Time Adaptive Sketches (Ada-Sketches) for Summarizing Data Streams

Anshumali Shrivastava (Rice), Arnd Christian König (Microsoft) and Misha Bilenko (Microsoft)

anshumali@rice.edu

30th June 2016

Motivation: Counting over Temporal Data Streams

Counting over temporal data streams is a common problem.

E.g.: Given query, user and context, predict probability of click.

Given:

- **Basic Features:** userid, query, **timestamp**, ad-url, ad content, more categorical features like browser type, etc.
- **Target:** (1/0) Click/Non-Click

Motivation: Counting over Temporal Data Streams

Counting over temporal data streams is a common problem.

E.g.: Given query, user and context, predict probability of click.

Given:

- **Basic Features:** userid, query, **timestamp**, ad-url, ad content, more categorical features like browser type, etc.
- **Target:** (1/0) Click/Non-Click

Most Information in Count Features:

- A significant burst of clicks on some "XYZ" category (or combinations) in past few minutes.
- Features based on frequencies of clicks/non-clicks conditional on categories, over time, is generally adopted in practice.

Motivation: Counting over Temporal Data Streams

Counting over temporal data streams is a common problem.

E.g.: Given query, user and context, predict probability of click.

Given:

- **Basic Features:** userid, query, **timestamp**, ad-url, ad content, more categorical features like browser type, etc.
- **Target:** (1/0) Click/Non-Click

Most Information in Count Features:

- A significant burst of clicks on some "XYZ" category (or combinations) in past few minutes.
- Features based on frequencies of clicks/non-clicks conditional on categories, over time, is generally adopted in practice.

Scale: Needle in haystack.

- Billions of users and tens of millions of ads.
- Combinations blow up. Just $(ad \times query) \sim 10^{10+}$
- 50ms response time, update statistics with observations.

The Classical (Non-Adaptive) Approximate Counting:

Setting:

- We are given a huge number of items (co-variates) $i \in I$ to track over time $t \in \{1, 2, ..., T\}$. T can be large as well.
- We only see increments (i, t, v), the increment v to item i at time t.

Goal: In limited space (hopefully $O(\log |I| \times T)$), we want to

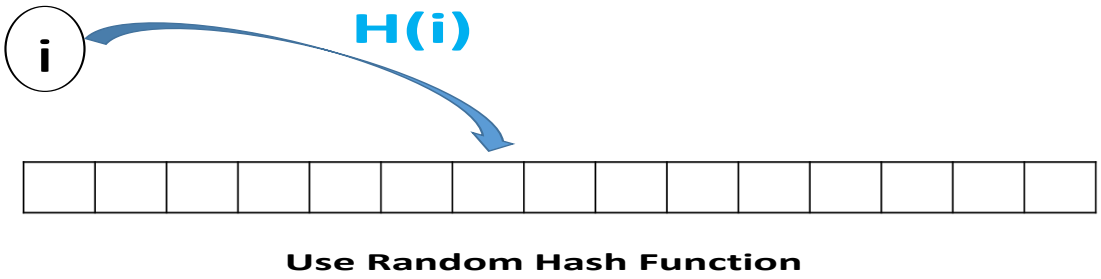
- **Point Queries:** Estimate the counts (increments) of item i at time t.
- **Range Queries:** Estimate the counts (increments) of item i during the given range $[t_1, t_2]$.

Classical Sketching: Count-Min Sketch (CMS), Lossy Counting, etc.

Basic Idea behind Sketching.

Randomly assign items to a small number of counters.

- It works! AMS 85, Moody 89, Charikar 99, MuthuKrishnana 02, etc.
- If no collisions, counts exact.

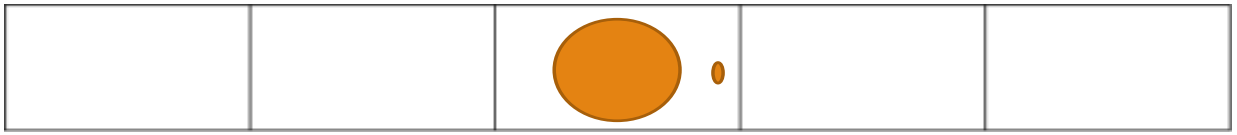


Handling Time:

- Treat each pair (i, t) (item, time) as different item.
- Hash pairs (i, t), instead of just items.
- Time only increases the number of items to $|I| \times T$.

What happens during Collision ?

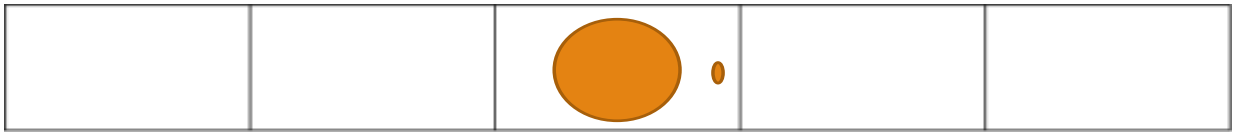
The Good



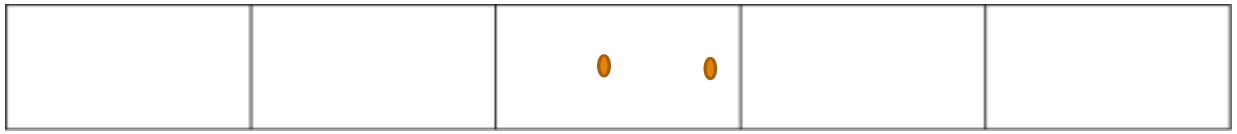
We typically care about heavy hitters.

What happens during Collision ?

The Good



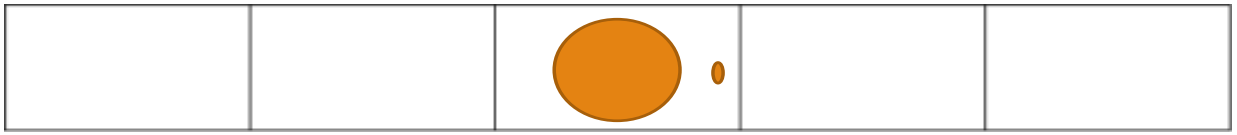
The Irrelevant



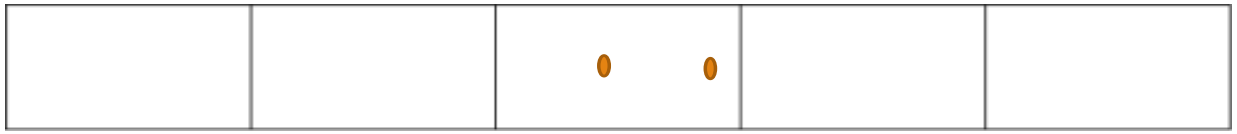
We typically care about heavy hitters.

What happens during Collision ?

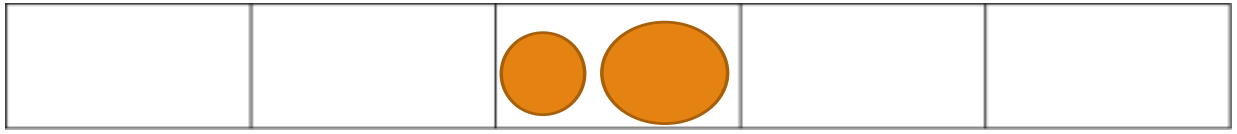
The Good



The Irrelevant



The Unlucky

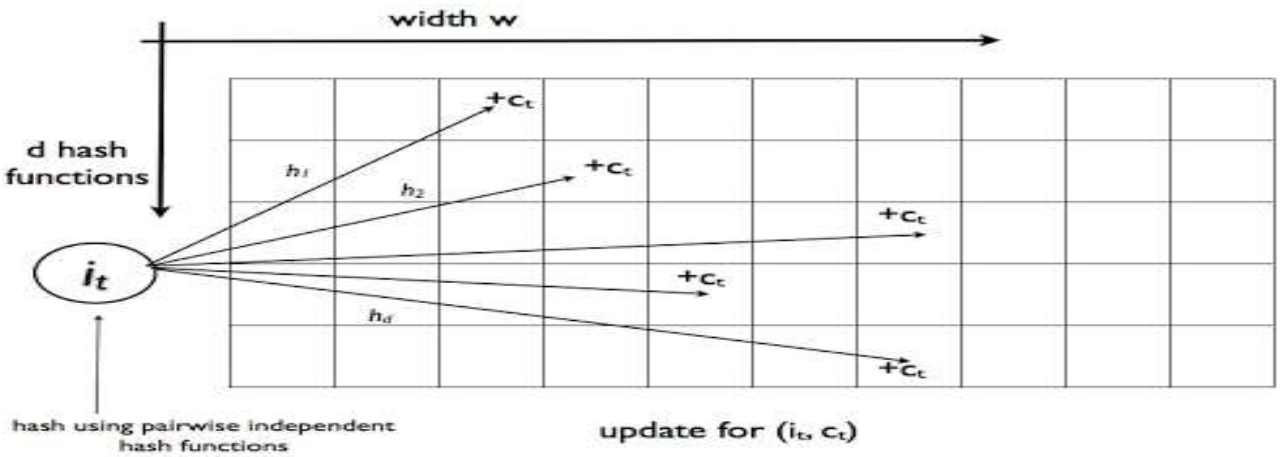


We typically care about heavy hitters.

Maximizing Luck : Count-Min Sketch (CMS)

Idea:

- We always overestimate, if unlucky, by a lot.
- Repeat independently d times and take minimum of all overestimates.
- Unless unlucky all d times, it will work. ($d = \log \frac{1}{\delta}$, $w = \frac{1}{\epsilon}$)



Theoretical Guarantee

- $c \leq \hat{c} \leq c + \epsilon \mathcal{M}^T$ with probability $1 - \delta$, where $\mathcal{M}^T$ is sum of all counts in the stream.
- Space $O(\log |I| \times T)$

New Requirement: Time Adaptability

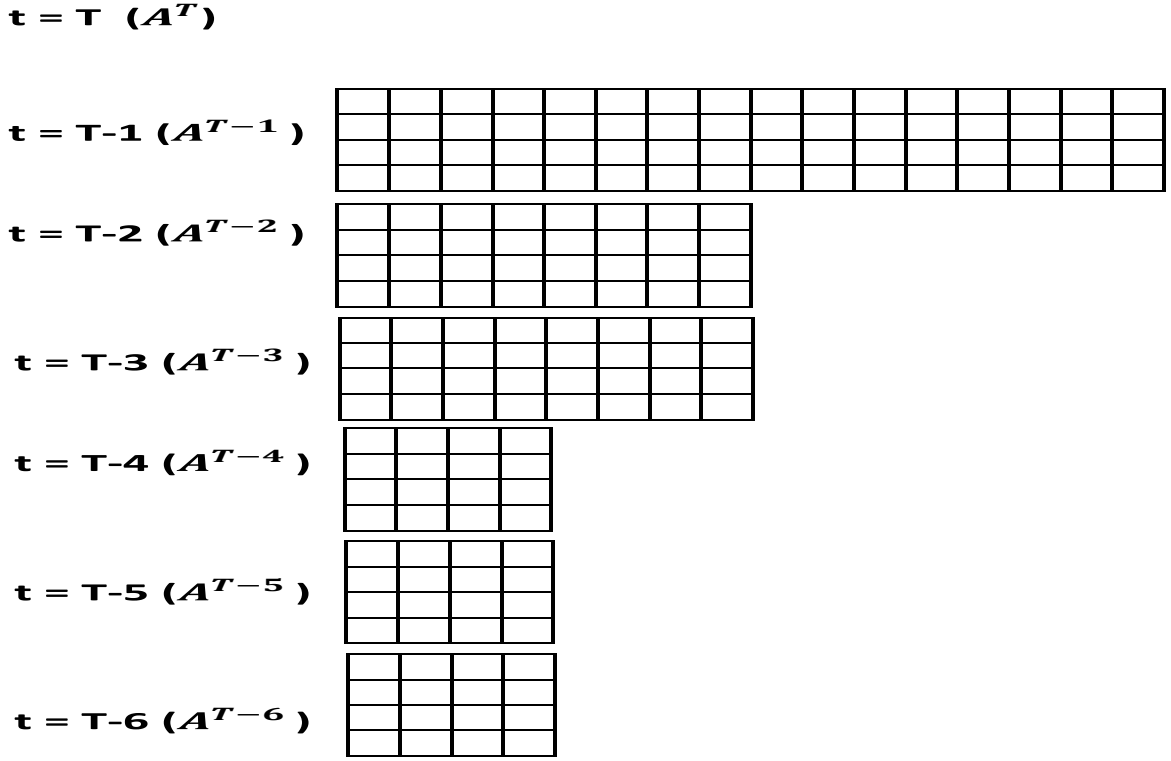
In Practice:

- Recent trends are more important.
- A burst in the number of clicks in the past few minutes more informative than similar burst last month.

Expectation: Time Adaptive Counting.

- Classical sketches do not take temporal effect into consideration.
- **Smart Tradeoff:** Given the same space, trade errors of recent counts with that of older ones.
- Like our memory, forget slowly.

Existing Solution: Hokusai[1]

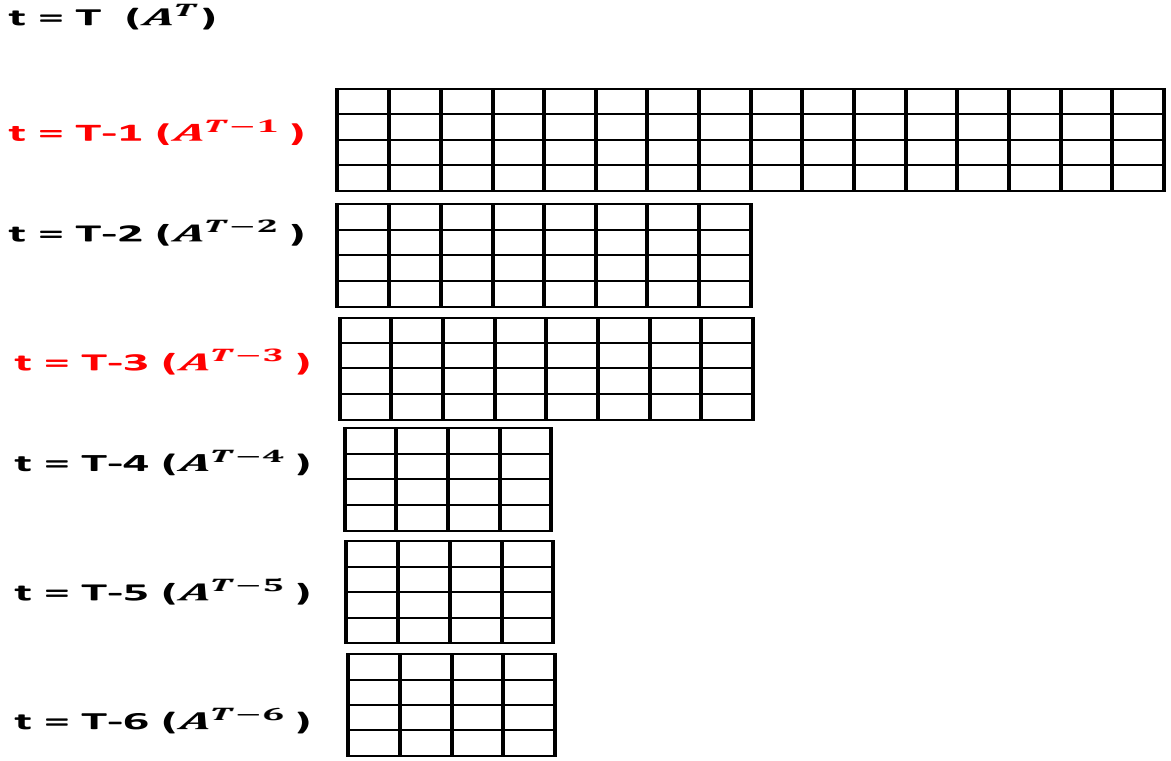


Idea: Disproportionate allocation over time.

- Accuracy of CMS dependent on memory allocated.
- More space for recent sketches and less for older.
- Keep a CMS sketch for every time. Shrink sketch size on fly.
  Clever Idea: Exploit Rollover.

[1] Matusevych, Smola and Ahmad 2012

Existing Solution: Hokusai[1]

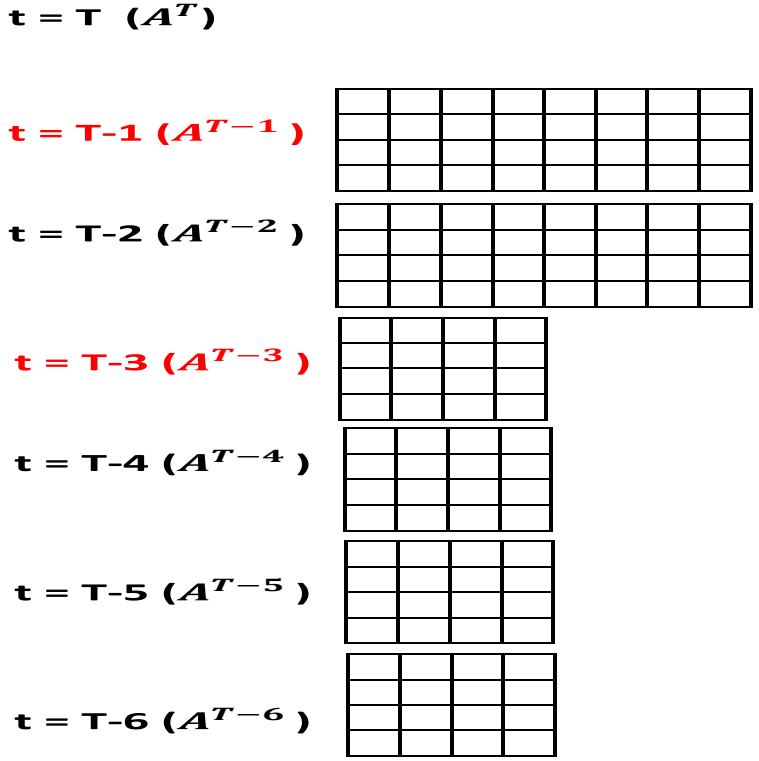


Idea: Disproportionate allocation over time.

- Accuracy of CMS dependent on memory allocated.
- More space for recent sketches and less for older.
- Keep a CMS sketch for every time. Shrink sketch size on fly.
  Clever Idea: Exploit Rollover.

[1] Matusevych, Smola and Ahmad 2012

Existing Solution: Hokusai[1]

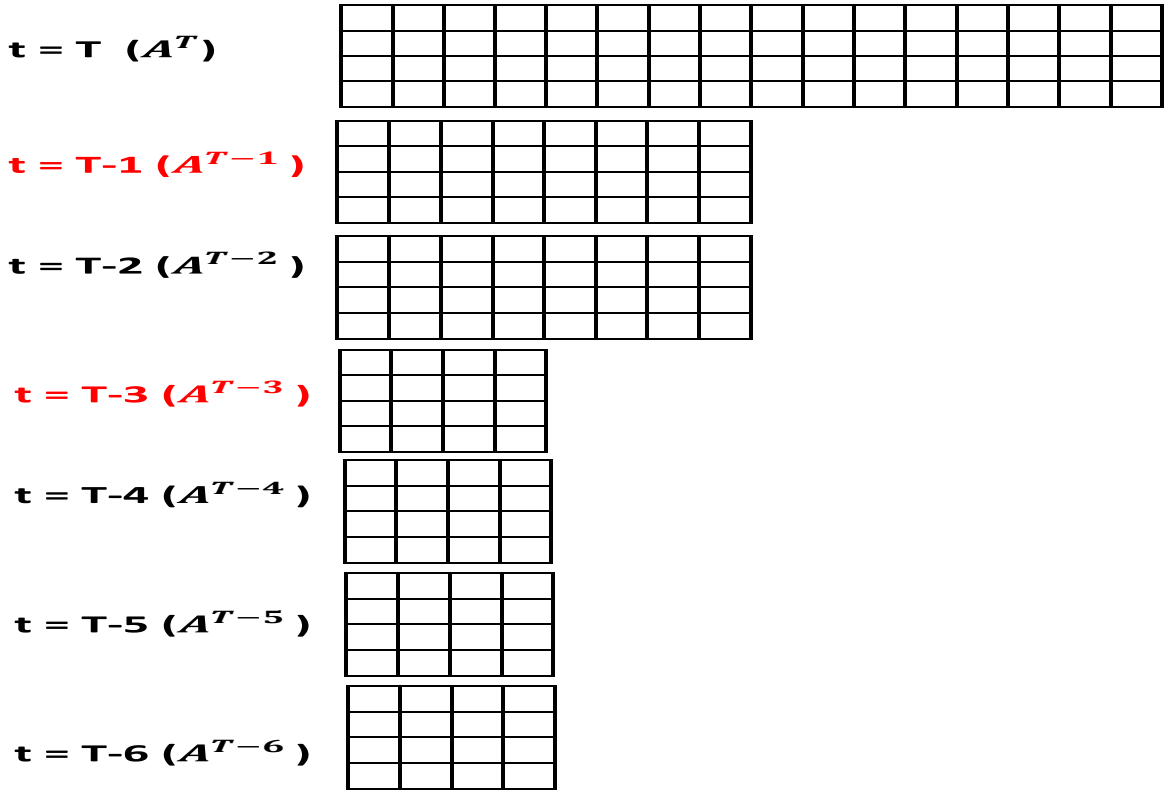


Idea: Disproportionate allocation over time.

- Accuracy of CMS dependent on memory allocated.
- More space for recent sketches and less for older.
- Keep a CMS sketch for every time. Shrink sketch size on fly.
  Clever Idea: Exploit Rollover.

[1] Matusevych, Smola and Ahmad 2012

Existing Solution: Hokusai[1]



Idea: Disproportionate allocation over time.

- Accuracy of CMS dependent on memory allocated.
- More space for recent sketches and less for older.
- Keep a CMS sketch for every time. Shrink sketch size on fly.
  Clever Idea: Exploit Rollover.

[1] Matusevych, Smola and Ahmad 2012

Problems with Hokusai

Issues:

- Discontinuity. If time t is empty, we still have to shrink sketch size for older times.
- O(T) memory. One for each t.
- Shrinking overhead. Shrink $\log t$ sketches for every transition from t to $t + 1$.
- No flexibility.

Detour: Dolby Noise Reduction (1960s)

High Level View

- In digital recording, the music signal compete with tape hiss (background noise).
- if Signal to Noise (SNR) ratio is high, the recording is noise free.
- While recording the frequencies in the music is artificially inflated (Pre-Emphasis).
- During playback a reverse transformation is applied which cancels pre-emphasis. (De-Emphasis)
- Overall effect of noise is minimized.

Proposal: (Adaptive)Ada-Sketches

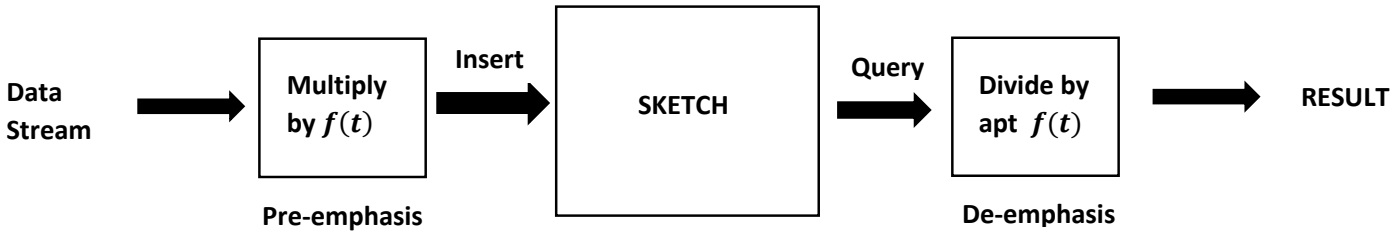
Analogy with Dolby Noise Reduction:

- Sketches preserves heavier counts more accurately.
- Artificially inflate recent counts (Pre-emphasis).
- Inflated counts will be preserved with less error.
- Deflate by exact same amount during estimation. (De-emphasis)

Proposal: (Adaptive)Ada-Sketches

Analogy with Dolby Noise Reduction:

- Sketches preserves heavier counts more accurately.
- Artificially inflate recent counts (Pre-emphasis).
- Inflated counts will be preserved with less error.
- Deflate by exact same amount during estimation. (De-emphasis)



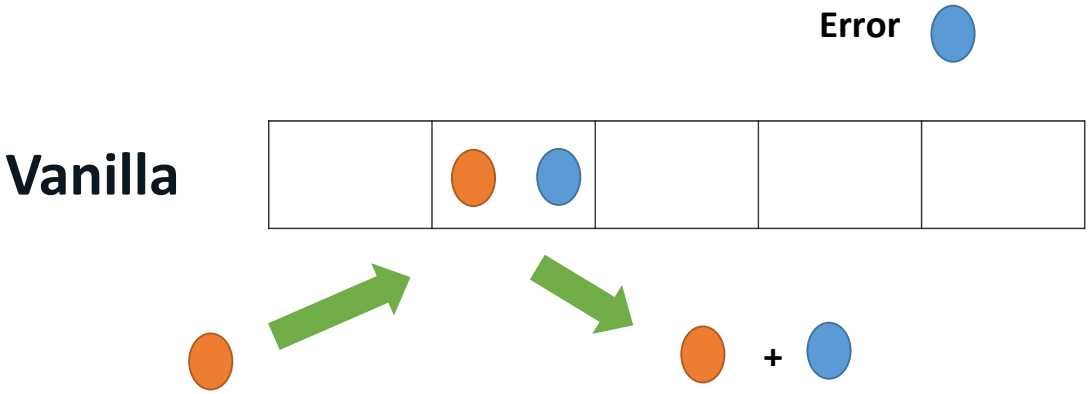
Proposal

- Let $f(t)$ be any (pre-defined) monotonically increasing sequence. ($f(t)$ can be chosen wisely)
- Multiply the count of (i, t) with $f(t)$ and then add to the sketch.
- While querying (i,t), get the estimate and divide by $f(t)$

Why it works ?

Observation

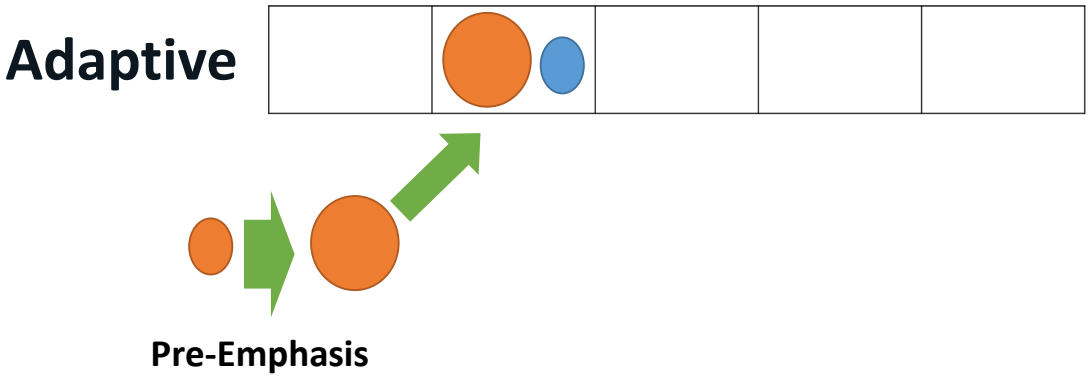
- If no collision then exact.
- During collision, errors or recent counts decrease due to greater de-emphasis.



Why it works ?

Observation

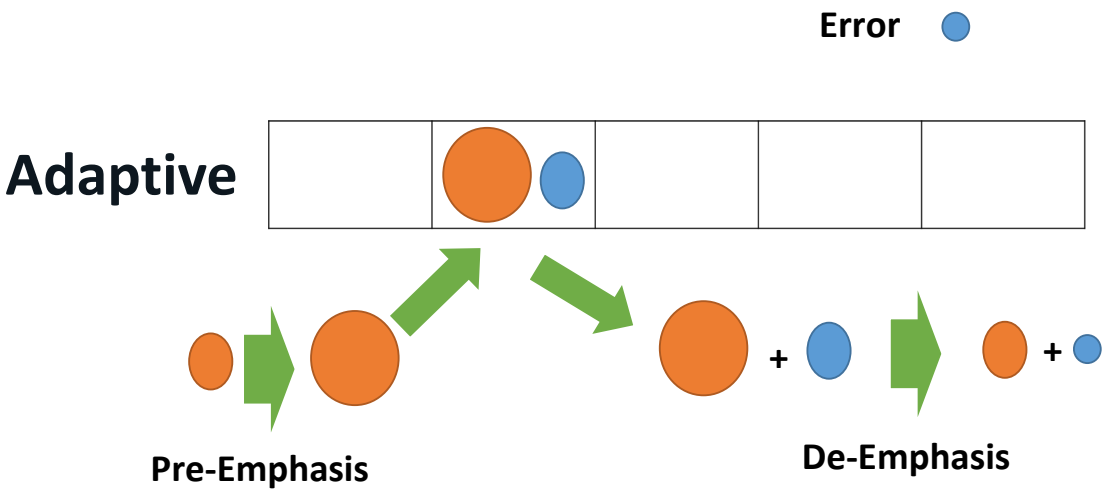
- If no collision then exact.
- During collision, errors or recent counts decrease due to greater de-emphasis.



Why it works ?

Observation

- If no collision then exact.
- During collision, errors or recent counts decrease due to greater de-emphasis.



Advantages

- No Discontinuity. If time t is empty, no addition, no extra collisions, no extra errors.
- $O(\log |I| \times T)$ memory just like CMS.
- No shrinking overhead. Minimum modification to CMS. (Strict Generalization)

Provable Time Adaptive Guarantees

Theorem

For $w = \lceil \frac{e}{\epsilon} \rceil$ *and* $d = \log \frac{1}{\delta}$, *given any* (i, t) *we have*

$$c_i^t \leq \widehat{c_i^t} \leq c_i^t + \epsilon \beta^t \sqrt{\mathcal{M}_2^T}$$

with probability $1 - \delta$. *Here* $\beta^t = \frac{\sqrt{\sum_{t'=0}^{T} (f(t'))^2}}{f(t)}$ *is the time adaptive factor monotonically decreasing with* t.

More..

Works with any Sketching Algorithm

- Adaptive Count Sketches, Adaptive Lossy Counting etc.
- Provable Time Adaptive Guarantees for all of them.

More..

Works with any Sketching Algorithm

- Adaptive Count Sketches, Adaptive Lossy Counting etc.
- Provable Time Adaptive Guarantees for all of them.

Flexibility in Choice of $f(t)$

- Any monotonic $f(t)$ works. Can be tailored
- Upper bound dependent on $\beta^t = \frac{\sqrt{\sum_{t'=0}^{T}(f(t'))^2}}{f(t)}$.
- Fine control over the error distributions.

Experiments: Accuracy for a given Memory

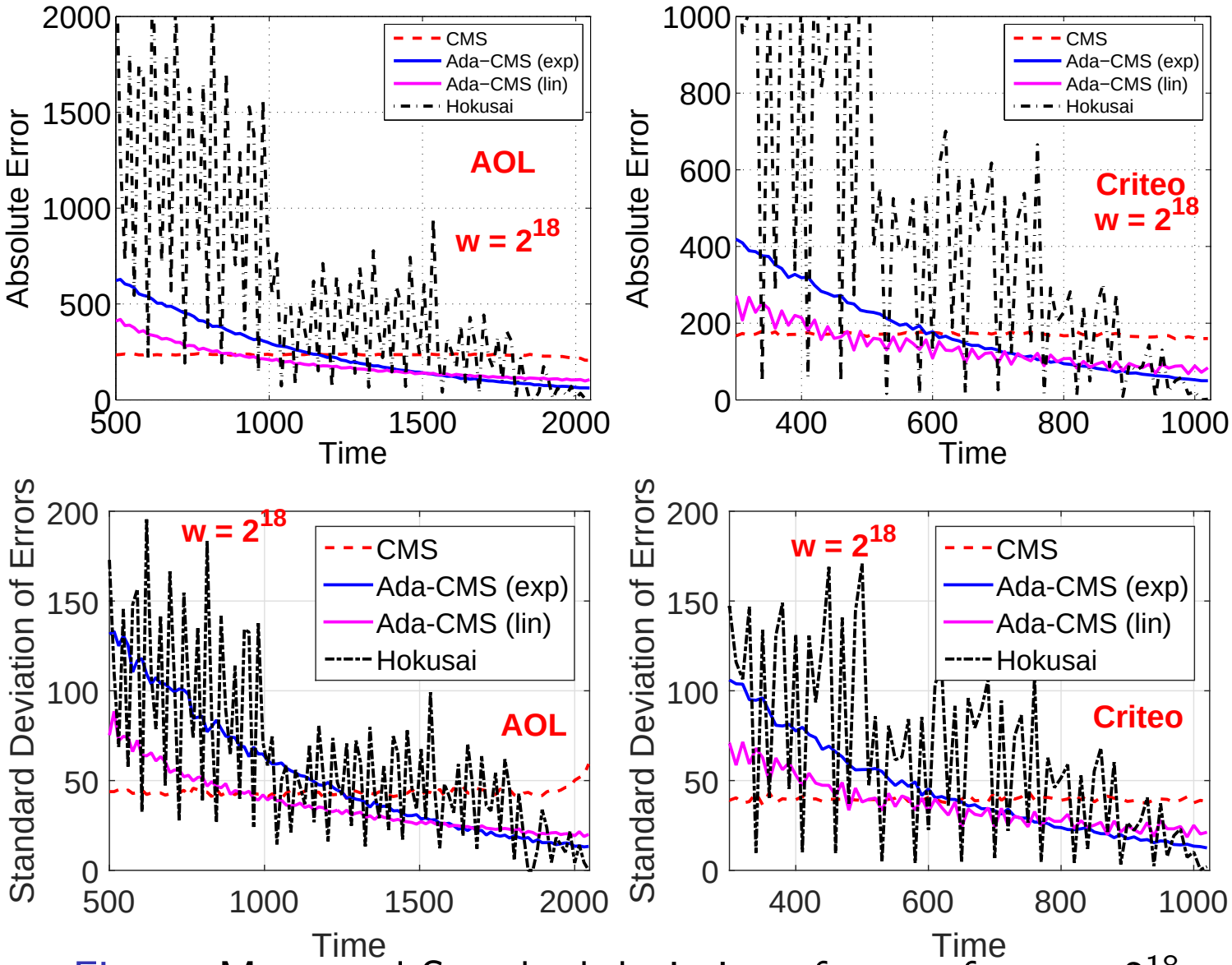


Figure: Mean and Standard deviation of errors for $w = 2^{18}$.

Scalability: Throughput

Table: Time in sec to summarize AOL dataset

	2^{20}	2^{22}	2^{25}	2^{27}	2^{30}
CMS	44.62	44.80	48.40	50.81	52.67
Hoku	68.46	94.07	360.23	1206.71	9244.17
ACMS (lin)	44.57	44.62	49.95	52.21	52.87
ACMS (exp)	68.32	73.96	76.23	82.73	76.82

Table: Time in sec to summarize Criteo Dataset

	2^{20}	2^{22}	2^{25}	2^{27}	2^{30}
CMS	40.79	42.29	45.81	45.92	46.17
Hoku	55.19	90.32	335.04	1134.07	8522.12
ACMS (lin)	39.07	42.00	44.54	45.32	46.24
ACMS (exp)	69.21	69.31	71.23	72.01	72.85

Summary

- Randomized counting algorithms are very powerful for massive data streams.
- Pre-emphasis and De-emphasis idea give more control over distributing errors more smartly.
- Adaptive Sketches are strict generalization of the existing sketching algorithms, which unify the tradeoffs between memory, accuracy and time adaptability in one framework.
- We see significant improvements both in accuracy and speed by smartly balancing this tradeoff.
- The final algorithm is principled with provable guarantees, and it can be easily tailored to incorporate a variety of application dependent constraints.