National College of Ireland

Data Warehousing and Business Intelligence Project

on

Impact of Revenue of Major Airlines on Passenger Sentiments and Ratings and the correlation between the performance of Major Airlines on Global Air Traffic Growth

Vikramaditya Tatke

17163668

MSc/PGDip Data Analytics – 2018/9

Submitted to: Dr. Simon Caton

National College of Ireland
Project Submission Sheet – 2017/2018
School of Computing

Student Name:	Vikramaditya Tatke
Student ID:	17163668
Programme:	MSc Data Analytics
Year:	2018/9
Module:	Data Warehousing and Business Intelligence
Lecturer:	Dr. Simon Caton
Submission Due Date:	26/11/2018
Project Title:	Impact of Revenue of Major Airlines on Passenger Sentiments and Ratings and the correlation between the performance of Major Airlines on Global Air Traffic Growth

I hereby certify that the information contained in this (my submission) is information pertaining to my own individual work that I conducted for this project. All information other than my own contribution is fully and appropriately referenced and listed in the relevant bibliography section. I assert that I have not referred to any work(s) other than those listed. I also include my TurnItIn report with this submission.

ALL materials used must be referenced in the bibliography section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is an act of plagiarism and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	
Date:	January 7, 2019

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
3. Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Table 1: Mark sheet – do not edit

Criteria	Mark Awarded	Comment(s)
Objectives	of 5	
Related Work	of 10	
Data	of 25	
ETL	of 20	
Application	of 30	
Video	of 10	
Presentation	of 10	
Total	of 100	

Project Check List

This section capture the core requirements that the project entails represented as a check list for convenience.

- ☒ Used LaTeX template
- ☒ Three Business Requirements listed in introduction
- ☒ At least one structured data source
- ☒ At least one unstructured data source
- ☒ At least three sources of data
- ☒ Described all sources of data
- ☒ All sources of data are less than one year old, i.e. released after 17/09/2017
- ☒ Inserted and discussed star schema
- ☒ Completed logical data map
- ☒ Discussed the high level ETL strategy
- ☒ Provided 3 BI queries
- ☒ Detailed the sources of data used in each query
- ☒ Discussed the implications of results in each query
- ☒ Reviewed at least 5-10 appropriate papers on topic of your DWBI project

Impact of Revenue of Major Airlines on Passenger Sentiments and Ratings and the correlation between the performance of Major Airlines on Global Air Traffic Growth

Vikramaditya Tatke
17163668

January 7, 2019

Abstract

The following report aims at discovering the correlation between between the performance of Major Airlines on Global Air Traffic Growth. We have selected Austrian Airlines, Etihad Airways, KLM, Lufthansa, Qatar Airways, Singapore Airlines, South African Airways and Turkish Airlines for this study. In addition to that, it also shows the relation between the performance measures of the airlines with the sentiments of the passenger and their ratings. It takes into consideration the global traffic growth for 5 years and relates it with he total number of passengers carried by these airlines for these selected years.

1 Introduction

The airline industry faces negative criticism even after its rigorous efforts to maintain standards and security. Passengers think that the satisfaction of air travel is in the class they select. Hence, Following report aims to identify the correlation between the sentiments shown by the passengers towards the major airlines in the flying industry and their selected class of travel. Passengers might resort to choosing different airlines if they are not satisfied with their experience (Subha R, 2012). In addition to that, with the constant growth in air traffic it is becoming increasingly difficult to manage the ever increasing data generated by the passengers. According to this article (IATA, 2017) the number of air passengers is going to doubly by 2036 at reach to 7.8 Billion per year. Furthermore, the effect of certain airline measures like load factor can directly affect the revenue of the company (Stalnaker, et al., 2017-2018). The higher the load factor the busier the aircraft. Certain studies mentioned in this article (Kunkle, n.d.) showed that crowded aircrafts can give rise to air rage among passengers. The following report tries to find a correlation between various airline measures and passenger ratings.

First requirement of our project would be to analyze the influence of the total number of passengers carried by the major airline on the air traffic growth worldwide.

Second requirement of our project would be to study the correlation between the travel satisfaction of the passengers and the class of the cabin by which they have travelled.

Source	Type	Brief Summary
Kaggle	Structured	Ratings on various parameters of airlines.
Statista	Structured	Scores the best airlines on a scale of 1 to 10. Contains global air traffic growth percentage
Twitter	Unstructured	275,000 tweets mentioning the selected airlines to analyze the sentiments of the passengers of these specific airlines.
Aviation Edge API	Structured	Provides parameters such as to calculate airline performance.
Wikipedia	Structured	Contains revenues for airlines. Contains number of passengers carried by airlines by years.
Airline Data Project	Structured	Contains amount spent on various factors by airlines in million dollars.

Table 2: Summary of sources of data used in the project

Final requirement would be to examine the impact of revenue and various other airline measures on passenger ratings.

2 Data Sources

All the data sources used in the project are listed below along with a description for each.

2.1 Source 1: Kaggle

The kaggle airline dataset downloaded this source from: `https://www.kaggle.com/arjhbholu/airline-dataset-mining/downloads/airline-dataset-mining.zip/5` This dataset has 14 columns of information on reviews and ratings given by 41218 passengers. We have selected the 8 airlines with a major presence in the industry. Hence, we will be using only the ratings for Austrian Airlines, Etihad Airways, KLM, Lufthansa, Qatar Airways, Singapore Airlines, South African Airways and Turkish Airlines.

2.2 Source 2: Statista

The Statista Air traffic Growth - Annual Passenger Demand is downloaded from the following source: `https://www.statista.com/statistics/193533/growth-of-global-air-traffic-` This dataset has 2 columns of information on passenger growth percentage by year. We have selected the 8 airlines with a major presence in the industry. This dataset will be used with the Wikipedia dataset mentioned below. The Statista Best Rated Airlines in the World, is downloaded from the following source: `https://www.statista.com/statistics/868128/best-airlines-according-to-airhelp-worldwide/` We have selected the 8 airlines with a major presence in the industry namely, Austrian Airlines, Etihad Airways, KLM, Lufthansa, Qatar Airways, Singapore Airlines, South African Airways and Turkish Airlines based on this dataset. It contains a score for each of these airlines.

2.3 Source 3: Twitter

The kaggle movies dataset downloaded this source from: This dataset has 14 columns of information on reviews and ratings given by 41218 passengers. We have selected the 8 airlines with a major presence in the industry. Hence, we will be using only the ratings for Austrian Airlines, Etihad Airways, KLM, Lufthansa, Qatar Airways, Singapore Airlines, South African Airways and Turkish Airlines.

2.4 Source 4: Aviation Edge API

The aviation edge dataset is downloaded from this source: `https://aviation-edge.com/v2/public/airlineDatabase?key=` It contains various airline measures such as Avg Fleet Age, Number of Airplanes with each company, etc for many different airlines. We will be using the data only for the select 8 airlines.

2.5 Source 5: Wikipedia

The first Wikipedia airline revenue dataset is downloaded from the source: `https://en.wikipedia.org/wiki/World27s_largest_airlines` %This dataset comprises of revenues of largest airlines in the world in Billion dollars. The second Wikipedia datast consisting of 'passengers carried by year' has also been downloaded from the same source. As the source contains years in individual columns we will transpose and convert the data in such a form that the all the years are accommodated in one single column for the ease of mapping.

2.6 Source 6: Airline Data Project

This is a yearly updated source of data. We have downloaded Gallons of Fuel per Block Hour and Load Factor of Airlines from this data source. We have combined the last columns of the dataset in one table. Only the last columns were chosen in order to prevent the violation of a project requirement. `http://web.mit.edu/airlinedata/www/Revenue&Related.html`

3 Related Work

On the one hand, as discussed in the (Subha R, 2012) airlines might lose their customers if they cannot live up to the expectations and provide adequate travel satisfaction for all classes of travel. Rapidly growing air traffic means a steep growth in customers and the data generated by customers. On the other hand, some studies show that the passengers did not perceive any difference between different carriers performing on the same level in the industry. It directly increases the amount of feedback received by the airlines. Our projects aims to convert this feedback given by the users into business critical knowledge. (IATA, 2017) discuss the increase in the of passengers boarding the airplanes to nearly double the current amount by the year 2036 and reach to 7.8 Billion per year. Furthermore, the effect of certain airline measures like load factor can directly affect the revenue of the company (Stalnaker, et al., 2017-2018). The higher the load factor the busier the aircraft. Certain studies mentioned in this article (Kunkle, n.d.) showed that crowded aircrafts can give rise to air rage among passengers. This certainly

is a negative effect of the revised airline policies wherein the legroom has been reduced to accommodate more seats in the economy and premium economy segments. This project aims at adding to the existing work and finding out the relation between user sentiments and revenue of the airlines. This could definitely lead to an increased profitability for the airlines if they make proper use of their social media presence. This would result in the complaints of the passengers actually being resolved and incremental growth with every feedback.

4 Data Model

We have modeled the data by using a star schema. It is designed to accommodate the selected entities and build relationships between them. Our model uses four conformed dimension tables, for Sentiments, Airline Details and Ratings. The names of these tables are 'DimAirline', 'DimRatings' and 'DimSentiments' and 'DimCabin' and 3 descriptive dimension tables, 'DimPassengers', 'DimTrafficGrowth' and 'DimCountry'.

The table 'DimAirline' contains the critical functional details of airline companies that include the number of airplanes owned by every company (fleet size), average fleet age, airline score, total revenue, operating revenue and operating expense, all in billions of dollars, gallons of fuel used per block hour. The table has been created by merging data from three different sources which are Wikipedia, Statista and Aviation Edge API. The tables can easily accommodate further additions without making changes in the star schema. 'AirlineID' holds 8 values starting from 1 for each airline. This column has been mapped to the 'AirlineID' column in the 'DimRatings'. Furthermore, this table is mapped to the fact table using the 'AirlineID' column.

The 'DimRatings' table consists of user ratings on a variety of facts which will be relevant for this report, such as ratings for Comfort, Staff, Food, Entertainment and Value for Money. It contains an ID for each rating which automatically increments with a new entry into the table. Again, this table can accommodate changes without changing the star schema. Sentiments of the passengers are contained within the table 'DimSentiments'. Every sentiment falls in a group and has a group ID. Also, the value of each sentiment decides its magnitude. Even this table can accommodate all the changes while keeping the star schema intact.

'DimCabin' tables comprises of the types of cabins available in each airline, each complimented with a 'CabinGroupID' which can be loaded into the fact table. This table also contains 'AirlineID' in order to map the types of cabins to the airline company. This table is formed from the Kaggle Airline ratings data source.

The table 'DimPassengers' includes the number of passengers in millions for the specified years by each airline. It is designed using the Wikipedia data source and the coloumn 'PID' is used for loading the table from this table into the fact table. Another of our dimension tables, 'DimTrafficGrowth' is formed from the Statista Data Source and contains the percentage of growth in worldwide air traffic for selected number of years. We can use this table to learn the effects of major airlines on the worldwide air traffic growth. The column 'YearID' is mapped to the column 'YearID' in the 'DimPassengers' table.

Origin countries of the selected airline companies are held in the 'DimCountry' table. The column 'CID' is used to map the values into the fact table. This table was designed by data from the Wikipedia data source and the airlines were mapped with their country of origin.

All the above mentioned tables are created from SSIS using the Execute SQL task. These tables have been designed to cater to our business requirements and we require all the tables in order to answer our queries. The revenue and the measures of airlines will be used to determine its effect on customer ratings and the correlation between the passengers carried by major airlines and the global air traffic will be obtained. Furthermore, we will use these tables to study the effect of the chosen class of travel has on the sentiments or emotions of the passengers.

The star schema is shows in Figure 1 below.

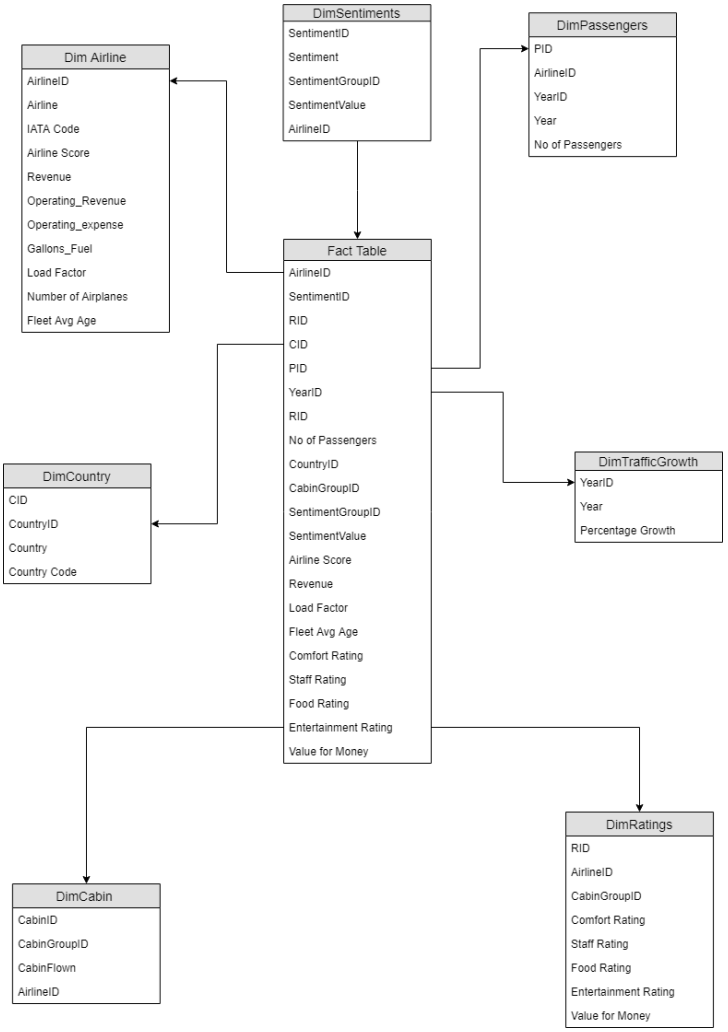

Figure 1: Deployed Star Schema

5 Logical Data Map

Table 3: Logical Data Map describing all transformations, sources and destinations for all components of the data model illustrated in Figure 1

Source	Column	Destination	Column	Type	Transformation
1	Airline	DimAirline	Airline	Dimension	Names of airlines, removed irrelevant names
1	Cabin	DimCabin	CabinFlown	Dimension	Distinct value for cabin class $
2	Score	FactTable	Airline Score	Fact	Score of airline
5	Year	FactTable	Airline Score	Fact	Removed irrelevant years $
2	No of Passengers	FactTable	No of Passengers	Fact	No of passengers carried in millions.
3	Sentiment	DimSentiment	Sentiment	Dimension	3 Calculated sentiments $
3	SentimentGroupID	FactTable	Sentiment	Fact	3 types of Calculated sentiments
3	SentimentValue	FactTable	Sentiment	Fact	3 Magnitude of sentiment. $
4	Country	DimCountry	Country	Dimension	Origin Country of Airline companies. $
1	Food Rating	FactTable	Food Rating	Fact	Rating for Food
1	Comfort Rating	FactTable	Staff Rating	Fact	Rating for Comfort $
1	Staff Rating	FactTable	Staff Rating	Fact	Rating for Staff
1	Entertainment Rating	FactTable	Entertainment Rating	Fact	Rating for Entertainment $
1	Value for Money	FactTable	Value for Money	Fact	Value for Money rating
5	Revenue	FactTable	Revenue	Fact	Rounded to nearest million $
4	Fleet Avg Age	FactTable	Fleet Avg Age	Fact	Primary Age of the fleet of the airline company.
6	Load Factor	FactTable	Load Factor	Fact	Percentage Load Factor $

6 ETL Process

We have carried out the Extraction, Transformation and Loading (ETL) of data using SQL Server Integration Services (SSIS) in Microsoft Visual Studio 2017. With the help of SSIS the data was transferred into a database which was deployed on the SQL Server using the SQL Server Management Services (SSMS) tool.

R and Python were used as the means of cleaning the data acquired from the above listed data sources. Execute Process Tasks handled the execution of R Scripts, one of which was internally linked to a Python script. Different Execute Process Tasks were created for each of the required raw data tables and everyone of them was linked to a Data Flow task. With every R script a data file was read from the local storage and then cleaned and transformed as per the requirements. Thereafter, CSV file was generated within the same script, which was then loaded into the database deployed on the SQL Server.

The first dataset was combined from the data files downloaded from Airline Data Project and Aviation Edge API, Wikipedia and Statista. These were considered as performance measures for the airlines. Data from the Airline Data Project data files contained periodically updated historical data and hence needed to be trimmed to fit the project requirements. Only the data from the last year (2017) was kept and the missing values for each data file were replaced with median of values from the previous years. The Aviation Edge API provided a variety of measures specific to the airlines. Again, only the required measures were kept, and others were removed. Revenue for the airline companies was extracted from Wikipedia tables and the scores for airlines were grabbed from the Statista data source. All these datafiles were cleaned and combined to form a large table consisting of performance measures of the chosen airlines. This was converted into a CSV file to be loaded into the database.

The second dataset downloaded from Kaggle contains reviews and user ratings for different aspects such as Entertainment, Staff, Food, Value for Money, etc. for several airlines. The irrelevant columns were removed, and the dataset was trimmed to accommodate only our airlines of choice. The result was stored in a CSV file in order to be loaded into the database.

The third dataset was a part extracted from the Kaggle dataset consisting of Cabin Details. Each Cabin was given a 'CabinGroupID' and all the rows were given individual 'CabinID's. Also, all the cabins were mapped to their specific airlines with an 'AirlineID'. This dataset was finally converted into a CSV to be transferred to the database on the SQL Server.

The fourth dataset was developed from the Twitter data. Firstly, twitter data was downloaded into MongoDB using Python and Tweepy. 200,000 tweets were collected into the database. Python was connected with the MongoDB collection and the database was queried to extract the data from the database and convert it into a dataframe only with the required columns, i.e., the 'text' field. The resultant column consisted of all the text fields which contained the name of our chosen airlines and every row was given a name (or ID) based on the airline name it contained. Upon inspection, it was observed that insufficient data had been downloaded for small subset of the chosen airlines and hence to compensate for the loss an R script was run to download another 75,000 tweets from twitter and store it in a dataframe. This dataframe was transformed in a similar manner to contain only the 'text' and 'airline name' (ID) columns. Every row was given an airline name to facilitate sentiment analysis for specific airlines. Moving on, sentiment analysis

was carried out on the data, which resulted in 6 different sentiments with specific values for each airline. Out of these 6 values 3 values, namely, ‘anger‘, ‘joy‘ and ‘trust‘ were kept while others were removed. Every sentiment was given a ‘SentimentGroupID‘ and a column of ‘AirlineID‘ was also added to the dataframe, which was further converted into a CSV to be loaded in the database.

The fifth dataset consisted of the percentage of global air traffic growth, which was again downloaded from Statista. Every year was given a ‘YearID‘ and then converted into a CSV to be loaded in the database.

The sixth dataset was put together from a Wikipedia table, which consisted of the number of passengers carried by the chosen airline. A ‘PID‘ column was added and the data was transposed to have all the years in one column and each year was provided with a ‘YearID‘. A CSV was prepared prepared from this data which would be transferred to the database.

The seventh dataset consisted of origin countries of each airline company along with an ‘AirlineID‘. Country code from the Aviation Edge API was also added to this dataset. Also, a column ‘CID‘ was added which was unique to every row. A CSV was created from this data to be transferred to the database.

All the CSV files are transferred to the SQL Server database by using SSIS using separate Execute SQL tasks for each CSV. Connection manager is specified for each file and relevant tables are created in the database. After the data is loaded in the raw data tables in the SQL Server database and Execute SQL task is created to create all the 7 dimension tables. Now, in order to insert the values from the raw data tables into the dimension tables another Execute SQL Task is created. This task selects the specified data from the raw dimension tables and inserts it into the dimension tables. Fortunately, no make major modifications had to be made in the structure of the tables because of the extensive and thoughtful ETL process.

Moving on to the second last step of our ETL, we create the fact table to accommodate all the INT values present in the dimension tables by using a third Execute SQL task. A fourth Execute SQL task created to transfer data from dimension tables to the fact tables. The primary keys of each of the dimension tables namely, ‘AirlineID‘, ‘SentimentID‘, ‘RID‘, ‘CID‘, ‘PID‘, ‘CabinID‘ and ‘YearID‘, which are referenced in the fact table as foreign keys are also presented in the fact table, as it helps in mapping of the values.

The ETL process is concluded by adding a final task to the SSIS flow. The Analysis Services Processing Task is used to select the necessary measure groups and it helps in ensuring the proper selection of dimension tables. A connection to the database on the SQL Server is specified and the correct user credentials are entered to provide proper authentication. Data Source and Data Source View are specified after carefully select the dimension and measure tables before creating and deploying a MOLAP (Multidimensional Online Analytical Processing) cube on the SQL Server.

Furthermore, in order to visualize the results of the BI Queries the cube is loaded into a visualization software, Tableau (2018.3). Here the values are plotted on graphs in order to visually represent all the BI Queries.

7 Application

In order to analyze the impact of the passengers carried by major airlines on the growth of air traffic (worldwide) we have used the data to visualize the trends in the industry

and make observations. We have selected all the 8 major airlines to demonstrate our observations. We selected the parameters essential for answering our BI queries and in turn satisfy the business requirements stated previously. We have used Tableau 2018.3 to as the visualization tool and to create an intuitive dashboard. It was connected directly with the Microsoft Analysis Services to import the data. While using this dashboard it is not necessary to incorporate all of the airlines and we can study on a subset of airlines as well, if required.

7.1 BI Query 1: How does the number of passengers carried by the major airline affect the worldwide traffic growth?

Statista and Wikipedia are the data sources that contributed to this query. The bar graphs show the comparison between the number of passengers carried by major airlines through 5 years with YearID 1 = 2011 and YearID 5 = 2015. We can see that for most airlines, for example AirlineID 1, which corresponds to Austrian Airlines the percentage growth in global air traffic increases with an exponential increase in the passengers carried. Similar observation can be made for AirlineID 2, which corresponds to Etihad Airways. This is illustrated in Figure 8.

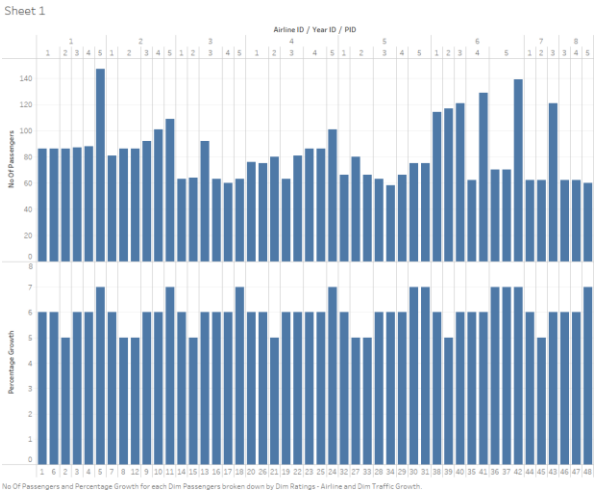

Figure 2: Results for BI Query 1

7.2 BI Query 2: Does the cabin class affect the sentiment of the passengers? In other words, is there a correlation between the travel satisfaction of the passengers and the class of the cabin by which they have travelled?

Kaggle and Twitter are the data sources that contributed to this query. We can see that for most airlines, the people who are travelling via Business Class (CabinGroupID = 4) have a very high positive sentiment towards the airline. This can be observed for the KLM Airlines (AirlineID = 3). We can also observe that a person travelling by the same airline in an Economy class (CabinGroupID = 1) as a very low sentiment of joy (SentimentGroupID = 2). It is illustrated in Figure **??**.

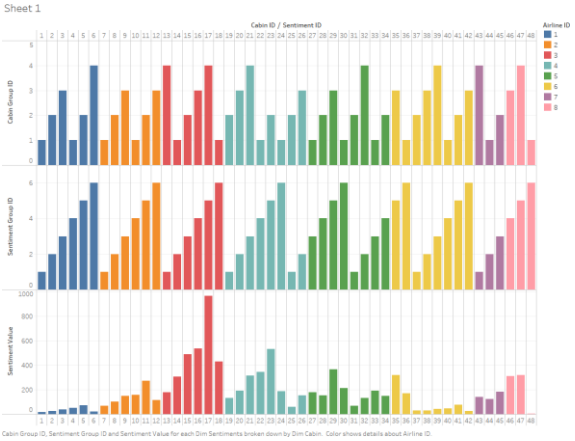

Figure 3: Results for BI Query 2

7.3 BI Query 3: What is the effect of revenue and airline measures on passenger ratings?

Statista, Wikipedia, Airline Data Project, Aviation Egde and Kaggle are the data sources that contributed to this query. Here we have combined the ratings for different airlines as all the airlines have similar number of ratings to study the correlation between revenue, airline measures and passenger ratings. Although the relation between revenue, measures are ratings is not plausible, the relation between load factor, revenue and avg fleet age appears to be directly proportional for most airlines. This is illustrated in Figure 4.

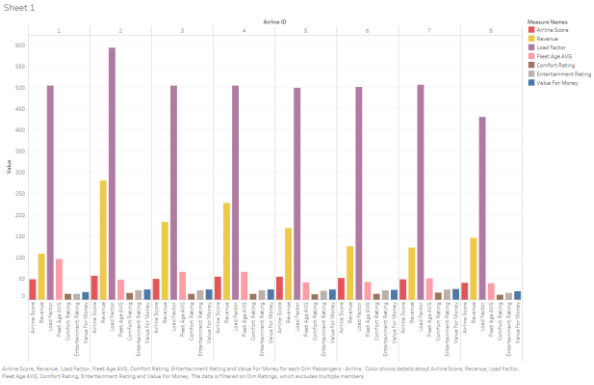

Figure 4: Results for BI Query 3

7.4 Discussion

We have found that the relation between travel satisfaction is dependent upon the class by which people chose to travel. This makes logical sense as the Business class is more spacious and provides better meals in general. These are the factors that strongly affect the mood of people. Also, as discussed previously in (Kunkle, n.d.) that crowded place tend to induce negative emotions. This can be proved by our findings. A more spacious environment means better travel satisfaction. Moreover, we can also observe that the last year in our data has the highest number of passengers for every airline. This indicates growth in the number of people flying by air which goes hand in hand with the results in these article (IATA, 2017).

Figure 5: Analysis on Twitter 'text' field

8 Conclusion and Future Work

We have tried to find the correlation between various user sentiments, which are a result of the overall satisfaction provided by the airline services to their passengers and the revenue generated for the airlines. This could also translate into a correlation of the quality of service provided by the airline and their respective profits. As we have limited data some of the observations could be less accurate. This study has found the that there is a directly proportional relationship between the airline measures and its revenue and there is little evidence to show that there is a relation between the actual ratings given by users and the revenue generated by the airlines. On the contrary, the user sentiments show a strong relation with the chosen cabin class and we can safely say that if passengers travel by paying a premium over the regular fees are happier travellers than the rest. Also, the best rated airlines according to Statista show a sufficient contribution to the global air traffic growth. This project can be extended further to inculcate factors which have a higher correlation with each other and help us to increase customer satisfaction and generate higher revenue for the airlines. Furthermore, instead of using Twitter as the only source of analyzing passenger sentiments we can accommodate news articles and more to more accurately analyze sentiments. Furthermore, it is easy to make changes in the schema of our Data Warehouse with the change in business requirements. IATA, 2017. 2036 Forecast Reveals Air Passengers Will Nearly Double to 7.8 Billion. [Online] Available at: https://www.iata.org/pressroom/pr/Pages/2017-10-24-01.aspx [Accessed January 2019]. Kunkle, F., n.d. Study shows crowded planes increases air rage in passengers. [Online] Available at: http://www.traveller.com.au/study-shows-crowded-planes-increases-air-rage-in-passengers-gwkdx1 [Accessed Janurary 2019]. Stalnaker, T., Khalid, U., Taylor, A. Alport, G., 2017-2018. Airline Economic Analysis, s.l.: Oliver. Subha, . D. M. V. R, A., 2012. A Study on service quality and passenger satisfaction on Indian airlines. International Journal of Multidisciplinary Research , 2(2).

References

Bonzanini, M., 2015. Mining Twitter Data with Python (Part 1: Collecting data). [Online]
Available at: https://marcobonzanini.com/2015/03/02/mining-twitter-data-with-python-part-1/
[Accessed 12 November 2018].

IATA, 2017. 2036 Forecast Reveals Air Passengers Will Nearly Double to 7.8 Billion. [Online]
Available at: https://www.iata.org/pressroom/pr/Pages/2017-10-24-01.aspx
[Accessed January 2019].

Kunkle, F., n.d. Study shows crowded planes increases air rage in passengers. [Online]
Available at: http://www.traveller.com.au/study-shows-crowded-planes-increases-air-rage-in-passengers-gwkdx1
[Accessed Janurary 2019].

Stalnaker, T., Khalid, U., Taylor, A. & Alport, G., 2017-2018. Airline Economic Analysis, s.l.: Oliver.

Subha, . D. M. V. & R, A., 2012. A Study on service quality and passenger satisfaction on Indian airlines. International Journal of Multidisciplinary Research , 2(2).

Appendix

R code

```
#CleaningstartsforBestairlines(Statista)dataset.
library(tidyverse)
library(dplyr)
library(data.table)
library(matrixStats)
library(tidyr)

bestAirlinesData <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.of.Ire
airlineRevenue <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.of.Irela
colnames(bestAirlinesData) <-c("Airline","Airline.Score")
bestAirlinesData <- bestAirlinesData[c(-1:-3, -10:-13, -15:-17),]
row.names(bestAirlinesData) <-1:nrow(bestAirlinesData)
bestAirlinesData$Airline <-gsub("(.*),.*","\\1", bestAirlinesData$Airline)
bestAirlinesData[7,1] <-"KLM"
bestAirlinesData <- bestAirlinesData %>% arrange(Airline)
setDT(bestAirlinesData)[, AirlineID := .GRP,by= Airline]
bestAirlinesData$Country <- airlineRevenue$Country
setDT(bestAirlinesData)[, CountryID := .GRP,by= Country]
bestAirlinesData <- bestAirlinesData %>% select(AirlineID, Airline, CountryI
write.csv(bestAirlinesData,file="C:\\Users\\vikra\\OneDrive.-.National.College.of
#CleaningendsforBestairlines(Statista)dataset.
```

```r
#Downloadingandcleaning IATAData
library(dplyr)
library(curl)
library(httr)
library(jsonlite)
library(tidyr)  lib
rary(data.table)

url<-"https://aviation-edge.com/v2/public/airlineDatabase?key=e5c06d-d3426f"
#GettingdatafromAviationEdgeAPI
differentAirlines <- fromJSON(url)
codelata <-c("OS","EY","KL","LH","QR","SQ","SA","TK")
alt_result <- differentAirlines %>% filter(codelataAirline%in%codelata)
alt_result[alt_result==""]<-NA
alt_result <- alt_result %>%drop_na() %>% select(nameAirline, codelataAirline
names(alt_result) <-c("Airline","IATA.Code","Number.of.Airplanes","Fleet.Age(AVG)"
alt_result[1,1] <-"Turkish.Airlines"
alt_result[2,1] <-"Lufthansa"
alt_result[6,1] <-"Singapore.Airlines"
alt_result[7,1] <-"Austrian.Airlines"
alt_result[8,1] <-"South.African.Airways"
alt_result <- alt_result %>% arrange(Airline)
```

```
setDT(alt_result)[, AirlineID := .GRP,by= Airline]
setDT(alt_result)[, CountryID := .GRP,by= Country]
alt_result <- alt_result %>% arrange(Airline) %>% select(AirlineID, Airline,
`IATA Code`, `Number of Airplanes`, `Fleet Age(AVG)`, `Country Code`)
write.csv(alt_result,file="C:\\Users\\vikra\\OneDrive.-.National.College.of.Irela


library(tidyverse)
library(dplyr)
library(data.table)
library(matrixStats)
library(tidyr)
library(stringr)
library(plyr)

#Cleaningbeginsforairlineratingsdataset.
AirlinesData <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland
segregatedAirlinesData <- AirlinesData[apply(AirlinesData,1,function(x) {any(c
segregatedAirlinesData$content <- NULL
segregatedAirlinesData$author <- NULL
segregatedAirlinesData$Month <-NULL                    s
egregatedAirlinesData$recommended <- NULL             se
gregatedAirlinesData$overall_rating <- NULL          se
gregatedAirlinesData$X <- NULL                       se
gregatedAirlinesData$`Author Country` <- NULL      sapp
ly(segregatedAirlinesData,function(x)sum(is.na(x)))
segregatedAirlinesData$airline_name <- str_replace_all(segregatedAirlinesData
                                                       c("austrian-airlines"="
                                                         "etihad-airways"="Eti
                                                         "klm-royal-dutch-airl
                                                         "lufthansa"="Lufthan
                                                         "qatar-airways"="Qata
                                                         "singapore-airlines"
                                                         "south-african-airwa
                                                         "turkish-airlines"="T
colnames(segregatedAirlinesData) <-c("Airline","Author.Country","Cabin.Flown","C
segregatedAirlinesData <- segregatedAirlinesData %>% arrange(Airline)
setDT(segregatedAirlinesData)[, AirlineID := .GRP,by= Airline]
setDT(segregatedAirlinesData)[, CabinID := .GRP,by= `Cabin Flown`]
segregatedAirlinesData <- segregatedAirlinesData %>%
                          select(AirlineID, Airline, CabinID, `Cabin Flown`,

f = segregatedAirlinesData$Airline
out <-split( segregatedAirlinesData , f)      `
Austrian Airlines` <- out$`Austrian Airlines`
`Austrian Airlines` <- ddply(`Austrian Airlines`, .(CabinID), numcolwise(med
`Austrian Airlines`$Airline <-c("Austrian.Airlines","Austrian.Airlines","Austri
`Austrian Airlines`$`Cabin Flown` <-c("Business.Class","Economy","Premium.Econo

`Etihad Airways` <- out$`Etihad Airways`
`Etihad Airways` <- ddply(`Etihad Airways`, .(CabinID), numcolwise(median), .
```

```
‘Etihad Airways‘$Airline <-c("Etihad.Airways","Etihad.Airways","Etihad.Airways")
‘Etihad Airways‘$‘Cabin Flown‘ <-c("Business.Class","Economy","First.Class")

‘KLM‘ <- out$‘KLM‘
‘KLM‘ <- ddply(‘KLM‘, .(CabinID), numcolwise(median), .drop=FALSE)
‘KLM‘$Airline <-c("KLM","KLM","KLM")
‘KLM‘$‘Cabin Flown‘ <-c("Business.Class","Economy","Premium.Economy")

‘Lufthansa‘ <- out$‘Lufthansa‘
‘Lufthansa‘ <- ddply(‘Lufthansa‘, .(CabinID), numcolwise(median), .drop=FALSE
‘Lufthansa‘$Airline <-c("Lufthansa","Lufthansa","Lufthansa","Lufthansa")      ‘
Lufthansa‘$‘Cabin Flown‘ <-c("Business.Class","Economy","Premium.Economy","Firs

‘Qatar Airways‘ <- out$‘Qatar Airways‘
‘Qatar Airways‘ <- ddply(‘Qatar Airways‘, .(CabinID), numcolwise(median), .dr
‘Qatar Airways‘$Airline <-c("Qatar.Airways","Qatar.Airways","Qatar.Airways","Qata
‘Qatar Airways‘$‘Cabin Flown‘ <-c("Business.Class","Economy","Premium.Economy","

‘South African Airways‘ <- out$‘South African Airways‘
‘South African Airways‘ <- ddply(‘South African Airways‘, .(CabinID), numcol
‘South African Airways‘$Airline <-c("South.African.Airways","South.African.Airw
‘South African Airways‘$‘Cabin Flown‘ <-c("Business.Class","Economy")

‘Singapore Airlines‘ <- out$‘Singapore Airlines‘
‘Singapore Airlines‘ <- ddply(‘Singapore Airlines‘, .(CabinID), numcolwise(m
‘Singapore Airlines‘$Airline <-c("Singapore.Airlines","Singapore.Airlines","Sin
‘Singapore Airlines‘$‘Cabin Flown‘ <-c("Business.Class","Economy","Premium.Econ

‘Turkish Airlines‘ <- out$‘Turkish Airlines‘
‘Turkish Airlines‘ <- ddply(‘Turkish Airlines‘, .(CabinID), numcolwise(media
‘Turkish Airlines‘$Airline <-c("Turkish.Airlines","Turkish.Airlines","Turkish.Ai
‘Turkish Airlines‘$‘Cabin Flown‘ <-c("Business.Class","Economy","Premium.Economs
egregatedAirlinesData2<-rbind(‘Austrian Airlines‘, ‘Etihad Airways‘, ‘KLM‘, se
gregatedAirlinesData2<- segregatedAirlinesData2%>%
                        dplyr::mutate(RID =row_number()) %>%
                        select(RID, AirlineID, Airline, CabinID, ‘Cabin Fl

segregatedAirlinesData2$Country <-ifelse(segregatedAirlinesData2$AirlineID ==1
                                  ifelse(segregatedAirlinesData2$AirlineID ==
                                  ifelse(segregatedAirlinesData2$AirlineID ==
                                  ifelse(segregatedAirlinesData2$AirlineID ==
                                  ifelse(segregatedAirlinesData2$AirlineID ==
                                  ifelse(segregatedAirlinesData2$AirlineID ==
                                  ifelse(segregatedAirlinesData2$AirlineID ==
                                  ifelse(segregatedAirlinesData2$AirlineID ==
write.csv(segregatedAirlinesData2,file="C:\\Users\\vikra\\OneDrive.-.National.Col
#Cleaningendsforairlineratingsdataset.


library(magrittr)
library(dplyr)
```

```
library(tidyr)  lib
rary(tidyverse) lib
rary(data.table)

#CleaningstartsforFuel(Gallons)dataset.
fuelData <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\\MS
fuelData$X<-NULL
complete.cases(fuelData)
colnames(fuelData) <-c("Airlines","1995","1996","1997","1998","1999","2000","2001","20
fuelData <-na.omit(fuelData)
fuelData <- fuelData[c(-8,-14,-19,-20),]
rownames(fuelData) <- NULL
fuelData <- fuelData %>% select(Airlines,`2017`)
names(fuelData) <-c("Airline","Gallons_Fuel")
fuelData <- fuelData %>% mutate(Gallons_Fuel = as.numeric(gsub(",","", Gallons
fuelMedian <- fuelData$Gallons_Fuel
fuelMedian[is.na(fuelMedian)] <-median(fuelMedian,na.rm=TRUE)
fuelData$Gallons_Fuel <- fuelMedian
fuelData$Airline <- as.character(fuelData$Airline)
fuelData[7,1] <-"America.West"
fuelData <- fuelData %>% arrange(Airline)
#CleaningendsforFuel(Gallons)dataset.

#CleaningstartsforTotalLoadFactordataset.
loadFactorData <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.of.Irela
loadFactorData$X<-NULL
complete.cases(loadFactorData)
names(loadFactorData) <-c("Airlines","1995","1996","1997","1998","1999","2000","2001",
rownames(loadFactorData) <- NULL
loadFactorData <- loadFactorData[c(-1:-4,-12,-13,-19,-20,-25:-30),]
rownames(loadFactorData) <- NULL
loadFactorData <- loadFactorData %>% select(Airlines,`2017`)
names(loadFactorData) <-c("Airline","Load_Factor")
loadFactorData <- loadFactorData %>% mutate(Load_Factor = as.nume
ric(gsub("%",          loadMedian <- loadFactorData$Load_Factor
loadMedian[is.na(loadMedian)] <-median(loadMedian,na.rm=TRUE)
loadFactorData$Load_Factor<- loadMedian
loadFactorData <- loadFactorData %>% arrange(Airline)
#CleaningendsforTotalLoadFactordataset.

#CleaningstartsforTotalOperatingRevenuedataset.
operatingRevenueData <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.o
operatingRevenueData$X<-NULL
complete.cases(operatingRevenueData)
colnames(operatingRevenueData) <-c("Airlines","1995","1996","1997","1998","1999","20
operatingRevenueData <-na.omit(operatingRevenueData)
rownames(operatingRevenueData) <- NULL
operatingRevenueData <- operatingRevenueData[c(-8,-14,-19,-20,-21),]
rownames(operatingRevenueData) <- NULL
operatingRevenueData <- operatingRevenueData%>% select(Airlines,`2017`)
names(operatingRevenueData) <-c("Airline","Operating_Revenue")
```

```
operatingRevenueData <- operatingRevenueData %>% mutate(Operating_Revenue =
revenueMedian <- operatingRevenueData$Operating_Revenue
revenueMedian[is.na(revenueMedian)] <-median(revenueMedian,na.rm=TRUE)
operatingRevenueData$Operating_Revenue <- revenueMedian
operatingRevenueData <- operatingRevenueData %>% arrange(Airline)  #C
leaningendsforTotalOperatingRevenuedataset.

#CleaningstartsforTotalOperatingExpensesdataset.
operatingExpensesData <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.
operatingExpensesData$X<-NULL
complete.cases(operatingExpensesData)
colnames(operatingExpensesData) <-c("Airlines","1995","1996","1997","1998","1999","2
operatingExpensesData <-na.omit(operatingExpensesData)
rownames(operatingExpensesData) <- NULL
operatingExpensesData <- operatingExpensesData[c(-8,-14,-19,-20,-21),]
rownames(operatingExpensesData) <- NULL
operatingExpensesData <- operatingExpensesData%>% select(Airlines,‘2017‘)
names(operatingExpensesData) <-c("Airline","Operating_Expense")                o
peratingExpensesData <- operatingExpensesData %>% mutate(Operating_Expense e
xpenseMedian <- operatingExpensesData$Operating_Expense
expenseMedian[is.na(expenseMedian)] <-median(expenseMedian,na.rm=TRUE)
operatingExpensesData$Operating_Expense <- expenseMedian                    o
peratingExpensesData <- operatingExpensesData %>% arrange(Airline)

operatingExpensesData$Airline <- as.character(operatingExpensesData$Airline)
operatingRevenueData$Airline <- as.character(operatingRevenueData$Airline)
loadFactorData$Airline <- as.character(loadFactorData$Airline)

airlineFinancial <-merge(operatingRevenueData, operatingExpensesData,by="Airl
airlinePerformance <-merge(fuelData, loadFactorData,by="Airline")
airlineDetails <-merge(airlineFinancial, airlinePerformance,by="Airline")     a
irlineRevenue <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.of.Irelab
estAirlinesData <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.of.Ires
egregatedAirlinesData <-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College
#bestAirlinesData$Country<-as.character(bestAirlinesData$Country)                a
irlineDetails[1,1] <-"Austrian.Airlines"
airlineDetails[2,1] <-"Etihad.Airways"
airlineDetails[3,1] <-"KLM"
airlineDetails[4,1] <-"Lufthansa"
airlineDetails[5,1] <-"Qatar.Airways"
airlineDetails[6,1] <-"Singapore.Airlines"
airlineDetails[7,1] <-"South.African.Airways"
airlineDetails[8,1] <-"Turkish.Airlines"          a
irlineDetails <- airlineDetails[c(-9:-16),]
row.names(airlineDetails) <-1:nrow(airlineDetails)
setDT(airlineDetails)[, AirlineID := .GRP,by= Airline]
airlineDetails <- airlineDetails %>% arrange(Airline) %>% select(AirlineID,
airlineDetails$Airline <- as.factor(airlineDetails$Airline)
temp1<- airlineDetails %>% right_join(airlineRevenue,by=c("AirlineID","Airline'
temp2<- temp1%>% right_join(bestAirlinesData,by=c("AirlineID","Airline","Country
temp3<- temp2%>% right_join(segregatedAirlinesData,by=c("AirlineID","Airline","C
```

```
temp3<- temp3%>% arrange(Airline) %>% select(AirlineID, Airline, IATA.Code, C
#write.csv(airlineDetails,file="C:\\Users\\vikra\\OneDrive-NationalCollegeofIrela
write.csv(temp3,file="C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\\MS

#CleaningendsforTotalOperatingExpensesdataset.

#DownloadingandCleaningDataFromWikipedia
library(htmltab)
library(tidyverse)
library(dplyr)
library(data.table)
library(matrixStats)
library(tidyr)

url<-"https://en.wikipedia.org/wiki/World%
27s_largest_airlines"                #Cleanin
gforairlineRevenueStarts
airlineRevenue <- htmltab(doc=url,which=1, stringsAsFactors =TRUE)
head(airlineRevenue)

airlineRevenue <- airlineRevenue %>%
  mutate(`Revenue(US$B)` = as.numeric(`Revenue(US$B)`),
         Country = as.character(sub('.','',Country))) %>%
  select(Airline, Country, `Revenue(US$B)`) %>%
  arrange(Airline)
names(airlineRevenue)[names(airlineRevenue) =="Revenue(US$B)"] <-"Revenue"
airlineRevenue[1,1] <-"KLM"
airlineRevenue[1,2] <-"Netherlands"
airlineRevenue[2,1] <-"Austrian.Airlines"
airlineRevenue[2,2] <-"Austria"
airlineRevenue[3,1] <-"South.African.Airlines"
airlineRevenue[3,2] <-"South.Africa"
airlineRevenue[4,1] <-"Singapore.Airlines"
airlineRevenue[4,2] <-"Singapore"
airlineRevenue[5,1] <-"Turkish.Airlines"
airlineRevenue[5,2] <-"Turkey"
airlineRevenue[6,1] <-"Etihad.Airways"
airlineRevenue[6,2] <-"United.Arab.Emirates"
airlineRevenue[7,1] <-"Qatar.Airways"
airlineRevenue[7,2] <-"Qatar"
airlineRevenue[8,2] <-"Germany"
airlineRevenue <- airlineRevenue %>% arrange(Airline)
airlineRevenue <- airlineRevenue[c(-8,-10),]
row.names(airlineRevenue) <-1:nrow(airlineRevenue)     a
irlineRevenue <- airlineRevenue %>% arrange(Airline)
setDT(airlineRevenue)[, AirlineID := .GRP,by= Airline]
setDT(airlineRevenue)[, CountryID := .GRP,by= Country]
airlineRevenue <- airlineRevenue %>% select(AirlineID, Airline, CountryID, C
write.csv(airlineRevenue,file="C:\\Users\\vikra\\OneDrive.-.National.College.of.I
#CleaningforairlineRevenueends
```

```
#DownloadingandCleaningTwitterData
library(feather)
library(textcat)
library(cld2)
library(cld3)
library(tidyverse)
library(syuzhet)
library(tidytext)
library(stringr)
library(wordcloud)
library(reshape2)
library(twitteR)
library(data.table)
library(base64enc)

api_key <-'yty8zHcHlsjCrYR8kn5mrLfyV'
api_secret <-'uXruq9Jq3ZgoOrwTqgFlPlR2NvvUS5HkE6cqsZ59dn5bhXVvTJ'     a
ccess_token <-'852005081197228032-HOClRwaQfio1pWAx4kQUW4fOetOpOeH'    a
ccess_token_secret <-'f3q7rgWcFmRv6wZqqGtHDnELedm97lnOR86qGVaglqHpA'
setup_twitter_oauth(api_key, api_secret, access_token, access_token_secret)

srch <-c('@_austrian','@IndonesiaGaruda','@FlySAA_US','@Lufthansa',"@EtihadAirway
airlineNames <-paste(srch, collapse =".OR.")
tweets <- searchTwitter(airlineNames, n=200000, lang ="en", retryOnRateLimit =

tweets_DF <-do.call("rbind",lapply(tweets, as.data.frame))
tweets <- tweets_DF %>%
          mutate(cld2= cld2::detect_language(text=text, plain_text=FALSE),
          cld3= cld3::detect_language(text=text)) %>%
          select(text, cld2, cld3) %>%
          filter(cld2=="en"& cld3=="en") %>%
          select(text)
tweets$text<-sapply(tweets$text,function(row) iconv(row,"latin1","ASCII",sub=""))

path1<-"C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\\MSc.Data.Analyti
write_feather(tweets,path1)
path1="C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\\MSc.Data.Analytic
twitter_df4<-read_feather(path1)

filtered_twitter_df4<- twitter_df4%>%
                       filter(!str_detect(twitter_df4$text,'RT.@')) %>%
                       select(Airline,text)
filtered_twitter_df4<- filtered_twitter_df4%>%
                       filter(!str_detect(filtered_twitter_df4$Airline,'None
                       select(Airline,text)
twitter_df4<- filtered_twitter_df4
twitter_df4$text<-gsub("http.*","",twitter_df4$text)
twitter_df4$text<-gsub("https.*","",twitter_df4$text)
twitter_df4$text<-gsub("#.*","",twitter_df4$text)

path<-"C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\\MSc.Data.Analytic
```

```
twitter_df3<-read_feather(path)

filtered_twitter_df3<- twitter_df3%>%
                       mutate(cld2= cld2::detect_language(text=text, plain_te
                       cld3= cld3::detect_language(text=text)) %>%
                       select(text, cld2, cld3, created_at, Airline) %>%
                       filter(cld2=="en"& cld3=="en") %>%
                       select(text, created_at, Airline)

twitter_text1<-  filtered_twitter_df3%>%
                  filter(!str_detect(filtered_twitter_df3$text,'RT.@'))%>%
                  select(Airline,text)
twitter_text1$text<-sapply(twitter_text1$text,function(row) iconv(row,"latin1","A
twitter_text1$text<-gsub("http.*","",twitter_text1$text)
twitter_text1$text<-gsub("https.*","",twitter_text1$text)
twitter_text1$text<-gsub("#.*","",twitter_text1$text)

tweets <-rbind(twitter_text1, twitter_df4)
noTweets <-table(tweets$Airline)

tweets <- data_frame(Airline = tweets$Airline,text= tweets$text)
tweets <- tweets %>%unnest_tokens(word,text)

nrc <-get_sentiments("nrc")
nrcSent <- tweets %>%
  inner_join(nrc) %>%
  count(index= Airline, sentiment) %>%
  spread(sentiment, n, fill =0) %>%
  mutate(sentiment = positive - negative)

sentimentTweets1<- data.frame(nrcSent)
names(sentimentTweets1)[names(sentimentTweets1) =="index"] <-"Airline"
setDT(sentimentTweets1)[, AirlineID := .GRP,by= Airline]
sentimentTweets1<- sentimentTweets1%>% select(Airline, AirlineID, anger, joy,
trust, negative, positive, sentiment)
sentimentTweets2<-read.csv("C:\\Users\\vikra\\OneDrive.-.National.College.of.Irel
Airline <- sentimentTweets2$V1
AirlineID <- sentimentTweets2$V2
sentimentTweets2$V1<- NULL
sentimentTweets2$V2<- NULL

temp <- data.frame(t(sentimentTweets2))
temp <-cbind(rownames(temp), temp)
rownames(temp) <- NULL
colnames(temp) <-c("1","Sentiment","2","3","4","5","6","7","8","9")
temp$`1` <- NULL
setDT(temp)[, SentimentID := .GRP,by= Sentiment]
temp2<- data.frame(t(temp))
wordcloud(tweets$word ,min.freq =10,colors=brewer.pal(8,"Dark2"),random.color =
write.csv(sentimentTweets1,file="C:\\Users\\vikra\\OneDrive.-.National.College.of
View(temp)
```

Python Code

(Bonzanini, 2015)

```
from __future__ importprint_function
import tweepy
import json
from pymongo import MongoClient
MONGO_HOST ="mongodb://localhost/twitterdb"
#assumingyouhavemongoDBinstalledlocally
#andadatabasecalled'twitterdb'
WORDS = ['#bigdata','#AI','#datascience','#machinelearning','#ml','#iot']
CONSUMER_KEY ="TzsprZxvgKGUTAWZRhzOPcdvk"
CONSUMER_SECRET ="cN9ngDq1dioGMJCYkCUO0LSS6z68YzblqjkwC6lVoUx3hjMcMQ"
ACCESS_TOKEN ="852005081197228032-egyblEdwCborYNzVE6bxL5tZPWmBje7"
ACCESS_TOKEN_SECRET ="xsU5i3Ju0Yc4ENb7lncdfr2EmwqlnmDJp5eTRzzDDMFz7"
classStreamListener(tweepy.StreamListener):#Thisisaclassprovidedbytweepytoacces
    defon_connect(self):
#Calledinitiallytoconnecttothestreamingapl
        print("You.are.now.connected.to.the.streaming.API.")

    defon_error(self,status_code):
#Onerror-ifanerroroccurs,displaytheerror/statuscode
        print('An.Error.has.occured:.'+ repr(status_code))
        returnFalse

    defon_data(self, data):                #Thisisthemeatofthescrip
t...itconnectstoyourmongoDBandstoresthetweet
        try:
            client = MongoClient(MONGO_HOST)  #Usetw
itterdbdatabase.Ifitdoesn'texist,itwillbecreated.
            db = client.twitterdb
#DecodetheJSONfromTwitter
            datajson = json.loads(data)
#grabthe'created_at'datafromtheTweettousefordisplay
            created_at = datajson['created_at']
#printoutamessagetothescreenthatwehavecollectedatweet
            print("Tweet.collected.at."+ str(created_at))  #in
sertthedataintothemongoDBintoacollectioncalledtwitter_search
#iftwitter_searchdoesn'texist,itwillbecreated.
            db.twitter_search.insert(datajson)
        except Exception as e:
            print(e)
auth = tweepy.OAuthHandler(CONSUMER_KEY, CONSUMER_SECRET)
auth.set_access_token(ACCESS_TOKEN, ACCESS_TOKEN_SECRET)
#Setupthelistener.The'wait_on_rate_limit=True'isneededtohelpwithTwitterAPIratelimi
listener = StreamListener(api=tweepy.API(wait_on_rate_limit=True))
streamer = tweepy.Stream(auth=auth, listener=listener)
print("Tracking:."+ str(WORDS))
streamer.filter(track=WORDS)

from pymongo import MongoClient
```

```
import pandas as pd
import feather
from _datetime import datetime
import re

client = MongoClient("mongodb://localhost")
twitterdb = client.twitterdb
cursor = twitterdb.twitter_search.find({"$text": {"$search":"austrian.etihad.klm
twitter_search= ["text","created_at"]

twitter_df1= pd.DataFrame(list(cursor), columns=twitter_search)
remove_ms = lambda x:re.sub("\+\d+\s","",x)
mk_dt= lambda x:datetime.strptime(remove_ms(x),"%a.%b.%d.%H:
%M:%S.%Y")         my_form = lambda x:"{:%Y-%m-%d}".format(mk
_dt(x))
twitter_df1.created_at = twitter_df1.created_at.apply(my_form)
twitter_df1['Airline'] = pd.np.where(twitter_df1.text.str.contains("austria",ca
                         pd.np.where(twitter_df1.text.str.contains("etihad",cas
                         pd.np.where(twitter_df1.text.str.contains("klm",case=
                         pd.np.where(twitter_df1.text.str.contains("lufthansa",
                         pd.np.where(twitter_df1.text.str.contains("qatar",case
                         pd.np.where(twitter_df1.text.str.contains("singapore",
                         pd.np.where(twitter_df1.text.str.contains("flysaa",cas
                         pd.np.where(twitter_df1.text.str.contains("flysaa",cas
                         pd.np.where(twitter_df1.text.str.contains("turkish",ca
                         pd.np.where(twitter_df1.text.str.contains("boeing",cas

path='C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\MSc.Data.Analytics\
feather.write_dataframe(twitter_df1,path)
path1='C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\MSc.Data.Analytics

twitter_df1['Airline'] = pd.np.where(tweets.text.str.contains("austria",case= F
                         pd.np.where(twitter_df1.text.str.contains("etihad",cas
                         pd.np.where(twitter_df1.text.str.contains("klm",case=
                         pd.np.where(twitter_df1.text.str.contains("lufthansa",
                         pd.np.where(twitter_df1.text.str.contains("qatar",case
                         pd.np.where(twitter_df1.text.str.contains("singapore",
                         pd.np.where(twitter_df1.text.str.contains("flysaa",cas
                         pd.np.where(twitter_df1.text.str.contains("flysaa",cas
                         pd.np.where(twitter_df1.text.str.contains("turkish",ca
                         pd.np.where(twitter_df1.text.str.contains("boeing",cas
```

```
import pandas as pd
import feather
import numpy
path1='C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\MSc.Data.Analytics
```

```
tweets = feather.read_dataframe(path1)

tweets['Airline'] = pd.np.where(tweets.text.str.contains("austria",case= False)
                          pd.np.where(tweets.text.str.contains("etihad",case= Fa
                          pd.np.where(tweets.text.str.contains("klm",case= False
                          pd.np.where(tweets.text.str.contains("lufthansa",case=
                          pd.np.where(tweets.text.str.contains("qatar",case= Fal
                          pd.np.where(tweets.text.str.contains("singapore",case=
                          pd.np.where(tweets.text.str.contains("flysaa",case= Fa
                          pd.np.where(tweets.text.str.contains("flysaa",case= Fa
                          pd.np.where(tweets.text.str.contains("turkish",case= F

path1='C:\\Users\\vikra\\OneDrive.-.National.College.of.Ireland\MSc.Data.Analytics
feather.write_dataframe(tweets,path1)
```

SQL code

```
CREATE TABLE [fact_table]([AirlineID] INT
      ,[SentimentID] INT
      ,[RID] INT
      ,[CID] INT
      ,[PID] INT
      ,[CabinID] INT
      ,[YearID] INT
      ,[Year] INT
      ,[No of Passengers] INT
      ,[Percentage Growth] INT
      ,[CountryID] INT
      ,[CabinGroupID] INT
      ,[SentimentGroupID] INT
      ,[SentimentValue] INT
      ,[Airline Score] INT
      ,[Revenue] INT
      ,[Operating_Revenue] INT
      ,[Operating_Expense] INT
      ,[Gallons_Fuel] INT
      ,[Load_Factor] INT
      ,[Number of Airplanes] INT
      ,[Fleet Age AVG ] INT
      ,[Comfort Rating] INT
      ,[Staff Rating] INT
      ,[Food Rating] INT
      ,[Entertainment Rating] INT
      ,[ValueforMoney] INT
)

TRUNCATE TABLE [fact_table]
INSERT INTO [fact_table] (
    [AirlineID],
    [SentimentID],
```

```
    [RID],
        [CID]
      ,[PID]
      ,[CabinID]
      ,[YearID]
          ,[Year]
      ,[No of Passengers]
      ,[Percentage Growth]
      ,[CountryID]
      ,[CabinGroupID],
    [SentimentGroupID],
    [SentimentValue],
    [Airline Score],
    [Revenue],
    [Operating_Revenue],
    [Operating_Expense],
    [Gallons_Fuel],
    [Load_Factor],
    [Number of Airplanes],
    [Fleet Age AVG ],
    [Comfort Rating],
    [Staff Rating],
    [Food Rating],
    [Entertainment Rating],
    [ValueforMoney]
)
SELECT [a].[AirlineID],
    [SentimentID],
    [RID],
        [CID]
      ,[PID]
      ,[CabinID]
      ,[].[YearID]
          ,[].[Year]
      ,[No of Passengers]
      ,[Percentage Growth]
      ,[n].[CountryID]
      ,[m].[CabinGroupID],
    [SentimentGroupID],
    [SentimentValue],
    [Airline Score],
    [Revenue],
    [Operating_Revenue],
    [Operating_Expense],
    [Gallons_Fuel],
    [Load_Factor],
    [Number of Airplanes],
    [Fleet Age AVG ],
    [Comfort Rating],
    [Staff Rating],
    [Food Rating],
```

```
    [Entertainment Rating],
    [ValueforMoney]
FROM [dimRatings] b
LEFT JOIN [dimAirline] a ON b.AirlineID=a.AirlineID
LEFT JOIN [dimSentiments] s ON b.RID=s.SentimentID
LEFT JOIN [DimPassengers] l ON b.RID=l.PID
LEFT JOIN [DimCabin] m ON b.RID=m.CabinID
LEFT JOIN [DimCountry] n ON b.RID=n.CID
LEFT JOIN [DimTrafficGrowth] o ON l.YearID=o.YearID
```

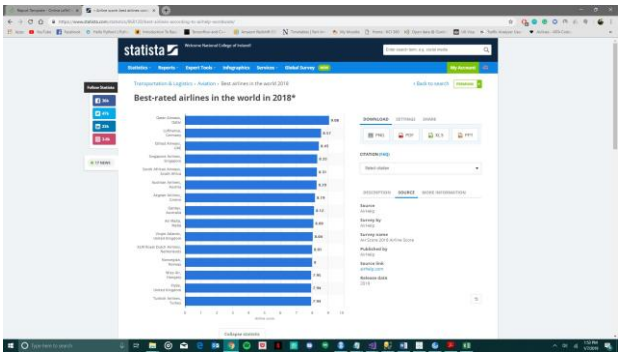

Figure 6: Statista1 Release Date

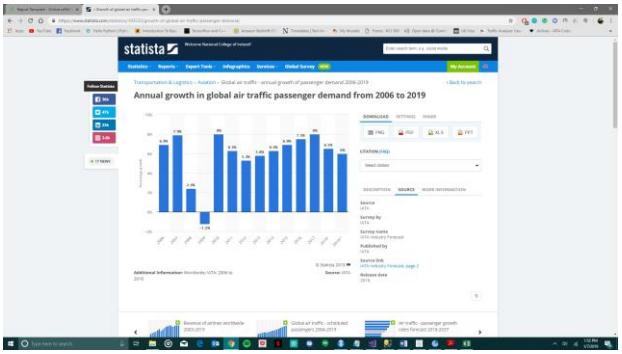

Figure 7: Statista 2 Release Date

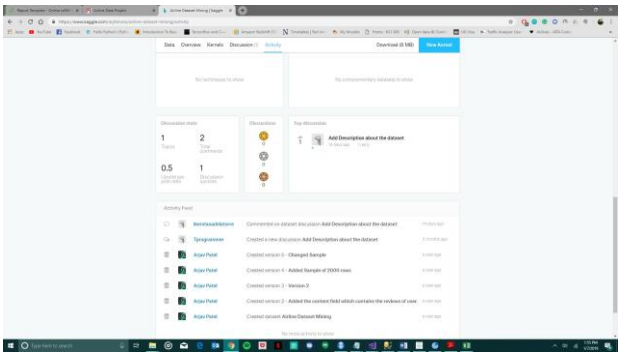

Figure 8: Kaggle release date. Was released ”9 months ago” at the time of starting the project.