Team: Prasuna Kumari Pothabolu, Nancy Divya JeyaKumar , Syed Rhythm Ahir Hussain

Flight Data Analysis Project Report

- In this project, we will develop cloud-based Big Data workflows to process and analyze a large volume of flight data using Hadoop/ Oozie on AWS VMS.

  Data : Airline On-time Performance data set (flight data set) from the period of October 1987 to April 2008

  Link: http://stat-computing.org/dataexpo/2009/the-data.html (Links to an external site.) .

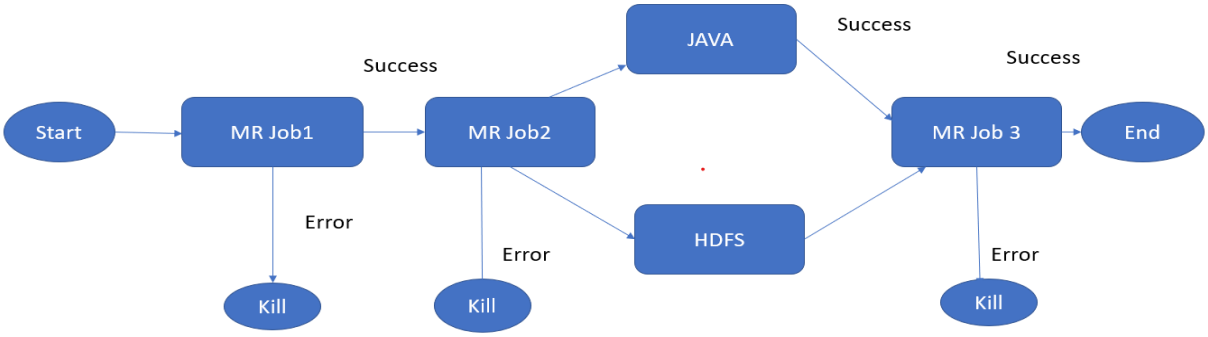
- Design, implement, and run an Oozie workflow to find out below.
  - the 3 airlines with the highest and lowest probability, respectively, for being on schedule;
  - the 3 airports with the longest and shortest average taxi time per flight (both in and out), respectively.
    - the most common reason for flight cancellations.
- workflow analyze the entire data set (total 22 years from 1987 to 2008) at one ime on two VMs first and then gradually increase the system scale to the maximum allowed

- workflow will analyze the data in a progressive manner with an increment of 1 year, i.e. the first year (1987), the first 2 years (1987-1988), the first 3 years (1987-1989), …, and the total 22 years (1987-2008), on the maximum allowed number of VMs, and measure each corresponding workflow execution time.

- We developed 3 jobs to compute below requirements:
  - Job1 to compute the 3 airlines with the highest and lowest probability, respectively, for being on schedule
  - Job2 to find the 3 airports with the longest and shortest average taxi time per flight (both in and out), respectively
  - Job3 to find the most common reason for flight cancellations

- Oozie workflow Diagram for 3 jobs:

Flight Data analysis workflow



1. Job1 to compute the 3 airlines with the highest and lowest probability, respectively, for being on schedule
2. Job2 to find the 3 airports with the longest and shortest average taxi time per flight (both in and out), respectively
3. Job3 to find the most common reason for flight cancellations

- Data : Airline On-time Performance data set (flight data set) from the period of October 1987 to April 2008
  Data Source Link: http://stat-computing.org/dataexpo/2009/the-data.html (Links to an external site.) .
  It has data in Separate CSV for each year.
  The data is about below.
  Year
  Month
  DayofMonth
  DayOfWeek
  DepTime
  CRSDepTime
  ArrTime
  CRSArrTime
  UniqueCarrier
  FlightNum
  TailNum
  ActualElapsedTime
  CRSElapsedTime
  AirTime
  ArrDelay
  DepDelay
  Origin
  Dest
  Distance
  TaxiIn
  TaxiOut
  Cancelled
  CancellationCode
  Diverted
  CarrierDelay
  WeatherDelay
  NASDelay
  SecurityDelay
  LateAircraftDelay

- **MapReduce:**
  - The computation takes set of key,value pairs and produces output key,value pairs
  - Mapreduce algorithm uses 2 functions Map, Reduce

- **A detailed description of Algorithm used to compute each problem values:**

- **Job1 : Find the 3 airlines with the highest and lowest probability, respectively, for being on schedule**
- **MapperOnSchedule class:**

  As each combination of airline carrier and flight number is unique, thus we can make them together as the key here. MapperOnSchedule calculate the sum of ArrDelay and DepDelay for each flight (each row in the table).When this sum is greater than the threshold time (15 min), then this flight can be seen as not on schedule. Then context write (total <carrier>, 1) and context write (count <carrier>, 1). Also each airline carrier's delayed flights and total flights will increase by 1. Otherwise, when the sum of ArrDelay and DepDelay is not larger than the threshold, which means the flight is on schedule, then we need to increase the on time flight by 1 with context write (count <carrier>, 1). This will not change the delayed fights of airline carriers and will only increase its total flight number by 1.

- **ReducerOnSchedule class:**

  To calculate the on_schedule probability, the total flights number of each airline carrier and the total number of flights which are not on_schedule of this carrier are needed. Then for each carrier, its total flights number as well as its not_on_schedule flight number are counted. Then here, by dividing the two count number, i.e. the number of not_on_schedule flights and the number of total flights, the delayed probability can be obtained for each carrier. In this reducer, these delayed probabilities are stored in linked list. Then by

sorting this linked list in ascending order we can get the airlines with highest probability for being on schedule, by sorting this linked list in descending order we can get the airlines with lowest probability for being on schedule. Maintain two size 3 sorted lists to sort highest percentage of on schedule flights and lowest percentage of on schedule flights.

- **Find the 3 airports with the longest and shortest average taxi time per flight (both in and out), respectively**

- MapperFlightTaxiTime class:

  This class use airport code as key and corresponding taxi time as value. It produce the Origin airport and departure taxi time(Origin, taxiTime) and taxi out time (Dest, taxiTime).

- ReducerFlightTaxiTime class:
  This class calculate the total taxi time (in time and out time) for each airport by adding taxiIn and taxiOut values. Also ReducerFlightTaxiTime count the total number of flight for each airport. Then the avg taxi time can be calculated from the two values by dividing total taxi time by number of flights. Then this value will be saved in a linkedlist. Then by sorting this linkedlist in ascending order we can get the airports with shortest average taxi time, by sorting this linkedlist in desending order we can get the airports with longest average taxi time. Maintain two size 3 sorted lists to store highest average taxi time airport and lowest average taxi time airport.

3. Find the most common reason for flight cancellations

- **MapperFlightCancel class:**

  MapperFlightCancel count this cancellation reason by context write the cancellation code and 1 when a flight is cancelled.

- **ReducerFlightCancel**

  This Reducer is used to find the highest reason for flight cancellations. All we need to do is to maintain a cancellation code with the highest number of cancel flights. Here count the number of flights which cancelled for the same reason (same cancel code), and when the number increased it needs to compare with the current code with highest cancel flights. If the count number is larger than current highest number then replace the cancel code and make this count

- Commands are mentioned Hadoop/oozie setup file

- Sample Output:

  Three airlines with Highest probability for being on schedule:

  YV7958 100.0%

  YV7003 100.0%

  YV6790 100.0%

Three airlines with lowest probability for being on schedule:

9E2072 0.0%

9E2076 0.0%

9E2102 0.0%

Three airports with the longest average taxi time per flight:

MKC 8.0

CBM 5.0

GLH 4.25

Three airports with the shortest average taxi time per flight:

BFF 0.0

CYS 0.0

EAU 0.0

The most common reason for flight cancellations:

A 317972

- **Performance report**

  We ran flight analysis jobs for 22 years data on one master , 2 slaves

  It took 15 minutes.

  The run time will be decreased if number of data nodes (slaves) are increased