Analysis of the Book Rating Dataset by Data Warehouse Implementation

Group 8

Dharani Aningi, Jiayi Liang, Rohini Sidharth Kulkarni, Sai Keerthana Kattige, Ying Liu

Abstract*—We are building a data warehouse implementation system for a historical dataset on cloud. A data warehouse can offer an insight into the business procedures and processes where the data is being used. However, a traditional data warehouse is not efficient for the data analysis needs as it cannot process the increase in number of users. So, there is a new and emerging method of data warehousing also supposed to be known as cloud data warehouse. The cloud data warehouse has evolved to an extent where it can handle huge influx of data. It can be scaled up or scaled down at any given point of time and it does not have any limitations or restrictions on increasing the number of users. A cloud-based system can deliver users with appropriate data utilization and much more like reliability, security, etc. So, we are using google cloud platform to orchestrate the entire process of data warehousing. We used a book analysis dataset to implement this entire process.

***Keywords*—MySQL, Cloud, ETL Process, Data Warehouse**

I. Introduction

The growing e-commerce and online marketing strategies in today's world is encouraging businesses to build efficient and cost-effective business models to target potential customers to improve their businesses. In this project, we want to analyze and build a system that effectively translates the data into more secure instances. We will use a Cloud Data Warehouse to implement our data, because a cloud data warehouse provides more powerful computing capabilities and is easy to scale as the data storage grows, and it does not require physical hardware and space. The purpose of this project is to analyze a historical complex dataset, and through ETL process a data warehouse will be implemented to develop an Online Transactional Processing database. In addition, SQL queries are used to analyze the related data and visualize them using interactive data visualization softwares such as Tableau.

II. Data Description

The dataset we used is called the Book Recommendation Dataset. It is important because it has a large amount of data that can be used to build a book recommendation system that can better target audiences and increase business profit. The dataset contains over two hundred thousands users with demographic information, over one million ratings on over two hundred thousands books. Our dataset contains three files. The Users file contains user ID, location and age. The user IDs were anonymized and mapped to integers. The Books file contains the books ISBN, Book-Title, Book-Author, Year-Of-Publication, Publisher. It also contains URLs to book cover images, in size small, medium, and large. The Ratings file contains book ratings expressed on a scale in range from one to ten with ten being the highest rating, or it can be expressed implicitly using zero.

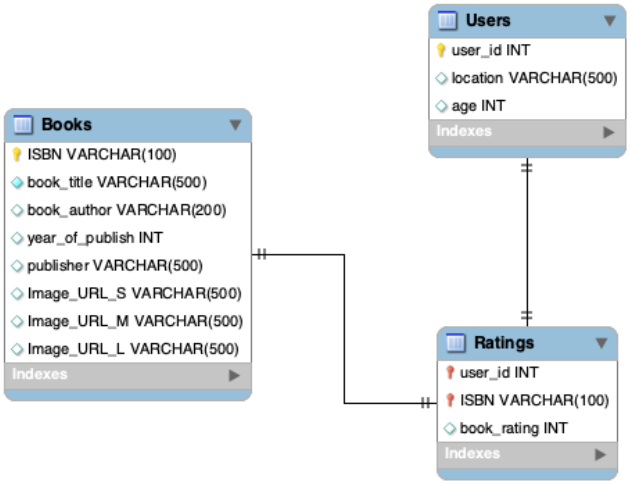


Figure 1 ER Diagram for the Source Data

III. Project Architecture

The purpose of this project is to build a book ratings model using a cloud data warehouse. We used the Google Cloud Platform for our project implementation. We downloaded the raw data in CSV files consisting of a flat-file format from the Kaggle website. Data was normalized into three 3NF tables, the books table, users table, and ratings table. Each table formed a CSV file. These three files were loaded into a Google Cloud Storage bucket. We decided to use Google Cloud SQL MySQL instance to be our source database. We created a relational database in the cloud SQL instance and loaded data records from the

Google Cloud Storage bucket. ETL scripts were developed and running in Google Cloud Composer using Apache Airflow. The cloud composer using Kubernetes clusters supports the horizontally and vertically auto-scaling. After running the ETL pipeline, the source data was extracted from the cloud SQL instance, transformed and loaded into a data warehouse in Google BigQuery by cloud client library. Then we used some SQL queries to achieve the data analytics and visualized the queries' results using interactive data visualization softwares such as Tableau.

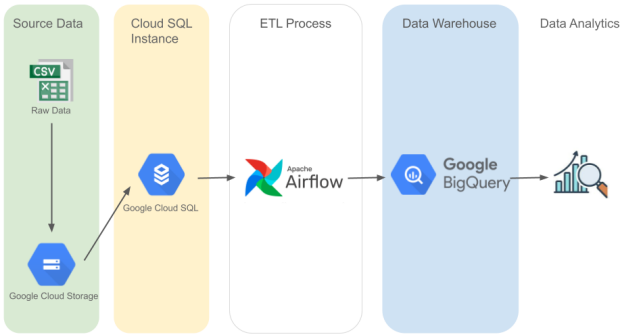


Figure 2 Project Architecture

IV. Cloud SQL Instance

The Cloud SQL instance is configured to have a private VPC network, it is hosted on a public IP address with limited authorized access. Only the accounts which are authorized and have permissions can access our Cloud SQL instance. The authorizations and permissions are managed using IAM roles. The instance's public IP address, an authorized user name and password, database name, etc. are needed to connect the Cloud SQL instance.

V. ETL Process

ETL scripts are developed in python to arrange the extract, transform, and load process.[1] The ETL pipeline is orchestrated using Google cloud composer. The cloud composer is built on Apache Airflow and operated using python to design, schedule, and monitor pipelines.[2] The source data is a cloud MySQL database hosted on the Cloud SQL instance, and the destination for the data analytics is a data warehouse on Google Bigquery.

- *Building ETL Scripts*

The first part was building ETL scripts. We built ETL scripts in python for extracting the source data from a cloud database using pymysql connector, transforming the data using pandas library, and loading the data into the data warehouse on Google Bigquery using cloud client libraries.

- *Extraction*

The source data was extracted from a Google Cloud SQL instance MySQL relational database using pymysql connector. The Cloud SQL instance had a private VPC network which was hosted on a public IP address and needed authorizations and permissions to access. The connection string was formed by some information about the source database, such as the Cloud SQL instance's public IP address, an authorized user and password, name of database, and port number. The authorization of a user was managed by the IAM role to get permission to access the Cloud SQL instance. After connecting the cloud SQL instance successfully, the source data can be extracted from the source relational database by using queries and converts the queries' results into pandas dataframe by read_sql function from pandas libraries.

- *Transformation*

After the data was extracted from the source database and stored in pandas dataframe, we needed to transform the extracted data into a single format. Several transformation tasks were performed in the transform process. For example, NULL values and non-integer values in the columns with integer data type were converted to some default values. Some unnecessary columns were removed. Some new columns were inserted to help future data analysis. A column named update_date was added in each table to record the current date timestamp of the data being transformed, in order to prepare for the BigQuery data warehouse with the Star schema model.

- *Loading*

After the data frame was transformed into a standard format, we can load the transformed data into the BigQuery data warehouse using cloud client libraries. Firstly, we created two datasets in BigQuery, a staging dataset for loading staging data, and a destination dataset for the data warehouse. Then we created a fact table and several dimension tables based on Star Schema. After creating datasets and tables, we firstly ingested the transformed data from the pandas dataframe to the staging dataset in BigQuery using client.load_table_from_dataframe function in cloud client libraries. Then we used SQL queries to fetch the data from the staging dataset to the corresponding columns in tables in the data warehouse.

VI. DAG Workflow

The ETL pipeline in Apache Airflow was written in python using Direct Acyclic Graph (DAG) script architecture. DAG files were used to design ETL processes and schedule workflows for data pipelines. In our project, the entire ETL pipeline was written in a single DAG file. Each process in the pipeline was regarded as a task. In Airflow, the DAG file defined dependencies between tasks using directed edges to link relational tasks.[3] The ELT pipeline was scheduled to run once a week at every Sunday midnight by setting the schedule_interval to "0 0 * * 0". [4] Dummy Operators were used as the start and end tasks in the DAG file.

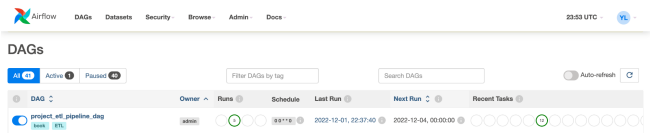


Figure 3 Airflow UI for DAGs List

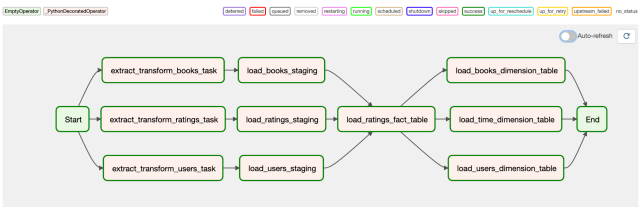


Figure 4 ETL Pipeline

VII. Data Warehouse Implementation

We designed a data warehouse in Google BigQuery using Star Schema Model with ratings table as the fact table, users table, books table, and time table as dimension tables.[7] The data values in these tables were ingested by running the ETL pipeline successfully.

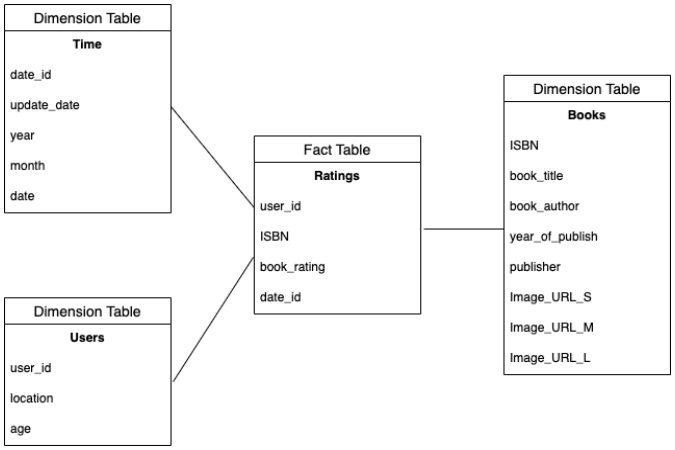


Figure 5 Star Schema Model for the Data Warehouse

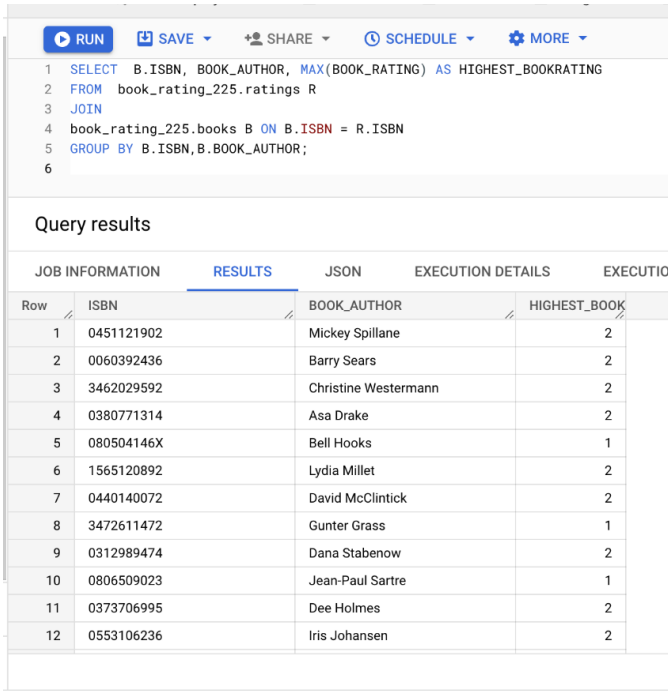
VIII. Data Analytics

After the data warehouse was established and data values were ingested in Google BigQuery, we performed several data analyses with SQL queries on our data warehouse.

- *Queries and Aggregations with Outputs using Google BigQuery*

a) Display highest-rated book author

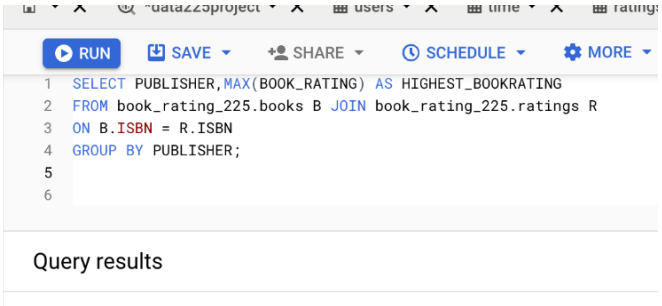
```
SELECT  B.ISBN, BOOK_AUTHOR,
MAX(BOOK_RATING) AS HIGHEST_BOOKRATING
FROM  book_rating_225.ratings R
JOIN
book_rating_225.books B ON B.ISBN = R.ISBN
GROUP BY B.ISBN,B.BOOK_AUTHOR;
```



Row	ISBN	BOOK_AUTHOR	HIGHEST_BOOK
1	0451121902	Mickey Spillane	2
2	0060392436	Barry Sears	2
3	3462029592	Christine Westermann	2
4	0380771314	Asa Drake	2
5	080504146X	Bell Hooks	1
6	1565120892	Lydia Millet	2
7	0440140072	David McClintick	2
8	3472611472	Gunter Grass	1
9	0312989474	Dana Stabenow	2
10	0806509023	Jean-Paul Sartre	1
11	0373706995	Dee Holmes	2
12	0553106236	Iris Johansen	2

b) Display highest book ratings for each publisher

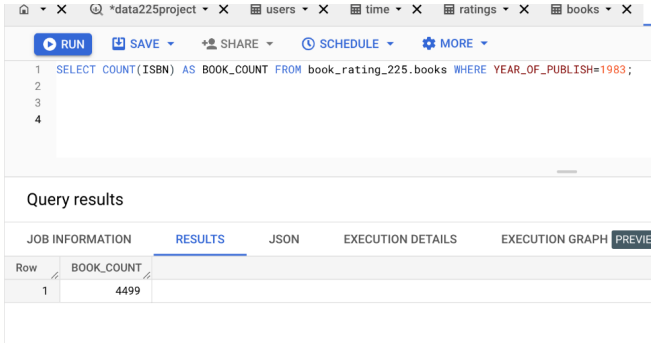
```
SELECT PUBLISHER,MAX(BOOK_RATING) AS
HIGHEST_BOOKRATING
FROM book_rating_225.books B JOIN
book_rating_225.ratings R
ON B.ISBN = R.ISBN
GROUP BY PUBLISHER;
```



Row	PUBLISHER	HIGHEST_BOOK
1	Hermann Luchterhand Verlag	1
2	Worldwide Church Of God	2
3	Grundle Ink Publications	2
4	Forlaget Oktober	1
5	Editorial de la Universidad de P...	2
6	Politicos Pub	2
7	stanyard creek publishing	2
8	Bridge City Books	2
9	SAS productions	1
10	Penmarin Books	2
11	Intouchables	2
12	VDE, Bln.	1

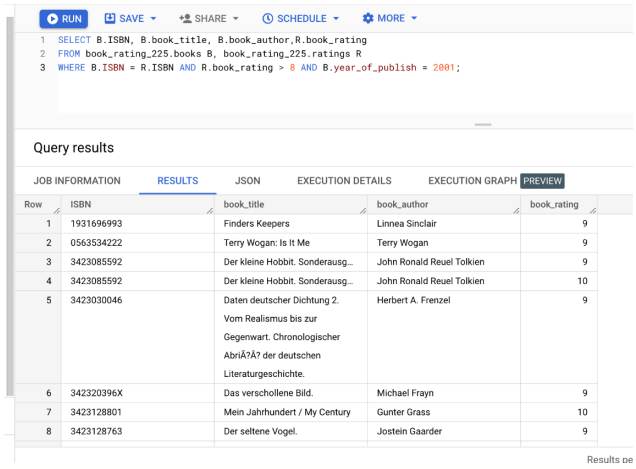
c) Display number of books that is published in the year 1983

```
SELECT COUNT(ISBN) AS BOOK_COUNT
FROM book_rating_225.books WHERE
YEAR_OF_PUBLISH=1983;
```



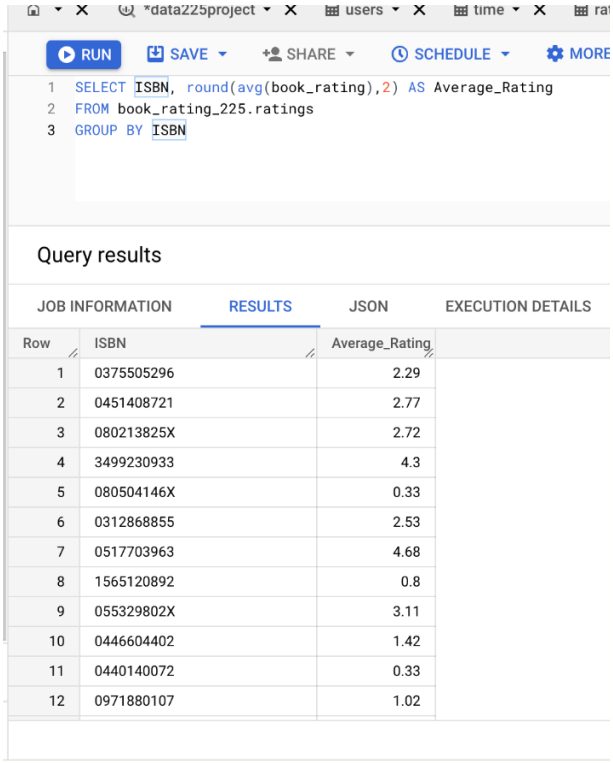
d) Display the 8+ rating books were published in 2001

```
SELECT B.ISBN, B.book_title,
B.book_author,R.book_rating
FROM book_rating_225.books B, book_rating_225.ratings
R
WHERE B.ISBN = R.ISBN AND R.book_rating > 8 AND
B.year_of_publish = 2001;
```



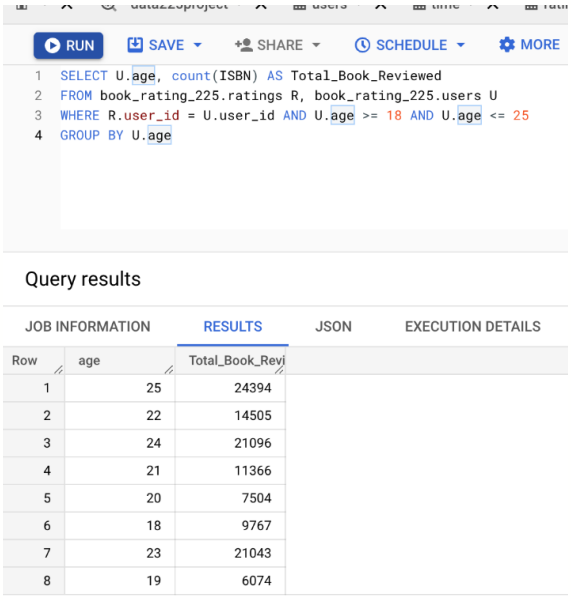
e) Display average book-ratings for each book

```
SELECT ISBN, round(avg(book_rating),2) AS
Average_Rating
FROM book_rating_225.ratings
GROUP BY ISBN
```



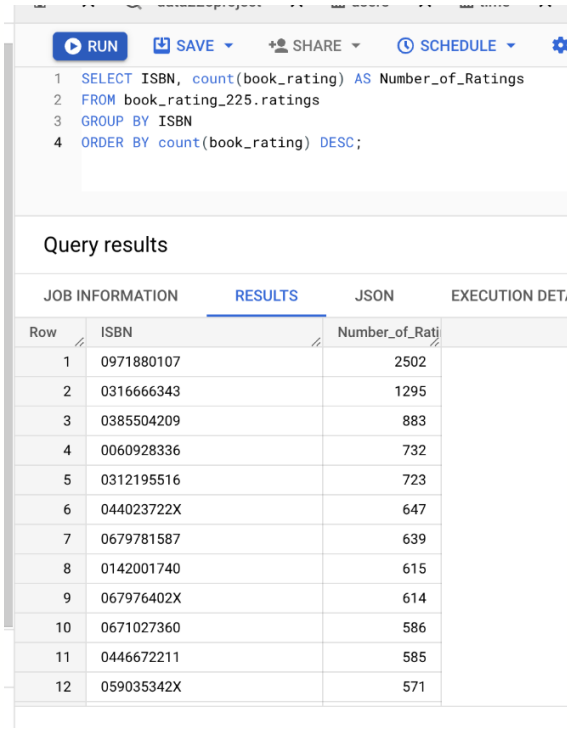
f) Display number of books that ages 18 - 25 users read

```
SELECT U.age, count(ISBN) AS
Total_Book_Reviewed
FROM book_rating_225.ratings R,
book_rating_225.users U
WHERE R.user_id = U.user_id AND U.age >= 18
AND U.age <= 25
GROUP BY U.age
```



g) Display number of ratings of the books in descending order

```
SELECT ISBN, count(book_rating) AS
Number_of_Ratings
FROM book_rating_225.ratings
GROUP BY ISBN
ORDER BY count(book_rating) DESC;
```



```
SELECT ISBN, count(book_rating) AS Number_of_Ratings
FROM book_rating_225.ratings
GROUP BY ISBN
ORDER BY count(book_rating) DESC;
```

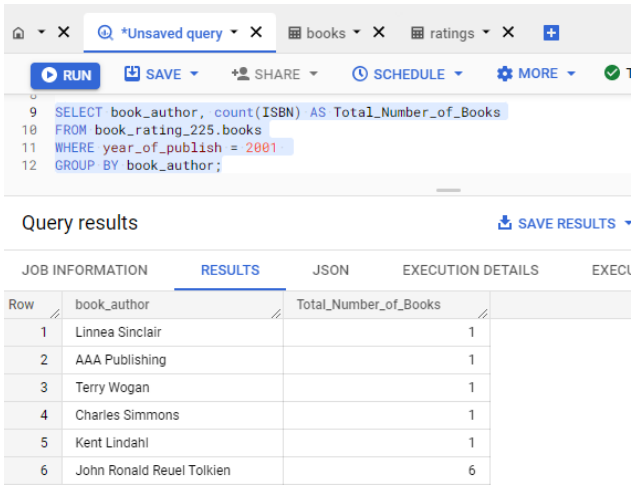
Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DET

Row	ISBN	Number_of_Rati
1	0971880107	2502
2	0316666343	1295
3	0385504209	883
4	0060928336	732
5	0312195516	723
6	044023722X	647
7	0679781587	639
8	0142001740	615
9	067976402X	614
10	0671027360	586
11	0446672211	585
12	059035342X	571

h) How many books did the authors publish in the year '2001'?

```
SELECT book_author, count(ISBN) AS
Total_Number_of_Books
FROM book_rating_225.books
WHERE year_of_publish = 2001
GROUP BY book_author;
```



```
SELECT book_author, count(ISBN) AS Total_Number_of_Books
FROM book_rating_225.books
WHERE year_of_publish = 2001
GROUP BY book_author;
```

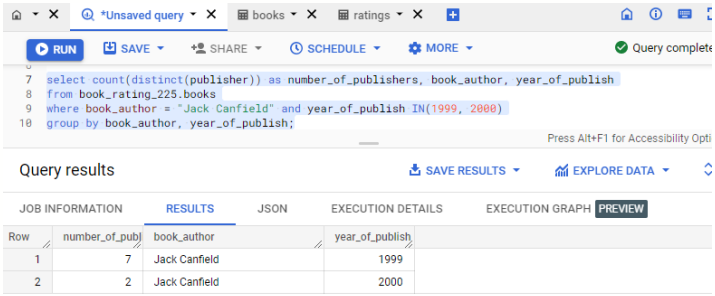
Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	book_author	Total_Number_of_Books
1	Linnea Sinclair	1
2	AAA Publishing	1
3	Terry Wogan	1
4	Charles Simmons	1
5	Kent Lindahl	1
6	John Ronald Reuel Tolkien	6

i) How many publishers published "Jack Canfiled" books in the year 1999 and 2000?

```
select count(distinct(publisher)) as number_of_publishers,
book_author, year_of_publish
from book_rating_225.books
where book_author = "Jack Canfield" and year_of_publish
IN(1999, 2000)
group by book_author, year_of_publish;
```



```
select count(distinct(publisher)) as number_of_publishers, book_author, year_of_publish
from book_rating_225.books
where book_author = "Jack Canfield" and year_of_publish IN(1999, 2000)
group by book_author, year_of_publish;
```

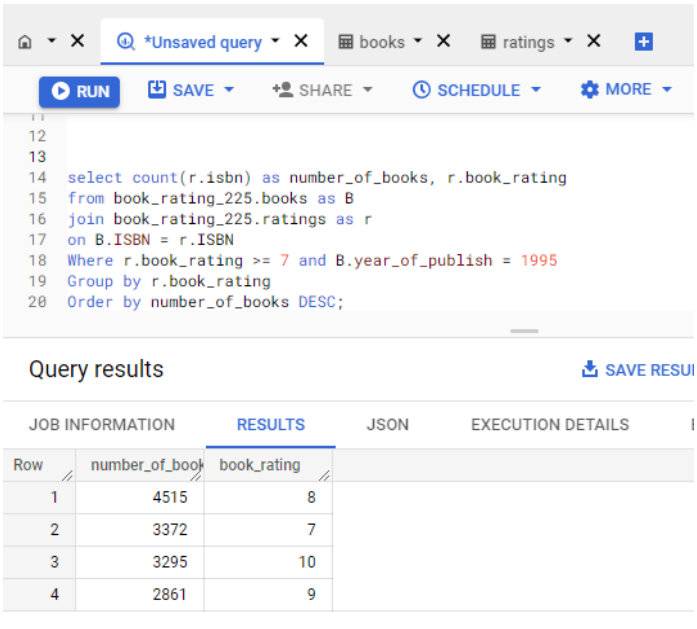
Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | EXECUTION GRAPH PREVIEW

Row	number_of_publ	book_author	year_of_publish
1	7	Jack Canfield	1999
2	2	Jack Canfield	2000

j) How many 7+ratings were published in the year '1995'?

```
select count(r.isbn) as number_of_books, r.book_rating
from book_rating_225.books as B
join book_rating_225.ratings as r
on B.ISBN = r.ISBN
Where r.book_rating >= 7 and B.year_of_publish = 1995
Group by r.book_rating
Order by number_of_books DESC;
```



```
select count(r.isbn) as number_of_books, r.book_rating
from book_rating_225.books as B
join book_rating_225.ratings as r
on B.ISBN = r.ISBN
Where r.book_rating >= 7 and B.year_of_publish = 1995
Group by r.book_rating
Order by number_of_books DESC;
```

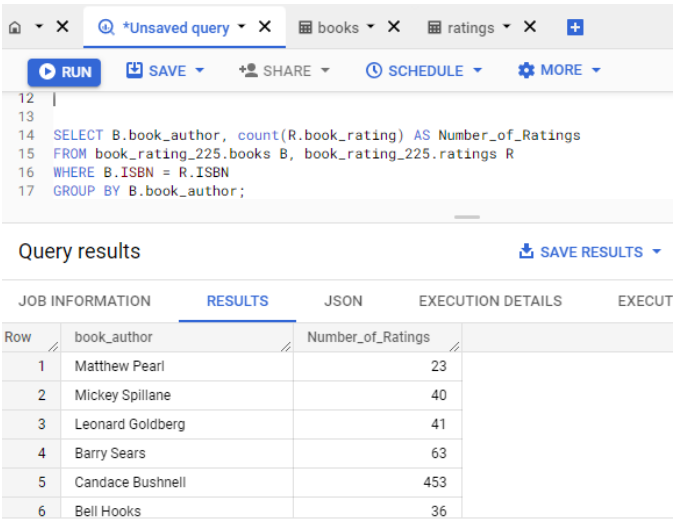
Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	number_of_book	book_rating
1	4515	8
2	3372	7
3	3295	10
4	2861	9

k) Find the number of ratings for each author

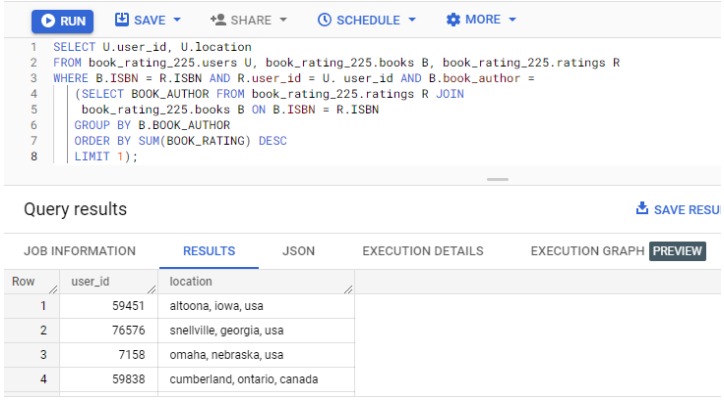
```
SELECT B.book_author, count(R.book_rating) AS
Number_of_Ratings
FROM book_rating_225.books B, book_rating_225.ratings
R
WHERE B.ISBN = R.ISBN
GROUP BY B.book_author;
```



Row	book_author	Number_of_Ratings
1	Matthew Pearl	23
2	Mickey Spillane	40
3	Leonard Goldberg	41
4	Barry Sears	63
5	Candace Bushnell	453
6	Bell Hooks	36

l) What Top cities and states do the users who read the books of the highest rated author?

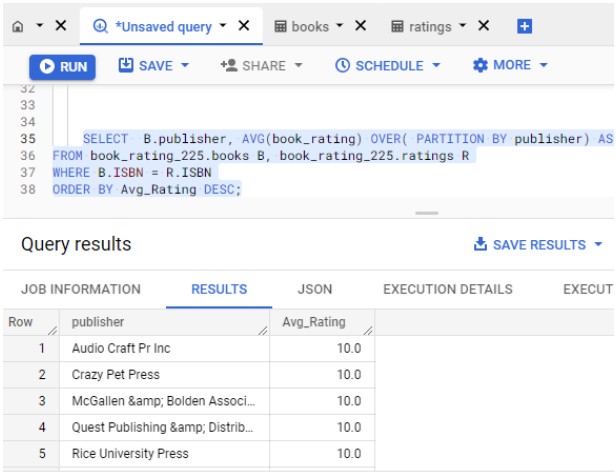
```
SELECT U.user_id, U.location
FROM book_rating_225.users U, book_rating_225.books
B, book_rating_225.ratings R
WHERE B.ISBN = R.ISBN AND R.user_id = U. user_id
AND B.book_author =
  (SELECT BOOK_AUTHOR FROM
book_rating_225.ratings R JOIN
   book_rating_225.books B ON B.ISBN = R.ISBN
  GROUP BY B.BOOK_AUTHOR
  ORDER BY SUM(BOOK_RATING) DESC
  LIMIT 1);
```



Row	user_id	location
1	59451	altoona, iowa, usa
2	76576	snellville, georgia, usa
3	7158	omaha, nebraska, usa
4	59838	cumberland, ontario, canada

m) Display the average book ratings partitioned by publishers?

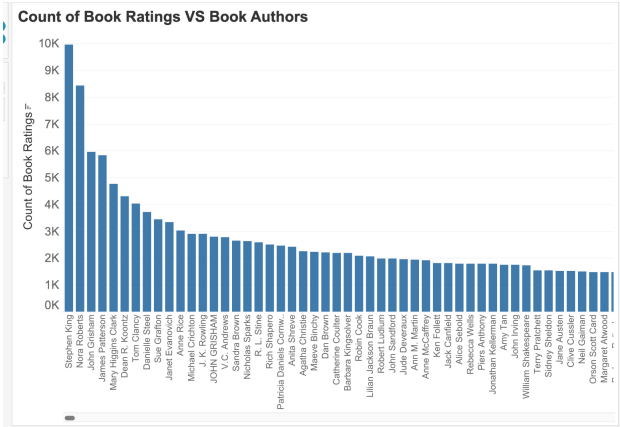
```
SELECT  B.publisher, AVG(book_rating) OVER(
PARTITION BY publisher) AS Avg_Rating
FROM book_rating_225.books B, book_rating_225.ratings
R
WHERE B.ISBN = R.ISBN
ORDER BY Avg_Rating DESC;
```



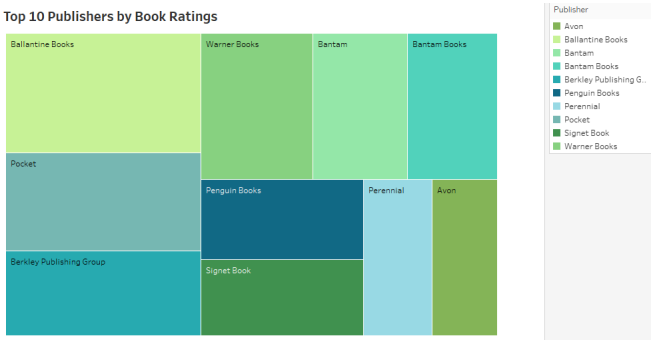
Row	publisher	Avg_Rating
1	Audio Craft Pr Inc	10.0
2	Crazy Pet Press	10.0
3	McGallen & Bolden Associ...	10.0
4	Quest Publishing & Distrib...	10.0
5	Rice University Press	10.0

IX. VISUALIZATION

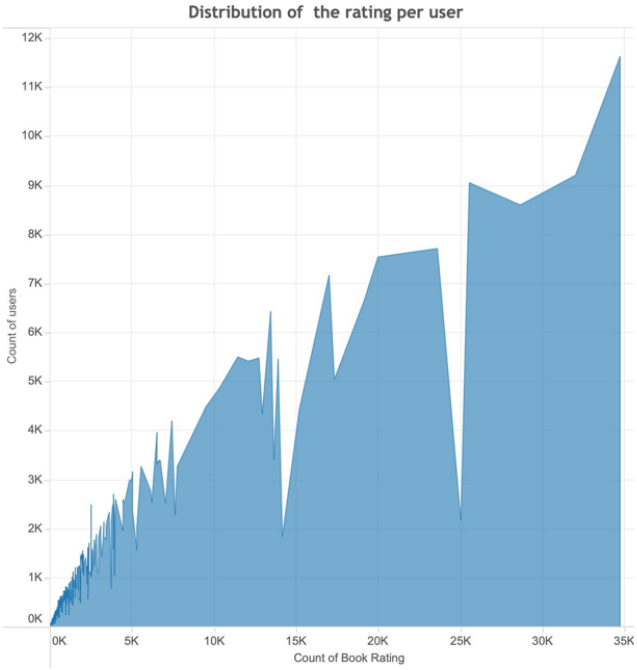
Using Tableau, we have used the data to create a visual representation. In order to use the data, we connected the book recommendation data with the BigQuery server and built a dashboard. The count of book ratings VS authors is represented in the first bar chart. From the chart, we can see that Stephen King has the most number of ratings, and it is followed by Nora Roberts and John Grisham, etc.



The Tree Map displays the Top 10 Publishers based on the book ratings. According to the chart, Ballentine Books are the top most publishers with overall book ratings of 97629 followed by Pocket and Berkley Publishing Group.



Distribution of the rating per user visualization states the number of times the user has read the book of the respected author published.Publisher Ballantine Books have the highest number of the count, i.e.,11,639, with the highest number of rating 34,724.



X. Learnings from Project

Implementing a data warehouse and performing Google BigQuery various analyses are challenging for us. We gained a lot of knowledge by studying on this project. Firstly, we deeply understood how the ETL process works.[5] We used this knowledge to design the ETL scripts for extracting data from the cloud SQL instance and ingesting into our data warehouse. Secondly, we got familiar with the Google cloud platform. We uploaded the source data in the Google cloud storage, worked on Google SQL instance as our source database, and designed the data warehouse on the Google BigQuery. Thirdly, we learned how Apache Airflow works and designed a DAG file to execute ETL pipeline and schedule the workflow.[6] Fourthly, we learned about data modeling with star schema in a data warehouse, working with a fact table and dimension tables.[7] Finally, we learned to connect the data warehouse on Google BigQuery to Tableau for our data visualization after we perform the data analytics.[8]

XI. Conclusion

In conclusion, our project successfully implemented a data warehouse on Google BigQuery and performed some analytics of the book recommendation database. We extracted data from a cloud database hosted on a private VPC network, and loaded data into the data warehouse, a dataset that allows scalability and flexibility. The functionality components include essential queries that we need for future analytics. We executed a python-based build ETL pipeline, arranged and monitored using cloud composer, which is based on Apache Airflow. Direct Acyclic Graph is the script architecture used by Airflow. The workflow is planned and scheduled using DAG files. Data is retrieved from a cloud SQL instance using the Python API and then loaded in Google Big Query using cloud client libraries as part of the ETL process. Then, several SQL queries were performed for data analysis and reporting. Key Performance Metrics (KPI) have been derived through business intelligence and displayed on a Tableau.

References

[1] “ETL Process Overview” (https://www.stitchdata.com/etldatabase/etl-process/)

[2] “Apache Airflow: Use Cases, Architecture, and Best Practices” (https://www.run.ai/guides/machine-learning-operations/apache-airflow)

[3] “Apache Airflow for Beginners - Build Your First Data Pipeline” (https://www.projectpro.io/article/apache-airflow-data-pipeline-example/610)

[4] “Scheduling & Triggers” (https://airflow.apache.org/docs/apache-airflow/1.10.1/scheduler.html)

[5] “ETL (Extract, Transform, and Load) Process in Data Warehouse” (https://www.guru99.com/etl-extract-load-process.html)

[6] “ApacheAirflow Documentation” (https://airflow.apache.org/docs/apache-airflow/stable/tutorial/index.html)

[7] “What is a Cloud Data Warehouse” (https://www.qlik.com/us/cloud-data-migration/cloud-data-warehouse)

[8] “A Complete Guide for Google BigQuery Authentication” (https://www.progress.com/tutorials/jdbc/a-complete-guide-for-google-bigquery-authentication)