Bloom filtering[1]

Filtering pattern (Challenge 4_b)

For this exercise you have to prepare a report including:

- Your data collections comment describing what they represent.
- Your source codes: a file precising in a comment the compiling and execution commands.
- Propose a test battery and report execution measures.
- Compress everything and sent a .zip or a .tar to genoveva.vargas@gmail.com

1.1 Problem statement

Given a list of user's comments, filter out a majority of the comments that do not contain a particular keyword.

For this example, a Bloom filter is trained with a hot list of keywords. We use this Bloom filter to test whether each word in a comment is in the hot list. If the test returns true, the entire record is output. Otherwise, it is ignored. Here, we are not concerned with the inevitable false positives that are output due to the Bloom filter.

1.2 Hot list

Look at page 53 of the book "Map Reduce design patterns" and see the proposed `Map` and `Reduce` codes. For understanding bloom filters you have to program a bloom filtering training (see page 53).

- Once you have programmed the filtering training, prepare collections of different sizes to run your tests trying to get to the limits of your solution.
- Explain the role of the `setup` method of the mapper code proposed in the book.
- Why do we have to use the Hadoop distributed cache?
- Explain and discuss why is it not possible (as stated in the book) to have a `combiner`. Support your arguments with examples.

1.3 HBase query using a bloom filter

Problem: Given a list of users' comments, filter out comments from users with a reputation of less than 1,500.

Bloom filters can assist expensive operations by eliminating unnecessary ones. For the following example, a Bloom filter was previously trained with IDs of all users that have a reputation of at least 1,500. We use this Bloom filter to do an initial test before querying HBase to retrieve more information about each user. By eliminating unnecessary queries, we can speed up processing time.

Look at page 56 of the book "Map Reduce design patterns" and see the proposed `Map`, `Reduce` and `Training` codes.

- Prepare your data collections for this exercise.

---

[1] This challenge is an example proposed in the book MapReduce design patterns, pp. 47.

- Explain the role and compare the use of the training phase for the bloom filtering process. Give an explanation that generally explains the purpose of this phase.
- Do the external calls to HBase create a representative overhead on the filtering process. Does this process really help HBase to reduce the query evaluation overhead? Discuss and give arguments.
- Implement the optimization proposed in page 58 and compare results.