Unmasking Vulnerabilities: Cardinality Sketches under Adaptive Inputs

Edith Cohen

Google Research & Tel Aviv University

Joint work with:

Sara Ahmadian

Google Research

Outline

Background

- Cardinality Queries
- Composable Sketches
- Adaptive vs Non-Adaptive Queries
- Known upper bounds on the number of queries as a function of sketch size

Our Contributions

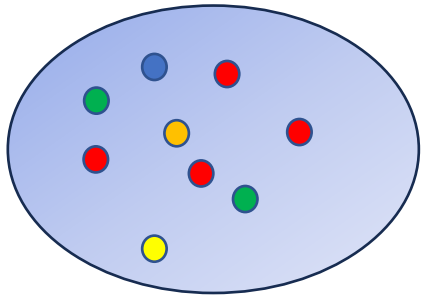
Attacks on the known composable sketch structures

- Single-Batch Linear-Size Attack on the optimal estimator (reveals all available information on the cardinality)
- Quadratic-size attack on any query response algorithm

Conclusion

Cardinality Queries

$F0$ frequency moment / ℓ_0 norm /**distinct count** statistic

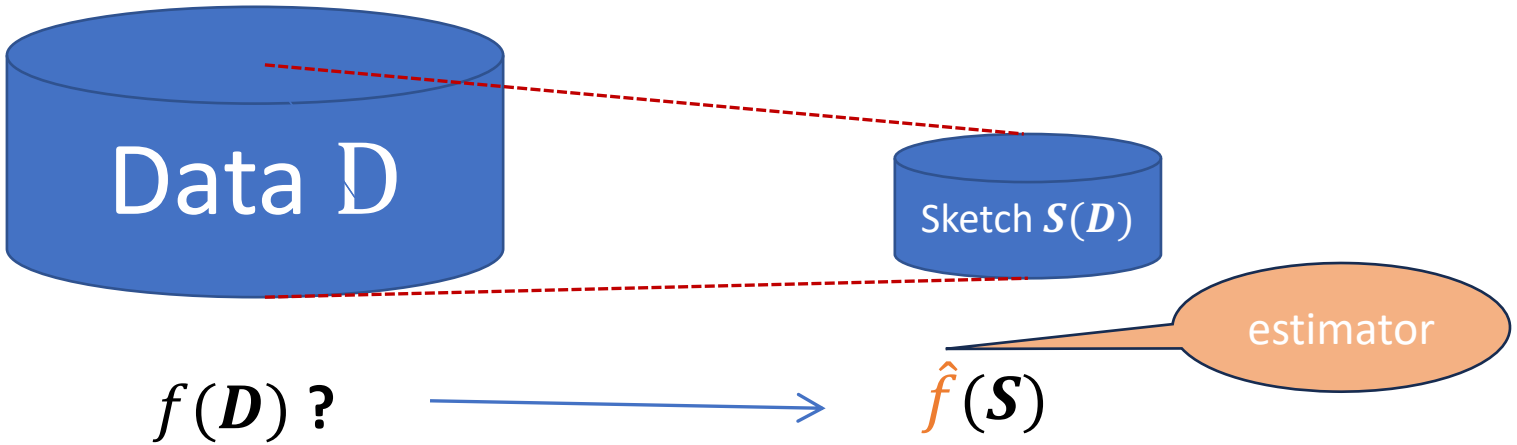
$\boldsymbol{A} =$  $\Rightarrow$  $\Rightarrow$ $|\boldsymbol{A}| = 5$

$\boldsymbol{V} = (0, 0, 3, -2, 1, 0, 0, -1, 10, 0, 0) \Rightarrow \|\boldsymbol{V}\|_{\boldsymbol{0}} = 5$

Applications: Distinct Search Queries, Users, Source-Destination pairs in IP flows……

Sketch Maps

Small representations that serve as surrogates $\mathrm{D} \mapsto S(D)$



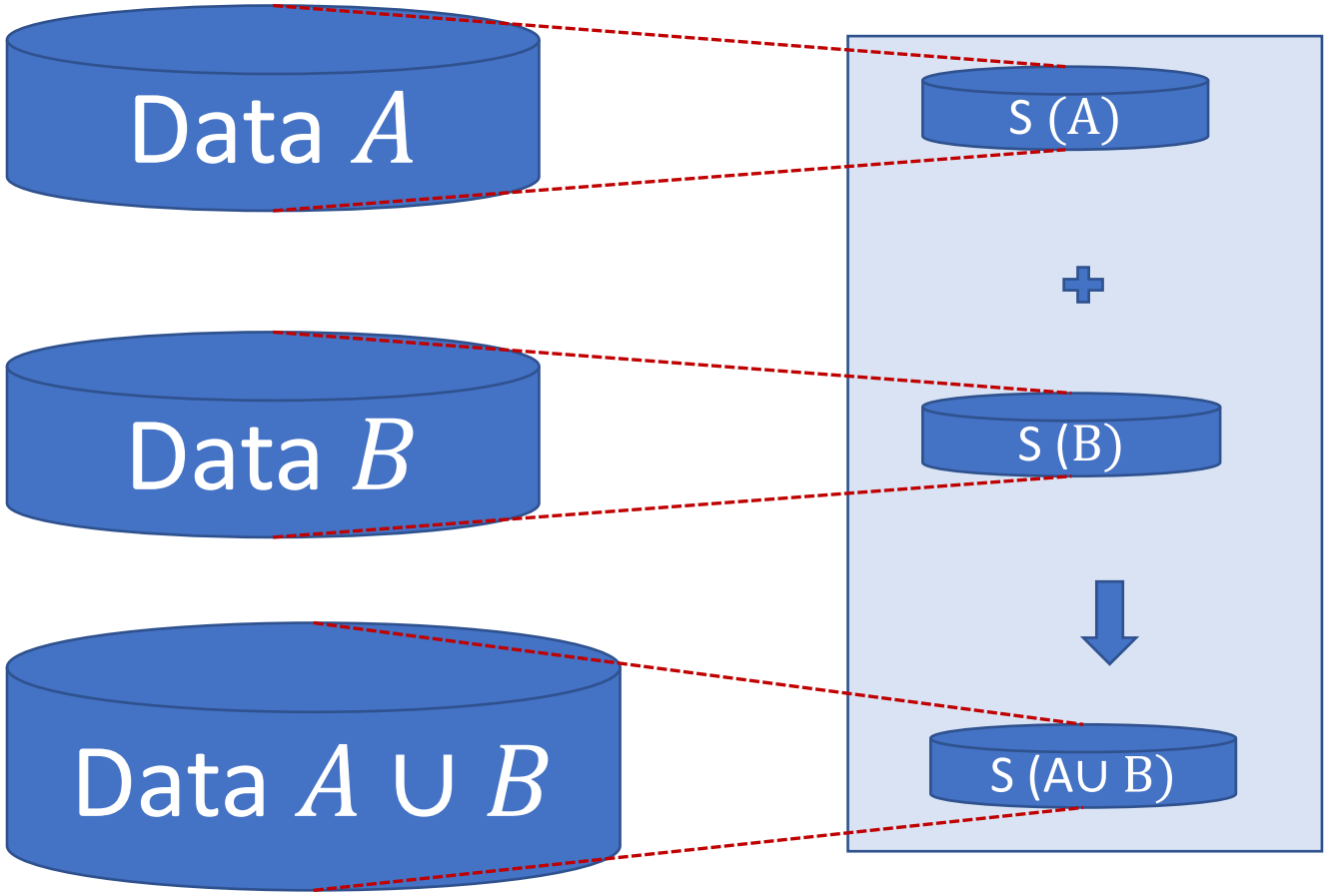
The cardinality of D can be estimated from its sketch $S(D)$

Design goals:

- Small size $|S(D)| \ll |D|$ (efficient storage/communication)
- Composable

Composable Sketch Maps

Map $\mathrm{D} \mapsto S(D)$

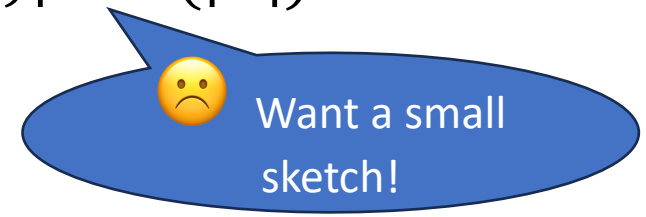


Why Composable?

- Efficient Distributed/Parallel/Streaming aggregation (operate in sketch space!)
- Composition queries (e.g. how many distinct users on certain combinations of days)

Composable sketches for Cardinality

First Try: Explicit representation or a Bloom Filter ⇒ $|S(\mathrm{D})| = \mathrm{O}(|\mathrm{D}|)$



Very small sketches! ☺️

Flajolet Martin '85
Cohen '97
Alon Marias Szegedy '99
Ganguly '07
Flajolet et al '07 (Hyperloglog)
.
.
.

Implementations
Apache DataSketches
Google BigQuery
.
.

Sketch size $\mathrm{loglog}\, n + k$ (n is dimension)

Statistical guarantees on estimate:

- NMSE: $\frac{1}{k}$
- $k = \frac{\log\left(\frac{1}{\delta}\right)}{\varepsilon^2}$ ⇒ $\Pr[\mathrm{RelError} > \varepsilon] < \delta$

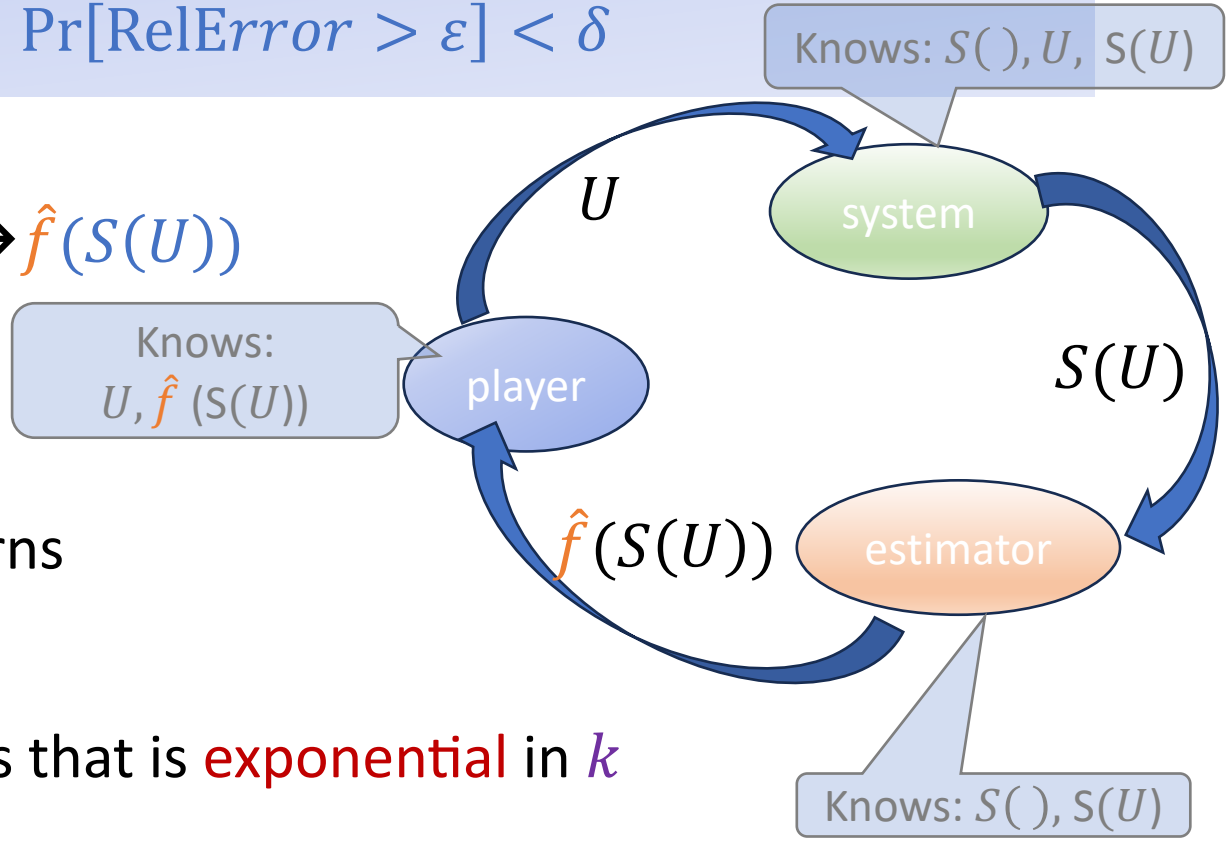
Composable sketches for Cardinality: Properties

Sketch size $\log\log n + k$ (n is dimension)

Statistical guarantees: $k = \frac{\log\left(\frac{1}{\delta}\right)}{\varepsilon^2} \quad \Rightarrow \quad \Pr[\text{RelError} > \varepsilon] < \delta$

Queries Processed in Sketch Space $U \rightarrow S(U) \rightarrow \hat{f}(S(U))$

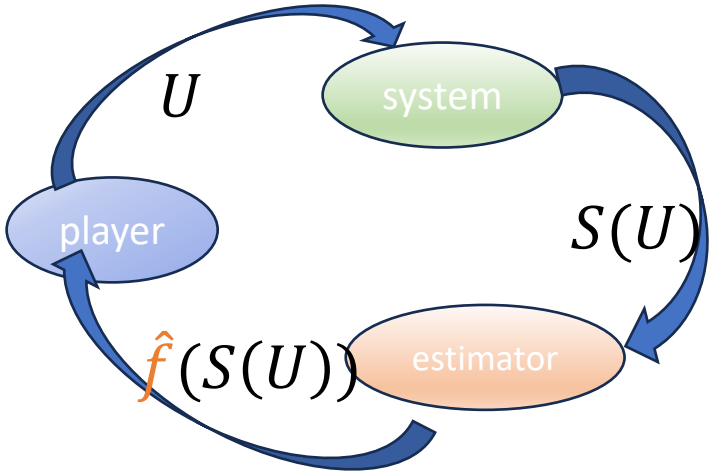
- "player" specifies query set U_i
- "system" : sketches $U \rightarrow S(U)$
- "estimator" (query response algorithm) returns estimate $\hat{f}(S(U_i))$ of $|U_i|$



⇒ Can answer accurately a number of queries that is exponential in k

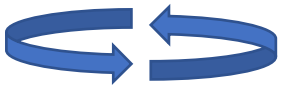
Caveat: Assumes inputs are not adaptive!

Adaptive Queries



Non-adaptive Setting:
The input sequence $(U_i)_{i=1}^{T}$ does not depend on the outputs (a_i)

Adaptive Setting:
Each input U_i may depend on $(U_j, a_j)_{j=1}^{i-1}$


A system with feedback


Adversarial: Aims to construct a bad input

?

What guarantees can we give for cardinality sketches **when inputs are adaptive**?

Outline

Background

- Cardinality Queries
- Composable Sketches
- Adaptive vs Non-Adaptive Queries
- Known upper bounds on the number of queries as a function of sketch size

Our Contributions

Attacks on the known composable sketch structures

- Single-Batch Linear-Size Attack on the optimal estimator (reveals all available information on the cardinality)
- Quadratic-size attack on any query response algorithm

Conclusion

Our Contributions

Sketch size k How many queries can be correctly answered?

Known Upper Bounds:

Non-adaptive queries: $2^{O(k)}$

Adaptive queries

- $\tilde{O}(k)$ (Use different randomness per query)
- $\tilde{O}(k^2)$ with the *DP robustness wrapper* [HKMMS]

We construct attacks that apply with all known composable sketch structures

➢ Against the optimal estimator (sufficient statistics):
Single batch attack with $\tilde{O}(k)$ queries

- Simple queries: random subsets of a ground set.
- Adversarial set produced by postprocessing

➢ Against **any** estimator: $\tilde{O}(k^2)$ queries

- Multiple $\tilde{\Omega}(k)$ batches are necessary!

Attacks work against powerful estimators and limited task

- Knows the attack algorithm, its state, all prior queries,…
- Only required to report correctly soft threshold queries:
  - If $|U| > 2A$ report "large"
  - If $|U| < A$ report "small"

Flajolet Martin ’85
Cohen ’97
Alon Marias Szegedy ’99
Ganguly ‘07
Flajolet et al ‘07 (Hyperloglog)
.
.
.

Composable Cardinality Sketches

One basic idea. Multiple designs.

- Assign random priorities $h(x)$ to keys $x \in \mathcal{U}$
- Sketch of set $U \subset \mathcal{U}$ is (derived from) the k keys of highest priority

$$\{ h(x) \mid x \in U \}_{(1:k)}$$

Analysis Idea: Larger cardinality implies Higher top priorities

Composable: The top priorities in $A \cup B$ can be recovered from top priorities in each of A, B

!! Composability requires **same random priorities** h for all queries

Analysis crucially depends on inputs being unrelated to h

But… adaptive Inputs may depend on prior outputs and hence on h

Attack Idea on Composable Cardinality Sketches

Cardinality Sketches

- Assign random priorities $h(x)$ to keys $x \in \mathcal{U}$
- Sketch of set $U \subset \mathcal{U}$ is (derived from) the k keys of highest priority

$$\{ h(x) \mid x \in U \}_{(1:k)}$$

Attack Approach:

- Fix a groundset $\mathcal{U}$
- Aim to identify the set $\mathrm{T} \subset \mathcal{U}$ of k keys of highest priority (by issuing queries $U_i \subset \mathcal{U}$)

Augmenting a query $U \subset \mathcal{U}$ with T poisons it by masking U
This because $S(\mathrm{U} \cup \mathrm{T}) = S(\mathcal{U})$ (regardless of the cardinality $|\mathrm{U}|$)!

? How to efficiently "zoom" on T ?

Warm Up: Adaptive Attack on Optimal Estimator

Optimal = returns a **sufficient statistic** of the cardinality

- Fix a groundset $\mathcal{U}$; take a random permutation $(x_1, x_2, \dots, x_n)$ of $\mathcal{U}$
- Let $c_i \leftarrow \hat{f}(S(x_1, \dots, x_i))$ % cardinality estimate for prefix
- Search to identify all positions $j \in [n]$ so that $c_j \neq c_{j-1}$ and collect x_j

Output keys at change points $\{x_j\}$

Claim: Sequence $c_1, \dots, c_n$ changes in expectation in $O(k \log n)$ positions

Attack uses $\tilde{O}(k)$ queries.

- queries produced adaptively.
- queries are contrived

Single Batch $\tilde{O}(k)$ Attack on Optimal Estimator

Fix a groundset $\mathcal{U}$; Initialize *scores* $c[x] \leftarrow 0$ for $x \in \mathcal{U}$

Repeat $\tilde{O}(k)$ times:

Select $\mathrm{U} \subset \mathcal{U}$ be independently including each $x \in \mathcal{U}$ with prob $\frac{1}{2}$

Get cardinality estimate $\hat{f}(S(U))$

For $x \in U:\ c[x] \mathrel{+}= \frac{1}{\hat{f}(S(U))}$

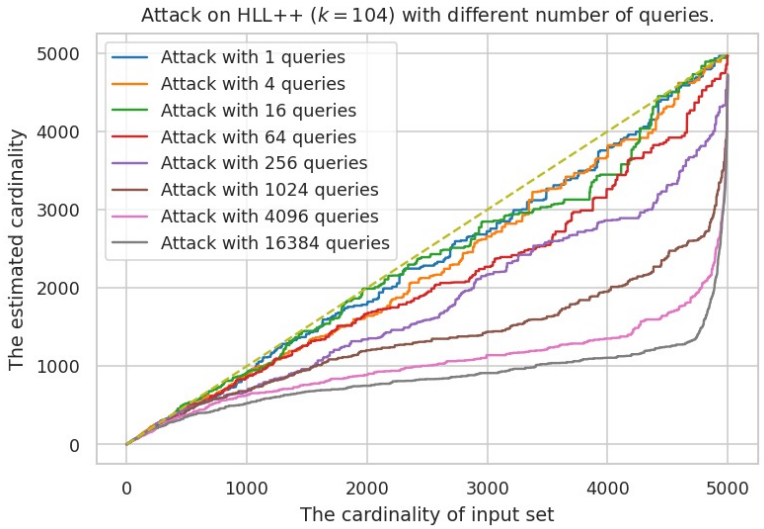
Output $\mathcal{U}$ ordered by score

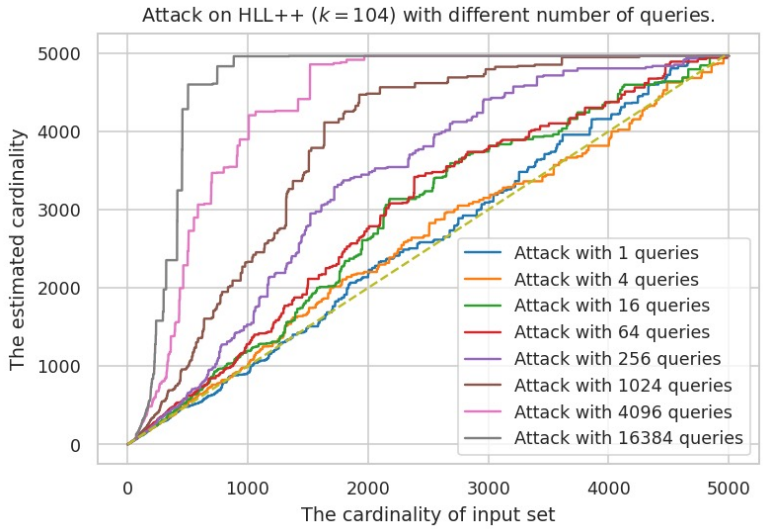
Only the post processing is dependent on prior outputs!

Lemma: The output order is *correlated* with priorities:
The $\tilde{O}(k)$ prefix includes the high priority keys

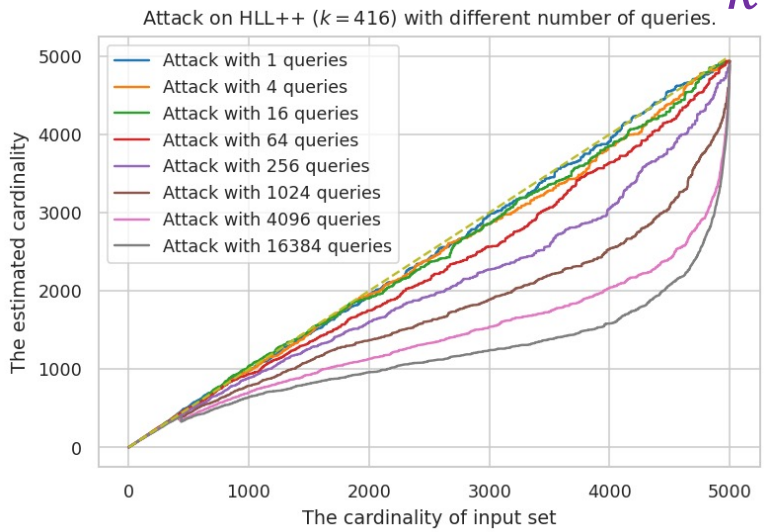
Single Batch Attack on HLL++

$k = 104$

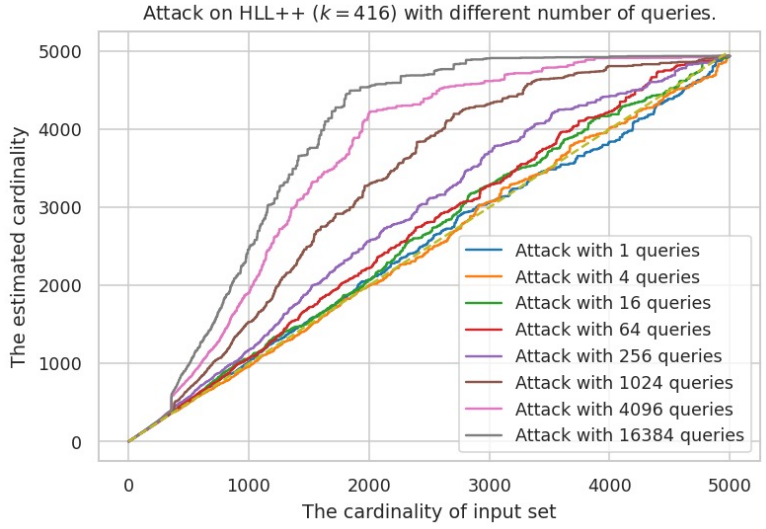




$k = 416$



Cardinality is underestimated for suffixes.



Cardinality is overestimated for prefixes.

Single Batch $\tilde{O}(k)$ Attack on Optimal Estimator

Fix a groundset $\mathcal{U}$; Initialize *scores* $c[x] \leftarrow 0$ for $x \in \mathcal{U}$

Repeat $\tilde{O}(k)$ times:

Select $\mathrm{U} \subset \mathcal{U}$ be independently including each $x \in \mathcal{U}$ with prob $\frac{1}{2}$

Get cardinality estimate $\hat{f}(S(U))$

For $x \in U:\ c[x] \mathrel{+}= \frac{1}{\hat{f}(S(U))}$

Output $\mathcal{U}$ ordered by score

Lemma: The output order is *correlated* with priorities:
The $\tilde{O}(k)$ prefix includes the high priority keys

Proof idea: When set includes more high priority keys, the cardinality estimate is slightly higher. So high priority keys get on average higher scores.

!! Attack can be heuristically applied when monitoring normal work loads. If most keys appear enough times in queries.

$\tilde{O}(k^2)$ Attack on Composable Sketches

any estimator (query response algorithm)

Effective even against a **powerful** defender (query response algorithm) and **weak** attacker (that gets 1-bit responses to a more specialized task than cardinality estimation)

Query Response Algorithm:
Knows the attack algorithm, its internal state, all prior query sets, the ground set, the distribution from which current query is selected, and after responding to a query obtains the query set

⇒ Attacker can not be deterministic, can not use fixed cardinality queries

Task: Soft threshold queries

- If $|U| > 2A$ ⇒ return 1 "large"
- If $|U| < A$ ⇒ return 0 "small"
- Otherwise ⇒ unrestricted 0 / 1

⇒ Soft Threshold is more specialize -- can be solved with Approximate Cardinality with $\sqrt{2}\times$ error.

$\tilde{O}(k^2)$ Attack on Composable Sketches

Fix a ground set $\mathcal{U}$; Initialize *scores* $c[x] \leftarrow 0$ for $x \in \mathcal{U}$; Initialize mask $M \leftarrow \emptyset$; Set soft threshold $A = \frac{|\mathcal{U}|}{16}$

Repeat $\tilde{O}(k^2)$ times:

- Sample a sampling rate q
- Select U by including each $x \in \mathcal{U}$ with probability q
- Get soft threshold $Z \in \{0,1\}$ for the sketch $S(U \cup M)$ from Query Response
- For each $x \in \mathcal{U}$,
  - $c[x] \leftarrow c[x] + Z$
  - If $c[x]$ is statistically above the median score, then $M \leftarrow M \cup \{x\}$

Type equation here.

- Simple unified attack for all **known** composable sketch types + Unified analysis
- Product of the attack is a small set M so that for any $U \subset \mathcal{U}$ $S(M \cup U) \approx S(\mathcal{U})$

  $\Rightarrow$ it is not possible for Query Response to estimate the cardinality $|M \cup U|$

A malicious player can use M to poison a much larger dataset U so that the sketches fail

$\tilde{O}(k^2)$ Attack on Composable Sketches

any estimator (query response algorithm)

Analysis Highlights

Unified view of all known composable sketch maps for cardinality estimation.

- The sketch of U is determined by a subset of U of size $\tilde{\Theta}(\min\{k, |U|\})$.
- We characterize the sketch distribution on our query distribution in terms of the sampling rate q. This thru a scan order over keys in the ground set so that at each point a key is determining for the sketch if sampled in. The sufficient statistics on q from the sketch is dominated by the sum of k iid $Geom[q]$ random variables. This corresponds to the number of keys scanned until one is sampled in. Repeated k times.
- We establish that when the sum statistic is lower than its expectation, ''1'' response to soft threshold is more likely and there is a higher fraction of top priority keys. Therefore, top priority keys get higher scores *on average*.

$\Rightarrow$ If some top priority keys get low scores, others must get higher scores faster as to balance the average and are selected earlier into the mask M.

Once placed in M, these keys do not contribute information on q and Query Response must rely on the presence of other top priority keys and expose them .

Conclusion

We demonstrated vulnerability to adaptive inputs by presenting attacks

- $\tilde{O}(k)$ queries to attack popular cardinality sketches and estimators
- $\tilde{O}(k^2)$ queries to attack any known composable sketch structure with any query response algorithm

Match upper bounds. Simple attack queries.

Open:

- (negative) Can we show that any composable cardinality sketch map MUST be based on prioritizing keys (that is, are essentially coordinated samples)?
- (positive) Formulate conditions (e.g. when keys participate in a limited number of queries) where the sketch is robust