SE ZG522: Big Data Systems

04 Analytics on Data Stream

Dr. Kamlesh Tiwari
Assistant Professor, Department of CSIS,
BITS Pilani, Pilani Campus, Rajasthan-333031 INDIA

Feb 19, 2022 ONLINE (WILP @ BITS-Pilani Jan-May 2022)

http://ktiwari.in/bds

Frequent items over data stream

- Let identity of items is drawn from the set $\{1, 2, 3, ..., n\}$.
- Frequency of item i be f_i
- Assume general arrival model, (i, v), $v > 0$ represents arrival and $v < 0$ is departure.
- Sum of frequencies $m = \sum_i f_i$ represent size of data stream
- **Frequent** item i, have **frequency** $f_i > m/(k+1)$ for some fixed k

Observations

- There could be at most k frequent items
  (why ? proof?) $m > k(k+1)$
- Any algorithm that finds all frequent and only frequent items requires at least $\log(\binom{n}{k})$ bits
  (how? $2^s \geq \binom{n}{k}$)

In action: Frequent items

Take $k = 4$, and consider following data stream

5	8	4	5	4	12	$\rightarrow$	
	6	5	2	8	3	5	$\rightarrow$
		4	5	4	12	6	13

Insert x in data structure

1. **IF** (A.ismember(x)) A[x]++
2. **ELSE** A.insert(x)
3. **IF** (A.size == k+1) **THEN** $\forall\ y \in A$
4. A[y]−−,
5. **IF** (A[y] ==0) A.delete(y);

Let us step by step execute the algorithm:

Index	Item	Frequency
1		
2		
3		
4		
5		

Data streams

Consider, stream of data. Where data in arriving in rapid succession. Re-scan is NOT possible. Even storage space is insufficient to accommodate all data points.

Without storing all the data one wish to estimate

- Set of frequent items
- Number of distinct items
- Frequent itemsets
- *etc*

Example: In continuous update model if you get n-1 numbers from first n Natural numbers, without repetition, in an arbitrary order. Can you report the missing one?

[Constraints are on memory and processing power]

Find approx frequent items

Wish to output a list of items such that

1. Every item in the list has frequency $f_i > (1-\epsilon)\frac{m}{k+1}$
2. All the items having frequency at least $(1+\epsilon)\frac{m}{k+1}$ is in the list

Output should satisfy above two properties with probability $(1-\delta)$

Algorithm maintains a data structure A over the stream.
Step to update an item x is as below

1. **IF** (A.ismember(x)) A[x]++
2. **ELSE** A.insert(x)
3. **IF** (A.size == k+1) **THEN** $\forall\ y \in A$
4. A[y]−−,
5. **IF** (A[y] ==0) A.delete(y);

Thank You!

Thank you very much for your attention!

Queries ?