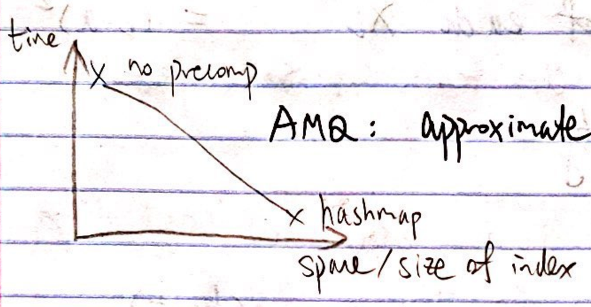
2/11/21 Hash-Based Data Structures

Membership query

① no precomputation is allowed O(N) time

② precomputation is allowed : hashmap O(1) time , O(N) space



AMQ: approximate membership query

AMQ Data Structure: Bloom Filter

Data item s

Hash functions h_i : $\{h_1, \cdots, h_k\}$

h : set elm → $\{0, \cdots, m-1\}$

item s → signature $[h_1(s)\ h_2(s)\ \cdots\ h_k(s)]$

binary vector of size m indicating that at least 1 hash ~~func~~ func has that value

sig(s) → m-dim binary vector has at most k ones

A set of items $S = \{s_1, \cdots, s_N\}$

$Sig(S) = \text{BitWise OR}_{i=1}^{N} \{sig(s_i)\}$

Ex. i → new item ; Sig(S) ; sig(i)

If sig(~~s~~i) is 1 somewhere where Sig(S) is 0

⇒ S does not contain i

Sig(S) can reject values not in the set

If sig(i) is 0 somewhere where sig(S) is 1

⇒ S might contain i

The 1 could've been set by another item

If Sig(i) has a 1 and Sig(S) has a 1 in the same spot

$\Rightarrow$ S might not contain i

$\because$ Could've been set by some other elements

BitwiseOR sigs : no false negatives, possibly false positives

usage : partition

vs $O(N)$ hashmap

Analysis $\epsilon = (1-e^{-\frac{kN}{m}})^k$ # of hash funcs $O(m)$ space

False positive rate & k & m ← size of bloom filter

① Each hash func : $\frac{1}{m}$ prob. to set a bit to 1

② Over k applications to a single data point, prob. of a given bit not 1 ?

$p = (1-\frac{1}{m})^k$

③ $e^{-1} = \lim_{m \to \infty} (1-\frac{1}{m})^m \Rightarrow p = (1-\frac{1}{m})^k \approx e^{-\frac{k}{m}}$

(ex. in database)

④ N datapoints : $p^N = e^{-\frac{kN}{m}}$

over k applications to N data points, prob. that a given bit not 1

⑤ A False Positive happens when all bits of a new item are 1

$(1-p^N)^k = \boxed{(1-e^{-\frac{kN}{m}})^k} = \epsilon$: false positive rate

space of data struct

Q How do we set m and k ? choose k to minimize FPR

$\varepsilon = \gamma^k$, $\gamma = (1-e^{-\frac{kN}{m}})$, want γ to have high entropy, i.e. $\gamma = \frac{1}{2}$

$\Rightarrow k = \frac{M}{N} \ln 2$

Q How to use a signature to estimate the number of input items ?

$E(\hat{r})$ = frac of bits that are 1

? $N = \frac{-m \log(1-E(\hat{r}))}{k}$