Lecture 4. Distributed sketching. Graph problems

Ricard Gavaldà

MIRI Seminar on Data Streams, Spring 2015

Contents

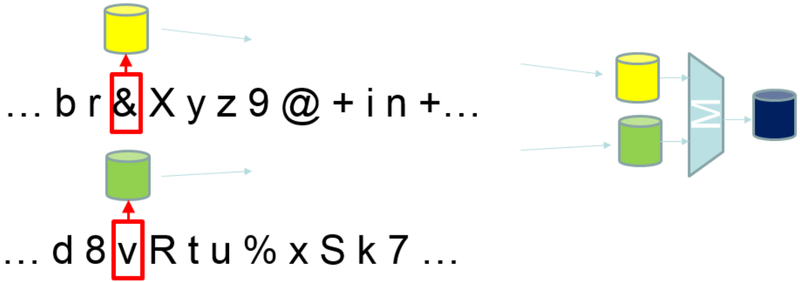
Distributed sketching

Distributed streams

Setting:

- Many sources generating streams concurrently
- No synchrony assumption
- Want to compute global statistics
- Streams can send short summaries to central

Merging sketches



Send the sketches, not the whole stream

Merging sketches

Mergeability

A sketch algorithm is mergeable if

- given two sketches $S1$ and $S2$ generated by the algorithm on two data streams $D1$ and $D2$,
- one can compute a sketch S that answers queries correctly with respect to any interleaving of $D1$ and $D2$

Note: For frequency problems, all interleavings of $D1$ and $D2$ should give the same query results. So for any interleaving = for one interleaving

For non-frequency problems, you give the "right" definition on a need basis

Merging sketches

CM-Sketch = Matrix $d \times w$, set h of d hash functions

Invariant:

$$CMS_h(S)[i,j] = \sum_{x \,:\, h_i(x)=j} \mathrm{freq}_S(x)$$

Merging sketches

Fact: Let S_1, S_2 be two streams. Let S be any interleaving of S_1 and S_2. Then for any h

$$CMS_h(S) = CMS_h(S_1) + CMS_h(S_2)$$

where $+$ is plain matrix addition

To distribute:

- At day 0, agree on common set of hash functions h
- Daily, sites send their $d \times w$ matrices to central
- Central adds them all; ready to answer queries

Merging sketches

Inner product sketch:

- Agree on common hash functions and bits b_i
- Keep $S_1 = \sum_i u_i b_i$ and $T_1 = \sum_i v_i b_i$
- Same for S_2 and T_2
- Merging: $S = S_1 + S_2$ and $T = T_1 + T_2$

Merging sketches

Min-based sketches: Cohen, HyperLogLog

- Store the min of some stream statistics
- The min on $D1 \cdot D2$ is is the min of both mins
- We keep r copies: sketch signature is $(min_1, \ldots, min_r)$
- Merging in time r

Merging sketches

Exercise 1

Show that space-saving is efficiently mergeable:

> Given two space-saving sketches of size k for two data streams, say how to merge them into a new size k space-saving sketch that fits their concatenation

Time should be $O(k)$

An application: distance distributions in large graphs

Distance distributions

For a directed graph $G = (V, E)$ and $u \in V$, neighborhood & distance functions

- $B(u, t)$ = set of vertices at distance $\leq t$ from u
- $N(u, t) = |B(u, t)| - |B(u, t-1)|$
- $N(t) = \sum_v N(u, t)$ = number of pairs (u, v) at distance t

Useful:

- $N(1)$ gives average degree
- $N(2)$ gives e.g. clustering coefficients
- max nonzero $N(t)$ gives diameter, etc.

Computing distance distributions

Traditional algorithm:

1. For each v, $B(v,0) = \{v\}$
2. For $t = 1, 2, \dots$

   for each $v \in V$,

   $B(v,t) = B(v,t-1)$

   for each $(v,u) \in E$, $B(v,t) = B(v,t) \cup B(u,t-1)$
3. For $t = 1, 2, \dots$

   output $N(t) = \sum_v |B(v,t)| - N(t-1)$

Problems:

- Random access to edges. Disk faults
- Memory $|V|^2$ in connected graphs, even if $|E| \ll |V|^2$

Computing distance distributions

Required:

- Access edges sequentially, as they are stored in disk
- Work well on small-world, sparse graphs
- Memory linear (or little more) in number of vertices
- Limited number of passes

HyperANF

ANF [Palmer,Gibbons,Faloutsos02]: Memory $O(n \log n)$

- Graph with 2 billion links $\rightarrow$ 30 minutes on 90 machines

HyperANF [Boldi,Rosa,Vigna11]: Memory $O(n \log \log n)$

- 15 minutes on a laptop

HyperANF

Key observation 1:
We eventually need only $|B(v,t)|$, not $B(v,t)$ itself

Key observation 2:
$|B(v,t)|$ is the number of distinct elements connected by one edge to nodes in $B(v,t-1)$

Key observation 3:
Hyperloglog keeps number of distinct elements, and can implement unions (is mergeable)

HyperANF

Keep a hyperloglog counter $H(v)$ for each $v \in V$. Then

$$B(v,t) = B(v,t-1)$$
for each $(v,u) \in E$, $B(v,t) = B(v,t) \cup B(u,t-1)$

$\rightarrow$

$$H'(v) = H(v)$$
for each $(v,u) \in E$, $H'(v) = \text{merge}(H'(v), H(u))$

Big Win: this can be done while reading edges sequentially!

HyperANF

Other ideas:

- Broadword programming: HLL registers are shorter than machine words. Pack several in a word and use bitwise operations to speedup merging
- Try to work only on changed counters. Large savings near the end, when most counters have stabilized.
- Systolic computation: A modified counter *signals* its predecessors that they must update

HyperANF: applications

Distinguishing Web-like and social-network-like networks [Boldi+11]

- Shortest-path-index of dispersion (SPID): Variance-to-mean ratio of distances
- < 1 for social networks, > 1 for web-like networks

Diameter of the Facebook graph [Backstrom+11]

- 720M active users, 69B friendship links
- Average distance is 4.74 (= 3.74 degrees of separation)
- 92% of users are at distance ≤ 5
- 10 hours on 256Gb RAM machine

Other graph problems studied in streaming

- Counting triangles (related to clustering coefficients, community finding ...)
- Counting arbitrary subgraphs (motifs, etc.)
- Computing pagerank
- Clustering (e.g., k-center)
- Computing spanning forests
- Graph partitioning