One Attack to Rule Them All: Cardinality Sketches under Adaptive Inputs

Edith Cohen

Google Research & Tel Aviv University

Sara Ahmadian

Jelani Nelson

Tamás Sarlós

Mihir Singhal

Uri Stemmer

Outline

Background

- Cardinality Queries
- Composable Sketches

  $2^{O(k)}$ non-adaptive queries for sketch size k
- Adaptive queries
  - Positive results: $\tilde{O}(k^2)$ adaptive queries via wrapper methods
  - Negative results via attacks

Our Contributions

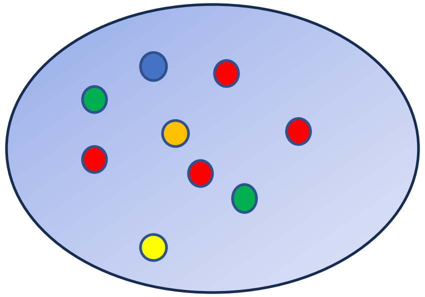
A unified universal attack on cardinality sketches

Structural properties of union-composable sketches

- Tight $\tilde{O}(k^2)$ attacks for monotone composable sketches and linear sketches (Boolean, Reals, Finite Fields) and (with some assumptions) Integers
- $\tilde{O}(k^4)$ attack on any composable sketch
- Single-batch $\tilde{O}(k)$ attack on optimal estimator

Cardinality Queries

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

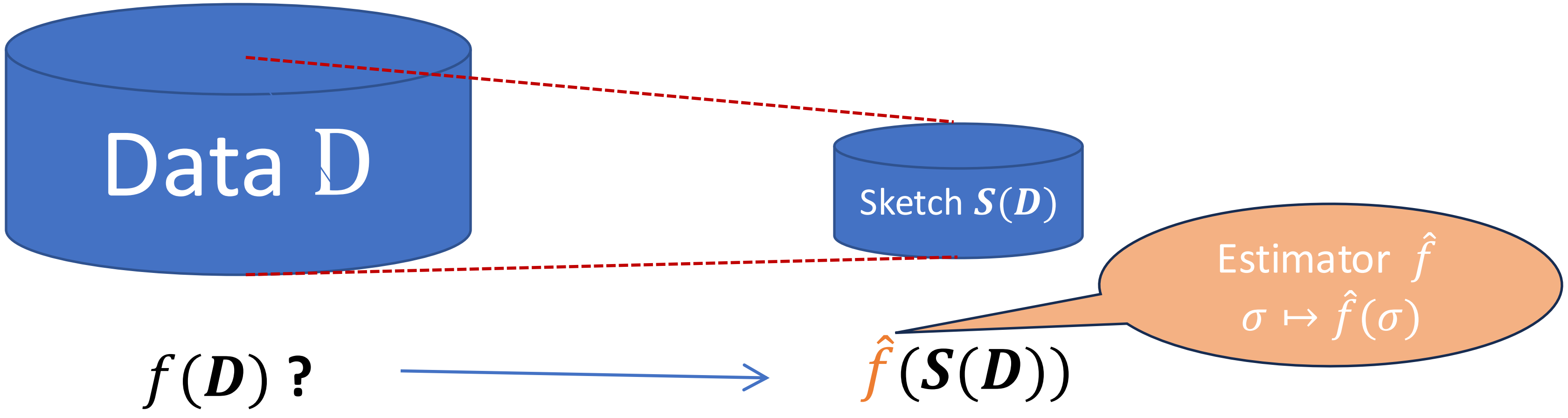
$\boldsymbol{D}$ =  $\Rightarrow$  $\Rightarrow$ $|\boldsymbol{D}| = 5$

$\boldsymbol{D} = (0, 0, 3, -2, 1, 0, 0, -1, 10, 0, 0) \Rightarrow \|\boldsymbol{D}\|_0 = 5$

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

Sketch Maps

Maps of data to small representations $\mathrm{D} \mapsto S(D)$



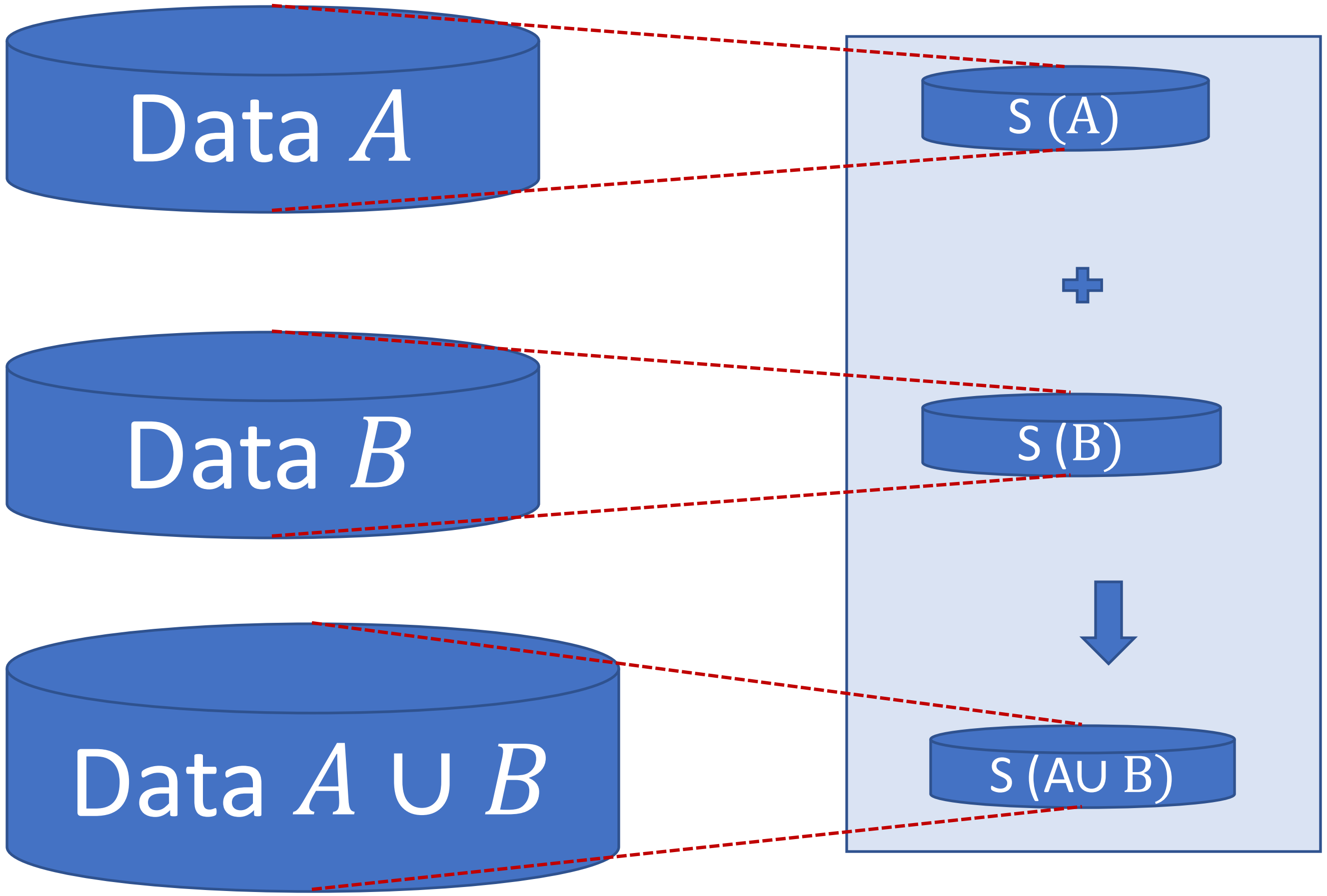
Cardinality sketch: The cardinality of $\boldsymbol{D}$ (or $\|\boldsymbol{D}\|_0$) can be estimated from $\boldsymbol{S}(\boldsymbol{D})$

Design goals:

- Small $|S(D)| \ll |D|$ (efficient storage/communication)
- Accurate $\hat{f}(\boldsymbol{S}(\boldsymbol{D})) \approx f(\boldsymbol{D})$
- Composable

Composable Sketch Maps

$$\mathbf{D} \mapsto S(\boldsymbol{D})$$



$$S(\boldsymbol{A} \cup \boldsymbol{B}) = S(\boldsymbol{A}) \oplus S(\boldsymbol{B})$$

Why Composable?
Efficiency on Distributed/ Streaming data (operate in sketch space!)

Practice: dataset in each location / time-period is sketched and then discarded. Queries are localized or on unions of datasets.

Composable sketches for Cardinality

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

☹ Want a small sketch!

Composable sketches for Cardinality

Very small sketches!

Flajolet Martin ’85
Cohen ’97
Alon Marias Szegedy ’99
Bar-Yoseff, Jayram, Kumar, Siva, Trevissan ‘02
Cormode, Datar, Indyk, Muthu ’03
Ganguly ‘07
Flajolet et al ‘07 (Hyperloglog)
.
Kane, Nelson, Woordruff ‘10
.
.

Implementations
Apache DataSketches
Google BigQuery
.
.

!! Randomness is **necessary**

Sketching map $S \sim D$ is sampled from a distribution

!! For composability, **same sampled map** S must be used for all sets

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

Statistical guarantees on accuracy:

- NMSE: $\frac{1}{k}$
- $k = \frac{\log\left(\frac{1}{\delta}\right)}{\varepsilon^2} \quad \Rightarrow \quad \Pr_{S\sim D}[\mathrm{RelError} > \varepsilon] < \delta$

Non-Adaptive Queries

Sketch size $\log\log n + k$

$k = \frac{\log\left(\frac{1}{\delta}\right)}{\varepsilon^2} \quad \Rightarrow \Pr_{S\sim D}[\text{RelError} > \varepsilon] < \delta$

Queries $U_1, U_2, U_3, \ldots$ **processed in Sketch Space** $U_i \rightarrow S(U_i) \rightarrow \hat{f}(S(U_i))$

How many queries can we answer accurately? ⇒ **exponential** in k

Caveat! We use the same **sampled map** S for all queries

⇒ Holds when inputs $U_1, U_2, U_3, \ldots$ are **non-adaptive – do not depend on** S !

? What about the adaptive setting?

Adaptive Queries

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

Adaptive Setting:
Each input U_i may depend on $\left(U_j, \hat{f}(S(U_i))\right)_{j=1}^{i-1}$

A system with feedback

Adversarial: Aims to construct a bad input

?

What guarantees can we give **when inputs are adaptive**?

Background: Positive Results
Quadratic boost via Wrapper Methods

$\mathcal{A}$ with nonadaptive guarantees $\Rightarrow$ adaptive guarantees

Simple: $\mathcal{A} \times k \Rightarrow \tilde{\Omega}(k)$ adaptive queries

Advanced: $\mathcal{A} \times k \Rightarrow \tilde{\Omega}(k^2)$ adaptive queries

- Statistical Queries: [Dwork et al., '15, Bassily et al., '21]
- General Application: [Hassidim et al. '20]
- Subsampling: [Blanc '23]

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

Negative Results on Cardinality Sketches

$\tilde{O}(k^2)$ universal attack for adaptive statistical queries (queries over samples of size k) [Hardt and Ullman'14 , Steinke and Ullman '15] based on Fingerprinting Codes [Boneh and Shaw '98].

Linear Sketches [Gribelyuk et al. 2024]

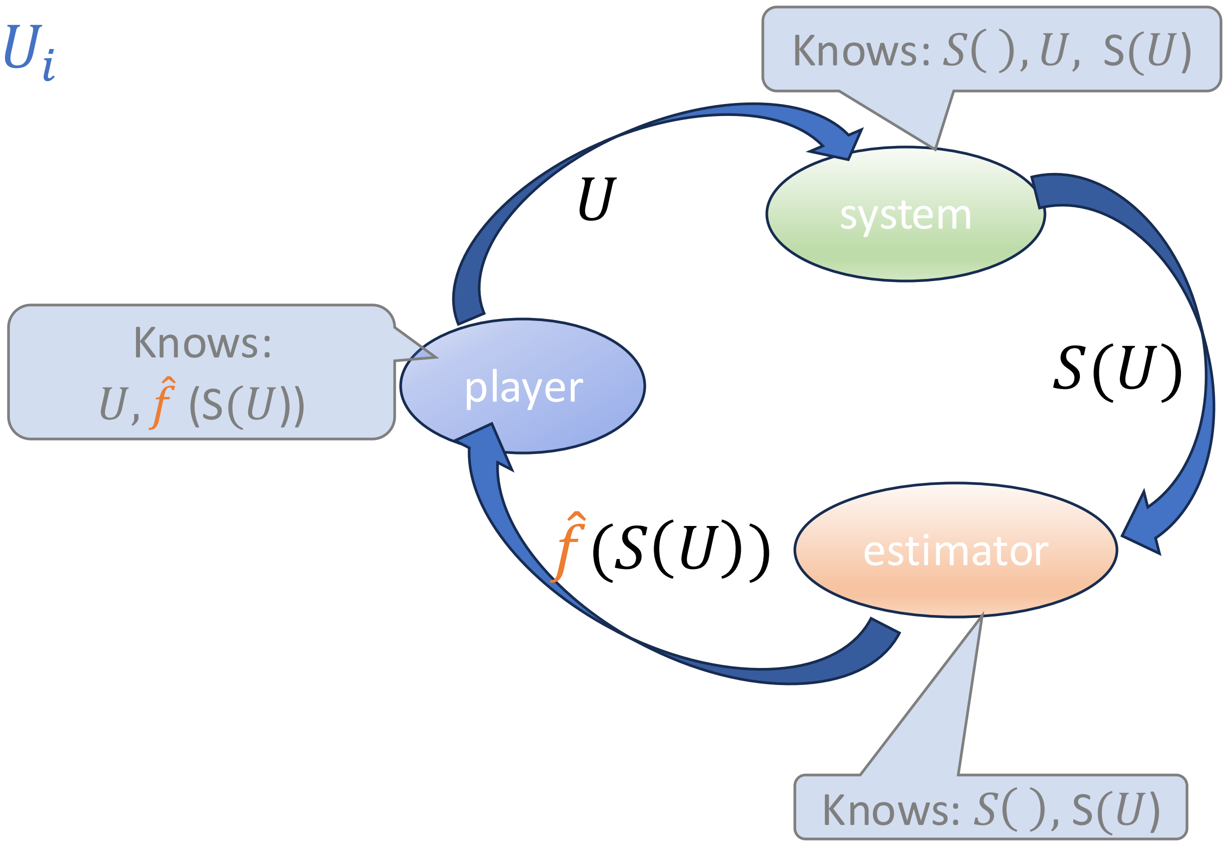
- $\tilde{O}(\text{poly}(k))$ over reals
- $\tilde{O}(k^8)$ over integers
- $\tilde{O}(k^3)$ over finite fields

Questions:
Gap – is there an $\tilde{O}(k^2)$ attack against any cardinality sketch?
Union-composable sketches (prevalent in practice)

Interaction Model

Queries $U_1, U_2, U_3, \ldots$ **processed in Sketch Space** $U_i \rightarrow S(U_i) \rightarrow \hat{f}(S(U_i))$

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

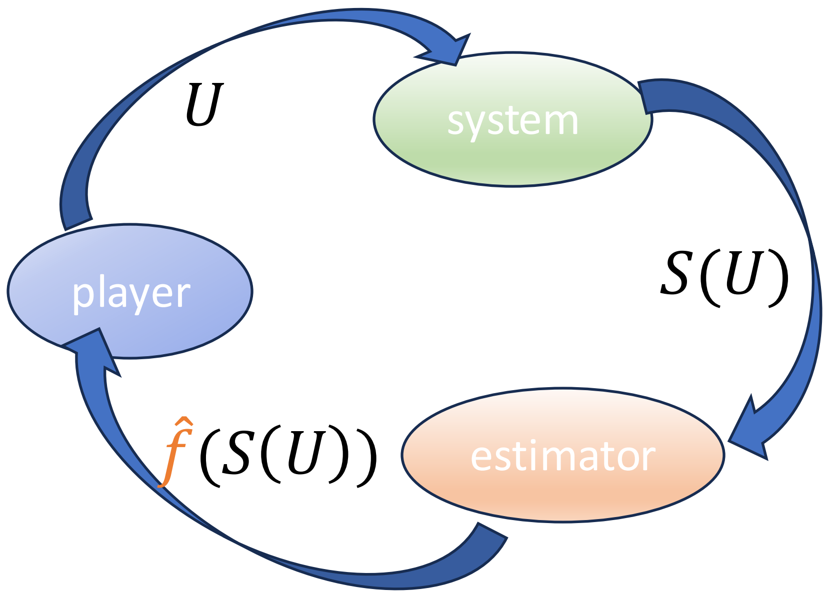


*Model corresponds to how sketches are used in practice

Attack on sketching map S

Queries $U_1, U_2, U_3, \ldots$ **processed in Sketch Space** $U_i \rightarrow S(U_i) \rightarrow \hat{f}(S(U_i))$

Attack Size: Number of adaptive queries needed to *compromise* (force incorrect responses) S of size k



Attack types:

- ***Tailored***: Applies with a specific estimator
- ***Universal***: Applies with any query responder

Our Results

A Unified Universal Attack (applies with any estimator)

Composable Sketch Map S :

- General: $\tilde{O}(k^4)$ adaptive queries
- Monotone: $\tilde{O}(k^2)$ adaptive queries

Linear Sketch Maps $\tilde{O}(k^2)$ adaptive queries

- Boolean, reals $\mathbb{R}$, Finite Fields F_{p}
- (with some assumptions) Integer

Tight!

Principled Technique: Structural properties of composable sketching maps

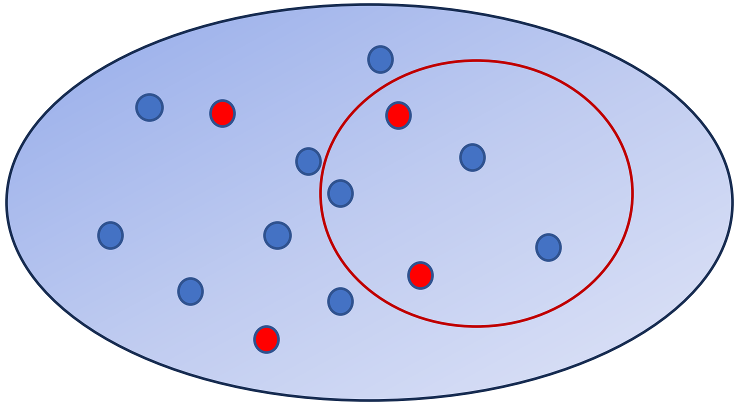
Tailored Attacks:

- Single-Batch $\tilde{O}(k)$ attack on the optimal estimator

Statistical Queries as Cardinality Sketches

Sketching map by a sample R of size k from the groundset $\mathcal{U}$

- $S(U) := U \cap R$



- Estimate $\frac{|U \cap R|}{|\mathcal{U}|}$ -- accurate on large inputs (a fraction of $\mathcal{U}$)
- Adaptive attacks aim to identify R , query responder aims to be accurate while protecting R

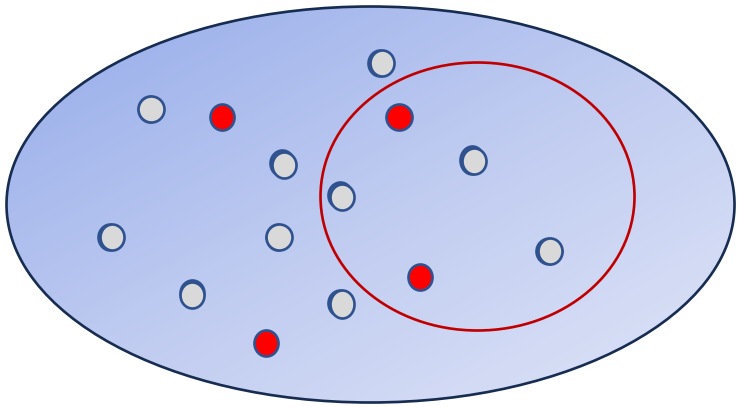
MinHash sketches (most used in practice) including Hyperloglog are glorified drilled-down samples

Cardinality Sketches

Property facilitating unified $\tilde{O}(k^2)$ attack:

Composable cardinality sketches (can be caused to) "behave like" statistical queries

Only few keys "determine" the sketch



Composable Cardinality Sketches

Multiple known designs. One basic idea*.

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

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

Sketching map S = **priorities** h

Analysis Idea: Larger cardinality $\Leftrightarrow$ Higher top priorities

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

* Implicit also in linear sketches

Composable Cardinality Sketches

Multiple known designs. One basic idea*.

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

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

Sketching map S = **priorities** h

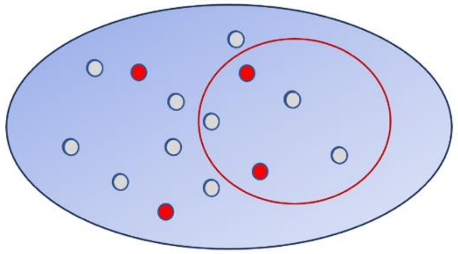
"Determining Pool" Property:

For random sets $U \sim \text{Bern}[q]^{\mathcal{U}}$, few keys "matter" , most keys are "transparent" to S

Just like SQ!

Theorem: Any composable sketching map has a "small" pool

Corollary: Inherent vulnerability to adaptive inputs (and privacy)



Determining Pool

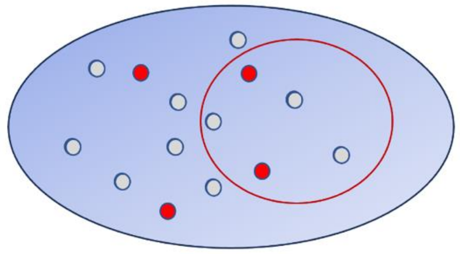
Groundset $\mathcal{U}$ Sketching map S

Set $L \subset \mathcal{U}$ such that for randomly sampled $U \sim \text{Bern}[q]^{\mathcal{U}}$ with $\mathrm{q} = \Omega(1)$

$$S(U) \approx S(U \cap L)$$

- A determining pool always exists (take $\mathrm{L} = \mathcal{U}$).
- To be useful, it needs to be small, depend on sketch size k not on ground set size $|\mathcal{U}|$

Example: SQ – the pool is the sample R of size k

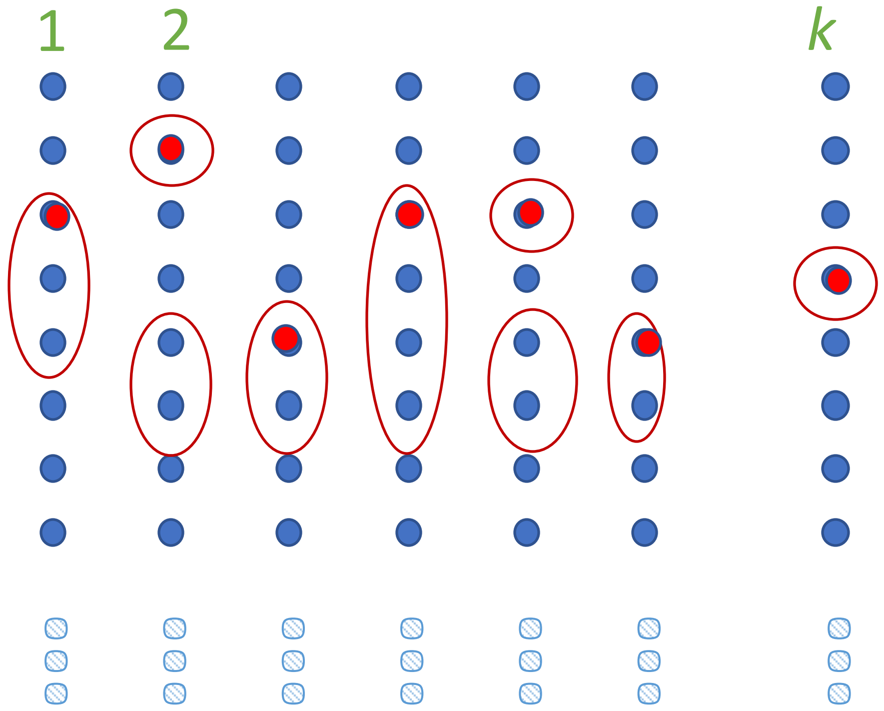


Example: Pool for MinHash Sketches HyperLogLog (Stochastic Averaging)

- Randomly prioritize keys
- Randomly partition universe to k bucket

Sketch: highest priority key in each bucket

Flajolet Martin '85
Flajolet et al '07 (Hyperloglog)

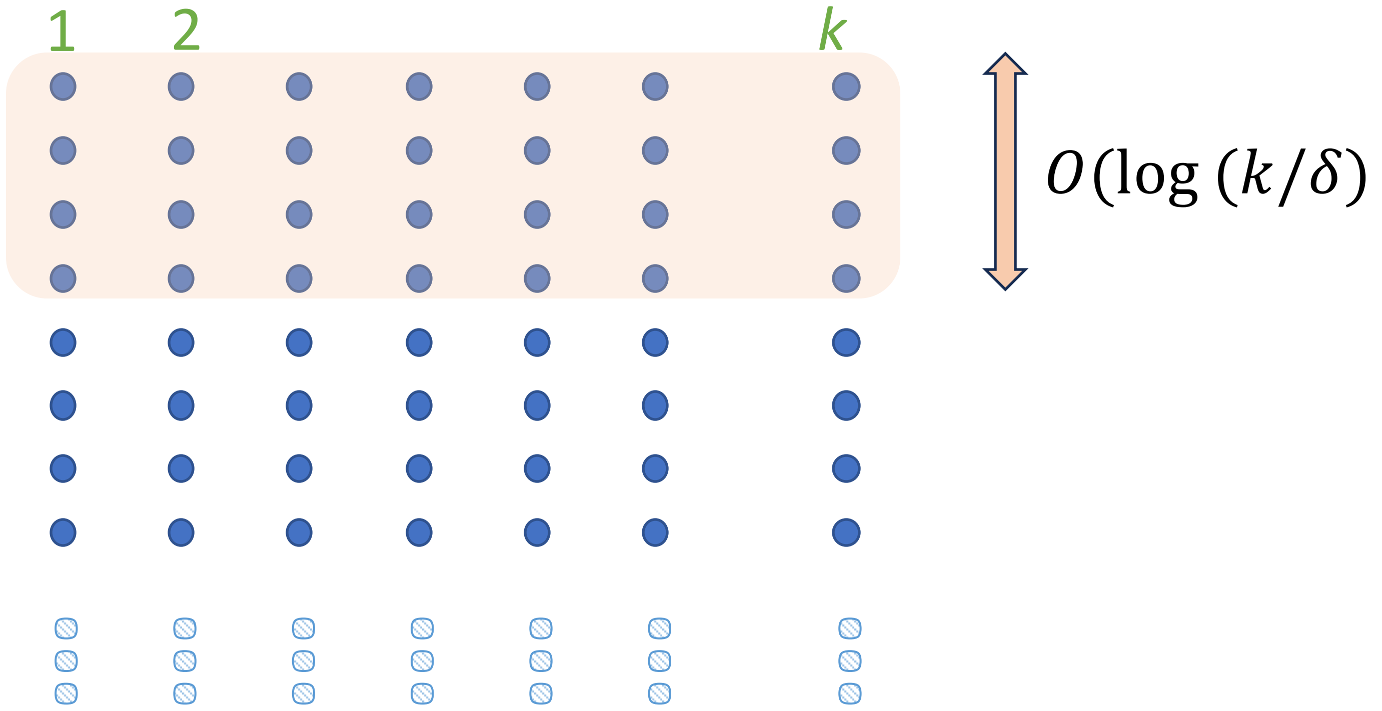


Example: Pool for HyperLogLog MinHash sketch

- Randomly prioritize keys
- Randomly partition universe to k bucket

Sketch: highest priority key in each bucket

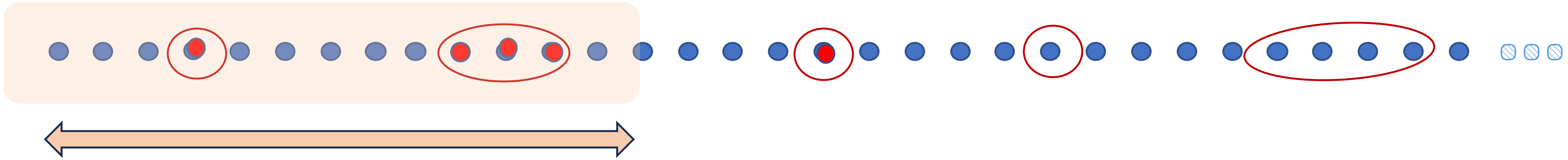
Flajolet Martin '85
Flajolet et al '07 (Hyperloglog)



Example: Pool for Bottom-k MinHash Sketches

- Randomly prioritize keys

Sketch: k highest priority keys



$O(k \log (k/\delta))$

Small Determining Pool L is a Vulnerability Attack Pradigm

- Fix a groundset $\mathcal{U}$ of size $1000 \cdot |L|$
- Attack identifies $M \approx L$ (approximate the determining pool)

For query sets

- $U \sim \text{Bern}[q]^{\mathcal{U}}$ for different $q > 0.2$
- $U' \leftarrow U \cup M$

We have $S(U') \approx S(M)$ ($\Rightarrow$ M **masks** U)

$\Rightarrow$ it is not possible to estimate $|U'|$ ($|U'| > 0.1\ |\mathcal{U}| \gg |L|$

Generalizes the Fingerprinting attacks of
[Hardt and Ullman’14 , Steinke and Ullman ‘15]

Composable Maps

Groundset $\mathcal{U}$ Sketching map S from $2^{\mathcal{U}}$ to Σ
Binary composition operation $\oplus$: $S(\boldsymbol{A} \cup \boldsymbol{B}) = S(\boldsymbol{A}) \oplus S(\boldsymbol{B})$

Core of a sketch $\sigma \in \Sigma$: Minimal $\mathrm{U} \subset \mathcal{U}$ such that $S(U) = \sigma$

Monotonicity: Core size can only increase with subset size
Rank of S : Minimum cardinality of a Core

Examples:

- Statistical Queries: Σ are subsets of the sample R. $\mathrm{S(U)} = \mathrm{U} \cap R$
- Vectors Spaces (σ is the spanned subspace, cores are basis)
- MinHash: Cores are the low priority keys

Monotone

Unique Core

Composable Maps

Groundset $\mathcal{U}$ Sketching map S from $2^{\mathcal{U}}$ to Σ

Binary composition operation $\oplus$: $S(\boldsymbol{A} \cup \boldsymbol{B}) = S(\boldsymbol{A}) \oplus S(\boldsymbol{B})$

Core of a sketch $\sigma \in \Sigma$: Minimal $\mathrm{U} \subset \mathcal{U}$ such that $S(U) = \sigma$

Monotonicity: Core size can only increase with subset size

Rank of S : Minimum cardinality of a Core

Lemma: Maximum sketch size $\max_{\sigma \in S(2^{\mathcal{U}})} |\sigma| \leq k \Rightarrow$ Rank $\leq k$

Thm: Pool size for composable maps of rank k

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

Monotone: $\tilde{O}(k)$

Constructive proof via Core Peeling, $\tilde{O}(k)$ for general, $\tilde{O}(1)$ for monotone S

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

Fix a groundset $\mathcal{U}$ of size $100 \cdot |L|$; Initialize *scores* $c[x] \leftarrow 0$ for $x \in \mathcal{U}$

Repeat $\tilde{O}(|L|)$ times:

Select $\mathrm{U} \subset \mathcal{U}$ to independently include 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

Single batch: Only the post processing is dependent on prior outputs!

Lemma: The $\tilde{O}(|L|)$ highest scores includes the pool keys

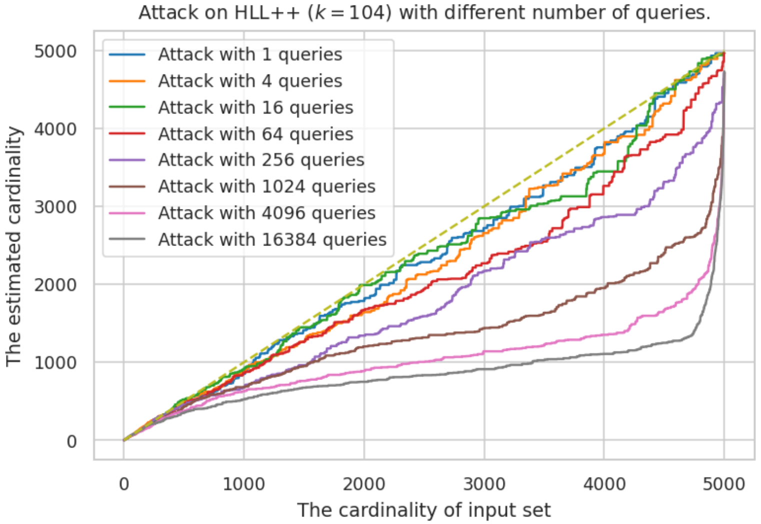
Optimal estimate depends only on intersection with L

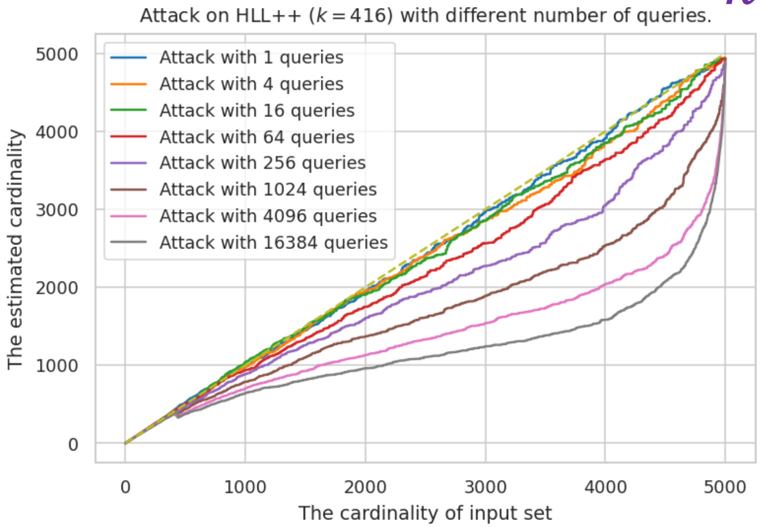
“Transparent” keys do not get biased scores, Pool keys more likely to be scores

Single Batch Attack on HLL++

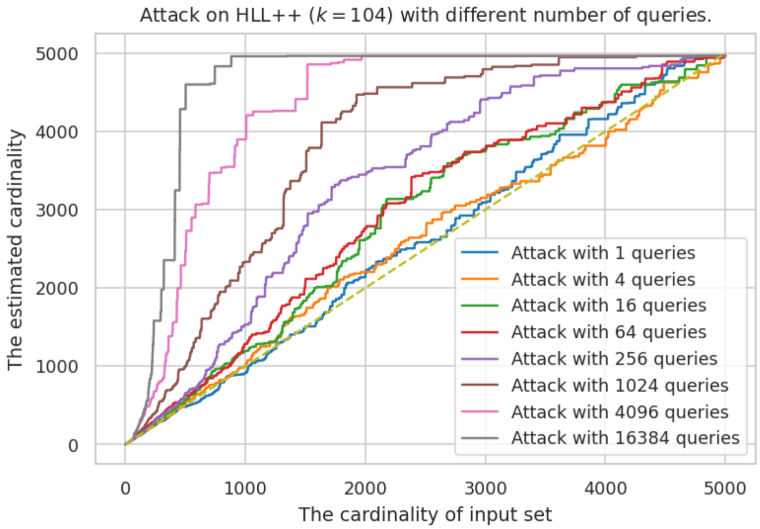
$k = 104$

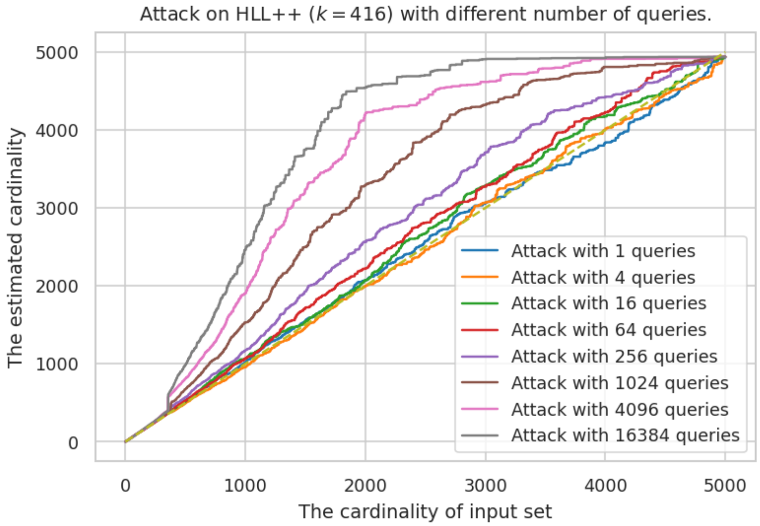
$k = 416$





Cardinality is underestimated for suffixes.





Cardinality is overestimated for prefixes.

Soft Threshold Queries

Task: Soft threshold queries

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

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

Unified Universal Attack

Fix a ground set $\mathcal{U}$; Initialize *scores* $c[x] \leftarrow 0$ for $x \in \mathcal{U}$; Initialize mask $M \leftarrow \emptyset$; Set threshold $A = 0.1\ |\mathcal{U}|$

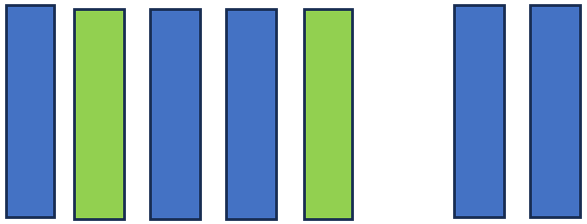
Repeat $\tilde{O}(|L|^2)$ times:

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

- Attack works against any query responder (powerful, strategic, adaptive)

Theorem: When Sketching map has a determining pool L, attack forces an error rate of $\frac{1}{4}$ after $\tilde{O}(|L|^2)$ queries

Linear Sketches

$A =$ 

Sketching matrix

$\boldsymbol{v} = (0, 3, 0, 0, -2, 0, 0, \cdots, 0, 0)$

Query vector $\boldsymbol{v}$ – the set are the nonzero entries

$S(\boldsymbol{v})$ = A $\boldsymbol{v}$

Sketch

Multiple representations for the same set, not union-composable

Linear Sketches: Boolean

Values are Boolean, ∨ instead of + , ∧ instead of *

Boolean linear sketches are monotone and composable ⇒ $\tilde{O}(k^2)$ attack

$\tilde{O}(k)$ Determining pool: All columns that have **1** value in some sparse measurement

$$A_i = (0,1,0,0,\cdots,0,1,1)$$
$$A_{i+1} = (1,0,0,0,\cdots,0,0,0)$$

Idea: Dense measurements do not matter, as whp they are hit with a member of a random set and sketch entry is **1** .

$$A_i = (1,0,1,1,\cdots,1,0,1)$$

Integer/Real Linear Sketches with sparsity pattern estimators

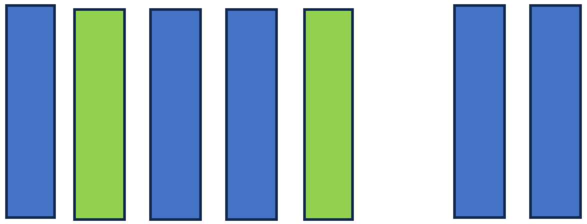
Boolean linear sketches are monotone and composable ⇒ $\tilde{O}(k^2)$ attack

Folklore and other linear sketches [Cormode, Datar, Indyk, Muthu ’03, Ganguly ’07] caused the sketch over integers to behave like Boolean. The estimator only uses the sparsity pattern (set of nonzero indices in the sketch and not values).

Result: $\tilde{O}(k^2)$ Attack on linear sketches on reals/integers that only use the sparsity structure in the sketch

Idea: We specify values randomly to attack queries ⇒ probability of any cancelation is small ⇒ sketch sparsity behaves like a Boolean sketch

Linear Sketches: Reals, Finite Fields

$A =$ 

$\boldsymbol{v} = (0, 3, 0, 0, -2, 0, 0, \cdots, 0, 0)$

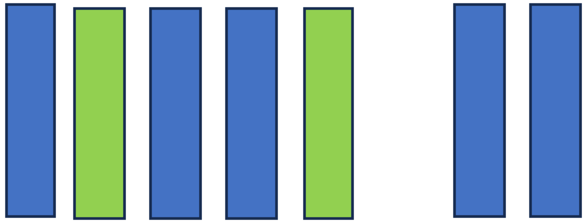
The attack queries are vectors, augment attack specs with values for the nonzero entries

1/0 don't work! sketch contains exact value

$A = (1,1,1,1, \cdots, 1, 1)$ $\boldsymbol{v} = (0, 1, 0, 0, 1, 0, 0, \cdots, 0, 0)$

Approach: We specify values $X(U)$ so that there is a determining pool L

Linear Sketches: Reals, Finite Fields

$A =$ 

$\boldsymbol{v} = (0, 3, 0, 0, -2, 0, 0, \cdots, 0, 0)$

Result: $\tilde{O}(k^2)$ Attack on linear sketches on reals/finite fields

Idea: span of vectors $U \mapsto span(U)$ is a monotone composable map

- Take L to be the determining pool for the column vectors of A for span.
- We specify a particular way $X(M, U)$ of sampling nonzero values to M, U in the attack queries so that L is a pool:

$$S(M, U, X(U, M)) \approx S(M, U \cap L, X(U \cap L, M))$$

Conclusion

Vulnerability to adaptive inputs by presenting attacks

- $\tilde{O}(k)$ queries to attack popular cardinality sketches and estimators
- Tight $\tilde{O}(k^2)$ Universal Attacks (against any query responder) on any monotone composable and linear sketches over reals, finite fields, Boolean, integers with limited estimators
- $\tilde{O}(k^4)$ for general composable sketches

Open:

- General composable sketches
- Determining pool property for other properties beyond cardinality
- Integer Linear Sketches

Follow up: When keys participate in a limited number of queries) where the sketch is robust (bound is in terms of key participation)

Thank you!