University of Zurich

Distributed Systems - Assignment 2

Deadline: 23:59, 21 November 2013

November 1, 2013

In this assignment, we will use Apache Hadoop to build inverted index for a corpus of documents. We will use this index and a simple scoring scheme to answer queries provided by the user.

Exercise:

Assignment 2 contains two parts: Theory part and Programming part. The theory part will be published during the first week of November. This is the description for the programming part.

In this assignment, we will build the inverted index using Hadoop Mapreduce jobs. The index will be written to the Hadoop Distributed File System (HDFS) and later to a Sqlite database. We will work with a dataset that contains information about movies and their actors.

1 How to begin:

1. Download the machine image we have provided.
2. Import the image in Virtualbox
3. Both username and password: ds2013

2 To use Hadoop pseudo-distributed cluster:

The virtual machine we have provided is configured with a pseudo-distributed Hadoop cluster. Usually a Hadoop cluster consists of multiple machines. A pseudo-distributed cluster is a cluster with only one node.

On the virtual machine, Hadoop files are located in /hadoop , please try not to change the files inside this directory.

On the virtual machine's command prompt,

- run start-dfs.sh to start the file-system processes (Namenode + Datanode)
- run start-mapred.sh to start Mapreduce processes (Jobtracker + Tasktracker)
- alternatively, run start-all.sh to start both the processes above
- to stop, you can run stop-all.sh
- on the web browser, you can see information related to the file-system at http://localhost:50070 and related to the jobs at http://localhost:50030

3 To run MapReduce programs:

- Use your favourite editor/IDE, use the Hadoop libraries and write MapReduce jobs
- Export your project as a jar and run the MapReduce job as follows: hadoop jar your-file-name.jar your-job-class

4 Dataset:

We use the Personas dataset. The dataset has already been copied to the HDFS. It is located at: /user/ds2013/data. The readme file present in the same location provides more information about the dataset. It contains three files, one with plot summaries for movies, another with various metadata about the movies and one more with various metadata about the actors.

5 Your job is to implement:

1. Mapreduce job(s) to count the total number of words in the plot-summaries dataset. This means the file plot_summaries.txt. The output file should contain only one information: the total number of words.
2. Mapreduce job(s) to find 10 most frequent words in the plot-summaries dataset. The words should be ordered in descending order by their number of occurrences in the dataset.
3. Mapreduce job(s) to create an inverted index for the plot-summaries dataset. The index should be a mapping from the term to the Wikipedia movie ID and the number of occurrences. For example, if the term "assignment" occurs 5 times in the plot summary of the movie whose Wikipedia movie ID is 42, then the index should contain a mapping from "assignment" to 42 and 5 (you are free to choose how to represent this information).
4. Mapreduce job(s) to join the character.metadata.tsv and movie.metadata.tsv datasets and create a file which has the following fields:
   i. Wikipedia Movie ID
   ii. Movie Name

iii, iv, v, ... Actor Name (possibly) multiple

5. Write a Java class that reads the above files from HDFS and writes them to a SQLite database. This database will be used to serve the queries mentioned below. Write another Java class that accepts a query file from the user on the command line. Each line of the query file will contain a query (possibly multiple terms). This class should read query on each line and return a top-10 list with following information by querying the SQLite database: the name of the movie in which these terms occur the highest number of times, followed by the actors of the movie. If the query contains more than one term, take the sum of the number of occurrences to rank.

6 Important points:

- Input location to all Mapreduce jobs will be the /user/ds2013/data directory inside HDFS
- Output location to all jobs will be provided by the user in the command line as the first parameter
- For the job to build SQLite database in task 5, three command line arguments will be provided: follows: the first parameter will contain the output of task 3, the second parameter will contain the output location of task 4, and the third parameter will contain a location for the SQLite database.
- While counting terms in step 1 above, do not count the movie id at the beginning of each line
- For all text processing, please do the following:
  a convert all text to lowercase
  b remove the following punctuation marks: , (comma), . (period), -, "". Replace - with a space and all others with empty string.
  c exclude stop-words provided at this HDFS location: /user/ds2013/stop_words , i.e., words that occur in this list should not be counted or processed in any of the steps mentioned above.

7 Submission Instruction:

You will have to submit a jar file. Because of the confusion regarding the submission process in the last assignment, specific submission instruction for assignment 2 will be updated later in the course forum.

8 Evaluation:

Your submission will be evaluated for all 5 of the above mentioned tasks. The score breakdown is as follows: Task 1 (15%), task 2 (15%), task 3 (15%), task 4 (15%), and task 5 (40%). Task 5 will be evaluated like in Assignment 1. There will be a theory part for Assignment 2 which will be published during the first week of November. The theory assignment will carry 20% weight. So, the total breakdown of scores for Assignment 2 is: 20% for theory assignment and 80% for this programming assignment.

9 Deadlines:

Please try to start working on the assignments on time so that you can submit early. However, in cases you need some extra time, everybody is provided a pool of 4 days. If you need extra days for solving the assignments, you can use them in any combination you want. For example, you can use 2 slack days for 1st assignment and 2 for 2nd. Then, you won't have any extra late days for the 3rd assignment.

10 More Information:

If you need to discuss more about the assignment, please use the course forum. For more help on HDFS or Mapreduce, please refer to the Apache documentations or ask in the course forum.