Finding similar items

CSE 344, section 10

June 2, 2011

In this section, we'll go through some examples of finding similar item sets. We'll directly compare all pairs of sets being considered using the Jaccard similarity. We'll also see small examples of minhashing and locality-sensitive hashing methods, which are intended to help make the similarity pairing tractable for many possible sets.

The examples we'll see in this assignment are taken from your textbook, specifically the exercises for Garcia-Molina section 22.3 (pages 1115-6) and section 22.4 (page 1122).

1 Jaccard similarity and minhashing

1. Compute the Jaccard similarity of each pair of the following sets: {1, 2, 3, 4, 5}, {1, 6, 7}, {2, 4, 6, 8} .

2. What are all the 4-grams of the following string?
   `abc␣def␣ghi`
   Remember that white space (denoted by ␣) counts!

3. Suppose that our universal item set is $\{1, 2, \ldots, 10\}$, and signatures for sets are constructed using the following list of permuations for the universal set:
   - $(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)$
   - $(10, 8, 6, 4, 2, 9, 7, 5, 3, 1)$
   - $(4, 7, 2, 9, 1, 5, 3, 10, 6, 8)$

   Construct minhash signatures for the following sets:
   (a) $\{3, 6, 9\}$
   (b) $\{2, 4, 6, 8\}$
   (c) $\{2, 3, 4\}$

4. Suppose that instead of using particular permutations to construct signatures for the three sets of the previous problem, we use hash functions to construct the signatures. The three hash functions we use are:

$$f(x) = x \mod 10$$
$$g(x) = (2x + 1) \mod 10$$
$$h(x) = (3x + 2) \mod 10$$

Compute the signatures for the three sets, and compare the resulting estimate of the Jaccard similarity of each pair with the true Jaccard similarity.

5. Suppose you have some documents, and have stored k-grams of these documents in a large table. Each column of the table represents all the k-grams for a single document, and each row r represents the r^{th} k-gram for all the documents. (Because documents vary in length, there may be empty cells in the bottom fringes of the table.) The "schema" of the table — that is, the mapping between row indexes in the table and document IDs — is stored separately.

   Show how you would use MapReduce to compute a minhash value for each of your documents, using a single hash function (*not* a permutation of a dictionary of possible k-grams). You can assume that every processor gets a copy of the schema, but:

   (a) The table must be partitioned across the processors by rows.

(b) The table must be partitioned by columns.

2 Locality-sensitive hashing

1. Suppose we have a table where each tuple consists of three fields/attributes (name, address, phone number), and we need to do an entity resolution on this table to find those sets of tuples that refer to the same person. For concreteness, suppose that the only pairs of tuples that could possibly be total edit distance 5 or less from each other consist of a true copy of a tuple and another *corrupted* version of the tuple. In the corrupted version, each of the three fields is changed independently. 50% of the time, a field has no change. 20% of the time, there is a change resulting in edit distance 1 for that field. There is a 20% chance of edit distance 2 and 10% chance of edit distance 10. Suppose there are one million pairs of ths kind in the table.
   (a) How many of the million pairs are within total edit distance 5 of each other?

   (b) If we hash each field of all the tuples to one million buckets, how many of these one million pairs will hash to the same bucket for at least one of the three hashings?

(c) How many false negatives will there be? That is, how many of the one million pairs are within total edit distance 5, but will not hash to the same bucket for any of the three hashings?

2. The function $p = 1 - (1 - s^r)^b$ gives the probability p that two minhash signatues that come from sets with Jaccard similarity s will hash to the same bucket at least once, if we use an LSH scheme with b bands of r rows each. For a given similarity threshold s, we want to choose b and r so that $p = 1/2$ at s. Suppose signatures have length 24, which means we can pick any integers b and r whose product is 24. That is, the choices for r are 1, 2, 3, 4, 6, 8, 12, or 24, and b must then be $24/r$.
   (a) If $s = 1/2$, determine the value of p for each choice of b and r. Which would you choose, if $1/2$ were the similarity threshold?

(b) For each choice of b and r, determine the value of s that makes $p = 1/2$.