MSc in Data Analytics

DWBI Project

Building Data Warehouse for Book Sales

Name – Tejal Gavate

Student ID- x17146895

Introduction

Business intelligence is group of technologies that transform the operational data into meaningful information. Business intelligence predicts and enhance the future by analysing the past records and make the better business decisions for the end users.

Reading has always been the most leisure activity for the people. There are millions of books that are been published all over the world. Here we have brought big data approach for success in sales of books by different selling approaches by authors. Amongst all the books, more copies of fiction were sold according to the public interest. According to different writing patterns of authors the books are given ratings and the highest ratings are been resulted into the great selling's of the book. There are also other genre books which are been sold at huge amount without having highest public ratings.

This paper explain the implementation and structure of data warehouse using SSIS for ETL procedure, SSAS for cube deployment and Power BI for visualisation of the case studies.

Tools and Technologies

Programming Languages:

R for twitter data extraction.

SQL for ETL

Programming API's:

Twitter API for sentiment analysis.

Tools:

Excel for basic operations and data cleaning.

Microsoft visual studio.

MS SQL Server-Management Studio.

Database Management:

SSIS for ETL.

SSAS for deploying cube.

Tableau for data visualisation and implemented queries.

Data Sources

There are 3 sources of dataset which is used in this project :

GitHub – 2 structured (.csv) dataset were downloaded. The attributes present such as authors, title of the book and the amount of book sales were used in my project. The book count in this file were used for showing the amount of the books been sold.

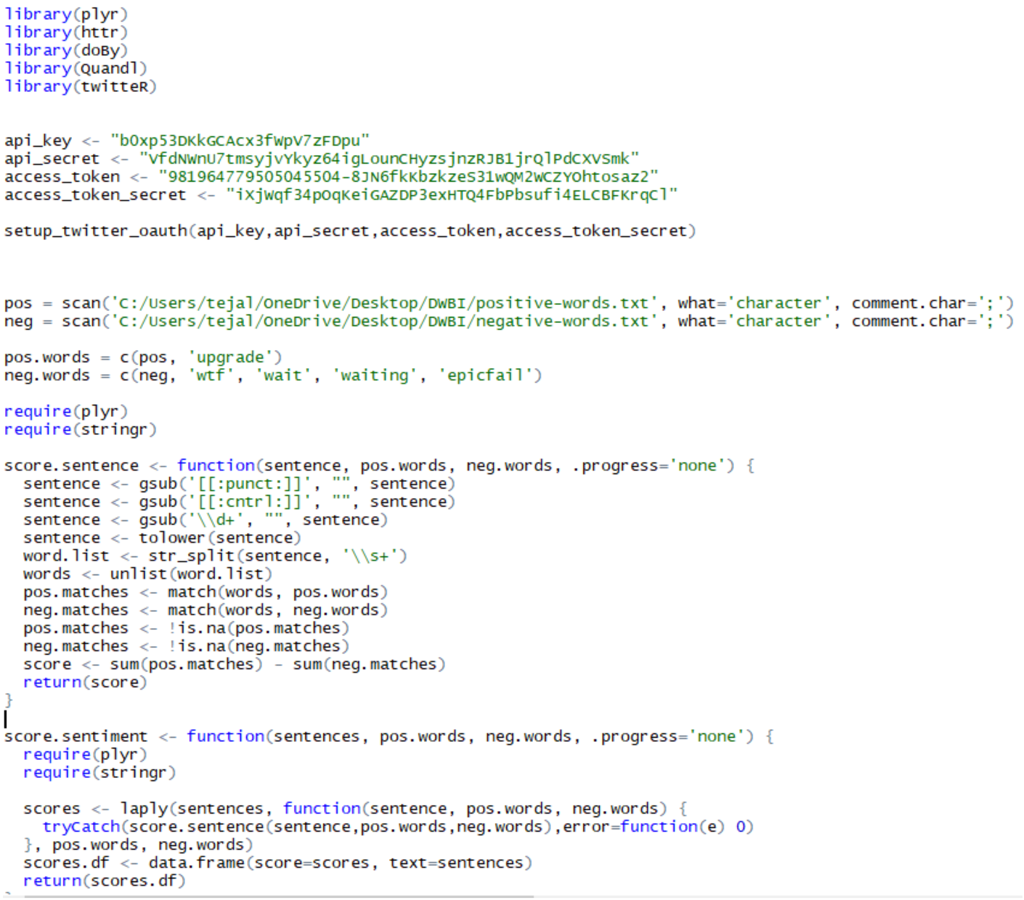
https://github.com/alexsanjoseph/goodreads-list-properties

https://github.com/zygmuntz/goodbooks-10k/blob/master/samples/books.csv

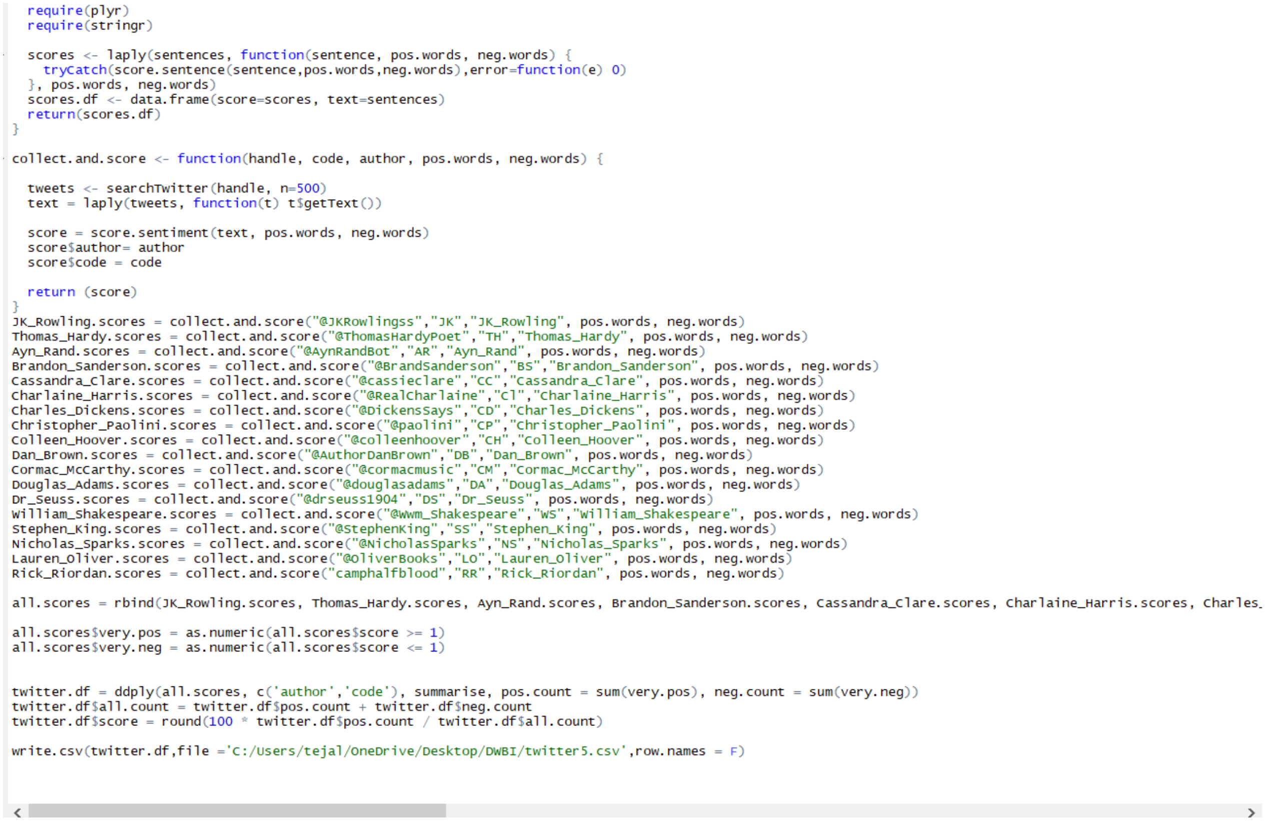
Data. World – 1 structure dataset was taken for implementing the project, this dataset consist of authors, titles and their ratings. Ratings were the main attribute in this project which results in good sales of books.

https://data.world/ssaudz/goodreads-review-of-350-k books/workspace/file?filename=br.csv

Twitter- Tweets were extracted by using R programming for various authors. The positive and negative count of the tweets where been used for visualising the sales of the books.



```
library(plyr)
library(httr)
library(doBy)
library(Quandl)
library(twitteR)


api_key <- "b0xp53DKkGCAcx3fWpV7zFDpu"
api_secret <- "VfdNWnU7tmsyjvYkyz64igLounCHyzsjnzRJB1jrQlPdCXVSmk"
access_token <- "981964779505045504-8JN6fkKbzkzeS31wQM2WCZYOhtosaz2"
access_token_secret <- "iXjWqf34pOqKeiGAZDP3exHTQ4FbPbsufi4ELCBFKrqCl"

setup_twitter_oauth(api_key,api_secret,access_token,access_token_secret)



pos = scan('C:/Users/tejal/OneDrive/Desktop/DWBI/positive-words.txt', what='character', comment.char=';')
neg = scan('C:/Users/tejal/OneDrive/Desktop/DWBI/negative-words.txt', what='character', comment.char=';')

pos.words = c(pos, 'upgrade')
neg.words = c(neg, 'wtf', 'wait', 'waiting', 'epicfail')

require(plyr)
require(stringr)

score.sentence <- function(sentence, pos.words, neg.words, .progress='none') {
  sentence <- gsub('[[:punct:]]', "", sentence)
  sentence <- gsub('[[:cntrl:]]', "", sentence)
  sentence <- gsub('\\d+', "", sentence)
  sentence <- tolower(sentence)
  word.list <- str_split(sentence, '\\s+')
  words <- unlist(word.list)
  pos.matches <- match(words, pos.words)
  neg.matches <- match(words, neg.words)
  pos.matches <- !is.na(pos.matches)
  neg.matches <- !is.na(neg.matches)
  score <- sum(pos.matches) - sum(neg.matches)
  return(score)
}

score.sentiment <- function(sentences, pos.words, neg.words, .progress='none') {
  require(plyr)
  require(stringr)

  scores <- laply(sentences, function(sentence, pos.words, neg.words) {
    tryCatch(score.sentence(sentence,pos.words,neg.words),error=function(e) 0)
  }, pos.words, neg.words)
  scores.df <- data.frame(score=scores, text=sentences)
  return(scores.df)
```



```
  require(plyr)
  require(stringr)

  scores <- laply(sentences, function(sentence, pos.words, neg.words) {
    tryCatch(score.sentence(sentence,pos.words,neg.words),error=function(e) 0)
  }, pos.words, neg.words)
  scores.df <- data.frame(score=scores, text=sentences)
  return(scores.df)
}

collect.and.score <- function(handle, code, author, pos.words, neg.words) {

  tweets <- searchTwitter(handle, n=500)
  text = laply(tweets, function(t) t$getText())

  score = score.sentiment(text, pos.words, neg.words)
  score$author= author
  score$code = code

  return (score)
}
JK_Rowling.scores = collect.and.score("@JKRowlingss","JK","JK_Rowling", pos.words, neg.words)
Thomas_Hardy.scores = collect.and.score("@ThomasHardyPoet","TH","Thomas_Hardy", pos.words, neg.words)
Ayn_Rand.scores = collect.and.score("@AynRandBot","AR","Ayn_Rand", pos.words, neg.words)
Brandon_Sanderson.scores = collect.and.score("@BrandSanderson","BS","Brandon_Sanderson", pos.words, neg.words)
Cassandra_Clare.scores = collect.and.score("@cassieclare","CC","Cassandra_Clare", pos.words, neg.words)
Charlaine_Harris.scores = collect.and.score("@RealCharlaine","Cl","Charlaine_Harris", pos.words, neg.words)
Charles_Dickens.scores = collect.and.score("@DickensSays","CD","Charles_Dickens", pos.words, neg.words)
Christopher_Paolini.scores = collect.and.score("@paolini","CP","Christopher_Paolini", pos.words, neg.words)
Colleen_Hoover.scores = collect.and.score("@colleenhoover","CH","Colleen_Hoover", pos.words, neg.words)
Dan_Brown.scores = collect.and.score("@AuthorDanBrown","DB","Dan_Brown", pos.words, neg.words)
Cormac_McCarthy.scores = collect.and.score("@cormacmusic","CM","Cormac_McCarthy", pos.words, neg.words)
Douglas_Adams.scores = collect.and.score("@douglasadams","DA","Douglas_Adams", pos.words, neg.words)
Dr_Seuss.scores = collect.and.score("@drseuss1904","DS","Dr_Seuss", pos.words, neg.words)
William_Shakespeare.scores = collect.and.score("@Wwm_Shakespeare","WS","William_Shakespeare", pos.words, neg.words)
Stephen_King.scores = collect.and.score("@StephenKing","SS","Stephen_King", pos.words, neg.words)
Nicholas_Sparks.scores = collect.and.score("@NicholasSparks","NS","Nicholas_Sparks", pos.words, neg.words)
Lauren_Oliver.scores = collect.and.score("@OliverBooks","LO","Lauren_Oliver", pos.words, neg.words)
Rick_Riordan.scores = collect.and.score("camphalfblood","RR","Rick_Riordan", pos.words, neg.words)

all.scores = rbind(JK_Rowling.scores, Thomas_Hardy.scores, Ayn_Rand.scores, Brandon_Sanderson.scores, Cassandra_Clare.scores, Charlaine_Harris.scores, Charles_

all.scores$very.pos = as.numeric(all.scores$score >= 1)
all.scores$very.neg = as.numeric(all.scores$score <= 1)


twitter.df = ddply(all.scores, c('author','code'), summarise, pos.count = sum(very.pos), neg.count = sum(very.neg))
twitter.df$all.count = twitter.df$pos.count + twitter.df$neg.count
twitter.df$score = round(100 * twitter.df$pos.count / twitter.df$all.count)

write.csv(twitter.df,file ='C:/Users/tejal/OneDrive/Desktop/DWBI/twitter5.csv',row.names = F)
```

Architecture

There are two approaches for building data warehouse. First, Inmon's and second Kimball. The Inmon's method is top-down approach and the Kimball approach is bottom-up. In this project the Kimball approach is been used. The data that is used for the business processes is been normalised and designed, were Kimball focus on report and analysis of the data and then building the data warehouse.

Reasons and advantages for selecting the Kimball approach for this project-

1. Easy to set-up and build.

2. The star schema helps the business users for understanding and reporting. Various BI tools use star schema for easy reporting.

3. Effective database operation. In this, star join is performed and by using all the dimension values and fact table, the cartesian product is created for querying for the rows.

4. Time taken for this approach is less than Inmon's.

5. In this model, data warehouse is used for faster querying. 4

6. Report generation is simple in this approach because many business processes can generate reports without any technical knowledge.

The Kimball Methodology

The Kimball approach identify the main features of the business process and the queries for which the data warehouse needs to answer and then the data warehouse is created for those business queries. Kimball's method identifies the business lines which are needed for the business intelligence.

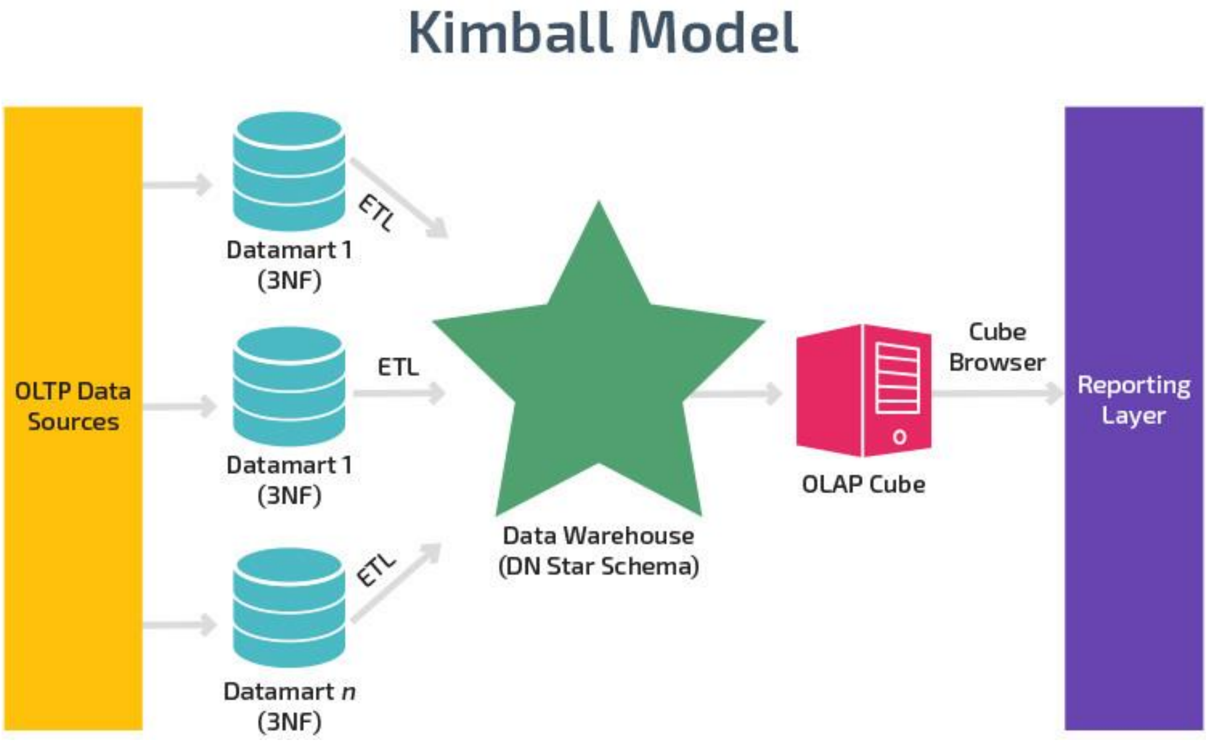
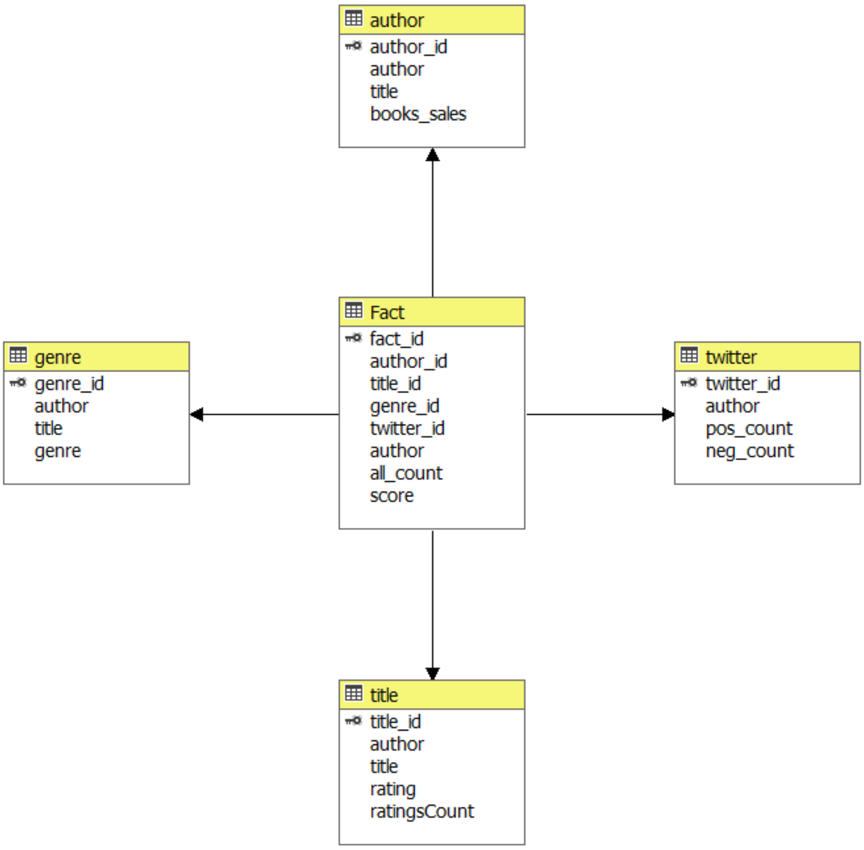


Figure 1: https://panoply.io/data-warehouse-guide/data-mart-vs-data-warehouse

This gives the strong view for enterprising data and analysing it. For this all the data should be merged into one data warehouse. This focuses on star schema which makes the user easy to understand and provides good performance to the data warehouse.

Data Model

For my Data Warehouse (Books), I have used star schema. It removes the complexity for building multidimensional model, it is faster and easier than snowflake schema. The star schema consists of a fact table which is linked with all the dimensions of the table. Before implementing data warehouse, the snow flake schema uses normalized data whereas star schema does not require normalized data, hence are more flexible and is used in this project.



The figure shows the example of star schema, in which at the centre of the schema there is a fact table which consist of all the numeric values and consist of four dimensions which contains text like attributes that are highly correlated with each other. In star schema dimensions have foreign keys in the fact table. For building the data model for books, star schema was more flexible.

ETL Strategies

ETL in snow flake model is very complex in design because it loads all the DataMart's. The parallel running of the processes is not possible between the dependency of dimension and facts. ETL is simple and easier in star schema as it loads the dimension table without dependency and gain higher parallelization.

ETL is very first step of building data warehouse. It involves extraction of various data from different data sources, transformation of the data and then loading it into the database.

Data Extraction

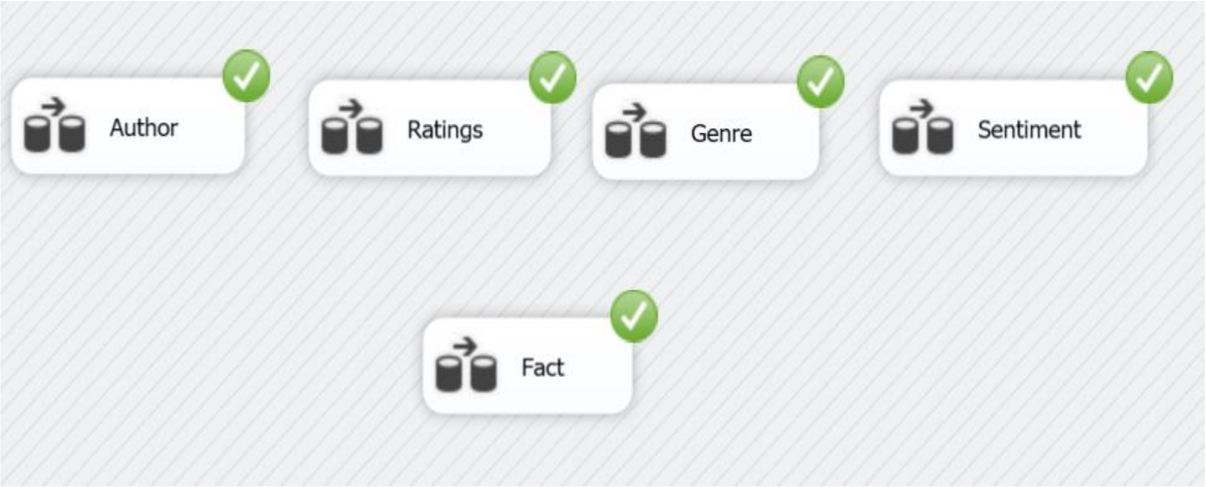
In this project various data sources were used for extracting data. Two structured data was taken from Data. World and GitHub sites. Another dataset was taken by Twitter, R programming was used for extracting tweets of authors from twitter. Libraries and function like “twitter” and “searchTwitter” helped to extract tweets.

Data Transformation-

This procedure is done to clean the data. Excel was used for cleaning the data. The missing values were identified and updated.

Data Loading-

After the data is transformed and cleaned, the data is been loaded into the database (books). Tables are created for all the attributes in SSMS and then the fact table is created using lookups and thus the fact values was populated in SSMS.

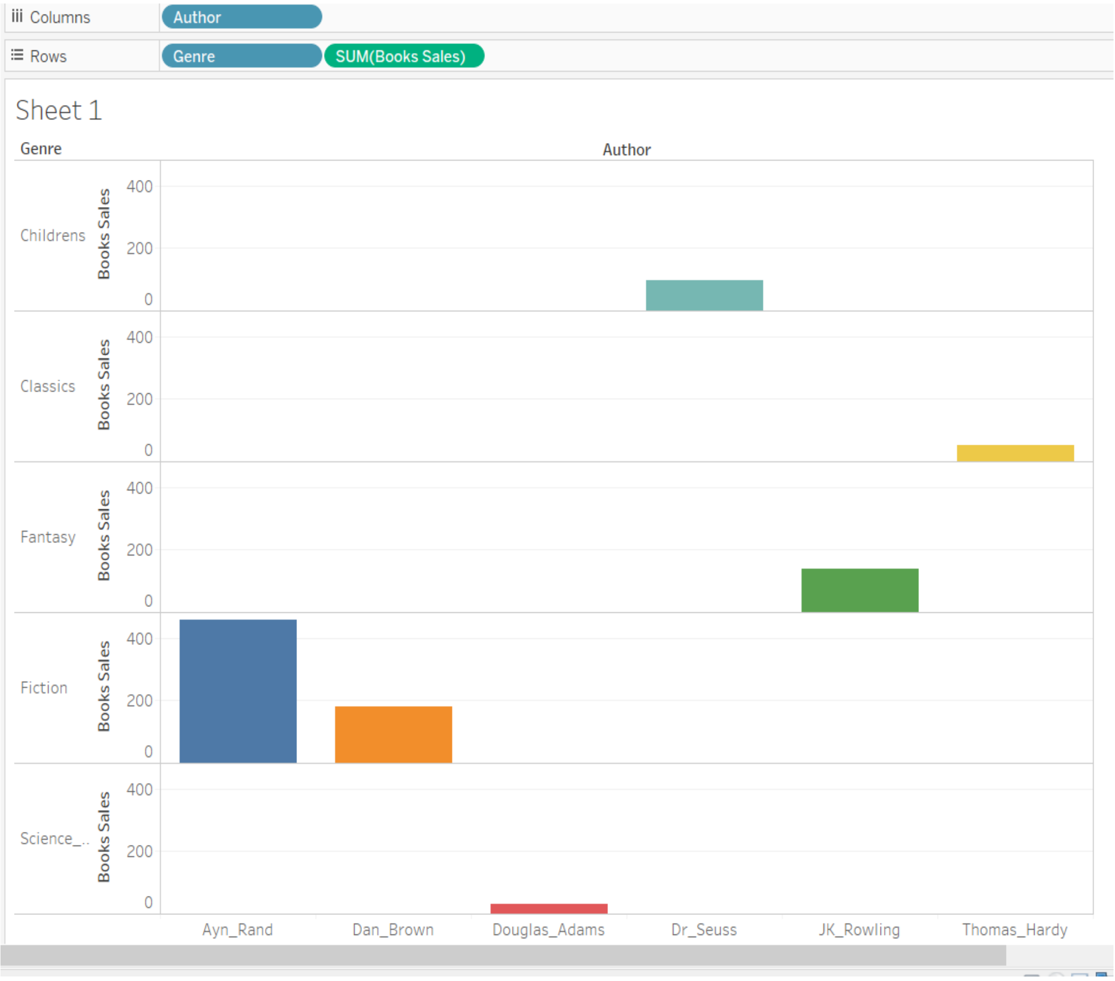


Applications of Data Warehouse

Reporting is important task in analysis of business queries. This is done after the cube is deployed. The cube has the values of fact table and the dimensions from which the visualisation is done. There are various tools used for data visualisation, such as SSRS, Power BI, Tableau. Here, in this project the visualisation is done with the help of tableau.

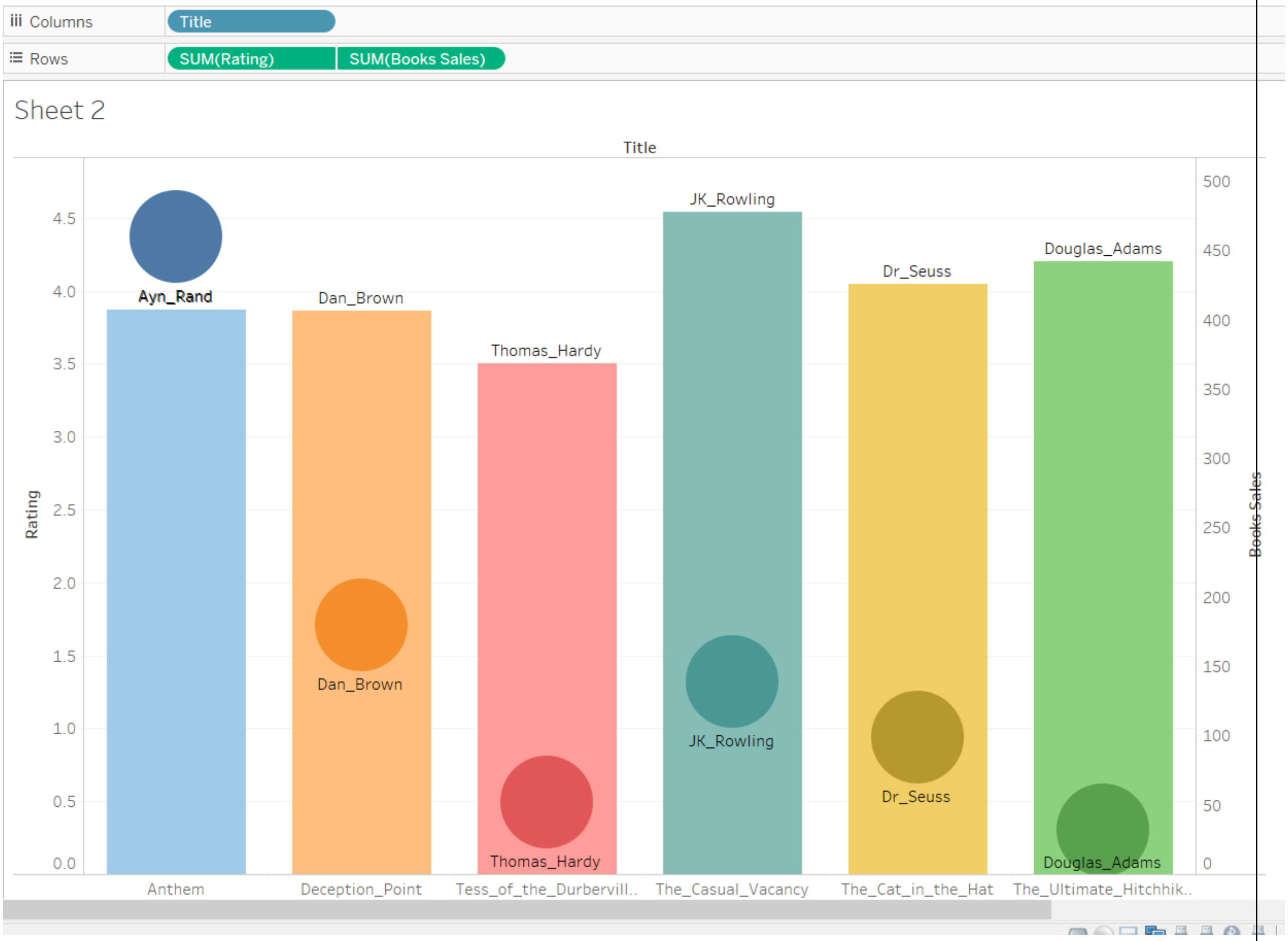
BI Queries-

1) Which type of genre books have made maximum sales in the market?



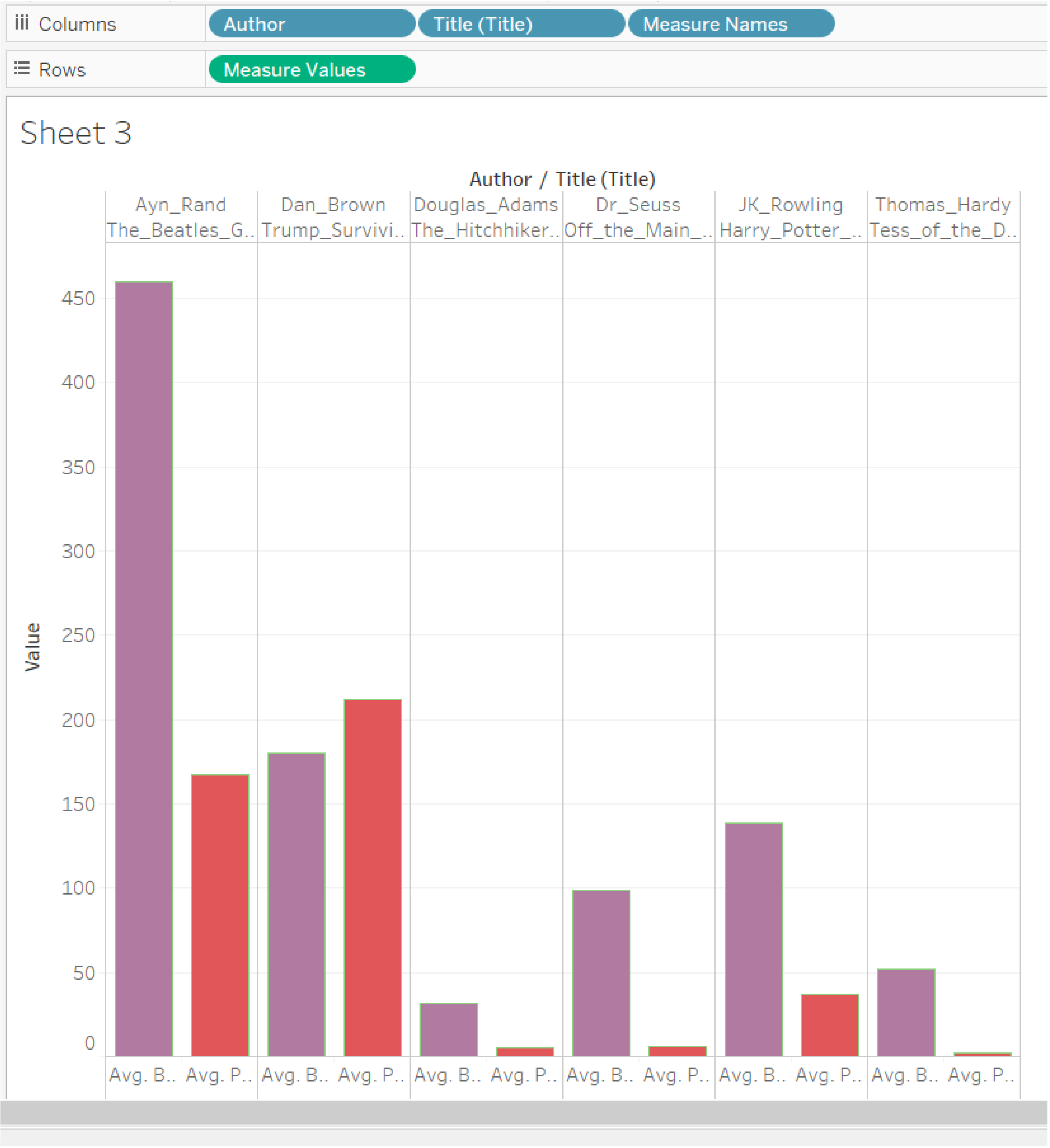
The first graph shows the authors that have made sales in the particular type of genre books. In this we can see that genre such as children's, classic, fantasy and science have less numbers of sales. Whereas, there are authors named Ayn Rand and Dan Brown who has contributed more for the Fiction books. This results that other authors should invest more in publishing fiction books for getting maximum numbers of sales.

2) Predicting sales of books according to user ratings given to the books?



In this graph we can see that Rating for the books of J.K.Rowling is 4.5 and the number of book sales is about 139k. Whereas, ratings for the books of Ayan Rand is 3.8 and number of books sold is maximum as compared to other authors. It is concluded that even if the user ratings are less for the book, the sales of books can be maximum accordingly. The factor ‘Rating’ is not affecting the sales of books.

3) How does positive scores of twitter affect the sales of the book?



Here in this graph we can see that for author Dan Brown the positive count is more but the sales are less.

Whereas the positive count for the authors, Ayn Rand, Douglas Adams, Dr Seuss, JK Rowling and Thomas Hardy has less number of positive count and the sales of books by these authors are still more. So, it is concluded that the number of positive counts does not always affect the sales of the books.

Conclusion-

The complete extraction, transformation, loading and analysis was shown from this architecture of data warehouse. Three business case study was obtained for showing about how visualisation can be done for gaining more business value.

References-

1. Kimball, R., & Caserta, J.,2004. The data warehouse ETL toolkit. John Wiley & Sons.
2. Inmon, W. H. (2005). Building the Data Warehouse. John wiley & sons.
3. Ralph Kimball, M. R. (2013). The data warehouse toolkit. Kimball Group.
4. Matloff, N., 2011. The art of R programming: A tour of statistical software design. No starch press.