Δομές Δεδομένων και Αλγόριθμοι

Χρήστος Γκόγκος

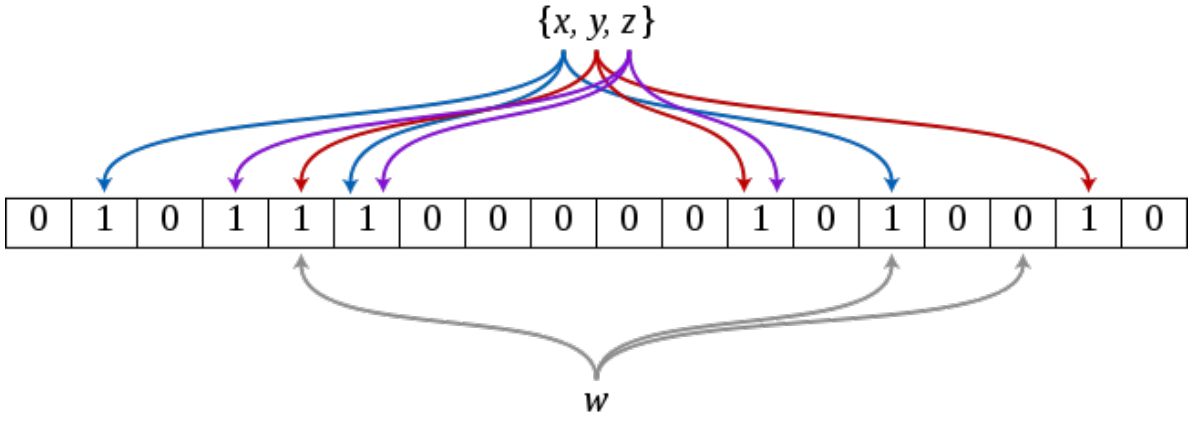
ΤΕΙ Ηπείρου

Χειμερινό Εξάμηνο 2014-2015
Παρουσίαση 21. Bloom Filters

Bloom Filters

Τα Bloom Filters προτάθηκαν από τον Burton Howard Bloom το 1970.Πρόκειται για μια δομή δεδομένων η οποία μπορεί να χρησιμοποιηθεί έτσι ώστε με μικρές απαιτήσεις χώρου και ταχύτατα να απαντά στο ερώτημα του εάν ένα στοιχείο key εμπεριέχεται σε ένα σύνολο S ή όχι.
Πρόκειται για μια πιθανοτική (probabilistic) δομή δεδομένων καθώς οι απαντήσεις που μπορούν να ληφθούν είναι: είτε ότι το στοιχείο key **δεν υπάρχει** στο σύνολο S είτε ότι το στοιχείο key **μπορεί να υπάρχει** στο σύνολο S.

Υλοποίηση Bloom Filter



http://en.wikipedia.org/wiki/Bloom_filter

- Η υλοποίηση ενός bloom filter χρησιμοποιεί ένα διάνυσμα δυαδικών ψηφίων (bit vector). Για να προστεθεί ένα στοιχείο στο bloom filter, το κλειδί του γίνεται hash από έναν αριθμό k hash functions και τα αποτελέσματα που προκύπτουν καθορίζουν ποια (<k) δυαδικά ψηφία του bit vector θα αλλάξουν τιμή από 0 σε 1 αν δεν είναι ήδη 1.
- Για να ελεγχθεί αν ένα στοιχείο με κλειδί key έχει εισαχθεί στο παρελθόν στο bloom filter, αρκεί να υπολογιστεί η hash value του κλειδιού key χρησιμοποιώντας τις ίδιες hash functions και εάν τα δυαδικά ψηφία που υποδεικνύονται από τα hash values δεν είναι όλα 1 στο bit vector τότε με σιγουριά γνωρίζουμε ότι το στοιχείο δεν είχε στο παρελθόν εισαχθεί στο bloom filter. Αν είναι τότε είτε το στοιχείο είχε εισαχθεί στο bloom filter είτε κάποιο άλλο στοιχείο που ενεργοποίησε τα ίδια δυαδικά ψηφία με αυτά που ενεργοποιεί το κλειδί key είχε εισαχθεί στο παρελθόν στο bloom filter.

False Positives - Ψευδώς Θετικές

False Positive

Υπάρχει πιθανότητα για ένα στοιχείο που δεν έχει εισαχθεί στο bloom filter να λαμβάνεται εσφαλμένη απάντηση ότι έχει στο παρελθόν εισαχθεί.
Για ένα bloom filter με bit vector μήκους m και k συναρτήσεις hash η πιθανότητα false positive μετά από n εισόδους είναι:

$$(1 - (1 - \frac{1}{m})^{kn})^k \approx (1 - e^{\frac{-kn}{m}})^k$$

- Η πιθανότητα false positive μειώνεται όταν αυξάνεται το μήκος k του bit vector.
- Οι συναρτήσεις hash που χρησιμοποιούνται από ένα bloom filter θα πρέπει να είναι ανεξάρτητες μεταξύ τους και να κατανέμουν ομοιόμορφα τα κλειδιά.
- Μεγαλύτερος αριθμός hash functions τυπικά συνεπάγεται μείωση των false positives, μείωση της ταχύτητας του bloom filter και ότι το bloom filter γεμίζει ταχύτερα.

Βέλτιστος αριθμός hash functions

Γνωρίζοντας τον αριθμό των στοιχείων n που πρόκειται να εισαχθούν στο bloom filter και το μήκος του bit vector m, η βέλτιστη τιμή για το πλήθος των hash functions που πρέπει να χρησιμοποιηθούν είναι:

$$\frac{m}{n} \ln(2)$$

Εφαρμογές των Bloom Filters

- Τα bloom filters χρησιμοποιούνται έτσι ώστε να ληφθεί με μεγάλη ταχύτητα απάντηση σχετικά με την απουσία ενός κλειδιού από ένα σύνολο. Για παράδειγμα πριν πραγματοποιηθεί η αναζήτηση για ένα κλειδί σε μια βάση δεδομένων μπορεί να γίνει ο έλεγχος σχετικά με το αν το κλειδί δεν υπάρχει στη βάση δεδομένων χρησιμοποιώντας ένα bloom filter.

Αναφορές

- http://billmill.org/bloomfilter-tutorial/
- http://www.jasondavies.com/bloomfilter/
- http://en.wikipedia.org/wiki/Bloom_filter
- Network Applications of Bloom Filters: A Survey by Andrei Broder and Michael Mitzenmacher, 2003