ID2221 – Lab 1
MapReduce, HDFS, and HBase

Dmytro Siniukov   siniukov@kth.se   951228-2436
Miquel Larsson   miquell@kth.se   920614-5998

September 23, 2018

1. **Code explanation**

   Main two classes are *TopTenMapper* and *TopTenReducer*.

   In the *map* method of the mapper we take a row (representation of a user), parse it, and put into *TreeMap* < *Integer, Text* > , which internally sorts the users by their reputation.

   In *cleanUp* we pass top 10 rows to a reducer (which is only one).

   So, each mapper task filters out top 10 users by reputation and passes them to the reducer. After that, the reducer does a similar job as the mappers, with the only difference that it filters top 10 rows among the partial lists of the mappers. Finally, in *cleanUp* we insert the list into ’topten’ HBase table.

2. **Running the script**

   After setting the environmental variables and adding the input folder to hdfs, we have to create the result table in HBase first: `create ’topten’, ’info’`

   Then:

   1) `javac -cp $ HADOOP_CLASSPATH -d topten_classes topten/TopTen.java`

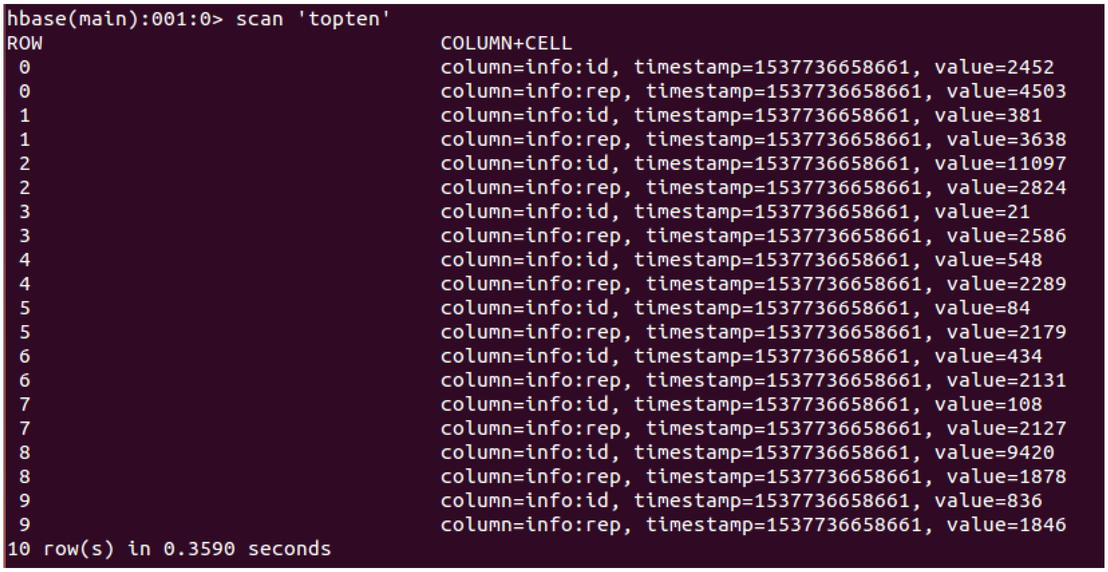
   2) `jar -cvf topten.jar -C topten_classes/ .`

   3) `$ HADOOP_HOME/bin/hadoop jar topten.jar topten.TopTen /topten_input`

3. **Results**

   After executing the script, the final table ’topten’ is obtained in HBase, as seen in the figure below.

   As we can observe, the top 10 records of reputation are displayed in descending order, which was the objective of the assignment.



```
hbase(main):001:0> scan 'topten'
ROW                                  COLUMN+CELL
 0                                   column=info:id, timestamp=1537736658661, value=2452
 0                                   column=info:rep, timestamp=1537736658661, value=4503
 1                                   column=info:id, timestamp=1537736658661, value=381
 1                                   column=info:rep, timestamp=1537736658661, value=3638
 2                                   column=info:id, timestamp=1537736658661, value=11097
 2                                   column=info:rep, timestamp=1537736658661, value=2824
 3                                   column=info:id, timestamp=1537736658661, value=21
 3                                   column=info:rep, timestamp=1537736658661, value=2586
 4                                   column=info:id, timestamp=1537736658661, value=548
 4                                   column=info:rep, timestamp=1537736658661, value=2289
 5                                   column=info:id, timestamp=1537736658661, value=84
 5                                   column=info:rep, timestamp=1537736658661, value=2179
 6                                   column=info:id, timestamp=1537736658661, value=434
 6                                   column=info:rep, timestamp=1537736658661, value=2131
 7                                   column=info:id, timestamp=1537736658661, value=108
 7                                   column=info:rep, timestamp=1537736658661, value=2127
 8                                   column=info:id, timestamp=1537736658661, value=9420
 8                                   column=info:rep, timestamp=1537736658661, value=1878
 9                                   column=info:id, timestamp=1537736658661, value=836
 9                                   column=info:rep, timestamp=1537736658661, value=1846
10 row(s) in 0.3590 seconds
```

Figure 1: Final obtained table in Hbase, which displays the top ten id values in reputation descending order, as desired.