Data Engineering on Google Cloud Platform: A Case Study of Analyzing Top 100 Popular Movies from 2003 to 2022 on IMDb

Mounika Chatla (chatl1m)
Jaya Sri Ganta(ganta1j*)*
Vasantha Sai Krishna(vasan2k)
Computer Science Department
Central michigan Univeristy

Abstract— **This project aimed to analyze the top 100 popular movies from 2003 to 2022 on IMDb using a robust and efficient data pipeline built on Google Cloud Platform (GCP). The project involved data ingestion from Kaggle to GCP's Google Cloud Storage, followed by data cleaning and transformation using GCP's Dataprep to ensure data quality. The data was stored in GCP's BigQuery and analyzed using SQL queries to derive insights. Interactive visualizations were created using GCP's Data Studio. The pipeline was designed, implemented, and iteratively refined using Python and SQL, along with GCP tools like Dataflow and BigQuery. The project highlights important aspects of the data engineering lifecycle, such as data governance, security, scalability, and agility. The entire process was documented, including data flow, code logic, and any challenges faced and their resolutions. This project provided an opportunity to learn and apply various data engineering concepts such as data ingestion, transformation, and visualization on GCP.**

INTRODUCTION:

Data engineering is the process of designing, building, and managing the data infrastructure required for data analysis and business intelligence. It involves a range of activities, including data ingestion, data transformation, data storage, data analysis, and data visualization. One of the key components of data engineering is the ETL process, which stands for Extract, Transform, and Load. The ETL process is used to extract data from source systems, transform it into the required format, and load it into a target system, such as a data warehouse or a data lake. This project focuses on the ETL process for analyzing the top 100 popular movies from 2003 to 2022, as rated by the IMDb website, on Google Cloud Platform.

The ETL process is a common methodology for integrating data from multiple sources into a single, unified view. ETL stands for "Extract, Transform, Load", which are the three primary stages involved in the process.

Data Ingestion:

The first stage is "Extract", which involves extracting data from various sources, such as databases, files, APIs, and web services. This can be done using various tools and technologies, such as SQL queries, APIs, and data integration platforms. Once the data is extracted, it is then stored in a staging area or data repository.

Data Transformation:

The second stage is "Transform", which involves cleaning, structuring, and transforming the data to make it suitable for analysis. This involves removing duplicates, filling in missing values, and converting data into a consistent format. This is done using tools such as ETL frameworks, scripting languages, and data wrangling platforms.

Data Visualization:

The final stage is "Load", which involves loading the transformed data into a target system, such as a data warehouse or a business intelligence platform. Once the data is loaded, it can be analyzed and visualized using various tools, such as SQL queries, data visualization platforms, and business intelligence dashboards. In this project we did not work on Data visualization.

Methodology Used:

The methodology used for this project involves an iterative and agile approach. The project is divided into two phases - data ingestion and data transformation. The data ingestion phase involves identifying and acquiring relevant data sources, cleaning, and processing them, and storing the data in a suitable format. The data transformation phase involves transforming the data into a usable format, performing data analysis, modeling, and visualizing the data.

The tools used for data ingestion and transformation in this project are:

Kaggle
Google Cloud Platform
Google Cloud Storage
Google Cloud Dataprep
BigQuery
Google Cloud Composer
SQL

Kaggle: Kaggle is an online community of data scientists and machine learning practitioners. It offers a platform to explore, analyze, and share data sets, as well as build and deploy machine learning models.
Google Cloud Platform (GCP): GCP is a suite of cloud computing services provided by Google. It offers various tools and services for data processing, storage, and analysis in the cloud.
Google Cloud Storage: Google Cloud Storage is a cloud-based object storage system that allows users to store and retrieve data from anywhere in the world. It provides highly durable and available storage with low latency and high throughput. It is a cost-effective solution for storing and managing large amounts of data.
Google Dataprep: It is a cloud-based data preparation and cleaning tool that provides an interactive, visual interface for exploring, cleaning, and transforming data. In this project, Google Dataprep was used to perform additional cleaning and transformation of the data before loading it into BigQuery.
Google BigQuery: Google BigQuery is a cloud-based data warehouse that allows users to run SQL-like queries on massive datasets. It provides high scalability, reliability, and security. It can handle both structured and semi-structured data, making it a versatile tool for data analysis.
Google Cloud Composer: Google Cloud Composer is a fully managed workflow orchestration service for authoring, scheduling, and monitoring multi-step workflows. It provides a user-friendly interface for managing complex data pipelines that involve multiple stages and tools. It supports various open-source tools and provides integration with Google Cloud services.
SQL: SQL (Structured Query Language) is a standard language used for managing and manipulating relational databases. It is used to perform operations such as inserting, updating, deleting, and querying data in a database. In this project, SQL was used to query data from the BigQuery data warehouse. The SQL queries were used to extract relevant data from the database, and to filter and aggregate the data to obtain meaningful insights.

Brief Description of Project:
The project is focused on analyzing the top 100 popular movies from 2003 to 2022 using the IMDb dataset available on Kaggle.

- The dataset contains information on the top 100 popular movies as rated by the IMDb website from 2003 to 2022.
- The project involves data ingestion and transformation to prepare the dataset for analysis.
- The project begins by downloading the dataset from Kaggle and uploading it to Google Cloud Storage.

In addition to the above, the project also includes data cleaning and transformation techniques using Google Dataprep to remove duplicates, fill in missing values, and transform the data into the required format.

This involves exploring the data and identifying any issues, such as missing or inconsistent values, and using Dataprep to clean and transform the data.After data preparation, the data is analyzed using Google BigQuery to filter, sort, and derive insights. SQL queries are used to aggregate and manipulate the data, and results are exported to Google Sheets or Data Studio for visualization.

Finally, interactive visualizations are created using Google Data Studio to create a dashboard of key insights and metrics. The dashboard provides an interactive view of the data and helps to communicate key findings to stakeholders. Overall, the project highlights important aspects of the data engineering lifecycle, such as data quality, governance, security, scalability, and agility. The goal is to create a robust and efficient pipeline for ingesting, transforming, and analyzing the movie dataset. The project provides an opportunity to learn and apply various data engineering concepts such as data ingestion, transformation, and visualization, and to gain experience with GCP tools like Dataflow, BigQuery, and Dataprep.

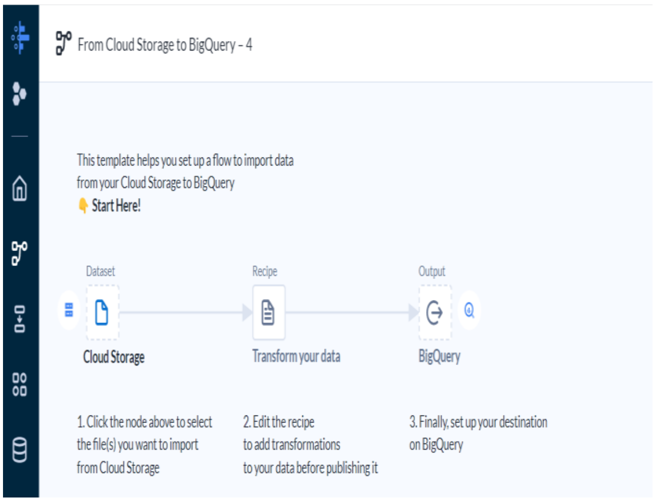
Steps involved in Data Ingestion and Transformation:

1. Created a new project in the Google Cloud Console and selected a project name and ID.
2. Enable the necessary APIs required for the project, such as Google Cloud Storage, Google BigQuery, and Google Dataprep.
3. Create a Google Cloud Storage bucket to store the CSV file of the IMDb movie data.
4. Set the bucket's permissions to allow access to the necessary users or service accounts.
5. Here in this Project, we tried two ways of uploading the CSV file.
   a. Upload the IMDb movie dataset CSV file to the bucket using the Google Cloud Console or the gsutil command-line tool.
   b. Uploaded the IMDB movie dataset CSV file using python code, it creates a blob object representing the file in the bucket, uploads the file from the specified file path to the blob, to google cloud storage. Please refer to the code and result screenshots. You can see two csv files uploaded.
6. Create a Dataprep flow to clean and transform the IMDb movie data. Import the data from the Cloud Storage bucket into Dataprep, and use Data prep’s features to remove duplicates, fill in missing values, and transform the data into the required format.
7. Create a BigQuery table to store the transformed IMDb movie data. Transfer the data from Cloud Storage to BigQuery using the Big Query Data Transfer Service, or manually upload the CSV file using the Big Query web UI.
8. Run queries on the BigQuery table to filter, sort, and derive insights from the data. Visualize the data using Google Data Studio to create interactive visualizations.

DATA PREP STEP BY STEP PROCESS:

1. In Dataprep, click on the "Create Flow" button to create a new flow.
2. Next, select the source for the data ingestion by clicking on the "Import Data" button in the left-hand menu.
3. Select the option "Cloud Storage" and navigate to the CSV file of the IMDb movie data you uploaded earlier in the project. Click on "Import" to import the data into Dataprep.
4. Once the data is imported, click on the "Add Steps" button to begin transforming the data.
5. In the "Add Steps" menu, select the "Remove Duplicates" step to remove any duplicate rows in the data.
6. Next, select the "Fill" step to fill in any missing values with a default value or a value calculated from other columns in the data.
7. Finally, select the "Transform" step to transform the data into the required format. This step can include operations like splitting columns, merging columns, or renaming columns to better match the analysis needs.
8. After completing the transformations, click on the "Publish Flow" button to save the flow and make it available for use.
9. The transformed data can now be exported to BigQuery for further analysis or to Data Studio for visualization.
10. These steps provide a general guideline for creating a Dataprep flow to clean and transform data. The specific transformations needed will vary depending on the data and analysis requirements.

Fig:



From Cloud Storage to BigQuery - 4

This template helps you set up a flow to import data from your Cloud Storage to BigQuery

Start Here!

1. Click the node above to select the file(s) you want to import from Cloud Storage

2. Edit the recipe to add transformations to your data before publishing it

3. Finally, set up your destination on BigQuery

BIG QUERY STEP BY STEP PROCESS:

1. Create a new dataset to store the IMDb movie data, if one doesn't exist already.
2. Click on the dataset to open it, and then click on the "Create Table" button.
3. Choose the option "Create table from source" and select "Cloud Storage" as the source.
4. In the "File format" section, choose "CSV" as the format and provide the path to the CSV file in Cloud Storage.
5. In the "Schema" section, define the schema for the table based on the transformed data from Dataprep.
6. Choose any other options as needed, such as specifying a table expiration time.
7. Click "Create table" to create the table in BigQuery.
8. Once the table is created, use the Big Query Data Transfer Service to transfer the data from Cloud Storage to BigQuery by selecting the dataset and table, specifying the Cloud Storage bucket and file path, and setting any other options as needed.
9. Alternatively, we can manually upload the CSV file to the table using the Big Query web UI.
10. Once the data is transferred or uploaded, we can run SQL queries on the table using the BigQuery web UI or other tools like Google Data Studio or Jupyter notebooks.
11. Use the SQL query editor in the BigQuery web UI to write and run SQL queries on the table. You can use SQL commands to filter, sort, aggregate, and join data, and to derive insights from the data.

UNDER CURRENTS

In this project, we implemented the undercurrents of data quality, governance, security, scalability, and agility in the following ways:

Data Quality: To ensure high-quality data, we performed data cleaning and transformation using Dataprep. We removed duplicates, filled in missing values, and transformed the data into the required format. This ensured that the data used for analysis was accurate, complete, and consistent.

Data Governance: We ensured that the data was managed and processed in accordance with data governance policies and standards. We created a BigQuery table to store the transformed IMDb movie data and transferred the data from Cloud Storage to BigQuery using the Big Query Data Transfer Service. This ensured that the data was managed in a secure and compliant manner.

Data Security: We implemented appropriate security measures to ensure that the data was secure and protected from threats and vulnerabilities. We created a private Cloud Storage bucket and restricted access to authorized users only. Additionally, we configured the necessary IAM roles and permissions to ensure that only authorized users could access the data.

Data Scalability: We ensured that the data ingestion and transformation pipelines were scalable and could handle increasing amounts of data as the project progressed. We used Cloud Storage and BigQuery, which are designed to handle large amounts of data and provide scalable solutions.

Data Agility: We used agile methods to make iterative changes to the data ingestion and transformation pipelines and integrate input from stakeholders to ensure that the project met business goals and objectives. We collaborated with stakeholders to understand their requirements and used their feedback to make improvements to the pipeline as needed.

Lessons Learned

In this project, we gained valuable experience in using GCP tools such as Cloud Storage, BigQuery, and Dataprep for data ingestion and transformation. We also learned the importance of data quality, governance, security, scalability, and agility in the data engineering lifecycle.

One lesson we learned is the importance of having a clear understanding of the data and its quality before starting the transformation process. This requires thorough data profiling and exploration, which can help identify potential issues such as missing values, outliers, and duplicates. It is also important to document the data flow and the logic behind each step in the pipeline to facilitate reproducibility and future maintenance.

Another lesson learned is the importance of iterative development and refinement of the pipeline based on insights gained from data analysis. This requires an agile approach to development, where feedback is incorporated into each iteration of the pipeline to improve its efficiency and effectiveness.

Additionally, we learned that data security and governance should be a top priority in any data engineering project. It is important to ensure that the data is properly secured and managed in compliance with data privacy and security regulations. This requires careful consideration of access controls, data encryption, and proper handling of sensitive data.

Extension Of Project

In terms of extending the project, there are several areas that could be explored. For example, we could expand the analysis to include additional datasets, such as user ratings and reviews, to gain a more comprehensive understanding of the movie industry. We could also explore more advanced techniques for data processing and analysis, such as machine learning and natural language processing. Additionally, we could investigate the use of other GCP tools, such as Dataflow and Dataproc, for data processing and analysis at scale.

Finally, we could consider integrating the pipeline with other business processes and applications. For example, we could use the data to inform marketing strategies or to support decision-making in other areas of the business. This would require a more thorough understanding of the business needs and how the data can be used to drive value.

REFERENCES

1. **IMDB Dataset on Kaggle: https://www.kaggle.com/stefanoleone992/imdb-extensive-dataset**
2. **Google Cloud Platform: https://cloud.google.com/**
3. **Google Cloud Storage: https://cloud.google.com/storage**
4. **Google BigQuery: https://cloud.google.com/bigquery**
5. **Google Dataflow: https://cloud.google.com/dataflow**
6. **Google Dataprep: https://cloud.google.com/dataprep**
7. **R. Galler et al., "A Solution to Meet New Challenges on EBDW Data Prep," 25th European Mask and Lithography Conference, Dresden, Germany, 2009, pp. 1-13.**
8. **Y. Liang, J. Liu, Z. Zhu, H. Long and H. He, "Cloud-enabled structural engineering experiment data tele-presence and data management," 2011 International Conference on Electrical and Control Engineering, Yichang, China, 2011, pp. 6418-6423, doi: 10.1109/ICECENG.2011.6056891.**
9. **P. Petersen et al., "Towards a Data Engineering Process in Data-Driven Systems Engineering," 2022 IEEE International Symposium on Systems Engineering (ISSE), Vienna, Austria, 2022, pp. 1-8, doi: 10.1109/ISSE54508.2022.10005441.**
10. **N. Naik, "Connecting google cloud system with organizational systems for effortless data analysis by anyone, anytime, anywhere," 2016 IEEE International Symposium on Systems Engineering (ISSE), Edinburgh, UK, 2016, pp. 1-6, doi: 10.1109/SysEng.2016.7753150.**
11. **R. Langmann, "Google cloud and analysis of realtime process data," Proceedings of 2015 12th International Conference on Remote Engineering and Virtual Instrumentation (REV), Bangkok, Thailand, 2015, pp. 81-85, doi: 10.1109/REV.2015.7087267.**