Cerca i Anàlisi d'Informació (CAI)
Grau en Ciència i Enginyeria de Dades, UPC

Session 5: Scaling up

Exercise List, Fall 2019

Exercise 1

Think of 12 information services you use often and say (discuss...) whether you prefer them to be more Available or more Consistent.

Examples: Instagram, a web searcher, the Racó, your online banking application, the Wikipedia, your favorite online newspaper, your google calendar, ... you think up the other 5.

Exercise 2

We know from theory that it is not possible to ensure simultaneously the properties of Consistency, Availability and Partitioning. If we consider them as boolean, 100% or nothing, properties.

However, if your boss asks you to design a database that more or less has these properties, it is probably a bad idea to tell him/her "Come on, boss, don't you know this is impossible? There is nothing I can do".

Give at least two versions of a quantitative, perhaps probabilistic, promise that a Partitioned database could make, regarding Consistency and Availability. Perhaps one is more concerned with Consistency, and the other with Availability.

Think of it in the spirit "Service Level Agreements" that cloud providers usually offer. (Look it up if you don't know what SLA is, you should).

Exercise 3

In an LSH system we use $k = 5$, $m = 10$. We want to get the neighbors of some item i_1 that are at least 0.9-similar to it.

1. If an item i_2 has similarity 0.9 to i_1, what is the probability that i_2 is in the candidate set?

2. If an item i_3 has similarity 0.45 to i_1, what is the probability that i_3 is in the candidate set?

3. Suppose that we have $1,000,000$ items in the database. Given the set of items that we would like to retrieve (0.9 similar at least), and those that we definitely want to exclude (0.45 similar at most) what is the recall, precision and f1-measure of this nearest-neighbor query?

   In the above, say that items in the grey zone between 0.9 and 0.45 do not count as false positives or as false negatives (this is new!).

4. Can you estimate the size of the candidate set? (The answer is NO, unless you suppose things about the distribution of similarities).

5. (Harder... for the braver) OK, let us suppose. Suppose that the probability that an item in the database has similarity s to i_1 decreases exponentially fast as s tends to 1. In particular i_1 has no copies, and 50% of the documents have similarity less than 0.9 with i_1. Can you now estimate the date of the candidate set? (Hint: First estimate how many documents have similarity above or below some given s. For that you need to estimate the parameters of a suitable exponential function of s ... an integral shows up...).

Exercise 4

Now let us look at it in the reverse direction. You probably need a numeric solver here.

1. We want that objects with similarity 0.9 appear as candidates 90% of the times or more, and that objects with similarity 0.7 appear as candidates 10% of the times at most. What values of k and m should we use?

   Of course, there are many pairs k, m that might work. Since the work performed by the algorithm is $O(km)$ we would like to choose a pair that, approximately, minimizes the product km.

2. We are now more strict. We want the items that are ≥ 0.99 similar to be recalled with probability 99% and we want items that are < 0.98 similar to be recalled only with probability 1%. What k, m should we use?

Exercise 5

We use consistent hashing where both items and servers are integers, and for simplicity we consider the circle to be divided into 360 degrees instead of being the interval [0,1].

Use the item function $h_I(i) = (317 * i) \mod 360$ and the server function $h_S(s) = (197 * s + 209) \mod 360$.

- Place the servers $s = 1, 2, 3$ in the circle, i.e, compute their h_S values.
- Place the pages $i = 1561, 2905, 3789, 4839, 5832, 6142, 7900, 8190, 9179, 10876, 11942, 12542, 13847, 14892, 15382$, i.e., compute their h_I values.
- Tell, for each of $s = 1, 2, 3$, how many pages are served by server s.
- What is the balance factor, the ratio of the number of pages served by the laziest server to that of the busiest server?
- We add two more servers, $s = 4, 5$, and remove server 1. Say how many pages are now served by each of the 4 servers.

Exercise 6

Consider the scenario of the previous problem before adding servers 4 and 5.

In order to improve the balance factor, we make two virtual copies of each server. That is, we use a second server hash function $h'_S(s) = (79 * s + 151) \mod 360$. Each server s is sent to both $h_S(s)$ and to $h'_S(s)$.

- Place the two virtual of each server $s = 1, 2, 3$ in the circle.
- Place the same pages as before.
- Say how many pages are served by each server.
- Compute the new balance factor.