CS 455: INTRODUCTION TO DISTRIBUTED SYSTEMS
Department of Computer Science
Colorado State University

SPRING 2018
URL: http://www.cs.colostate.edu/~cs455
Instructor: Shrideep Pallickara

Homework 3: Programming Component

ANALYZING ON-TIME PERFORMANCE OF COMMERCIAL FLIGHTS IN THE UNITED STATES USING MAPREDUCE
VERSION 1.1

DUE DATE: Wednesday, April 11th, 2018 @ 5:00 pm

OBJECTIVE

The objective of this assignment is to gain experience in developing MapReduce programs. As part of this assignment, you will be working with datasets released by the United States Bureau of Transportation Statistics. You will be developing MapReduce programs that parse and process the on-time performance records for flights operated by major carriers in the U.S. from October, 1987 to April, 2008.

You will be using Apache Hadoop (version 2.7.3) to implement this assignment. Instructions for accessing datasets and setting up Hadoop clusters are available on the course website.

This assignment may be modified to clarify any questions (and the version number incremented), but the crux of the assignment and the distribution of points will not change.

1 Cluster setup

As part of this assignment you are responsible for setting up your own Hadoop cluster with HDFS running on every node. We will be staging datasets on a *read-only* cluster. You should use your ***own*** cluster to write outputs produced by your MapReduce programs. MapReduce clients will be able to access namespaces of both clusters through Hadoop ViewFS federation. Your programs will process the staged datasets; data locality will be preserved by the MapReduce runtime.

2 Analysis of Airline On-time Data

You should develop MapReduce programs that process the main *and* supplementary datasets to answer the following questions. Each question accounts for **2 points each for Q1-Q5, 1 point for Q6, and 3 points for Q7**.

Q1.	What is the best time-of-the-day/day-of-week/time-of-year to fly to minimize delays?
Q2.	What is the worst time-of-the-day / day-of-week/time-of-year to fly to minimize delays?
Q3.	What are the major hubs (busiest airports) in continental U.S.? Please list the top 10. Has there been a change over the 21-year period covered by this dataset?
Q4.	Which carriers have the most delays? You should report on the total number of delayed flights and also the total number of minutes that were lost to delays. Which carrier has the highest average delay?
Q5.	Do older planes cause more delays? Contrast their on-time performance with newer planes. Planes that are more than 20 years old can be considered old.
Q6.	Which cities experience the most weather-related delays? Please list the top 10.
Q7.	Come up with an innovative analysis program for this dataset. For this component, think of yourself as the lead data scientist at a start-up firm. What would do with this dataset that is cool? You are allowed to: (1) combine your analysis with other datasets, (2) use other frameworks such as Mahout for performing your analyses, and/or (3) perform visual analytics. **Restrictions:** Note that there should be NO DISCUSSIONS about Q7 on Piazza or Canvas. Your analysis must be something that you have come up with on your own. Q7 is quite open-ended and you have a lot of freedom. That freedom comes with the responsibility that you manage your own problems and don't except someone else (be it the Professor, GTA, or your peers) to solve your problems for you. You have to iron out all problems that you are facing on your own.

2.1.1 Main Dataset

The dataset contains information about flight arrivals and departures for all commercial flights within USA from October, 1987 to April, 2008. Records for each year is stored in a separate CSV file. The year is part of the file name. For example, records for 1990 are stored in a file named 1990.csv. There are 22 files for the entire time period which is being considered. Each line in a file corresponds to a single record comprising of 29 fields separated by commas. There are approximately 120 million records in the entire dataset.

The table below summarizes the fields. Fields in the following table are appearing in the same order as in a record. The complete documentation including the data dictionary for the dataset is available at http://www.transtats.bts.gov/Fields.asp?Table_ID=236. Please note that we are using a reduced dataset in which the derivable fields are removed. This dataset contains only a subset of the fields from the list of fields described in the data dictionary provided in the above link.

The dataset is available under directory `/data/main` in the shared HDFS.

Index	Field Name	Description
1	Year	Between 1987 - 2008
2	Month	Between 1 - 12
3	DayOfMonth	Between 1 – 31
4	DayOfWeek	1 (Monday) – 7 (Sunday)
5	DepTime	Actual Departure Time (local time, hhmm)
6	CRSDepTime	Scheduled Departure Time (local time, hhmm)
7	ArrTime	Actual Arrival Time (local time, hhmm)
8	CRSArrTime	Scheduled Arrival Time (local time, hhmm)
9	UniqueCarrier	Carrier Code
10	FlightNum	Flight Number
11	TailNum	Plane tail number
12	ActualElapsedTime	Actual elapsed time for the journey (in minutes)
13	CRSElapsedTime	Scheduled elapsed time for the journey (in minutes)
14	AirTime	Flight time (in minutes)
15	ArrDelay	Arrival delay (in minutes)
16	DepDelay	Departure delay (in minutes)
17	Origin	Origin Airport (international airport abbreviation code)
18	Dest	Destination Airport (international airport abbreviation code)
19	Distance	Distance between origin and destination (in miles)
20	TaxiIn	Taxi in time (in minutes)
21	TaxiOut	Taxi out time (in minutes)
22	Cancelled	Was the flight cancelled? (0 = No, 1 = Yes)
23	CancellationCode	Reason for cancellation (A = carrier, B = weather, C = NAS (National Air System), D= Security)
24	Diverted	Was the flight diverted? (0 = No, 1 = Yes)
25	CarrierDelay	Carrier delay (in minutes)
26	WeatherDelay	Weather delay (in minutes)
27	NASDelay	National air system delay (in minutes)
28	SecurityDelay	Security delay (in minutes)
29	LateAircraftDelay	Delay due to a late aircraft (in minutes)

2.1.2 Supplementary Datasets

The main dataset outlined above is supplemented by two other datasets. Answering some of the queries may require referring to these additional datasets in addition to the main dataset. Supplementary datasets are staged in the directory `/data/supplementary` in the read-only cluster.

airports.csv

Describes the locations of the US airports.

Index	Field Name	Description
1	iata	International airport abbreviation code
2	airport	Name of the airport
3	city	City
4	state	State
5	country	Country
6	lat	Latitude
7	long	Longitude

carriers.csv
Lists carrier codes and their full names.

Index	Field Name	Description
1	Code	Carrier code
2	Description	Full name of the carrier

plane-data.csv
Details about individual planes are available in this dataset. Please note that this dataset is incomplete. You only need to work with the data available in this dataset.

Index	Field Name	Description
1	tailnum	Tail Number of the plane
2	type	Type of the Manufacturer
3	manufacturer	Manufacturer
4	issue_date	Date the flight was handed over to the airline
5	model	Model of the plane
6	status	Status
7	aircraft_type	Type of the aircraft
8	engine_type	Type of the engine
9	year	Manufactured Year

Note: 'NA' (Not Applicable) or empty strings are used to represent missing data.

3 Provided Resources

Datasets required for both components are shared through a viewfs based federated HDFS setup running on CS department machines. A complete guide on setting up your own Hadoop cluster and connecting to the shared HDFS will be provided through course website.

4 Grading

Homework 3 accounts for 20 points towards your final course grade. The programming component accounts for 80% of these points with the written element (to be posted later) accounting for the remaining 20%. This programming assignment will be graded for 16 points. The point distribution for this assignment is listed below.

2 points	For setting up the Hadoop cluster
11 points	Knowledge extraction and developing programs to answer questions Q1 through Q6. You will also be judged on the elegance of your MapReduce programs. While getting the answers is important, your design matters as well.
3 points	Your solution to Q7.

The grading for this assignment will be done based on a one-on-one interview and will include a code review.

5 Deductions

There will be a **16-point deduction** (i.e. a 100% penalty) if any of the restrictions are violated:

1. You should not implement this assignment as a stand-alone program.
2. You should not implement this assignment using anything other than Hadoop MapReduce. Implementing your own framework or using a 3rd party framework (that is not Hadoop) to implement this assignment is not allowed.

6 Milestones:

You have 5 weeks to complete this assignment. The weekly milestones below correspond to what you should be able to complete at the end of every week.

Milestone 1: You should be able to set up a Hadoop cluster and get started with basic processing for the Airlines Census data.

Milestone 2: Programs to answer Q1, Q2, and Q3 are completed. Come up with the core idea for Q7.

Milestone 3: Programs to answer Q4, Q5, and Q6 are complete. Work on Q7 underway with significant progress.

Milestone 4: Programs to answer Q7 are complete. Iron out bugs in any of the other components.

7 What to Submit

Use the CS455 checkin program to submit a single .tar file that contains:

- All the Java files related to the assignment (please document your code)
- You should use **`ant`** to compile your codebase and provide the corresponding `build.xml` file that is used for compiling the codebase. Please make sure that your `build.xml` works! You may modify the sample build.xml file that we have provided to do this.
- A `README.txt` file containing a description of each file and any information you feel the GTA needs to grade your program.

The folder set aside for this assignment's submission using checkin is **HW3-PC**

8 Change History

This section will reflect any changes that were made to a particular version of the assignment. Generally, these changes are made to better clarify the spirit of the assignment.

Version	Date	Change
1.1	3/8/2018	First public release of the assignment