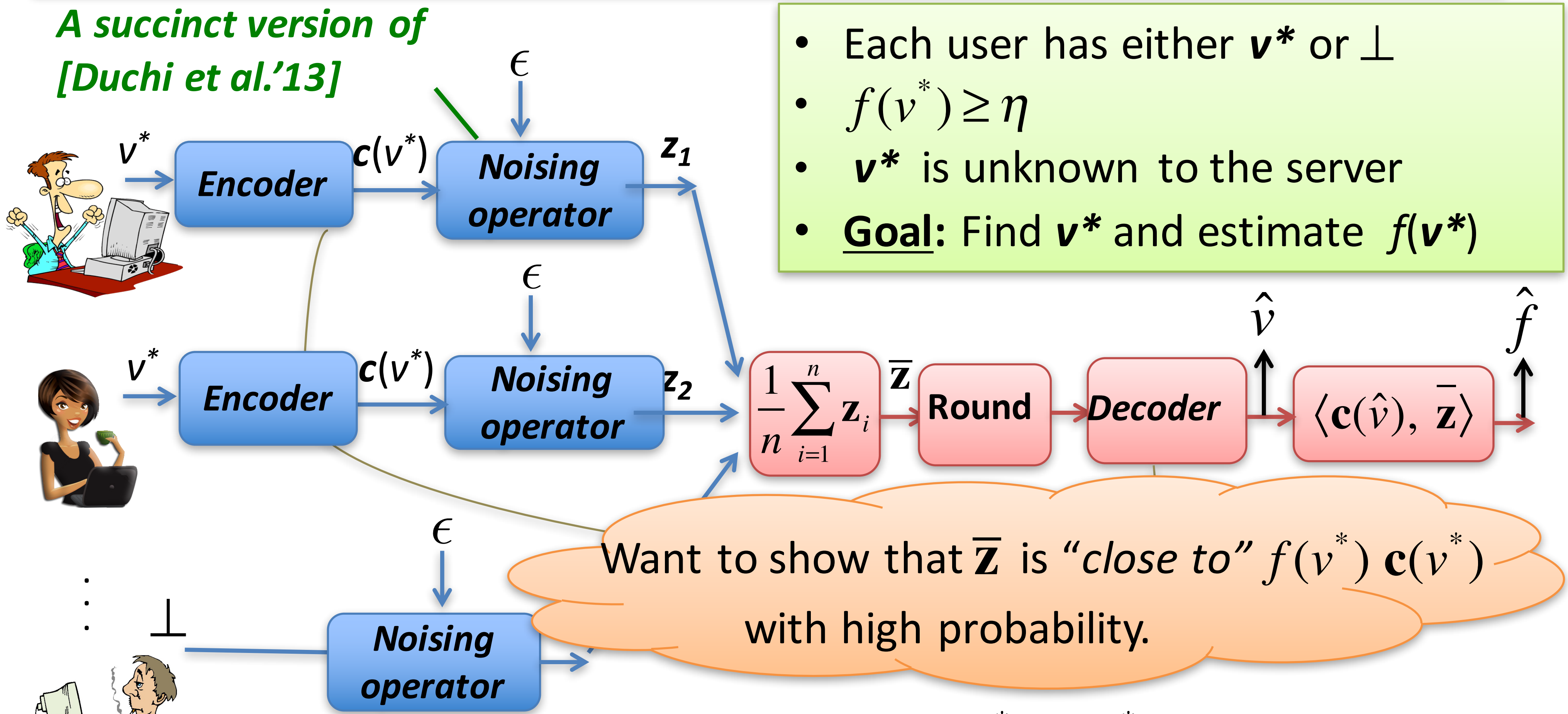
We give

- Efficient LDP algorithm for SHH with optimal accuracy
- Reduction from general case to SHH that **preserves privacy and accuracy**

Inspired by low-space algorithms, e.g. [Gilbert et al.’02].

Construction for the SHH problem

- Each user has either ***v**** or $\perp$
- $f(v^*) \geq \eta$
- ***v**** is unknown to the server
- **<u>Goal</u>:** Find ***v**** and estimate *f*(***v****)



Write $\bar{\mathbf{z}} = f(v^*)\,\mathbf{c}(v^*) + \mathbf{e}$

Bound $\langle \mathbf{e}, \mathbf{c}(v^*) \rangle$ and $\|\mathbf{e}\|_2$

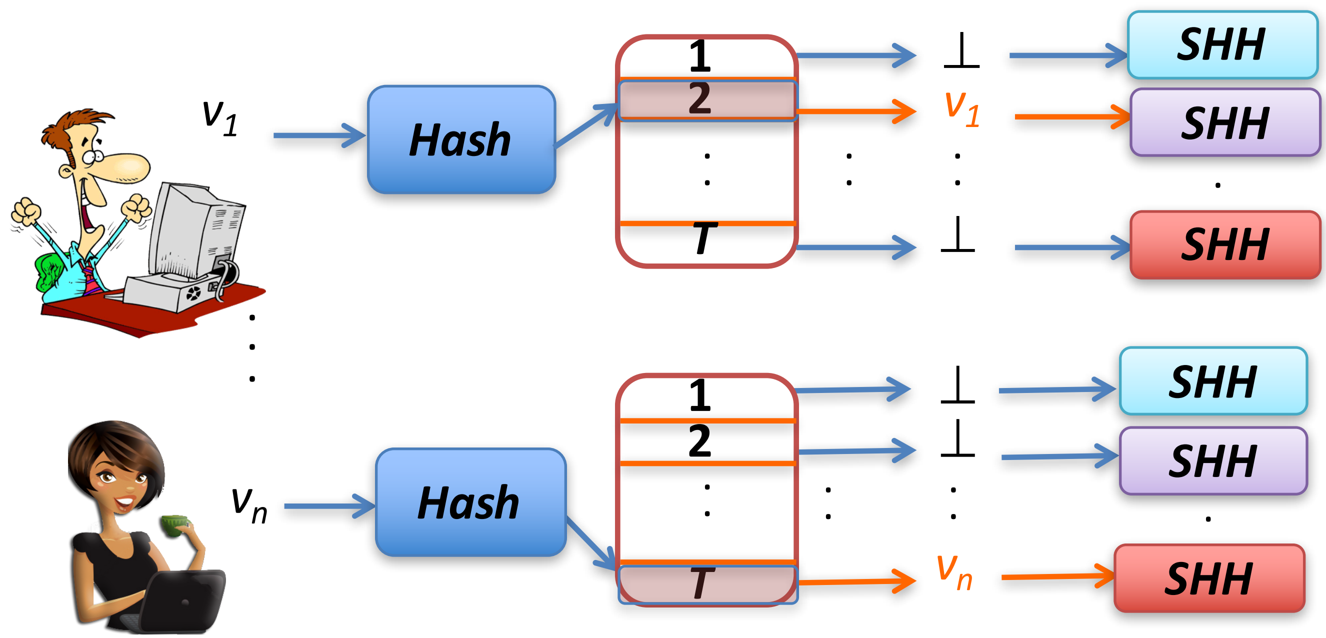
Key step: Show decoding succeeds (i.e., $\hat{v} = v^*$) w.h.p. when

$$\eta \geq \text{const} \times \sqrt{\frac{\log(d)}{\epsilon^2 n}}$$

Construction for the general setting

Key insight:

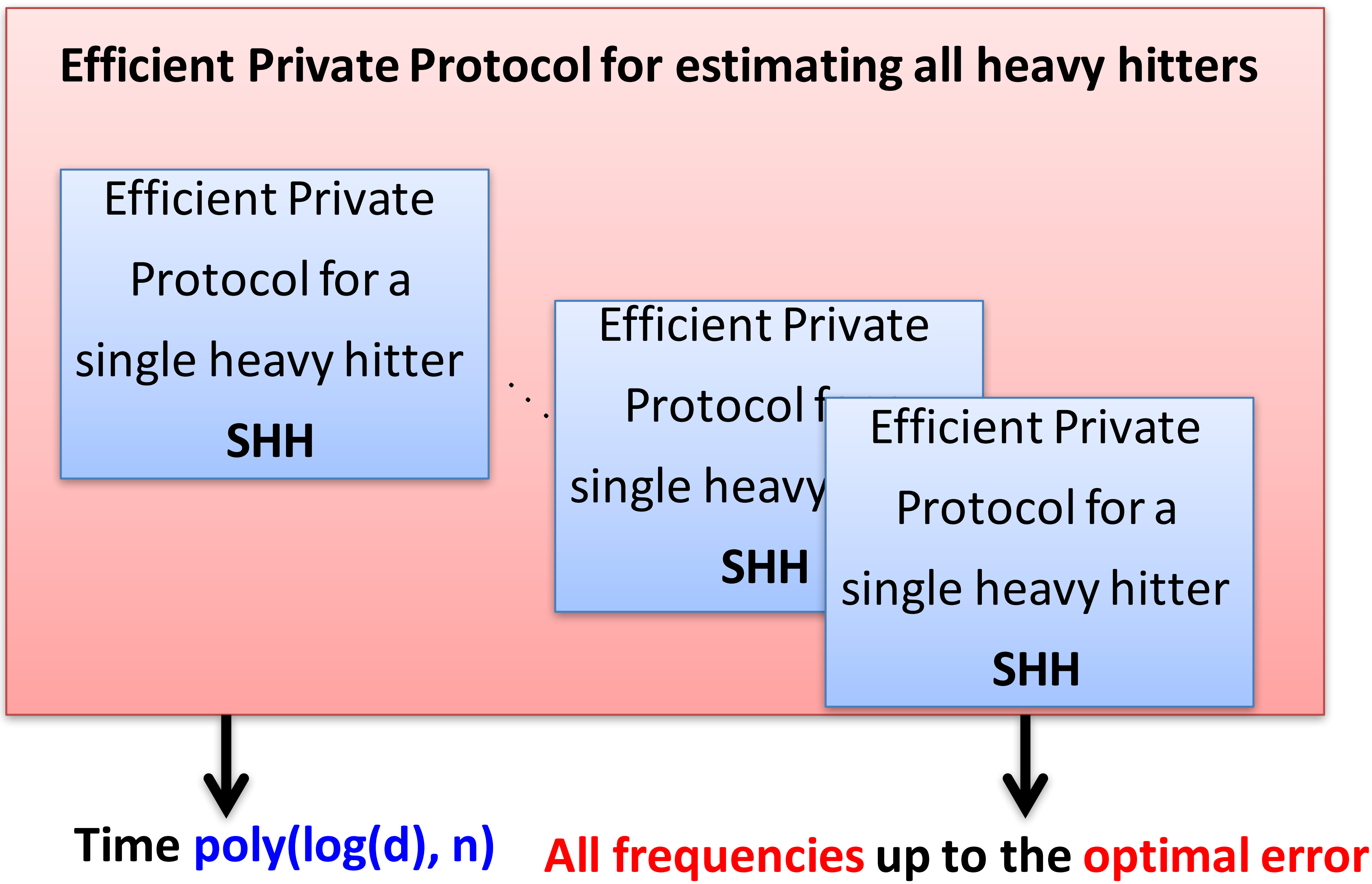
- Run multiple copies of the SHH protocol.
- Isolate *every* heavy hitter into a separate copy via hashing.



- W.h.p., every heavy hitter is alone in at least one SHH protocol.
- Same privacy co…
appears i…

item whose frequency $\geq \eta = \text{const} \cdot \sqrt{\frac{\log(d)}{\epsilon^2 n}}$

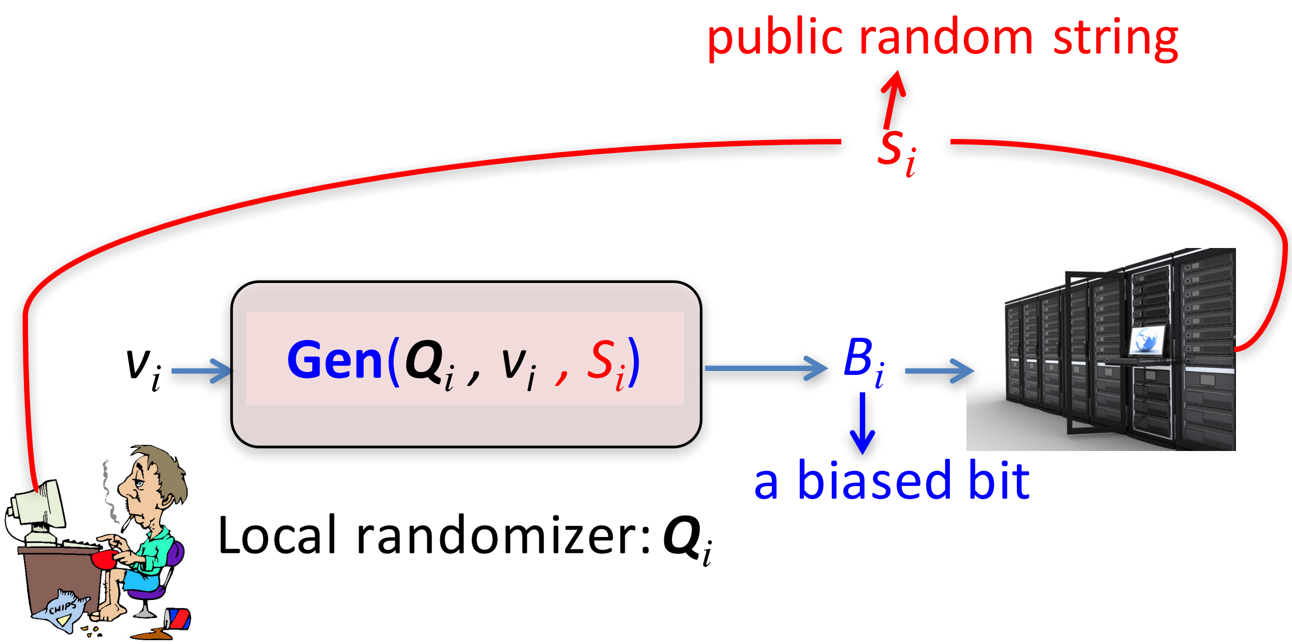
Recap: Construction of succinct histograms



Transforming to a protocol with 1-bit reports

Theorem: In a public coin model, *any ϵ-LDP protocol* can be transformed into another ϵ-LDP protocol with 1-bit reports.

- We modify a generic compression technique of [McGregor et al.'10].
- For our protocols, this transformation is
  - computationally efficient and
  - yields essentially same (optimal) error.



Conditioned on $B_i = 1$, S_i is distributed identically to $\boldsymbol{Q}_i(v_i)$

Conclusions

- Local, private protocols for succinct histograms that:
  - attain optimal worst-case error
  - are computationally efficient.
  - have low communication complexity
- More evidence of connections between differential privacy and low-space algorithms [Gilbert et al.'02, Dwork et al.'10, Blocki et al.'12,…]

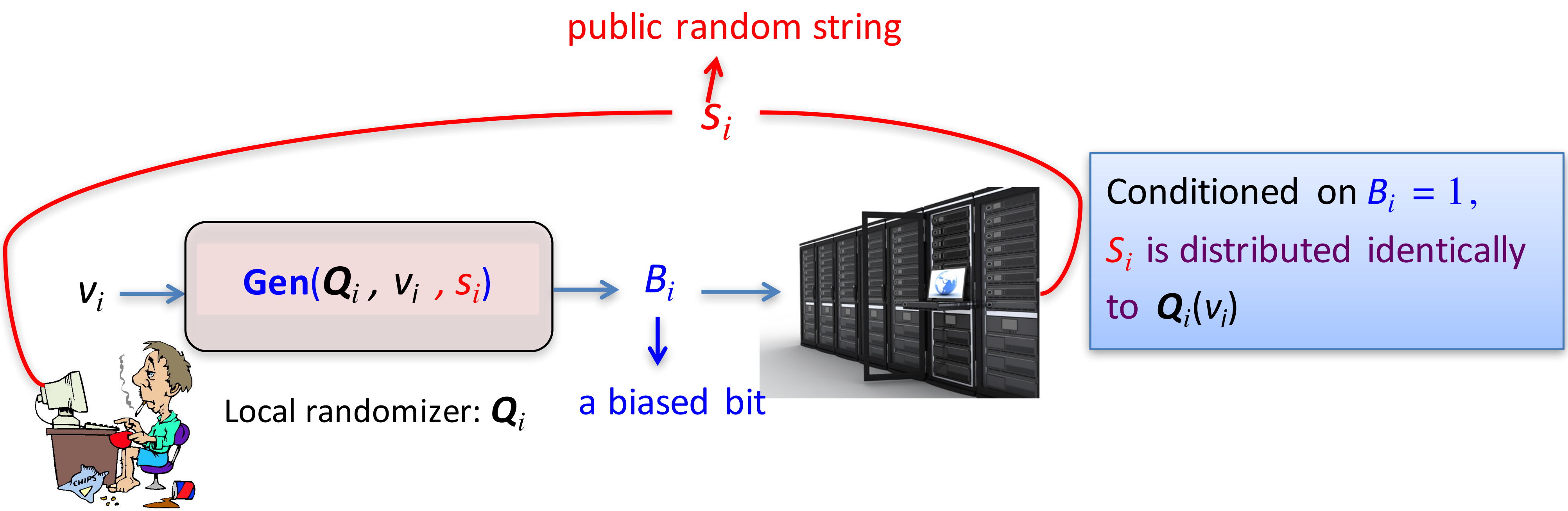
Not in this talk:

Lower bound on error

- Give a proof approach that adapts/simplifies a framework by [Duchi et al.'13].
- Show it applies also to the relaxed version of (ϵ,δ)-LDP for all $\delta \ll 1/n$.

Transforming to a protocol with 1-bit reports

Key idea: In public coin model, each user sends a single bit that enables the server to simulate the view of the user's differentially private report.



- This transformation is generic and adapts/modifies the technique of [McGregor et al.'10].
- Public string does not depend on private data: can be generated by untrusted server.
- For our HH protocol, this transformation gives essentially same error and computational efficiency (**Gen** can be computed in $O(log(log(d))+\log(n))$).

Transforming to a protocol with 1-bit reports

- In a public coin model, *any ϵ-LDP protocol* can be transformed into another ϵ-LDP protocol with 1-bit reports.

- Our transformation is a modification to a generic compression technique of [McGregor et al.'10].

- When applied to our protocol for histograms, this transformation gives a protocol that:
  - is computationally efficient protocol *(essentially same run time).*
  - has optimal error *(essentially same error).*

Key idea: Each user generates a single *biased* bit that enables the server to simulate the view of the user's differentially private report using the public coins.