INFO 7250 U.S Flight On-time Performance Analysis

Team 16

Bowei Wang, Dongyue Li, Zelong Zhao, Xiaoyu Zheng

1.Introduction

1.1 Idea

Flight delay is a challenging problem for both passengers and airline companies, which will lead to financial loss and negative impact on airline companies' reputation. As a passenger, we get stuck in the airports all the time because of flight delays. So we decided to dig into the big data and use the flight records to analyze the flight on-time performance and predict whether your flight will be delayed.

1.2 Objective

In this project we are trying to answer some of the questions, for example:

- What are the reasons for such flight delays and cancellations?
- What are the Top 10 infamous airports which have serious delay rate?
- What's the reputation of this carrier?
- How was my flight perform in the past?
- Will my flight be late?

By using techniques like R, Hadoop, HBase, Machine Learning. We answered all these questions and provide more information for customers when they try to buy tickets on our website.

1.3 Teamwork

- Bowei Wang: Build Front End, AWS, HBase, Writing report.

- Zelong Zhao: Hadoop, Machine Learning, Hbase, Writing report.

- Dongyue Li: R Analysis, Data visualization, UI Design, Writing report.

- Xiaoyu Zheng: Data collection, Hadoop, Writing report.

2.Dataset

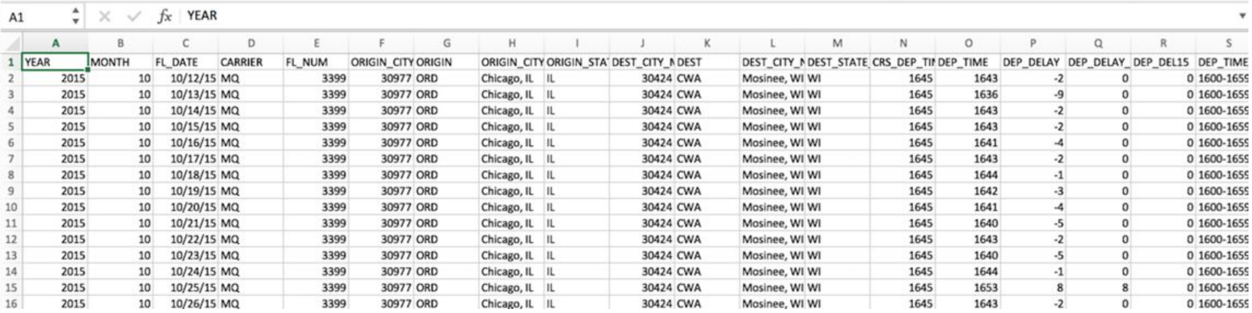
In this project, we use 3 big datasets related to flights, price, airline reviews:

1. Flights dataset, from BTS
   http://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=236
2. Real-time flight price, from Google QPX Express API
   https://developers.google.com/api-client-library/java/apis/qpxExpress/v1
3. Airline Reviews, from SKYTRAX
   http://www.airlinequality.com/review-pages/a-z-airline-reviews/

2.1 Flights dataset:

The flight dataset which is used for our investigation is derived from the Bureau of Transportation Statistics. Our goal is to investigate flights data of past year (from Oct 2015 to Sep 2016) provided by BTS.

This dataset contains on-time arrival data for non-stop domestic flights by major air carriers, and provides such additional items as departure and arrival delays, origin and destination airports, flight numbers, scheduled and actual departure and arrival times, cancelled or diverted flights, taxi-out and taxi-in times, air time, and non-stop distance.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	YEAR	MONTH	FL_DATE	CARRIER	FL_NUM	ORIGIN_CITY	ORIGIN	ORIGIN_CITY	ORIGIN_STA	DEST_CITY_N	DEST	DEST_CITY_N	DEST_STATE_	CRS_DEP_TI	DEP_TIME	DEP_DELAY	DEP_DELAY_	DEP_DEL15	DEP_TIME
2	2015	10	10/12/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1643	-2	0	0	1600-1659
3	2015	10	10/13/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1636	-9	0	0	1600-1659
4	2015	10	10/14/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1643	-2	0	0	1600-1659
5	2015	10	10/15/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1643	-2	0	0	1600-1659
6	2015	10	10/16/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1641	-4	0	0	1600-1659
7	2015	10	10/17/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1643	-2	0	0	1600-1659
8	2015	10	10/18/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1644	-1	0	0	1600-1659
9	2015	10	10/19/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1642	-3	0	0	1600-1659
10	2015	10	10/20/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1641	-4	0	0	1600-1659
11	2015	10	10/21/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1640	-5	0	0	1600-1659
12	2015	10	10/22/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1643	-2	0	0	1600-1659
13	2015	10	10/23/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1640	-5	0	0	1600-1659
14	2015	10	10/24/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1644	-1	0	0	1600-1659
15	2015	10	10/25/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1653	8	8	0	1600-1659
16	2015	10	10/26/15	MQ	3399	30977	ORD	Chicago, IL	IL	30424	CWA	Mosinee, WI	WI	1645	1643	-2	0	0	1600-1659

2.2 Real-time flight price

The Google QPX Express API allows developers to access information on global airline pricing and availability. By integrating the API into their applications, developers can provide their customers with airfare pricing and shopping services. With one query, QPX Express searches airline schedules, fares, tax rules, and seat availability in order to return fully-priced, availability-checked flight options and booking information.

2.3 Airline Reviews:

SKYTRAX as the leading global guide to Passenger reviews and ratings of airlines throughout the world, featuring customer trip experiences, ratings and opinions. Traveller airline ratings include seat comfort, Cabin staff service, Inflight Entertainment, onboard catering, Airport services and Value For money.

3. R

3.1 Purpose

In this project, we used R to analysis flights dataset and do text mining on airline reviews dataset. We investigated the two datasets from several aspects:

1) Number of flights per carrier
2) Flight cancellation rate per carrier
3) Number of flights operated by day of the week
4) Number of flights operated by month
5) Top 10 Worst Airports by Average Arrival Delay Time and Delay Rate
6) Top 10 Worst Airports by Average Departure Delay Time and Delay Rate
7) On-time arrival performance
8) Departure delay distribution over the day period
9) Departure and arrival airport count by carrier
10) Text mining on passengers reviews for each airline(carrier) and plot word cloud for each airline(carrier).

3.2 Dataset description

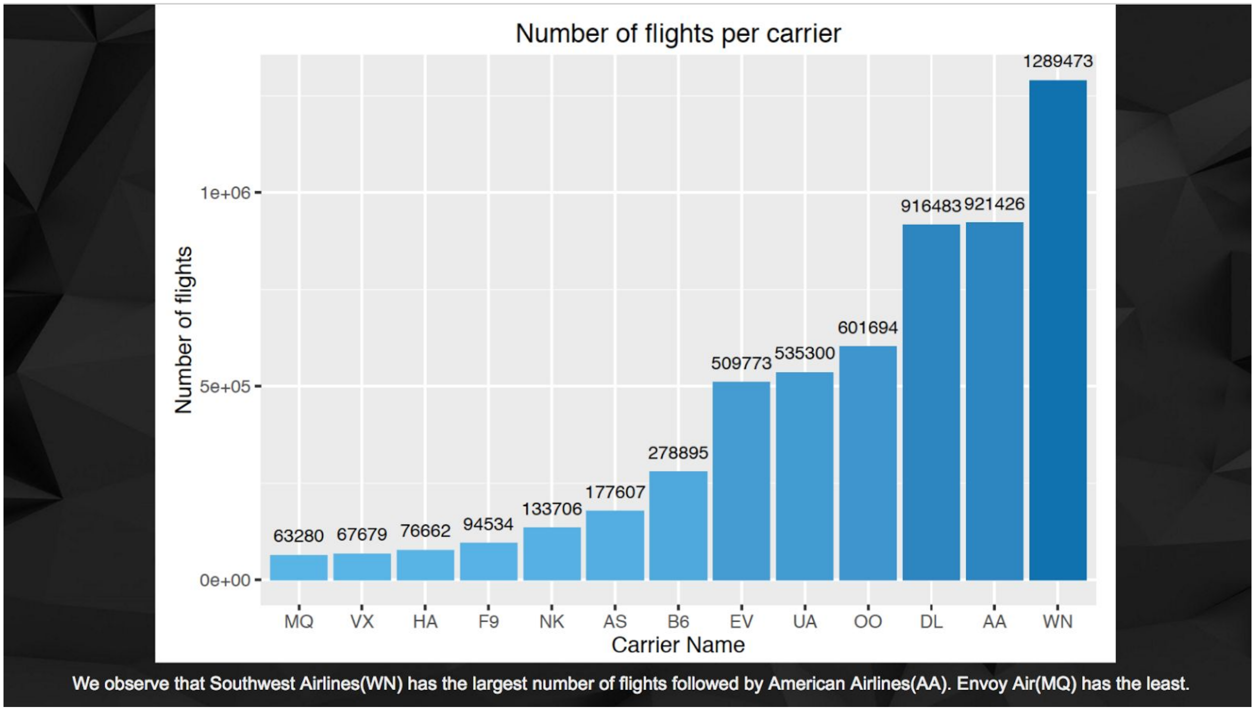
The flight dataset which is used for our investigation is derived from the Bureau of Transportation Statistics. Our goal is to investigate flights data of past year (from Oct 2015 to Sep 2016) provided by BTS.

This dataset contains on-time arrival data for non-stop domestic flights by major air carriers, and provides such additional items as departure and arrival delays, origin and destination airports, flight numbers, scheduled and actual departure and arrival times, cancelled or diverted flights, taxi-out and taxi-in times, air time, and non-stop distance.

Another dataset we used for text mining is passengers airline reviews from SKYTRAX. It contains 12 airlines reviews including AA, AS, B6, DL, F9, HA, MQ, NK, OO, UA, VX, WN.

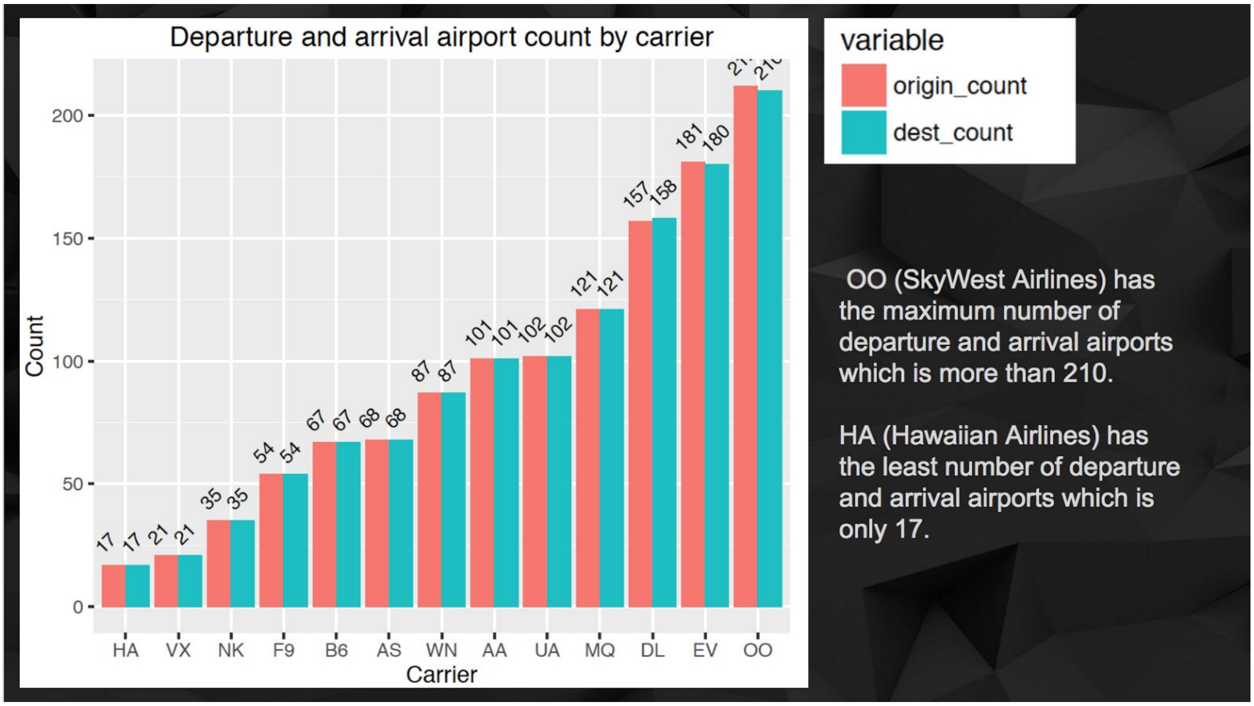
3.3 Graphs

1) How many flights did each carrier operate during past year? Which carrier has the largest/least number of flights?



We observe that Southwest Airlines(WN) has the largest number of flights followed by American Airlines(AA). Envoy Air(MQ) has the least.

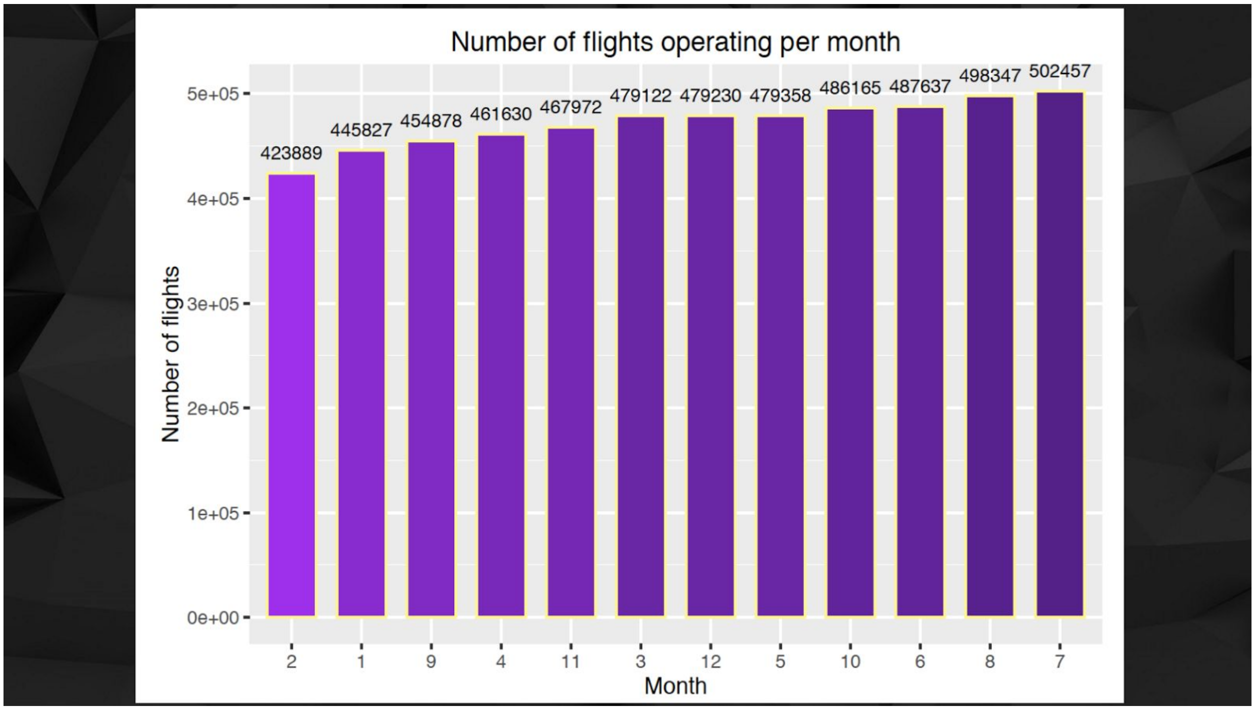
2) Which carrier is the most widely distributed?



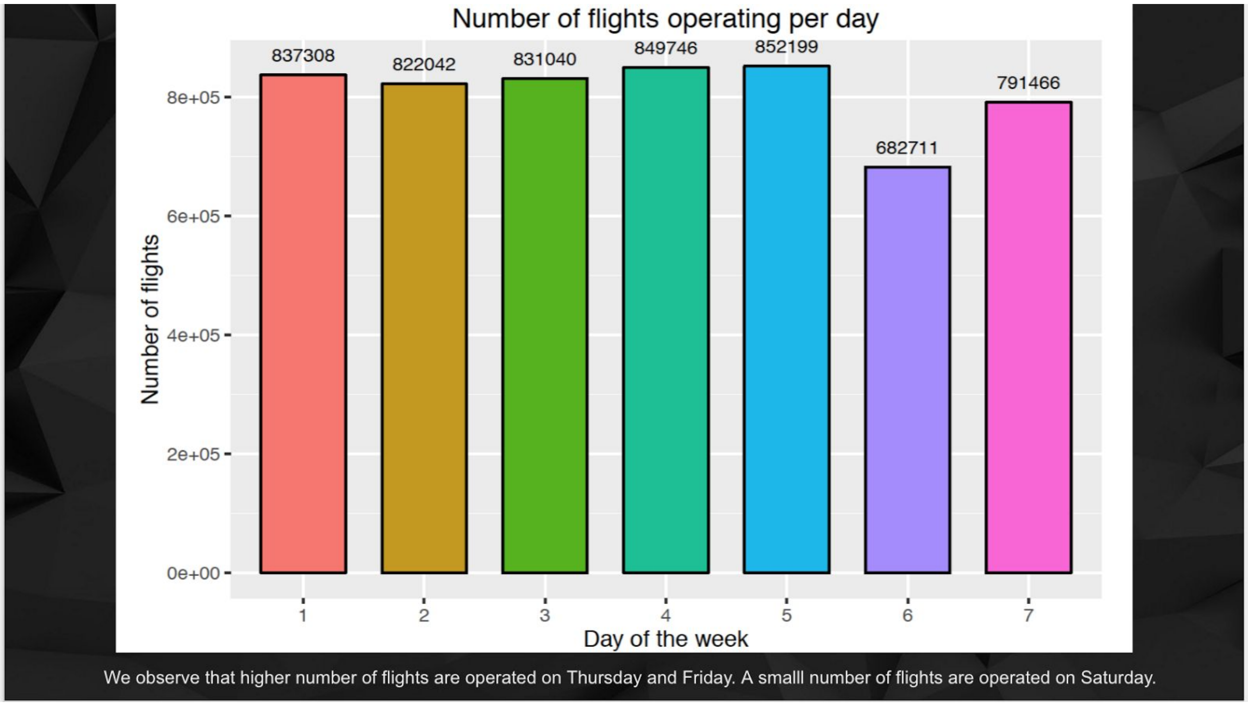
OO (SkyWest Airlines) has the maximum number of departure and arrival airports which is more than 210.

HA (Hawaiian Airlines) has the least number of departure and arrival airports which is only 17.

3) Which month is the busiest month of the year?

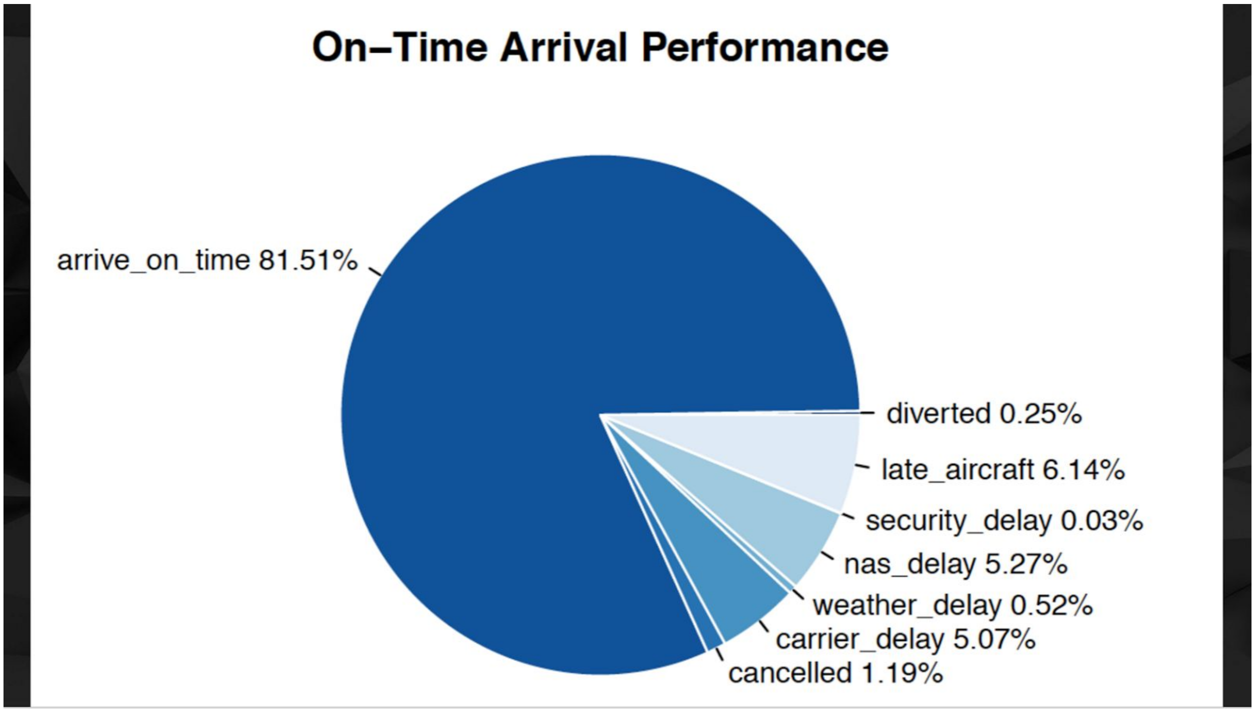


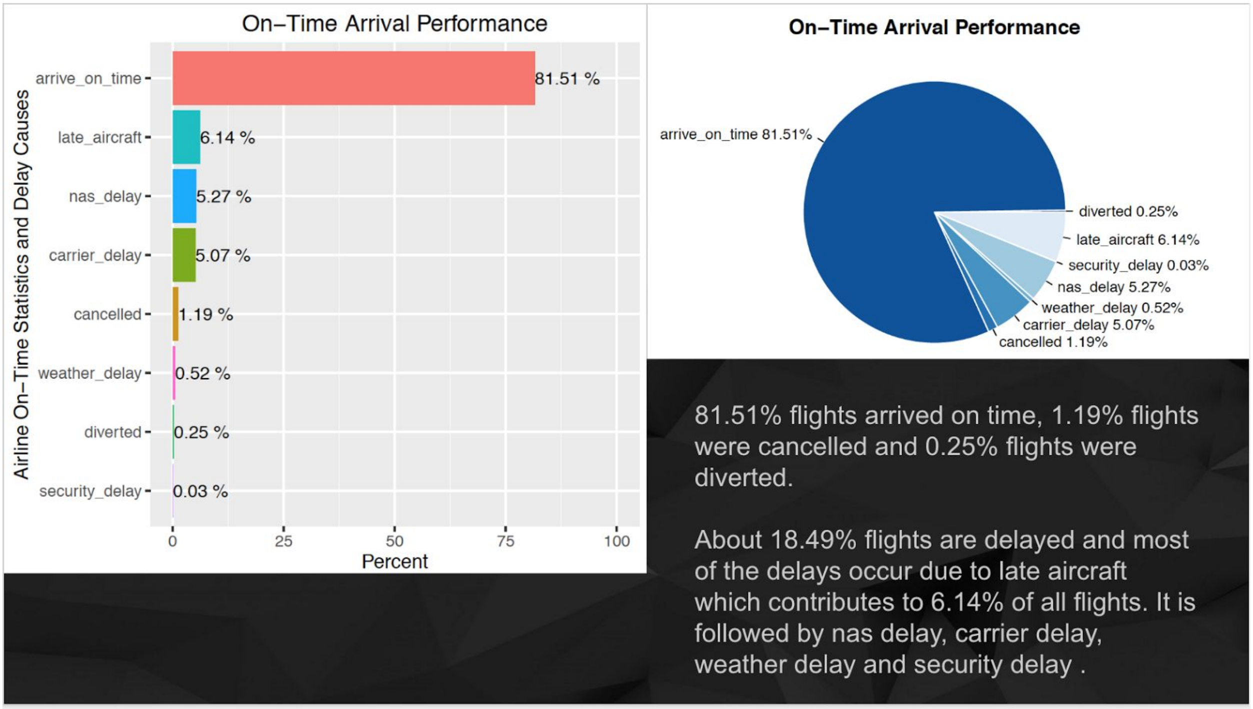
4) Which day is the busiest day of the week?



We observe that higher number of flights are operated on Thursday and Friday. A smalll number of flights are operated on Saturday.

5) How about the On-time Arrival Performance of last year?

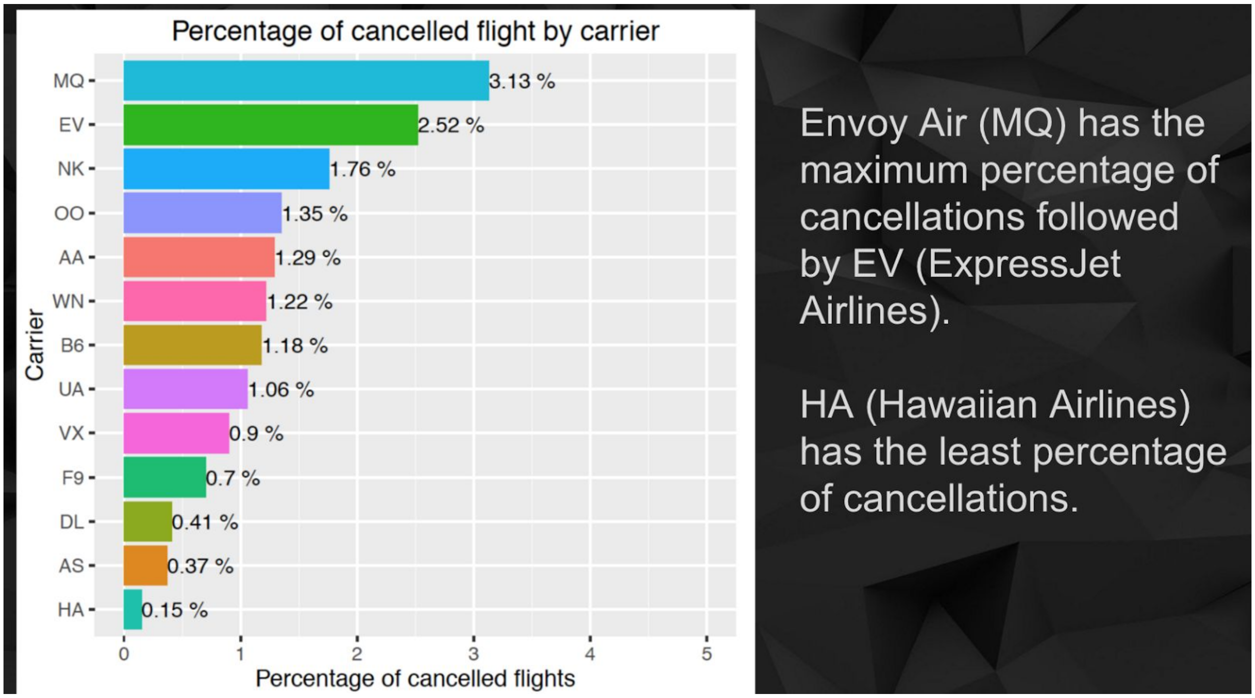




81.51% flights arrived on time, 1.19% flights were cancelled and 0.25% flights were diverted.

About 18.49% flights are delayed and most of the delays occur due to late aircraft which contributes to 6.14% of all flights. It is followed by nas delay, carrier delay, weather delay and security delay .

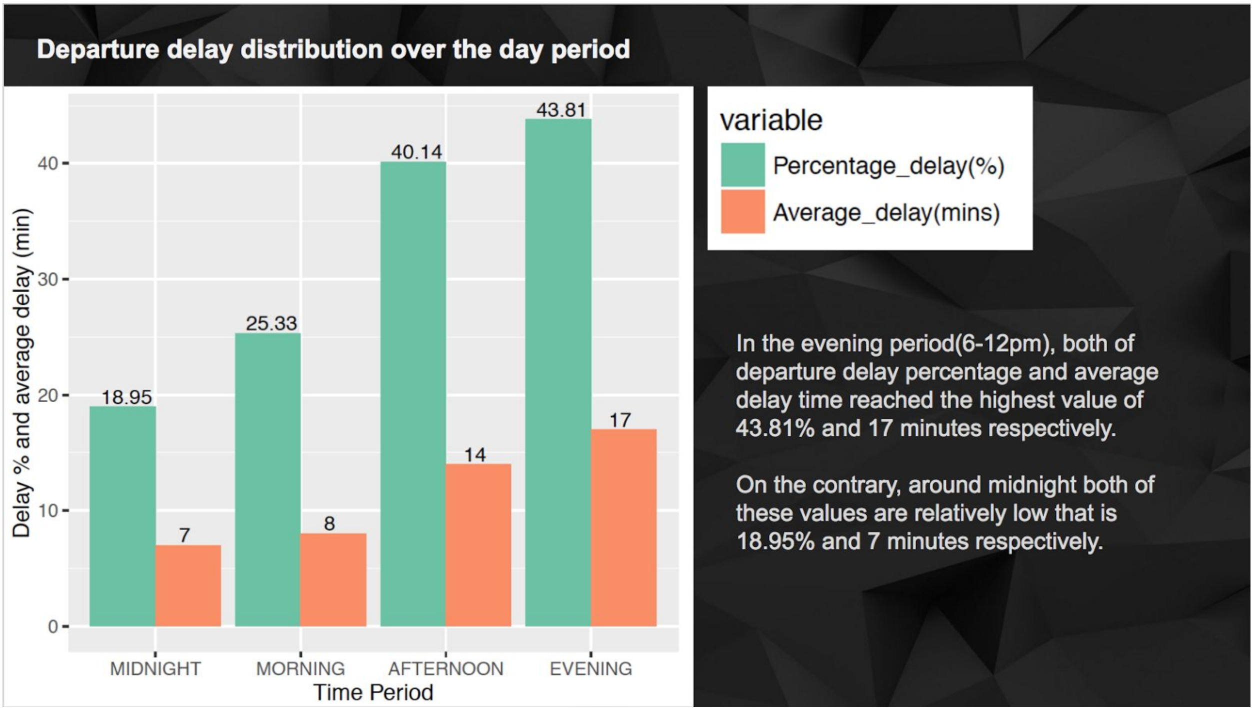
6) So, How about the flights cancellation rate per carrier?
Which carrier most frequently cancel their flights?



Envoy Air (MQ) has the maximum percentage of cancellations followed by EV (ExpressJet Airlines).

HA (Hawaiian Airlines) has the least percentage of cancellations.

7) So, How about the flights delay time and delay rate?
Which time period of the day is most likely to take off late?

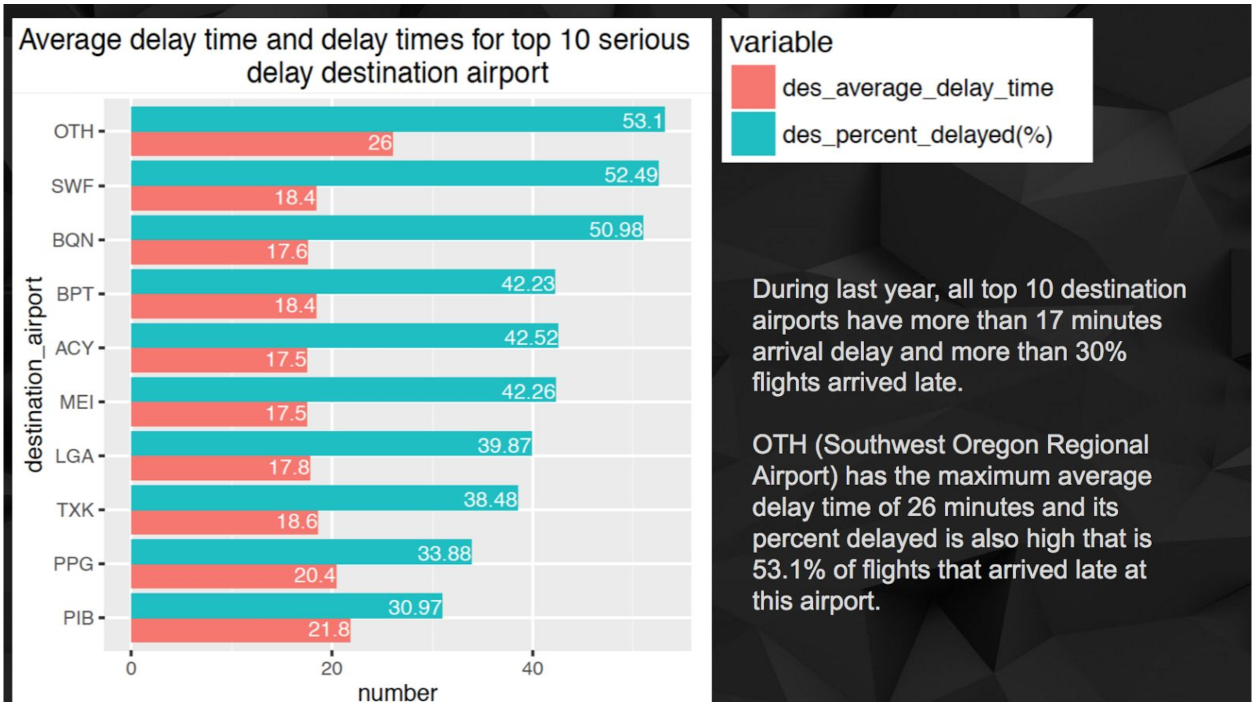


Departure delay distribution over the day period

In the evening period(6-12pm), both of departure delay percentage and average delay time reached the highest value of 43.81% and 17 minutes respectively.

On the contrary, around midnight both of these values are relatively low that is 18.95% and 7 minutes respectively.

8) Top 10 Worst Destination Airports ranked by Average Arrival Delay Time and Delay Rate

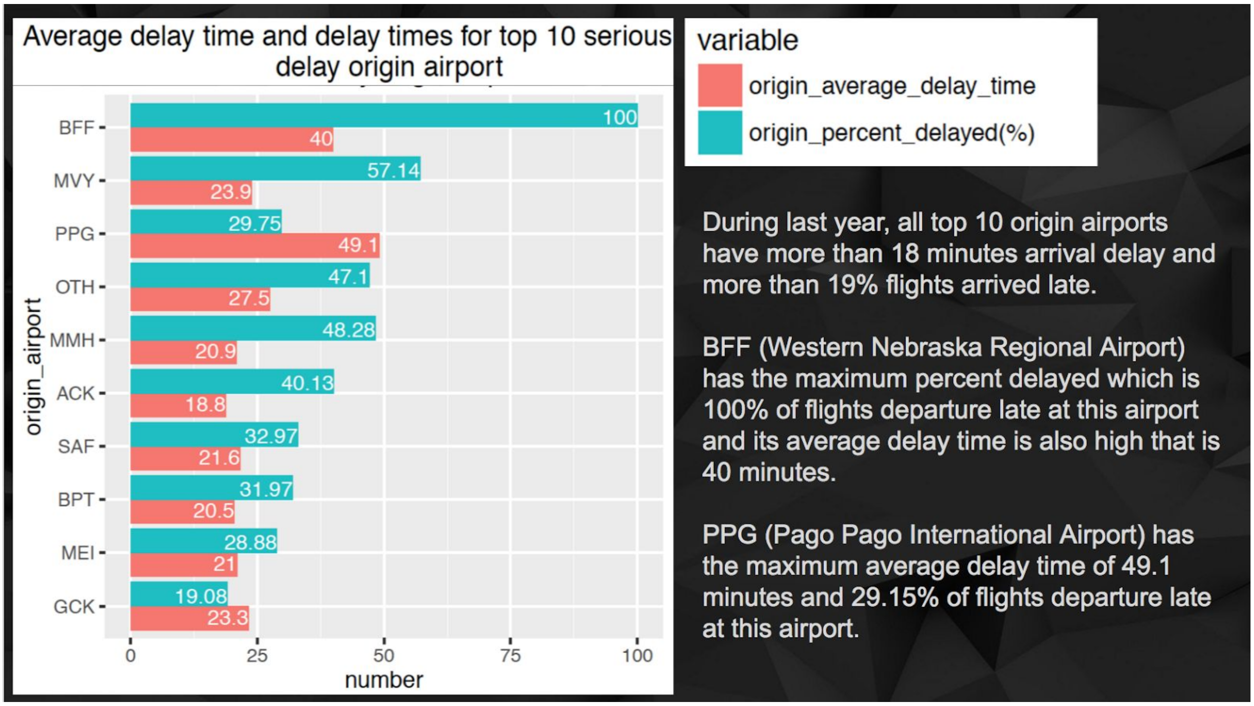


Average delay time and delay times for top 10 serious delay destination airport

During last year, all top 10 destination airports have more than 17 minutes arrival delay and more than 30% flights arrived late.

OTH (Southwest Oregon Regional Airport) has the maximum average delay time of 26 minutes and its percent delayed is also high that is 53.1% of flights that arrived late at this airport.

9) Top 10 Worst Origin Airports ranked by Average Arrival Delay Time and Delay Rate

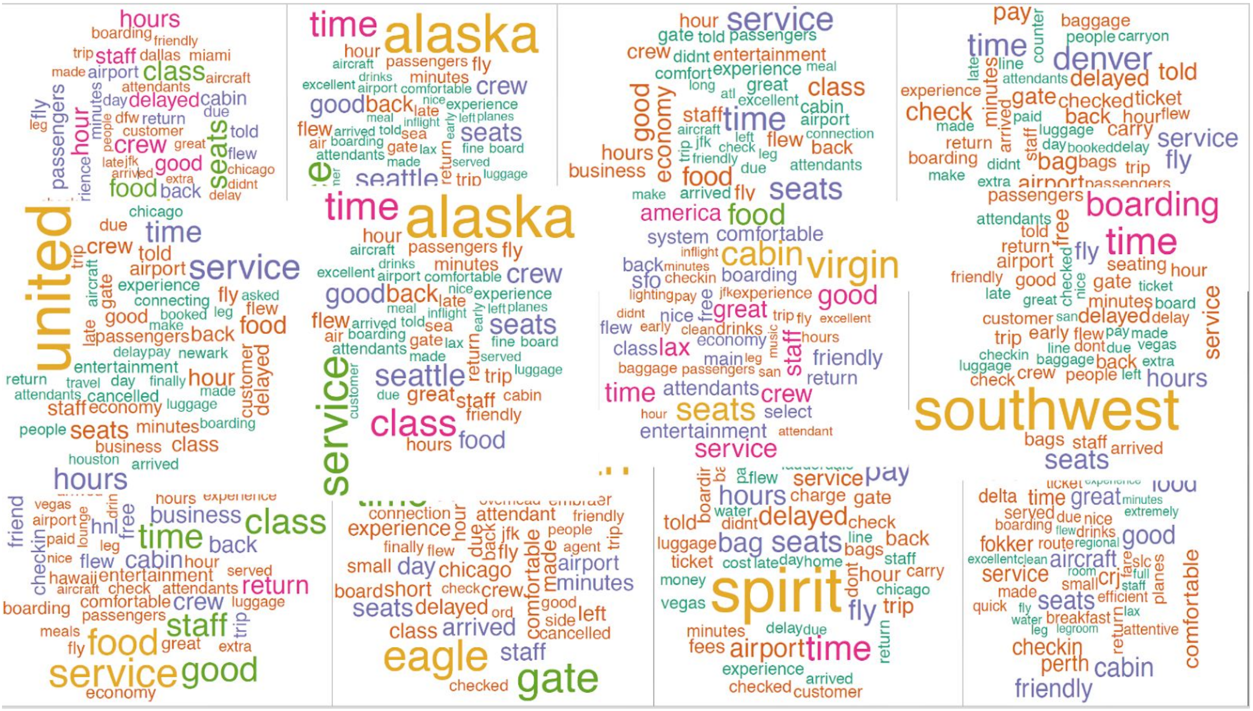


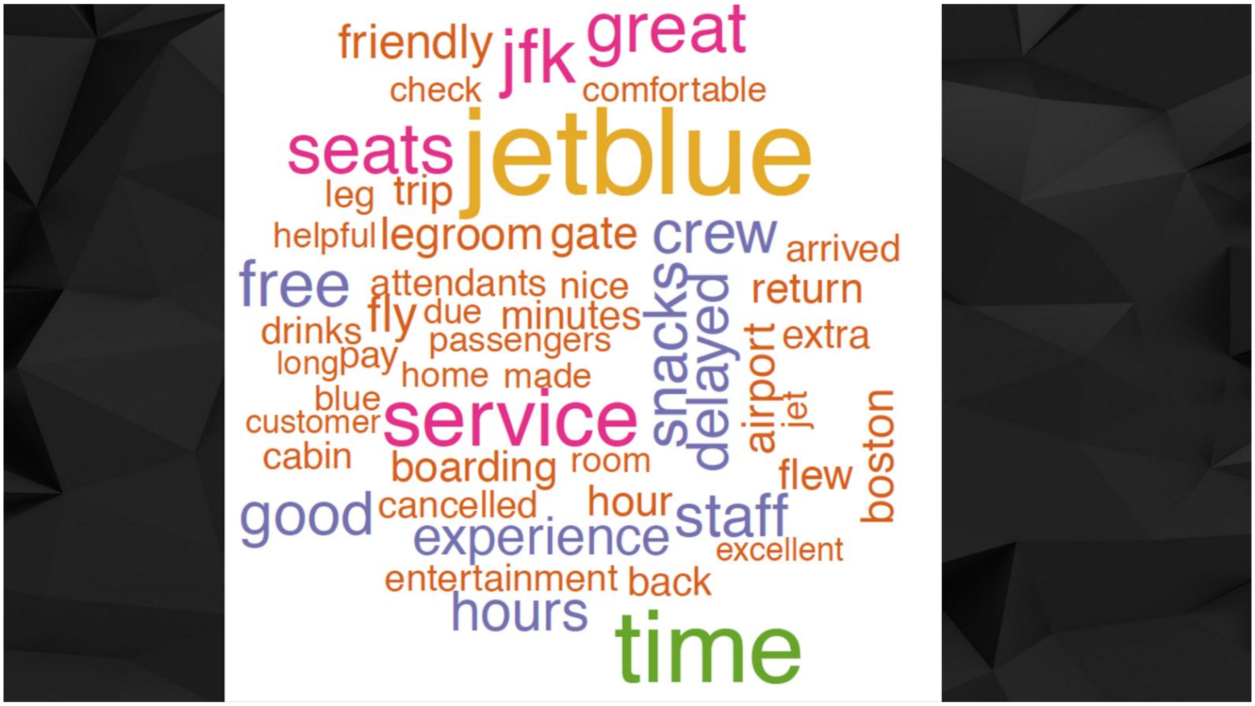
During last year, all top 10 origin airports have more than 18 minutes arrival delay and more than 19% flights arrived late.

BFF (Western Nebraska Regional Airport) has the maximum percent delayed which is 100% of flights departure late at this airport and its average delay time is also high that is 40 minutes.

PPG (Pago Pago International Airport) has the maximum average delay time of 49.1 minutes and 29.15% of flights departure late at this airport.

10) Text mining on passengers reviews for each airline(carrier) and plot word cloud for each airline(carrier). What about passengers' impressions of each airline(carrier)?





4. Hadoop

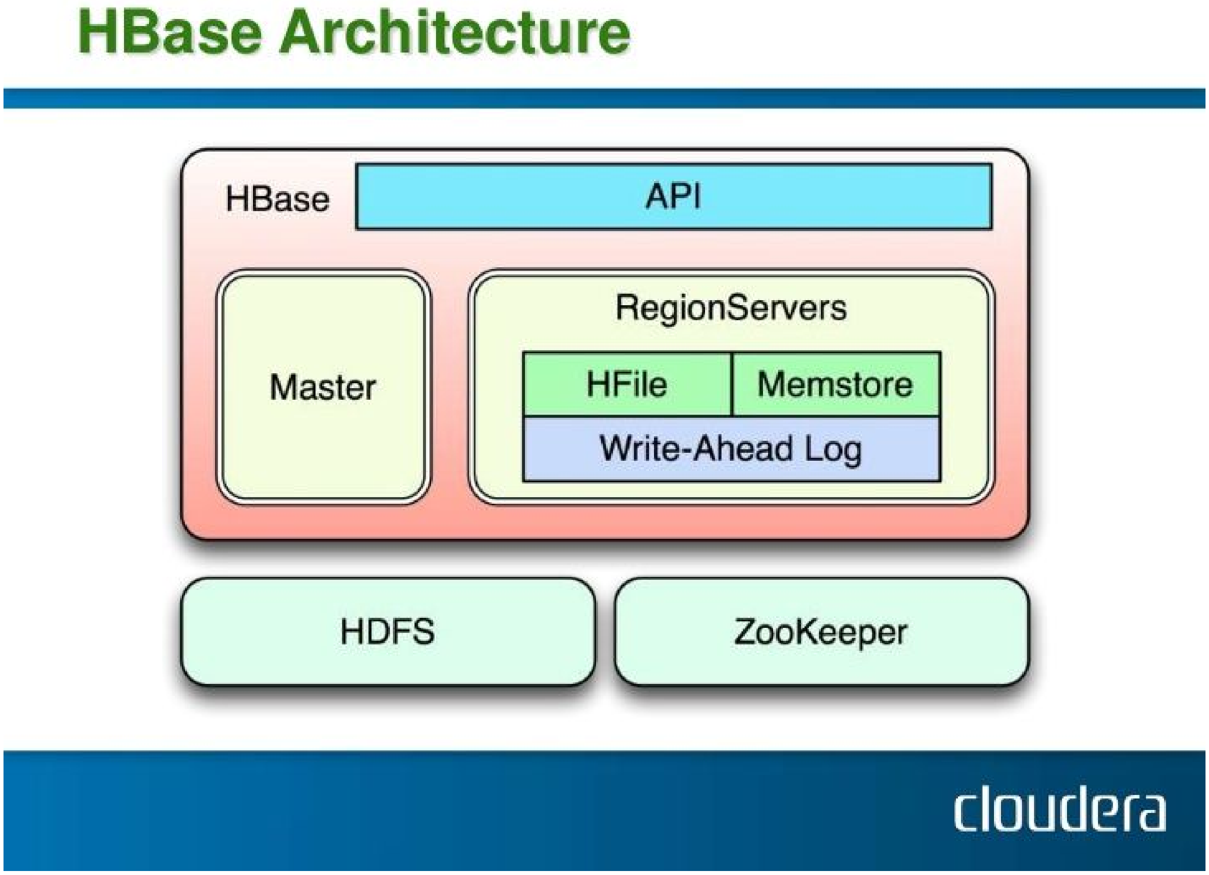
4.1 Purpose

Using over 5 million flight records we downloaded from the Bureau of Transportation Department. We use Hadoop to achieve 2 main goals:

a. Calculate the delay rate, average delay time and average air time for each flight.
b. Store the result into HBase. So that we can query the data from the front-end in a very fast way.

4.2 Hbase

Apache HBase is the Hadoop database that is used for a distributed, scalable, big data store. It is different from traditional relational database. It is a NoSQL database that runs on top the Hadoop cluster and provides random real-time read/write access to the data. it stores key/value pairs in columnar fashion.



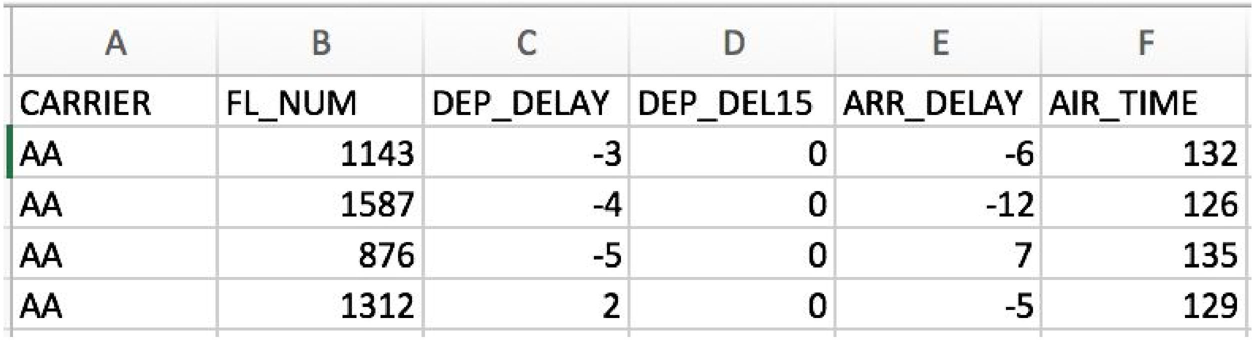
HBase Architecture

4.3 Data Preparation

Before we use MapReduce to process data. We choose several attributes from the dataset:

1. Carrier
2. FL_Num: Flight number.
3. DEP_DELAY: Departure delay time.
4. ARR_DELAY: Arrival delay time.
5. DEP_DEL15: Delay indicator
6. AIR_TIME: Time that the airplane is on air

Here is a screenshot of selected data:



A	B	C	D	E	F
CARRIER	FL_NUM	DEP_DELAY	DEP_DEL15	ARR_DELAY	AIR_TIME
AA	1143	-3	0	-6	132
AA	1587	-4	0	-12	126
AA	876	-5	0	7	135
AA	1312	2	0	-5	129

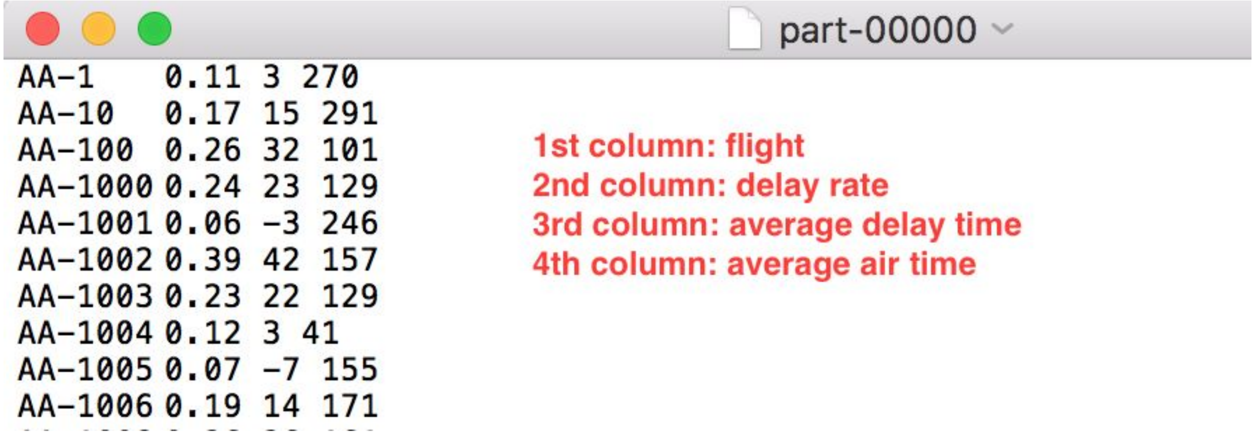
The negative number in DEP_DELAY and ARR_DELAY means that the flight departs early or arrive early.

4.3 MapReduce

We analyzed the delay rate of each flight using MapReduce. The mapper read the csv file and generating key value pair. Where key is the carrier name plus flight number. So that we can avoid two carriers have the same flight number. And value is a text Writable containing whether it's delay or not, and delay time. Mapper generate key and values like:

- Key: <Carrier + flightNum>
- Value: <delayIndicator, arrDelayTime, depDelayTime, airTime>

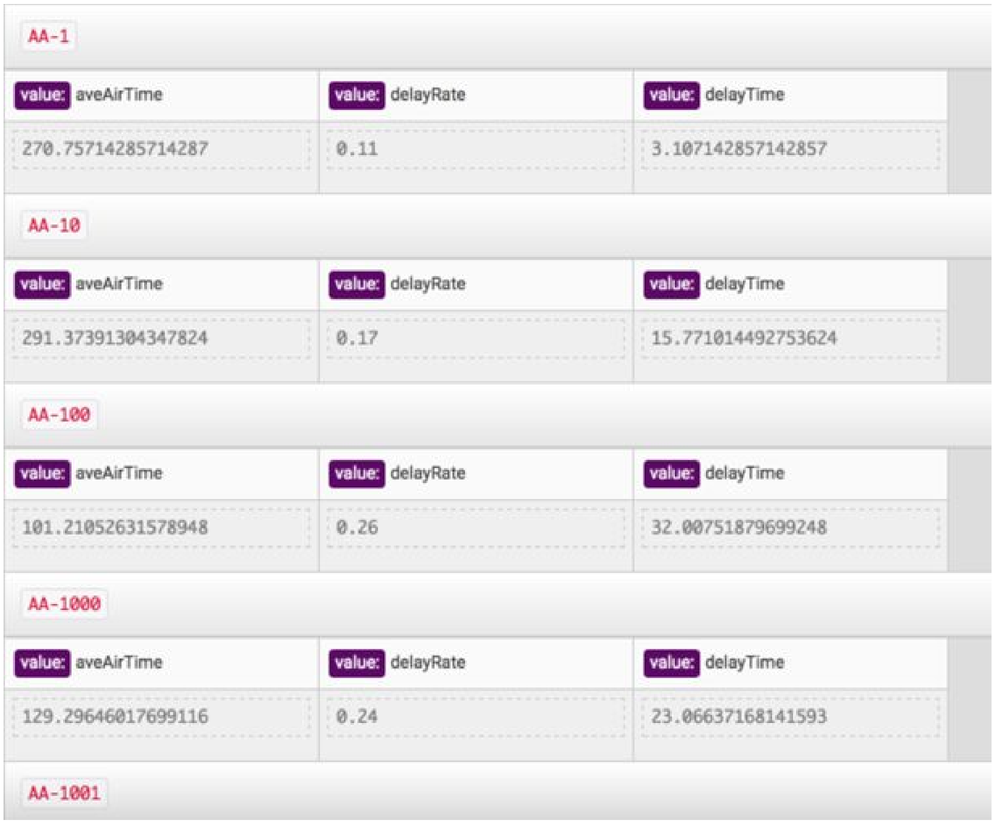
In reducer I do the summation and calculation to generate delay rate and average delay time. These information can be piped up on the website when customer try to search the flight. Help them make better decision. Here is a screenshot of MapReduce output before we insert it into HBase:



```
AA-1    0.11 3 270
AA-10   0.17 15 291
AA-100  0.26 32 101
AA-1000 0.24 23 129
AA-1001 0.06 -3 246
AA-1002 0.39 42 157
AA-1003 0.23 22 129
AA-1004 0.12 3 41
AA-1005 0.07 -7 155
AA-1006 0.19 14 171
```

1st column: flight
2nd column: delay rate
3rd column: average delay time
4th column: average air time

And here is what we got after insert data into HBase:



AA-1

value: aveAirTime	value: delayRate	value: delayTime
270.75714285714287	0.11	3.107142857142857

AA-10

value: aveAirTime	value: delayRate	value: delayTime
291.37391304347824	0.17	15.771014492753624

AA-100

value: aveAirTime	value: delayRate	value: delayTime
101.21052631578948	0.26	32.00751879699248

AA-1000

value: aveAirTime	value: delayRate	value: delayTime
129.29646017699116	0.24	23.06637168141593

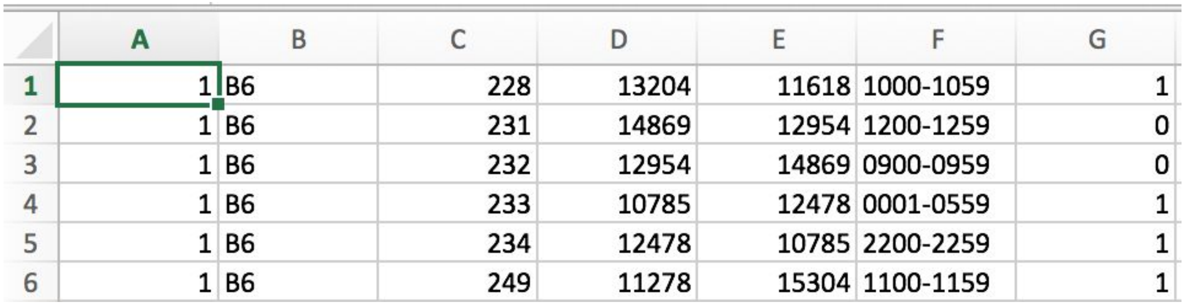
AA-1001

5. Machine Learning

We also do some machine learning based on our dataset. We use logistic Regression to predict whether a flight will be delayed. The independent variables are:

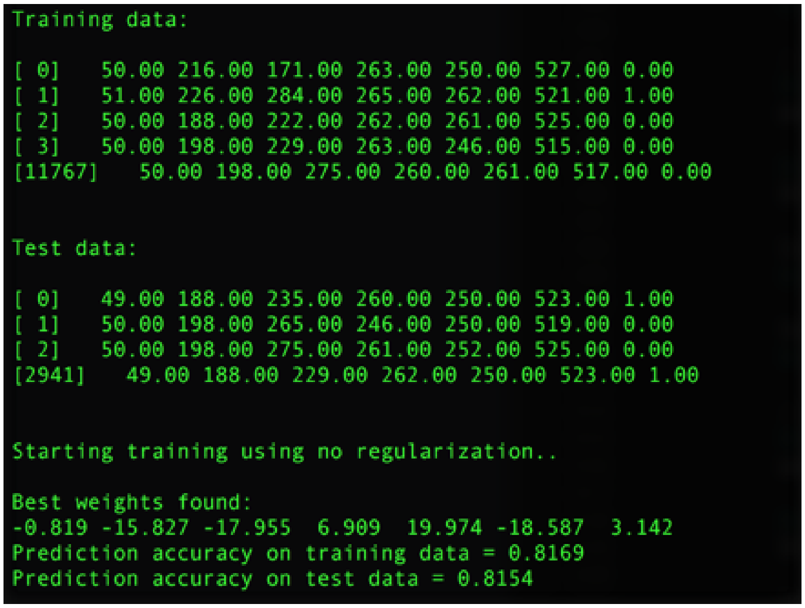
1. Day of the Week
2. Origin Airport
3. Destination Airport
4. Flight Number
5. Departure Time Block

the dependent variable is the flight will be delay or not: 0 means on time, 1 means delay. The program read about 17 thousand records. And here is the input data:



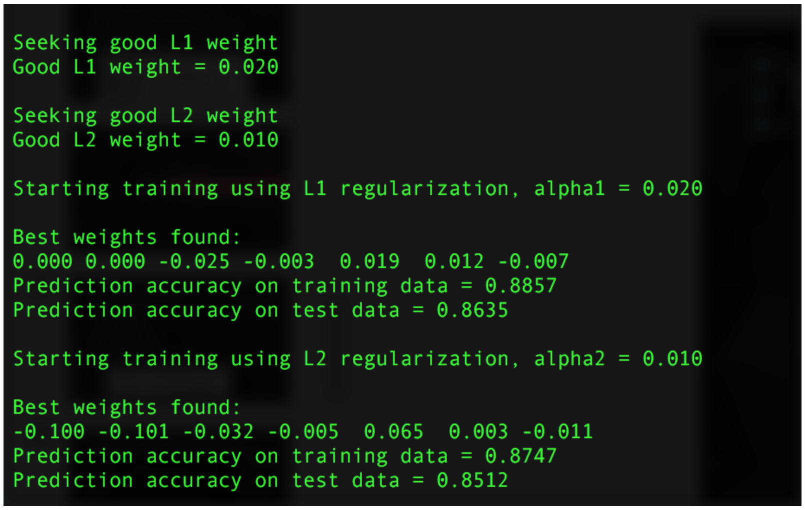
	A	B	C	D	E	F	G
1	1	B6	228	13204	11618	1000-1059	1
2	1	B6	231	14869	12954	1200-1259	0
3	1	B6	232	12954	14869	0900-0959	0
4	1	B6	233	10785	12478	0001-0559	1
5	1	B6	234	12478	10785	2200-2259	1
6	1	B6	249	11278	15304	1100-1159	1

First we train the data with no regularization. and here is some of the training data and test data. the prediction accuracy is about 80%.



```
Training data:

[ 0]   50.00 216.00 171.00 263.00 250.00 527.00 0.00
[ 1]   51.00 226.00 284.00 265.00 262.00 521.00 1.00
[ 2]   50.00 188.00 222.00 262.00 261.00 525.00 0.00
[ 3]   50.00 198.00 229.00 263.00 246.00 515.00 0.00
[11767]   50.00 198.00 275.00 260.00 261.00 517.00 0.00


Test data:

[ 0]   49.00 188.00 235.00 260.00 250.00 523.00 1.00
[ 1]   50.00 198.00 265.00 246.00 250.00 519.00 0.00
[ 2]   50.00 198.00 275.00 261.00 252.00 525.00 0.00
[2941]   49.00 188.00 229.00 262.00 250.00 523.00 1.00


Starting training using no regularization..

Best weights found:
-0.819 -15.827 -17.955  6.909  19.974 -18.587  3.142
Prediction accuracy on training data = 0.8169
Prediction accuracy on test data = 0.8154
```

Then we optimized our algorithm by using L1 regularization and L2 regularization. The prediction accuracy on test data improved by 5%.



```
Seeking good L1 weight
Good L1 weight = 0.020

Seeking good L2 weight
Good L2 weight = 0.010

Starting training using L1 regularization, alpha1 = 0.020

Best weights found:
0.000 0.000 -0.025 -0.003  0.019  0.012 -0.007
Prediction accuracy on training data = 0.8857
Prediction accuracy on test data = 0.8635

Starting training using L2 regularization, alpha2 = 0.010

Best weights found:
-0.100 -0.101 -0.032 -0.005  0.065  0.003 -0.011
Prediction accuracy on training data = 0.8747
Prediction accuracy on test data = 0.8512
```

And the final prediction accuracy we got is 86%.

6. Front End development: Airline database search engine.

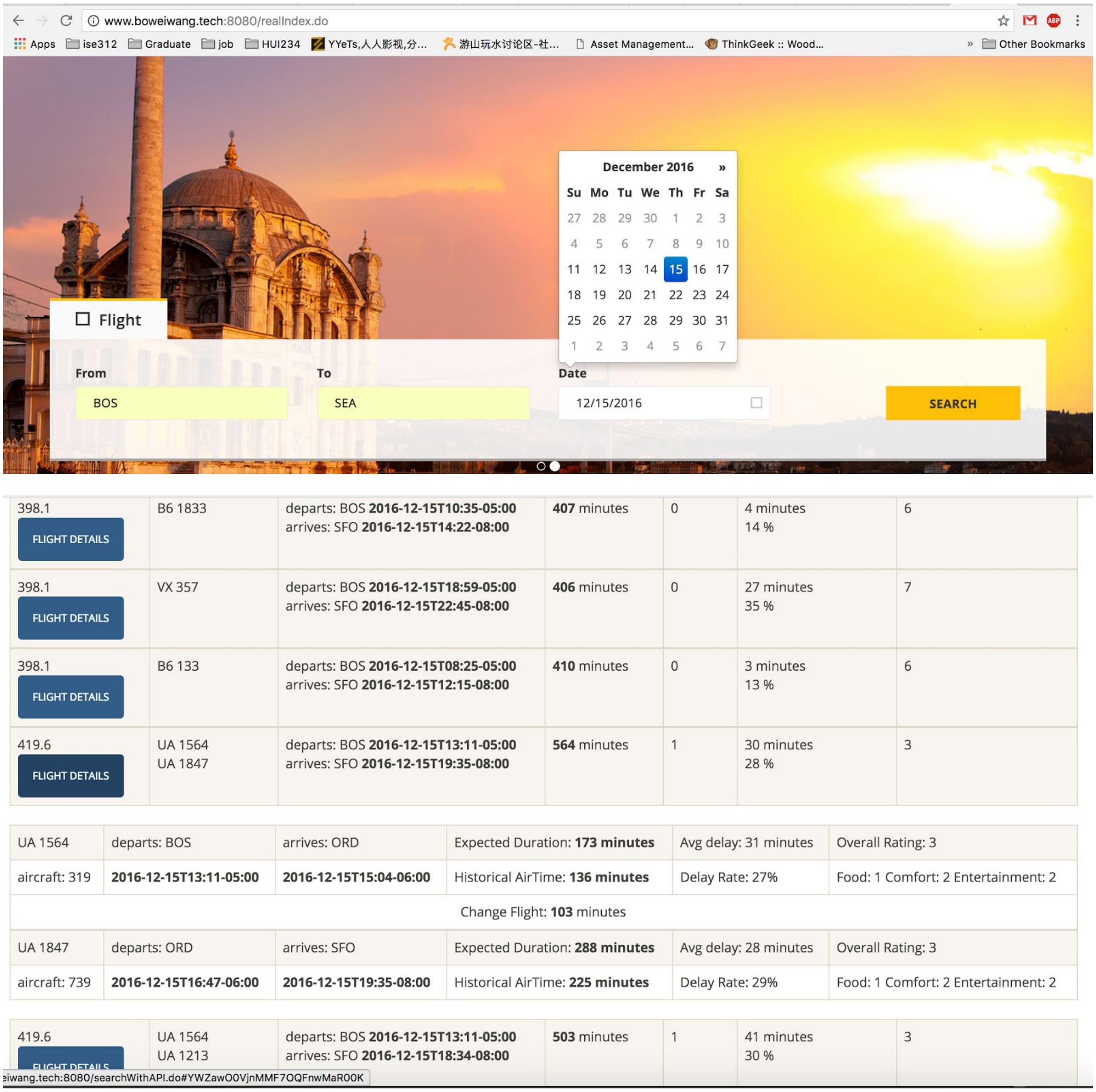
In addition to the previous analysis, we created a front end system to better serve the user. In this system, data comes from two main source:

1. Real time pricing and routing data from Google QPX Express API.
2. Historical flight delay data and airline reputation rating data from HBase.

For the Backend architecture, we used Spring MVC framework the handle frontend request. The frontend web page was created using HTML, Javascript, JQuery, AngularJS and BootStrap. The system features automatic frontend data binding and column sort using AngularJS.

End user could sort the result based on price, time, airline, duration, stops, delay time and probability and airline rating.

The website has been fully deployed to Amazon AWS EC2 instance with glassfish servlet container. I also added my personal domain to the website for easier access.



December 2016						»
Su	Mo	Tu	We	Th	Fr	Sa
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

From: BOS

To: SEA

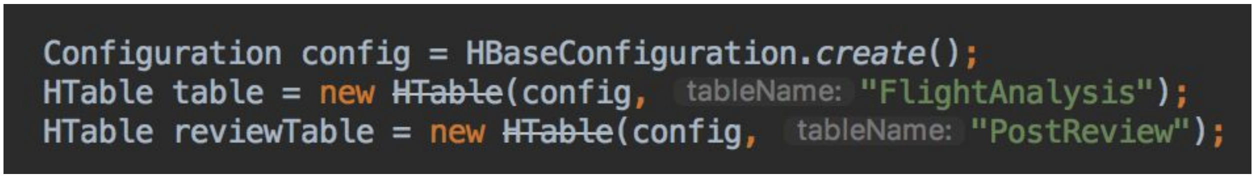
Date: 12/15/2016

398.1 FLIGHT DETAILS	B6 1833	departs: BOS **2016-12-15T10:35-05:00** arrives: SFO **2016-12-15T14:22-08:00**	**407** minutes	0	4 minutes 14 %	6
398.1 FLIGHT DETAILS	VX 357	departs: BOS **2016-12-15T18:59-05:00** arrives: SFO **2016-12-15T22:45-08:00**	**406** minutes	0	27 minutes 35 %	7
398.1 FLIGHT DETAILS	B6 133	departs: BOS **2016-12-15T08:25-05:00** arrives: SFO **2016-12-15T12:15-08:00**	**410** minutes	0	3 minutes 13 %	6
419.6 FLIGHT DETAILS	UA 1564 UA 1847	departs: BOS **2016-12-15T13:11-05:00** arrives: SFO **2016-12-15T19:35-08:00**	**564** minutes	1	30 minutes 28 %	3

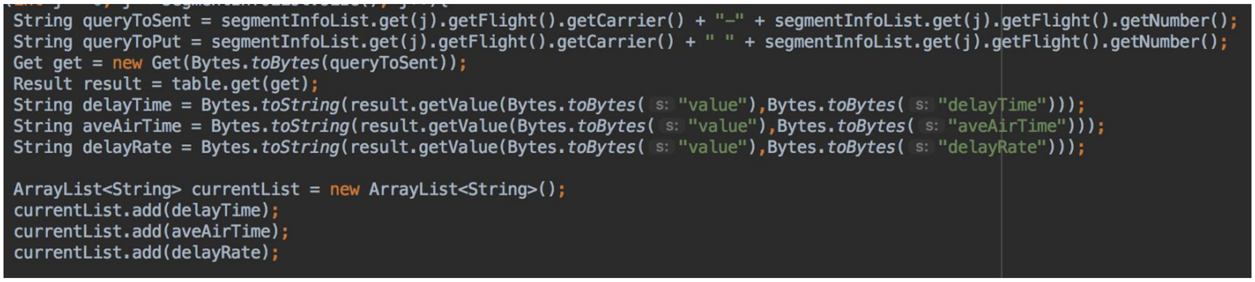
UA 1564	departs: BOS	arrives: ORD	Expected Duration: **173 minutes**	Avg delay: 31 minutes	Overall Rating: 3
aircraft: 319	**2016-12-15T13:11-05:00**	**2016-12-15T15:04-06:00**	Historical AirTime: **136 minutes**	Delay Rate: 27%	Food: 1 Comfort: 2 Entertainment: 2
Change Flight: **103** minutes					
UA 1847	departs: ORD	arrives: SFO	Expected Duration: **288 minutes**	Avg delay: 28 minutes	Overall Rating: 3
aircraft: 739	**2016-12-15T16:47-06:00**	**2016-12-15T19:35-08:00**	Historical AirTime: **225 minutes**	Delay Rate: 29%	Food: 1 Comfort: 2 Entertainment: 2

419.6 FLIGHT DETAILS	UA 1564 UA 1213	departs: BOS **2016-12-15T13:11-05:00** arrives: SFO **2016-12-15T18:34-08:00**	**503** minutes	1	41 minutes 30 %	3

eiwang.tech:8080/searchWithAPI.do#YWZawO0VjnMMF7OQFnwMaR0OK



```
Configuration config = HBaseConfiguration.create();
HTable table = new ~~HTable~~(config, tableName: "FlightAnalysis");
HTable reviewTable = new ~~HTable~~(config, tableName: "PostReview");
```



```
String queryToSent = segmentInfoList.get(j).getFlight().getCarrier() + "-" + segmentInfoList.get(j).getFlight().getNumber();
String queryToPut = segmentInfoList.get(j).getFlight().getCarrier() + " " + segmentInfoList.get(j).getFlight().getNumber();
Get get = new Get(Bytes.toBytes(queryToSent));
Result result = table.get(get);
String delayTime = Bytes.toString(result.getValue(Bytes.toBytes( s: "value"),Bytes.toBytes( s: "delayTime")));
String aveAirTime = Bytes.toString(result.getValue(Bytes.toBytes( s: "value"),Bytes.toBytes( s: "aveAirTime")));
String delayRate = Bytes.toString(result.getValue(Bytes.toBytes( s: "value"),Bytes.toBytes( s: "delayRate")));

ArrayList<String> currentList = new ArrayList<String>();
currentList.add(delayTime);
currentList.add(aveAirTime);
currentList.add(delayRate);
```

7.Conclusion

By doing this project we analyzed why flight delayed and the overall on-time performance of U.S flights. By using R we achieved data visualization so that we can observe the data in different perspectives. By using Hadoop we analyzed the past performance for each flight and calculate their delay rate, average review score and so on. Then we use logistic regression to predict whether a flight will be delayed in the future. At end we designed and implemented an user interface that provides informations we get while he is buying tickets online.

8. Future scope.

For the future development, we are planning to extend another weather data source to help the customer have a better understanding about what should him/her expect during the travel.
We also planning to combine all the different delay factors like historical delay time, delay probability, weather factors, routing and airline reputation to build a comprehensive calculation model to make the decision for the customer.
Additionally, we would like to add another purchase link for the end user to directly purchase the ticket through our web site.