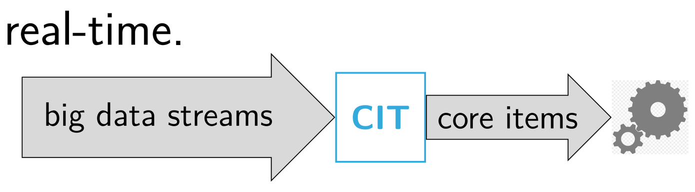
Continuously Tracking Core Items in Data Streams with Probabilistic Decays

Junzhou Zhao[1] Pinghui Wang[1] Jing Tao[1] Shuo Zhang[1] John C.S. Lui[2]

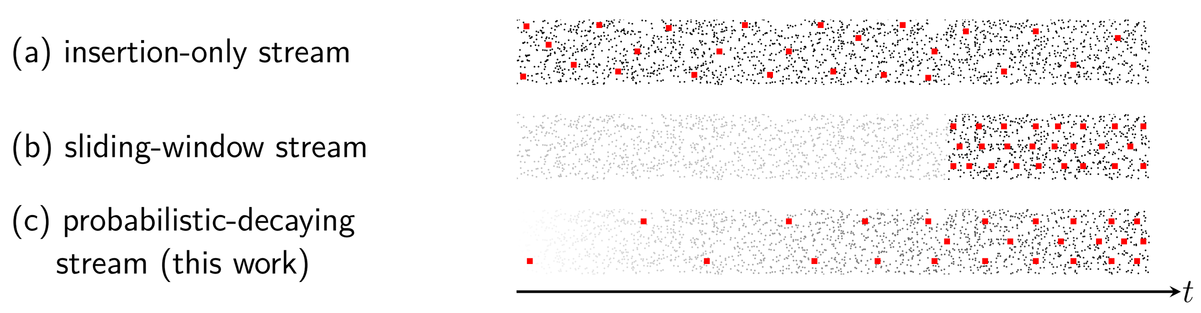
[1]Xi'an Jiaotong University [2]The Chinese University of Hong Kong

Background & Motivation

- Data streams are ubiquitous:
  - email stream, tweets stream, news stream, network traffic stream, etc
  - geo-location stream generated by taxis, IoT devices, LBSNs, etc
  - user consuming record stream from Amazon, Taobao, etc
- Applications:
  - real-time trending topic detection
  - network security monitoring
  - online collaborative filtering
- However, their high speed and large volume cause troubles.
- **Core Items**: informative or representative items in a data stream.
- **Core Items Tracking (CIT)**: a streaming algorithm that can continuously track core items in a data stream in real-time.

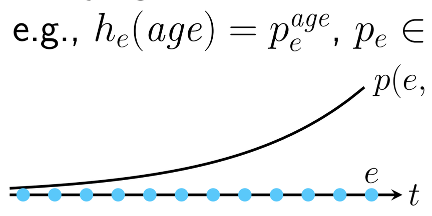


- The right to be forgotten:



Problem Formulation

- Utility Function: measuring the informativeness of a set of items: $f\colon 2^V \mapsto \mathbb{R}_{\geq 0}$
- Monotonicity: $f(S) \leq f(T), \forall S \subseteq T \subseteq V$.
- Submodularity: $f(S \cup \{e\}) - f(S) \geq f(T \cup \{e\}) - f(T), \forall S \subseteq T \subseteq V, e \in V$.
  - aka the *dimension return* property [Nemhauser et al. 1978]
- **Probabilistic-Decaying Stream (PDS) model**:
  - At time t, an item e arrived at time $t_e \leq t$ participates in analysis with probability $p(e,t) = h_e(t - t_e)$
  - $h_e\colon \mathbb{Z}_{\geq 0} \mapsto [0,1]$ is an item-specific decaying function.
  - $h_e(age)$ decreases as age increases, e.g., $h_e(age) = p_e^{age}$, $p_e \in (0,1)$.



- **The Core Items Tracking (CIT) problem**:
  - **Given** a monotone submodular utility function f, a PDS with item-specific decaying function h_e, and a budget $k > 0$
  - **Want** to find a subset $S_t^* \subseteq V$ at any query time t, s.t.

$$S_t^* = \operatorname*{arg\,max}_{S \subseteq V \wedge |S| \leq k} \mathbb{E}_{h_e}[f(S)|\mathcal{D}_t]$$

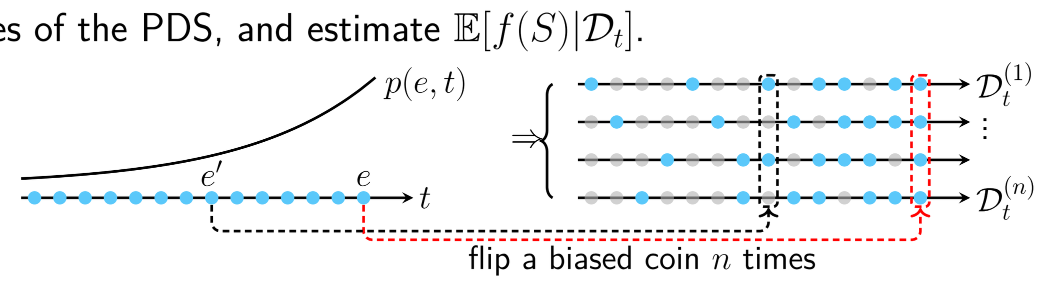
where $\mathcal{D}_t \triangleq \{e\colon t_e \leq t\}$ denotes the items arrived before t.

A Monte-Carlo Framework

- Expensive to calculate $\mathbb{E}[f(S)|\mathcal{D}_t]$ exactly
  - need to consider the participation possibility of each item in S, e.g.,

$$\mathbb{E}[f(\{a,b\})|\mathcal{D}_t] = \underbrace{p(a,t)p(b,t)f(\{a,b\})}_{\text{both } a \text{ and } b \text{ participate in the analysis}} + \underbrace{p(a,t)(1-p(b,t))f(\{a\})}_{\text{only } a \text{ participates in the analysis}} + \underbrace{(1-p(a,t))p(b,t)f(\{b\})}_{\text{only } b \text{ participates in the analysis}}$$

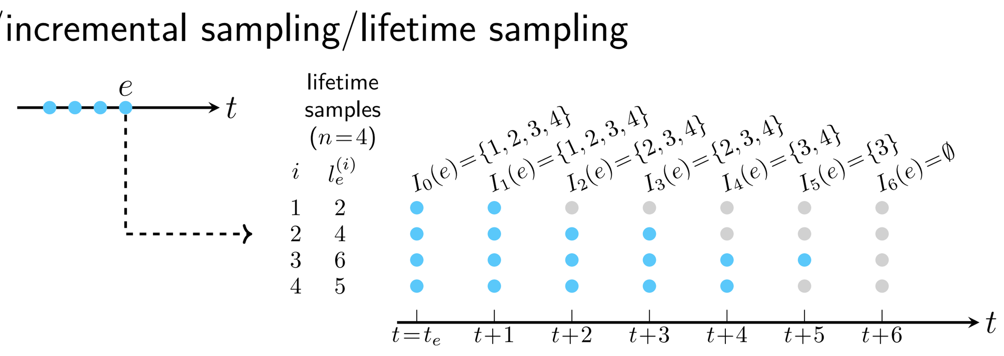
  - exactly calculating $\mathbb{E}[f(S)|\mathcal{D}_t]$ requires $O(2^{|S|})$ oracle calls.
- **Monte-Carlo Approximation**:
  - Generate n samples of the PDS, and estimate $\mathbb{E}[f(S)|\mathcal{D}_t]$.



  - By Monte-Carlo approximation, we have

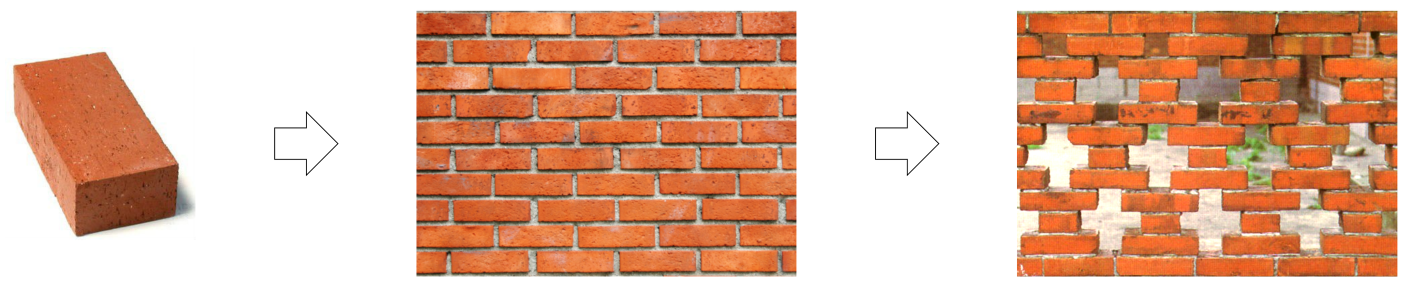
$$F(S) \triangleq \frac{1}{n}\sum_{i=1}^{n} f(S \cap \mathcal{D}_t^{(i)}) \xrightarrow{a.s.} \mathbb{E}[f(S)|\mathcal{D}_t], \quad n \to \infty.$$

  - The number of oracle calls reduces from $O(2^{|S|})$ to $O(n)$.
  - $F(S)$ is still monotone and submodular.
- Maintaining data stream samples:
  - naive sampling/incremental sampling/lifetime sampling



Algorithms

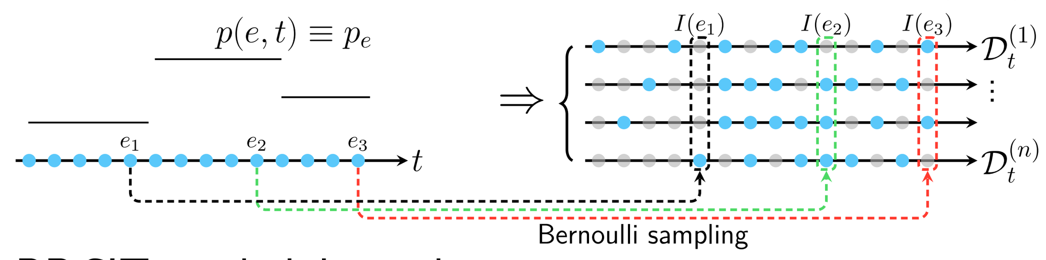
- Overview



PNDCIT PDCIT PDCIT+

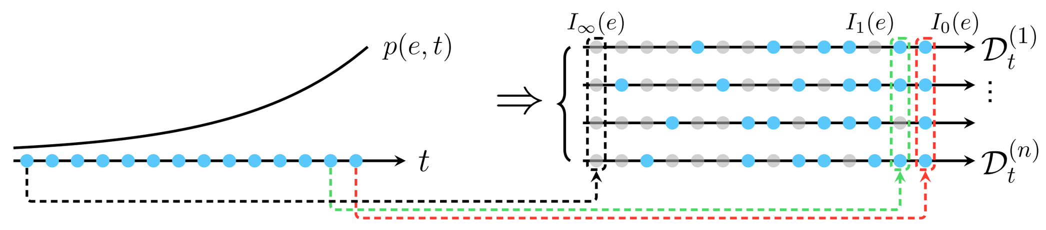
Algorithm	Update Time	Memory	Approximate Ratio
PNDCIT	$O(n\epsilon^{-1}\log k)$	$O(nk\epsilon^{-1}\log k)$	$1/2-\epsilon$
PDCIT	$O(Ln\epsilon^{-1}\log k)$	$O(Lnk\epsilon^{-1}\log k)$	$1/2-\epsilon$
PDCIT+	$O(n\epsilon^{-2}\log^2 k)$	$O(nk\epsilon^{-2}\log^2 k)$	$1/4-\epsilon$

- PNDCIT: probabilistic non-decaying case, i.e., $p(e,t) \equiv p_e$

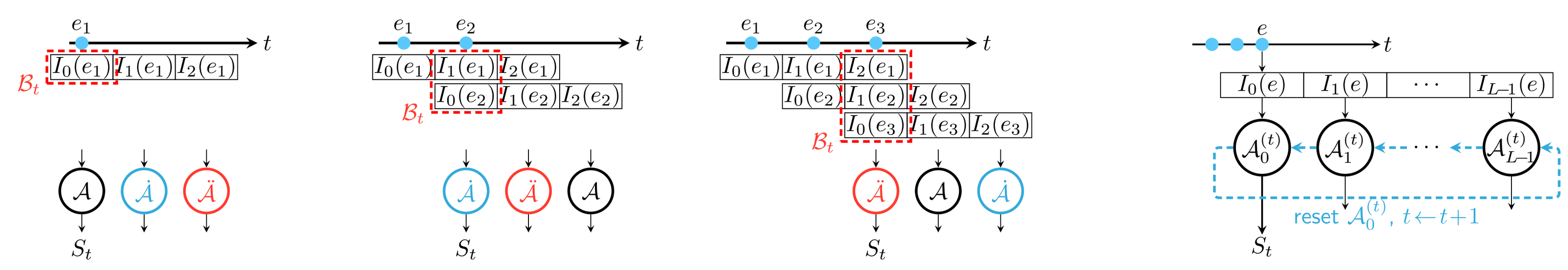


$$I(e_1) = [0,0,0,1]^T \quad I(e_2) = [0,1,1,1]^T \quad I(e_3) = [1,0,1,0]^T$$

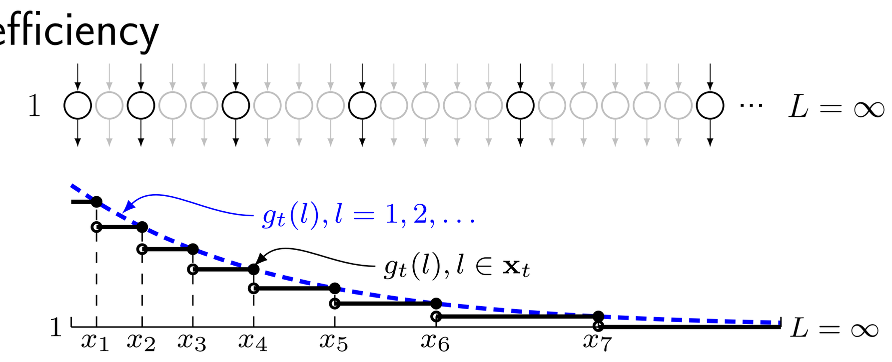
- PDCIT: probabilistic decaying case



$$I_0(e) = [1,1,1,1]^T \quad I_1(e) = [1,1,0,1]^T \quad I_\infty(e) = [0,0,0,0]^T$$



- PDCIT+: improve efficiency

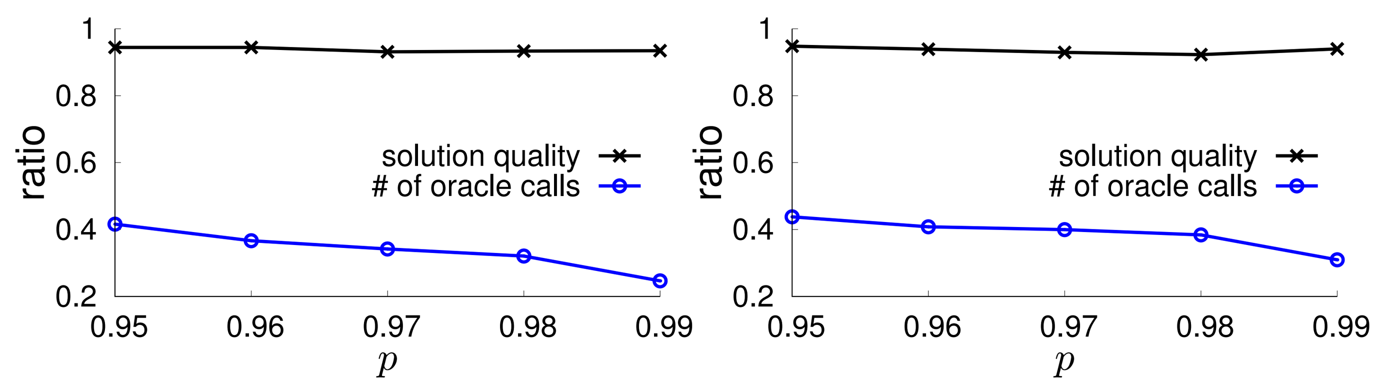


Experiments

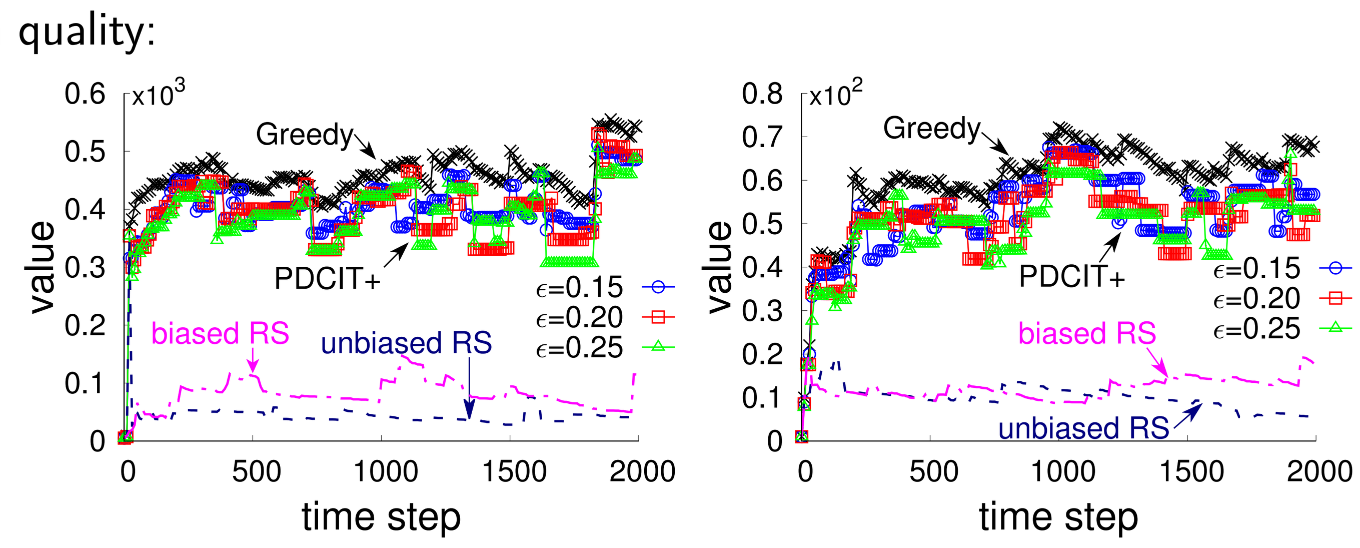
- Data

data stream	item	length	time period
DBLP	author	$371,690$	1936 - 2018
MemeTracker	article	$714,072$	1/2009 (one month)
math.StackExchange	question	$955,284$	7/2010 - 6/2018
StackOverflow	question	$2,904,450$	1/2015 - 3/2016

- **Goal:** maintain k most representative items that jointly have the maximum coverage, i.e., $f(S) = |\cup_{e\in S} e|$.
- PDCIT vs PDCIT+:



- Solution quality:



- Scalability: