Hadoop training: http://courses.coreservlets.com

coreservlets.com – Hadoop Course

Streaming

In this exercise, you will have a chance to develop Hadoop Streaming MapReduce Job(s).

Approx. Time: 45 minutes

Perform

1. Develop a streaming job that will count up each unique token
   - Persist a file with tab-separated results, one token, and corresponding occurrence count per a line:

     ```
     Airline	20
     Airport	7
     ...
     ```
   - Use `/training/data/war_and_peace.txt` as input to your job; the file already exists in HDFS
   - I suggest using python, starting with an example in the lecture
   - Don't forget that you can test your scripts on the command line:

     ```
     $ cat inputTest.txt | <mapperScript> | sort | <reducerScript>
     ```
   - If you place your script in the Exercises project then $HADOOP_EXERCISES_SRC environment variable may be useful

Extra Credit

1. Develop a MapReduce job that given a text file input will produce the two counts: (1) Number of tokens whose character length is greater than or equals to five characters (2) Number of tokens whose character length is less than five characters
   - Persist results to a file that should looks something like this:

     ```
     greaterOrEqualsToFiveChars	236865
     lessThanFiveChars	329372
     ```
   - Use `/training/data/war_and_peace.txt` as input to your job; the file already exists in HDFS
   - I suggest using python, starting with an example in the lecture
   - Don't forget that you can test your scripts on the command line:

     ```
     $ cat inputTest.txt | <mapperScript> | sort | <reducerScript>
     ```
   - If you place your script in the Exercises project then $HADOOP_EXERCISES_SRC environment variable may be useful

Solution

1. The code can be found in the Solutions project

    ```
    src/resources/mapRed/streaming/CountUniqueMapper.py
    src/resources/mapRed/streaming/CountUniqueReducer.py
    ```

    First test our the scripts with command line:

    ```
    cat $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/inputTest.txt | \
            $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/CountUniqueMapper.py | \
            sort | $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/CountUniqueReducer.py
    ```

    Finally run them on the cluster:

    ```
    yarn jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-*.jar \
            -D mapred.job.name="Count Job via Streaming" \
            -files $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/CountUniqueMapper.py,\
            $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/CountUniqueReducer.py \
            -input /training/data/war_and_peace.txt \
            -output /training/playArea/streaming/CountUnique \
            -mapper CountUniqueMapper.py \
            -combiner CountUniqueReducer.py \
            -reducer CountUniqueReducer.py
    ```

Extra Credit Solution

1. The code can be found in the Solutions project:

    ```
    src/resources/mapRed/streaming/LengthDividerMapper.py
    src/resources/mapRed/streaming/CountUniqueReducer.py
    ```

    First test our the scripts with command line:

    ```
    cat $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/inputTest.txt | \
            $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/LengthDividerMapper.py | \
            sort | $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/CountUniqueReducer.py
    ```

    Finally run them on the cluster:

    ```
    yarn jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-*.jar \
            -D mapred.job.name="Count Job via Streaming" \
            -files $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/LengthDividerMapper.py,\
            $HADOOP_SOLUTIONS_SRC/resources/mapRed/streaming/CountUniqueReducer.py \
            -input /training/data/war_and_peace.txt \
            -output /training/playArea/streaming/LengthDivider \
            -mapper LengthDividerMapper.py \
            -combiner CountUniqueReducer.py \
            -reducer CountUniqueReducer.py
    ```