DATA 228 - Big Data Tech & App

Term Project Report

May 2021

Data Analytics Pipeline in Google Cloud Platform for Consumer Reviews of Amazon Products

Group #1

Viritha Vanama - 015356991

Aneshaa Kasula - 014558427

Mohini Patil - 015359188

Project Abstract

Our project deals with capturing customer reviews about products sold on the Amazon website and streamlining the captured data using GCP, which could be leveraged for various analytics. Amazon hosts many variants of any given product from different manufacturers, it is the reviews & ratings from fellow customers which help others in making quick decisions on future purchases. As data enthusiasts, in this project, we are trying to analyze a couple of scenarios about consumer behavior.

Introduction & Background

The primary purpose of any Data Analysis is finding valuable insights and the whole purpose of data collection in the BigData context is to perform data analysis and make better decisions based on the information. The biggest challenge when handling Big Data is the hardware architecture. This is where cloud technologies like Google Cloud, AWS, etc come into the picture and which liberates cluster management from data science.

Google Cloud platform hides its internal architecture and helps in streamlining the BigData Life Cycle easy to implement. Most of its products like Google BigQuery, Cloud Dataflow, Cloud Pub/Sub, and Cloud ML Engine which are commonly used in data pipelines are all serverless and autoscaling.

We have chosen Amazon Customer Reviews which is one of the iconic datasets. In a period of over two decades since the first review in 1995, millions of Amazon customers have contributed over a hundred million reviews to express opinions and describe their experiences regarding products on the Amazon.com website. This makes Amazon Customer Reviews a rich source of information for academic researchers in the fields of Natural Language Processing (NLP), Information Retrieval (IR), and Machine Learning (ML), amongst others.

Project Description & Requirements

Description & Requirements

This Project deals with capturing CSV Data (i.e. both historical and streaming) into BigQuery via the DataFlow pipeline and then visualizing with the help of Data-Studio. Incoming Data could either be a CSV file on Google Storage Bucket or Streaming data that's published to GCP Pub/Sub.

- Capture the Input data.
- Colocate all the incoming data using Data-Flow Pipeline
- Data-Flow Pipeline will ingest data into BigQuery.
- End data hosted in BigQuery is visualized using Data-Studio.

The requirements to achieve the above steps are:

1. VM in GCP
    a. Python venv, with all required GCP packages
    b. Apache Beam - DataFlow
2. Google Storage Buckets
3. GCP Pub/Sub
4. Big Query
5. Data Studio

Problems & constraints

As the project scope is restricted to reviews, ratings and doesn't include the orders and returns information, the insights we will be delivering would vary in real-world scenarios.

Objectives

Our objective is to build a scalable and efficient data pipeline architecture that completes the data-to-information transformation process which is the key factor in the success of any analytics. We are building a Data Analytics pipeline in Google Cloud Platform on amazon product reviews dataset to achieve the following goals:

- ❏ Which are the most reviewed products?
- ❏ Top 10 rated products
- ❏ The popularity of free shipping products
- ❏ Popular category based on gender

End-Users will be Marketing & Sales Department along with Sellers.

Scope

The scope of the project includes the products, categories, shipping, gender, reviews, and rating information based on each customer. And also data ingestion is automated using a python program rather than using API.

Boundaries

The scope of the project is limited to descriptive analysis of multiple features and does not include the financial and machine learning aspects.

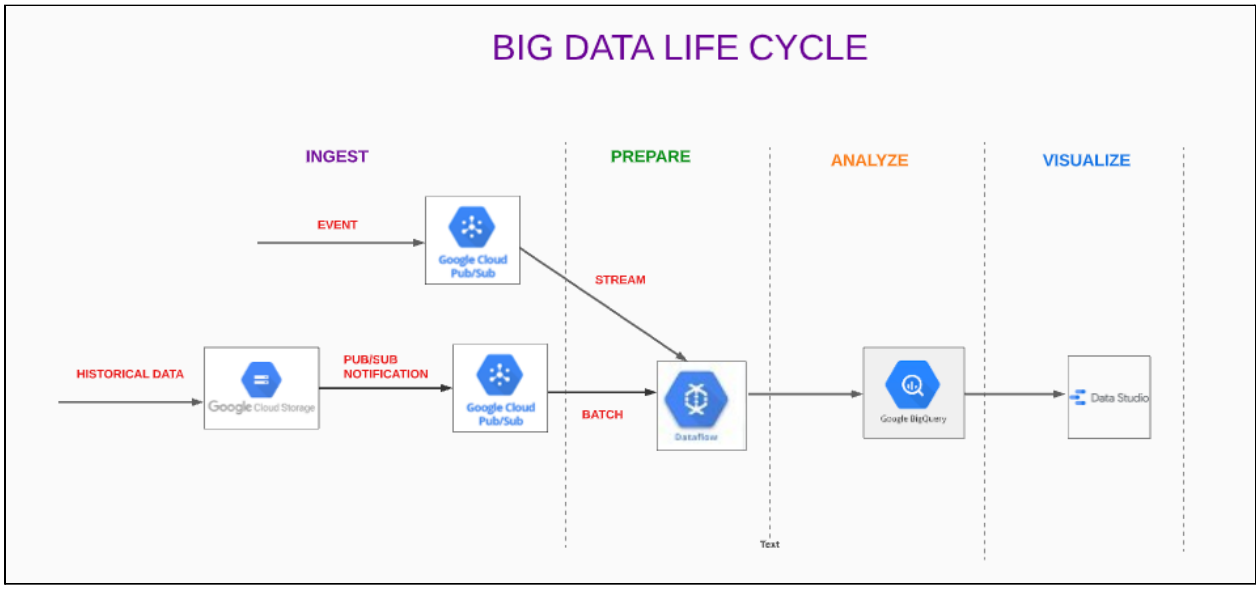
Architecture for Data Analytics Pipeline

A data pipeline is architectured to capture raw data from different sources, process it, and route the data so that it can be used to gain insights. It streamlines data events, making it easier for reporting, analysis, etc.

The GCP components we have used for data analytics pipeline milestones are:

- Google Cloud Pub/Sub
  - for streaming data ingestion
  - for publishing batch file notifications
- Google Cloud Storage
  - Storing Batch files
  - Storing staging and temp data for data flow
- Data Flow
  - to prepare and process batch and streaming data
- Google BigQuery
  - Data Warehouse
- Data Studio
  - to visualize information

The data analytics pipeline architecture diagram is as shown below:



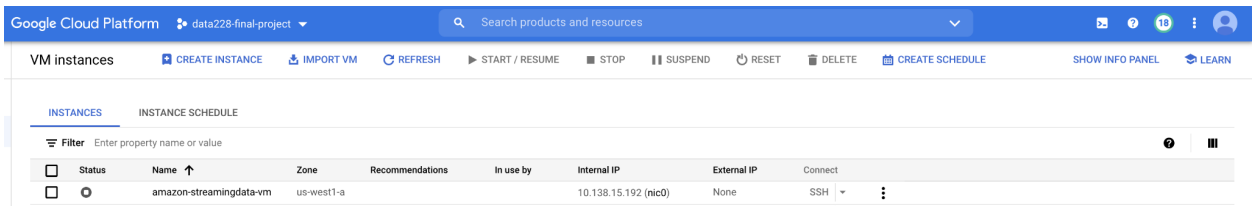
Data Ingestion

Before data ingestion, we have set up an environment to build a data analytics pipeline in the Google cloud platform (GCP).

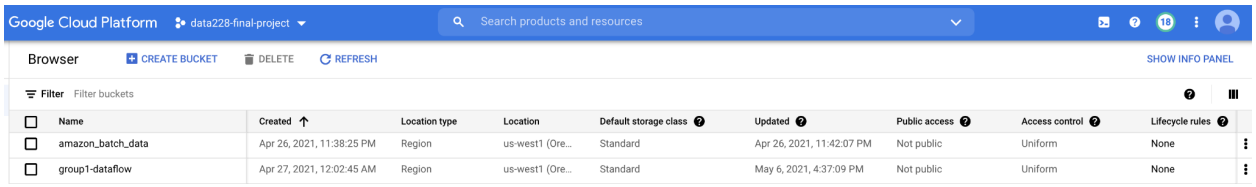
GCP environment setup

We have created a new project "'data228-final-project" in the google cloud platform (GCP).

- 1 Compute Instance: **amazon-streamingdata-vm**



- 2 Storage Buckets :
  - Batch Files: **amazon_batch_data**
  - Data Flow (Stage and Temp): **group1-dataflow**



Python virtual environment setup

- **Installed python virtual environment using pip**
- **Activated venv and installed cloud storage, pub/sub, dataflow, and big query packages.**

```
(venv) aa@amazon-streamingdata-vm:~/term_project$ pip list
Package                          Version
-------------------------------- ---------
apache-beam                      2.2.0
avro                             1.10.2
cachetools                       4.2.2
certifi                          2020.12.5
cffi                             1.14.5
chardet                          4.0.0
crcmod                           1.7
dill                             0.2.6
future                           0.16.0
gapic-google-cloud-pubsub-v1     0.15.4
google-api-core                  1.26.3
google-apitools                  0.5.11
google-auth                      1.30.0
google-auth-httplib2             0.0.4
google-cloud-bigquery            0.25.0
google-cloud-core                0.25.0
google-cloud-dataflow            2.2.0
google-cloud-pubsub              0.26.0
google-cloud-storage             1.38.0
google-crc32c                    1.1.2
google-gax                       0.15.16
google-resumable-media           1.2.0
googleapis-common-protos         1.5.5
googledatastore                  7.0.1
grpc-google-iam-v1               0.11.4
grpcio                           1.37.1
httplib2                         0.9.2
idna                             2.10
libcst                           0.3.18
mock                             2.0.0
mypy-extensions                  0.4.3
oauth2client                     3.0.0
packaging                        20.9
pbr                              5.6.0
pip                              21.1.1
pkg-resources                    0.0.0
ply                              3.8
proto-google-cloud-datastore-v1 0.90.4
proto-google-cloud-pubsub-v1     0.15.4
proto-plus                       1.18.1
protobuf                         3.3.0
pyasn1                           0.4.8
pyasn1-modules                   0.2.8
pycparser                        2.20
pyparsing                        2.4.7
pytz                             2021.1
PyYAML                           3.13
requests                         2.25.1
rsa                              4.7.2
setuptools                       40.8.0
six                              1.10.0
typing                           3.6.6
typing-extensions                3.10.0.0
typing-inspect                   0.6.0
urllib3                          1.26.4
wheel                            0.36.2
(venv) aa@amazon-streamingdata-vm:~/term_project$
```

About Data

Source: Datafiniti's datasets

Dataset: Consumer reviews of Amazon products

Filetype : CSV

This dataset contains over 28,000 lists of consumer reviews and ratings for Amazon products with various categories such as electronics, home & garden, health & beauty, animals & pet supplies, etc. We have selected around 27,400 records subset of records and required columns for our analysis from Jan 2015 to Mar 2019. Also generated a couple of columns and data which are required for the objectives.

Batch Data Ingestion

We considered the batch as historical data (initial and subsequent data) concerning the date when customers have reviewed products i.e.records from Jan 2015 to Dec 2017 in the dataset which includes basic product information, user details with reviews, and ratings given to Amazon products. This historical data will be stored in a separate GCS bucket.

We created a topic **batch_fileupload_notification** in GCP pub/sub to publish a message when a new batch file is triggered in GCS bucket **amazon_batch_data**. This message contains metadata of the file in JSON format as shown below:

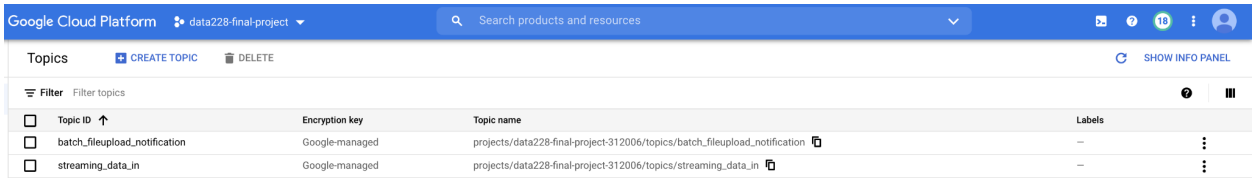
```
(data228-final-project-312006)$ gsutil notification create -t  batch_fileupload_notification -f json -e OBJECT_FINALIZE gs://amazon_batch_data
projects/_/buckets/amazon_batch_data/notificationConfigs/1
(data228-final-project-312006)$
```

Publish time	Attribute keys	Message body	Ack

May 1, 2021, 6:54:04 PM	bucketId
	eventTime
	eventType
	notificationConfig
	objectGeneration
	objectId
	payloadFormat
		{
		"kind": "storage#object",
		"id": "amazon_batch_data/batch.csv/1619875444037369",
		"selfLink":
		"https://www.googleapis.com/storage/v1/b/amazon_batch_data/o/batch.cs
		"name": "batch.csv",
		"bucket": "amazon_batch_data",
		"generation": "1619875444037369",
		"metageneration": "1",
		"contentType": "application/vnd.ms-excel",
		"timeCreated": "2021-05-01T13:24:04.043Z",
		"updated": "2021-05-01T13:24:04.043Z",
		"storageClass": "STANDARD",
		"timeStorageClassUpdated": "2021-05-01T13:24:04.043Z",
		"size": "28500",
		"md5Hash": "uYhQ4yFmwsLaALmVlUf0KA==",
		"mediaLink":
		"https://www.googleapis.com/download/storage/v1/b/amazon_batch_data/
		?generation=1619875444037369&alt=media",
		"crc32c": "e1mFcA==",
		"etag": "CPmtsKPKqPACEAE="
		}
			Deadline exceeded

Streaming Data Ingestion

We are publishing messages through a python program from VM (venv) which reads the data from a CSV file and publishes messages to a pub-sub topic. Created a topic **streaming_data_in** to publish this streaming data.



Data Preparation

As part of data preparation, we have filtered the fields from the original dataset as shown in the table with their descriptions to meet our primary objectives:

COLUMN NAME	DESCRIPTION
id	product id
dateAdded	date the product was first added to the product database
dateUpdated	most recent date the product was updated or seen by our system
name	product's name
asins	The ASIN (Amazon identifier) used for the product, Ex."B0009XCFRE"
brand	The brand name of the product
categories	A list of category keywords used for the product across multiple sources
primaryCategories	A list of standardized categories to which the product belongs
reviews_date	The date the review was posted
reviews_rating	start rated value for products (0-5)
reviews_text	The full (or available) text of the review
reviews_title	Title of review given by user
reviews_username	The reviewer's username
reviews_users_gender	reviewer's gender (Male/ Female)
shipping	shipping status either free or paid of the product

The snippet of CSV data is shown below screenshot:

id	dateAdded	dateUpdated	name	asins	brand	categories	primaryCategories	reviews_date	reviews_rating	reviews_text	reviews_title	reviews_username	reviews_users_gender	shipping
AVpf2cQm1cnluZ0-sb5y	2017-01-30T18:40:57	2019-02-24T04:01:52	Amazon Kindle R	B001NIZB5M	Amazon	Computers & Acc	Electronics	2019-02-26T00:00:00.000	5	Since the details for the iten	Kindle Power Adapter Specs	WingNut/Pilot	Female	Free
AVpiGyzGLieJMLi43iUd	2015-11-29T21:24:13	2019-02-24T21:41:04	AmazonBasics Bl	B005OOKNP4	AmazonBasics	Electronics Featu	Electronics	2015-12-05T00:00:00Z	5	It's a bit smaller than Apple'	Great keyboard	debugy2k	Female	Paid
AVpgyLH7ilAPnD_xxQ07	2016-07-21T02:17:37	2019-04-19T07:30:55	Kindle PowerFast	B006GWO7UA	Amazon	Chargers & Adap	Electronics	2018-12-17T00:00:00.000	5	I do a lot of international tra	Very Nice International Charge	Just Wondering	Male	Paid
AVpe7JuRilAPnD_xQ_M6	2015-10-17T00:32:56	2019-03-26T02:07:55	Oem Amazon Kir	B008GIQS1A	Amazon	Tablet & eBook R	Electronics	2017-10-10T00:00:00.000	1	Is Amazon kidding me They	cannot believe this is not inclu	jetgrrrl	Female	Paid
AVpe7nGV1cnluZ0-aG2o	2014-10-28T11:14:38	2019-04-25T09:05:28	AmazonBasics N	B00DIHVMEA,B0	Amazonbasics	Audio & Video Ac	Electronics	2016-06-14T05:00:00Z	5	After discarding and getting	It was a much needed storage	Diablita	Male	Paid

Both historical and streaming data go through the same steps of cleaning, filtering, and formatting in the dataflow we created earlier.

We have created a table in Bigquery to store and analyze the Amazon customer reviews processed data from dataflow. The SQL query of creation of table, schema, and details of the table before ingestion of data are as follows:

Dataset and Table Name: term_project.amazon_data_raw

```sql
CREATE TABLE term_project.amazon_data_raw (
id STRING,
dateAdded DATETIME,
dateUpdated DATETIME,
name STRING,
asins STRING,
brand STRING,
categories STRING,
primaryCategories STRING,
reviews_date DATETIME,
reviews_rating FLOAT64,
reviews_text STRING,
reviews_title STRING,
reviews_username STRING,
reviews_users_gender STRING,
shipping STRING
);
```

Query results

Query complete (0.2 sec elapsed, 0 B processed)

Job information | Results

This statement created a new table named data228-final-project-312006:term_project.amazon_data_raw.

amazon_data_raw

Schema | Details | Preview

Field name	Type	Mode	Policy tags	Description
id	STRING	NULLABLE		
dateAdded	DATETIME	NULLABLE		
dateUpdated	DATETIME	NULLABLE		
name	STRING	NULLABLE		
asins	STRING	NULLABLE		
brand	STRING	NULLABLE		
categories	STRING	NULLABLE		
primaryCategories	STRING	NULLABLE		
reviews_date	DATETIME	NULLABLE		
reviews_rating	FLOAT	NULLABLE		
reviews_text	STRING	NULLABLE		
reviews_title	STRING	NULLABLE		
reviews_username	STRING	NULLABLE		
reviews_users_gender	STRING	NULLABLE		
shipping	STRING	NULLABLE		

Edit schema

amazon_data_raw

Schema | Details | Preview

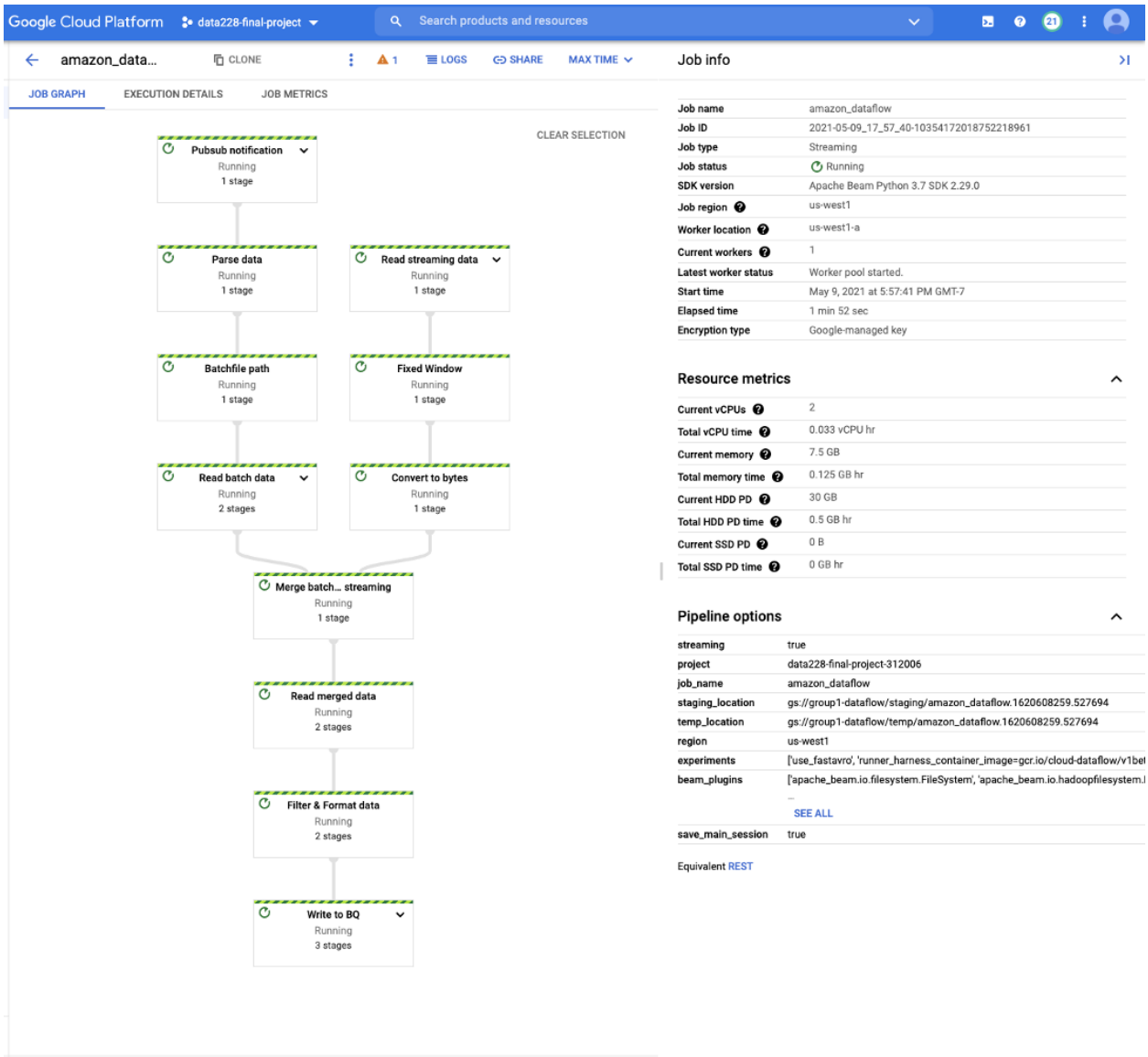
Description

None

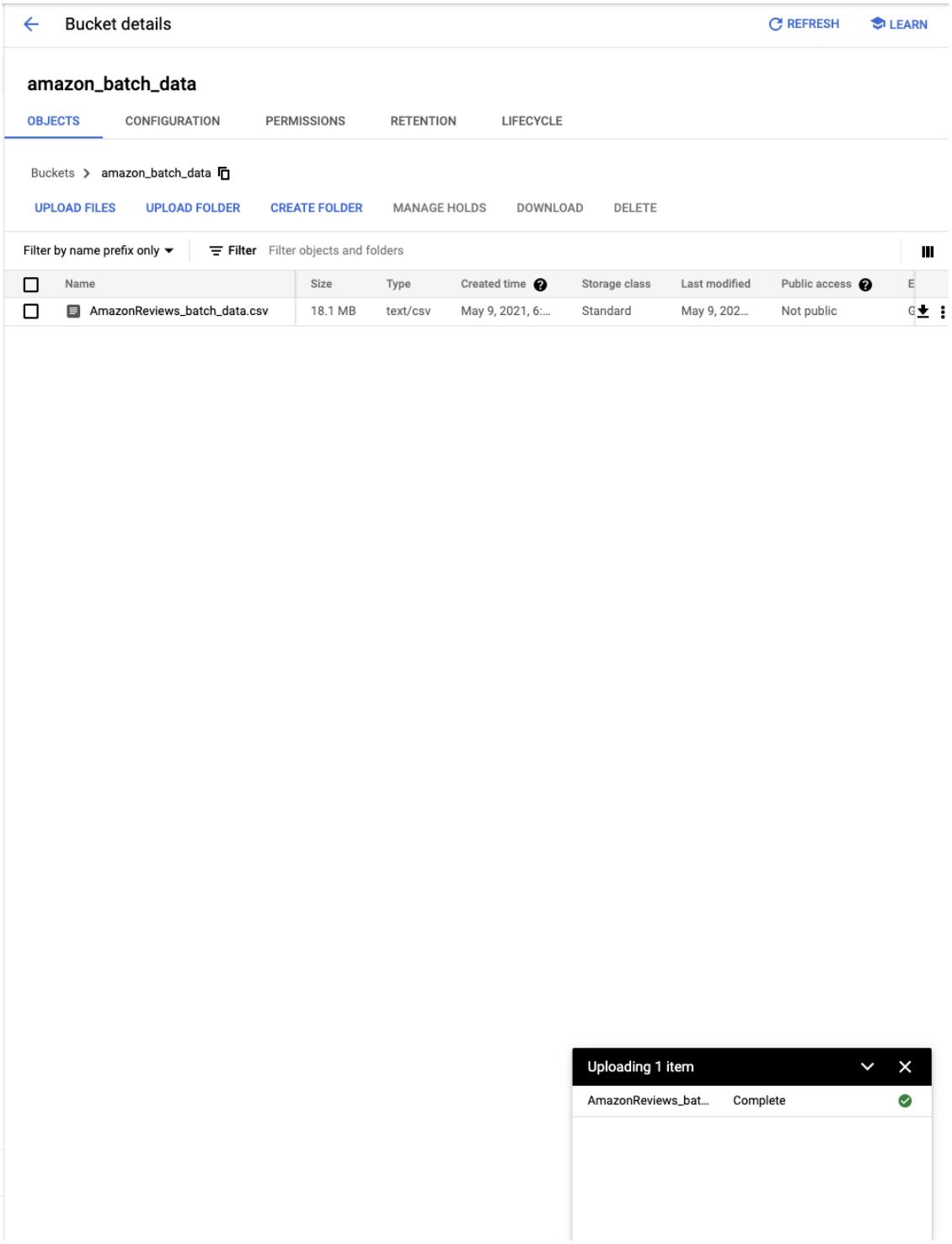
Table info

Table ID	data228-final-project-312006:term_project.amazon_data_raw
Table size	0 B
Number of rows	0
Created	May 9, 2021, 5:04:19 PM
Table expiration	Never
Last modified	May 9, 2021, 5:04:19 PM
Data location	US

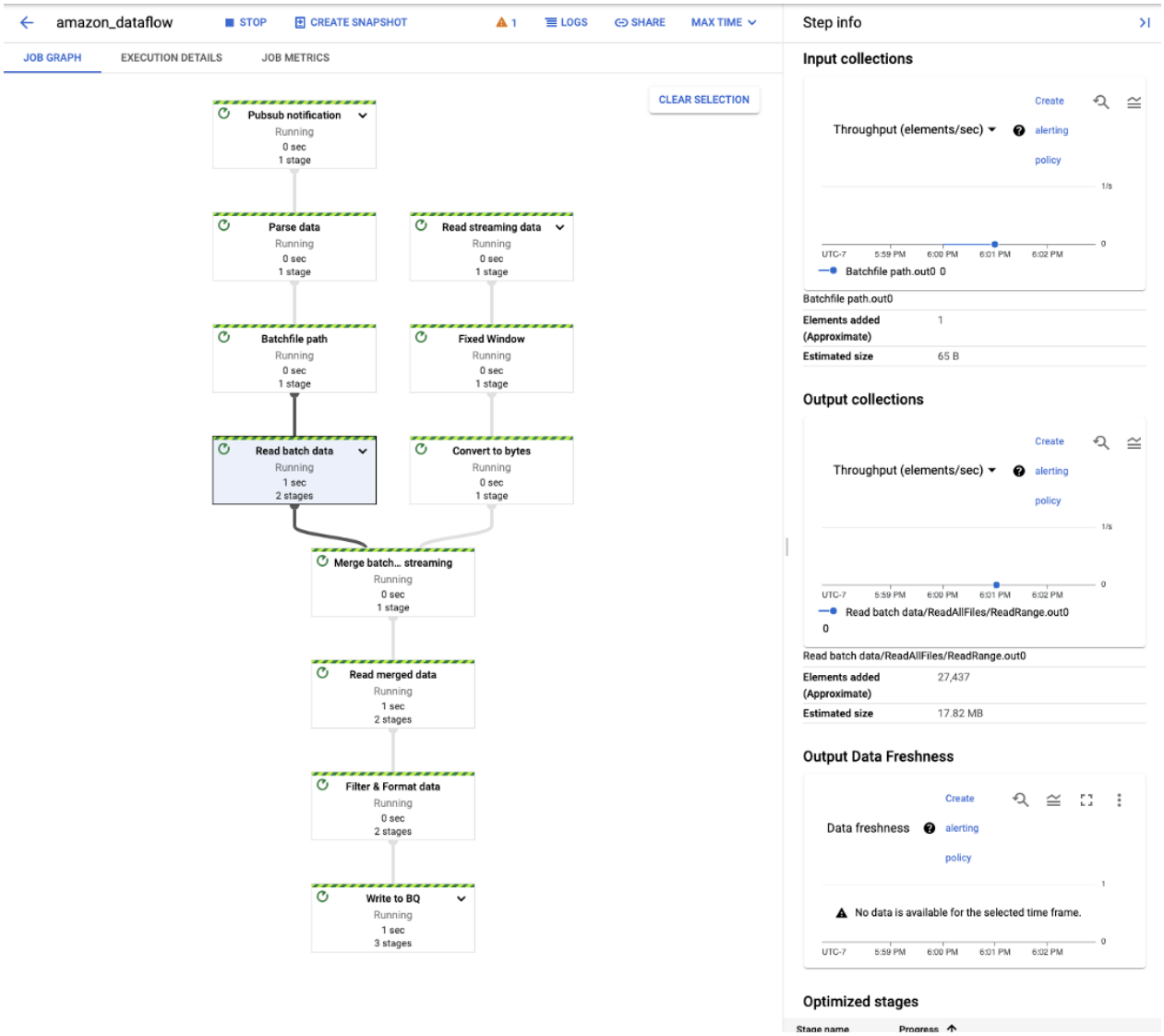
For batch data, we have used Dataflow (Apache Beam Python) for the ETL processing. Dataflow subscribes to the topic that publishes the metadata of the batch file. From the metadata, the bucket and file name details are extracted and the file path is constructed and accessed. The screenshot below shows the dataflow job started and waiting for incoming data:



Then batch file is uploaded into **amazon_batch_data** bucket in Google cloud storage as shown below:



Once the file is uploaded successfully, pub/sub notification that was created earlier gets triggered and sends the metadata of the file which is uploaded into the specified bucket to dataflow that listens to the message with the file path and reads the records from the file as shown below:



Now dataflow writes all the filtered and formatted batch data to Bigquery table, we can see the details of the table which shows an estimated count of rows in buffer statistics as shown below:

amazon_data_raw

Schema Details Preview

Description

None

Table info

Table ID	data228-final-project-312006:term_project.amazon_data_raw
Table size	0 B
Number of rows	0
Created	May 9, 2021, 5:04:19 PM
Table expiration	Never
Last modified	May 9, 2021, 6:03:16 PM
Data location	US

Streaming buffer statistics

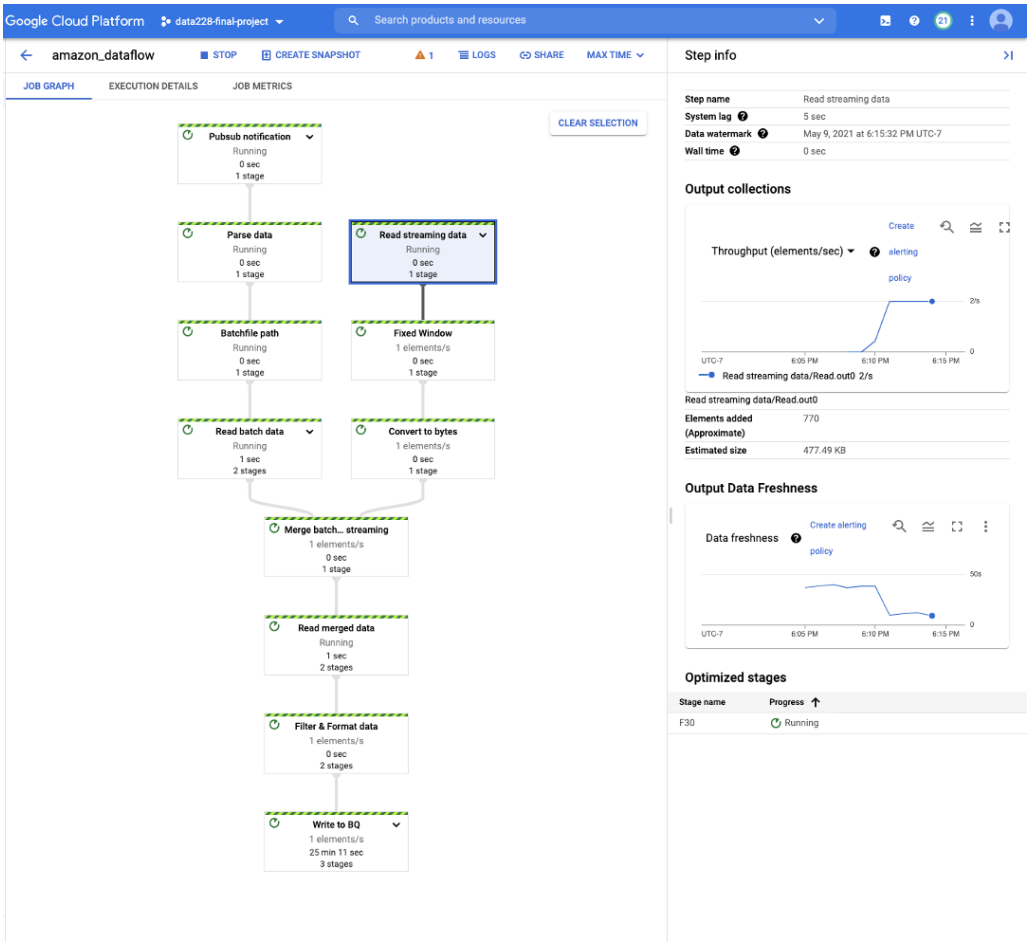
Estimated size	15.21 MB
Estimated rows	27,437
Earliest entry time	May 9, 2021, 6:03:00 PM

For streaming data, we have used the same dataflow for processing and this subscribes to a topic that is created specifically for streaming data with a fixed time window of an hour. Below is the screenshot of publishing streaming messages to dataflow through python code (first few records and last few records).

```
(venv) aa@amazon-streamingdata-vm:~/term_project$ python publish.py
Publishing file object 1 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:02.308909...
Publishing file object 2 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:02.810021...
Publishing file object 3 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:03.311092...
Publishing file object 4 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:03.812221...
Publishing file object 5 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:04.313286...
Publishing file object 6 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:04.814345...
Publishing file object 7 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:05.315385...
Publishing file object 8 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:05.816466...
Publishing file object 9 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:06.317532...
Publishing file object 10 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:06.818632...
Publishing file object 11 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:07.319705...
Publishing file object 12 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:07.820811...
Publishing file object 13 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:08.321824...
Publishing file object 14 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:08.822985...
Publishing file object 15 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:09.324061...
Publishing file object 16 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:09.825212...
Publishing file object 17 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:10.326296...
Publishing file object 18 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:10.827396...
Publishing file object 19 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:11.328491...
Publishing file object 20 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:09:11.829634...
```

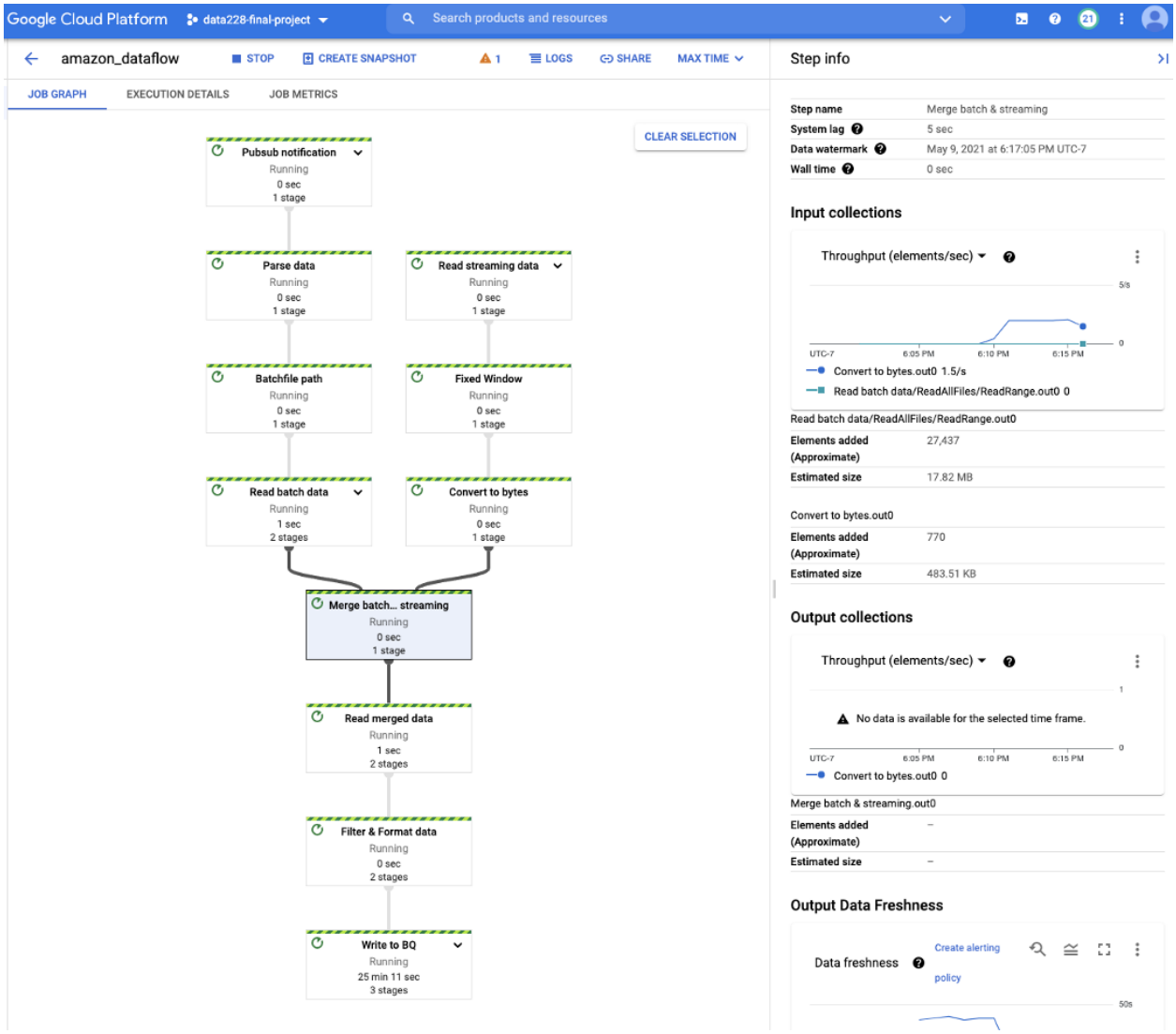
```
Publishing file object 750 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:17.642664...
Publishing file object 751 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:18.143752...
Publishing file object 752 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:18.644827...
Publishing file object 753 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:19.146016...
Publishing file object 754 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:19.647145...
Publishing file object 755 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:20.148226...
Publishing file object 756 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:20.649392...
Publishing file object 757 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:21.150478...
Publishing file object 758 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:21.651546...
Publishing file object 759 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:22.152665...
Publishing file object 760 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:22.653761...
Publishing file object 761 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:23.154764...
Publishing file object 762 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:23.655935...
Publishing file object 763 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:24.157138...
Publishing file object 764 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:24.658321...
Publishing file object 765 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:25.159432...
Publishing file object 766 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:25.660565...
Publishing file object 767 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:26.161734...
Publishing file object 768 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:26.662835...
Publishing file object 769 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:27.163923...
Publishing file object 770 to projects/data228-final-project-312006/topics/streaming_data_in at 2021-05-10 01:15:27.665023...
(venv) aa@amazon-streamingdata-vm:~/term_project$
```

We can also see that dataflow started receiving streaming messages from topic "streaming_data_in" as shown below:



The screenshot below shows merged data step info, as proof that all the data ingested is processed successfully in the dataflow.

Elements added: 27,437 (batch count) and 770 (streaming count)



The screenshot below shows that dataflow processed data (batch: 27,437 & streaming: 770) is loaded into the Bigquery table:

amazon_data_raw

Schema **Details** Preview

Description

None

Table info

Table ID	data228-final-project-312006:term_project.amazon_data_raw
Table size	0 B
Number of rows	0
Created	May 9, 2021, 5:04:19 PM
Table expiration	Never
Last modified	May 9, 2021, 6:03:16 PM
Data location	US

Streaming buffer statistics

Estimated size	15.59 MB
Estimated rows	28,207
Earliest entry time	May 9, 2021, 6:03:00 PM

The details of the table after all the data processed into the Bigquery table:

amazon_data_raw

SCHEMA DETAILS PREVIEW

Table info

Table ID	data228-final-project-312006:term_project.amazon_data_raw
Table size	16.12 MB
Long-term storage size	
Number of rows	28,207
Created	May 9, 2021, 5:04:19 PM UTC-7
Last modified	May 9, 2021, 7:34:18 PM UTC-7
Table expiration	NEVER
Data location	US
Description	

Data Analytics

For data analytics, we have decided to create separate views for different use cases so that data access could be segregated for each end-user such as the Marketing team, Strategy team, Inventory team, etc.

- **Most_Reviewed_products:**
  This view captures products with the top number of reviews

  SQL Query:

```
Create or replace view term_project.Most_Reviewed_products
as
SELECT
name AS Name_of_the_Product,
COUNT(reviews_title) AS Count_of_Reviews
FROM term_project.amazon_data_raw
GROUP BY name
ORDER BY COUNT(reviews_title) DESC;
```

Output:

Schema Details

Field name	Type	Mode	Policy tags	Description
Name_of_the_Product	STRING	NULLABLE		
Count_of_Reviews	INTEGER	NULLABLE		

Edit schema

Displaying most reviewed products view output:

```
select * from term_project.Most_Reviewed_products
```

Run | Save query | Save view | Schedule query | More | This query will process 2.6 MB when run.

Query results | SAVE RESULTS | EXPLORE DATA

Query complete (0.3 sec elapsed, 2.6 MB processed)

Job information | Results | JSON | Execution details

Row	Name_of_the_Product	Count_of_Reviews
1	AmazonBasics AAA Performance Alkaline Batteries (36 Count)	8343
2	AmazonBasics AA Performance Alkaline Batteries (48 Count) - Packaging May Vary	3728
3	Fire HD 8 Tablet with Alexa, 8 HD Display, 16 GB, Tangerine - with Special Offers	2443
4	All-New Fire HD 8 Tablet, 8 HD Display, Wi-Fi, 16 GB - Includes Special Offers, Black	2370
5	Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16 GB, Pink Kid-Proof Case	1676
6	Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16 GB, Blue Kid-Proof Case	1425

Rows per page: 100 | 1 - 60 of 60 | First page | Last pag

❑ Products_with_FiveRatings

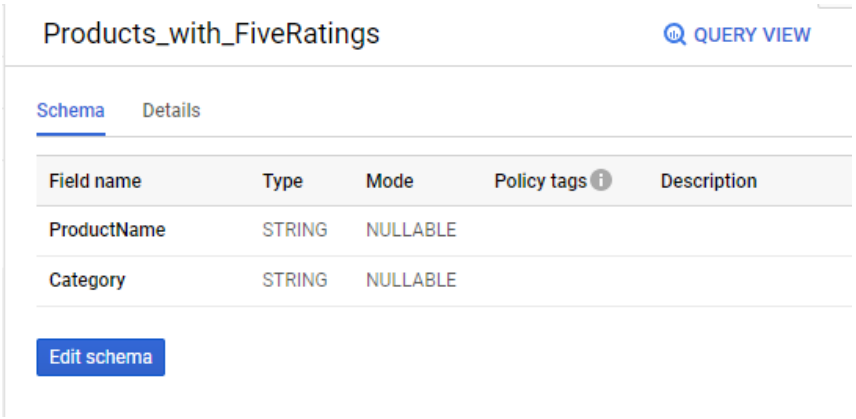
This view contains information about the 5 star rated products and their categories

SQL Query:

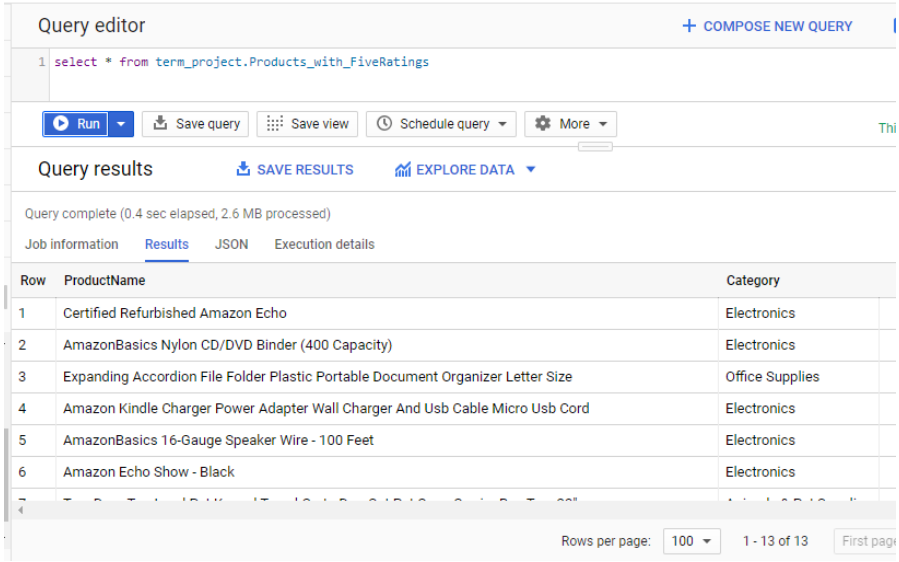
```
Create or replace view term_project.Products_with_FiveRatings
as
SELECT ProductName, Category
FROM (
```

```
SELECT
name AS ProductName,
primaryCategories AS Category,
AVG(reviews_rating) AS Rating
FROM term_project.amazon_data_raw
GROUP BY name, primaryCategories
HAVING AVG(reviews_rating) = 5.0
ORDER BY AVG(reviews_rating) DESC
);
```

Output:



Displaying the list of products by category with five ratings view output:



❑ freeShipping_productCategories

This view provides information about the popularity of each free shipping product and its category

SQL Query:

```
create or replace view
term_project.freeShipping_productCategories as
SELECT name, primaryCategories,
count(name) as FreeShippingProductCount
FROM term_project.amazon_data_raw
WHERE shipping = 'Free'
group by name, primaryCategories;
```

Output:

freeShipping_productCategories QUERY VIEW

Schema Details

Field name	Type	Mode	Policy tags	Description
name	STRING	NULLABLE		
primaryCategories	STRING	NULLABLE		
FreeShippingProduct Count	INTEGER	NULLABLE		

Edit schema

Displaying popularity of the products with free shipping by category view output:

```
1  select * from term_project.freeShipping_productCategories
```

Query results SAVE RESULTS EXPLORE DATA

Query complete (0.4 sec elapsed, 2.6 MB processed)

Job information Results JSON Execution details

Row	name	primaryCategories	FreeShippingProductCount
1	Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16 GB, Blue Kid-Proof Case	Electronics	725
2	Fire Kids Edition Tablet, 7 Display, Wi-Fi, 16 GB, Pink Kid-Proof Case	Toys & Games,Electronics	847
3	AmazonBasics AAA Performance Alkaline Batteries (36 Count)	Health & Beauty	4203

❑ Popularcategory_gender

This view is for analyzing the trends or popularity of a specific category based on gender

SQL Query:

```
Create or replace view term_project.popularcategory_gender as
SELECT
primaryCategories,
reviews_users_gender,
count(*) as Count
FROM
`data228-final-project-312006.term_project.amazon_data_raw`
GROUP BY primaryCategories,reviews_users_gender
ORDER BY primaryCategories
```

Output:

popularcategory_gender

Schema Details

Field name	Type	Mode	Policy tags	Description
primaryCategories	STRING	NULLABLE		
reviews_users_gender	STRING	NULLABLE		
Count	INTEGER	NULLABLE		

Displaying the number of reviews written by each gender under different categories view output:

```
1   select * from term_project.popularcategory_gender
```

Query results SAVE RESULTS EXPLORE DATA

Query complete (0.5 sec elapsed, 626.7 KB processed)

Job information Results JSON Execution details

Row	primaryCategories	reviews_users_gender	Count
1	Animals & Pet Supplies	Male	3
2	Animals & Pet Supplies	Female	3
3	Electronics	Male	7045
4	Electronics	Female	6843
5	Electronics,Furniture	Female	1
6	Electronics,Furniture	Male	1
7	Electronics,Media	Male	92
8	Electronics,Media	Female	93
9	Health & Beauty	Male	6072
10	Health & Beauty	Female	5999

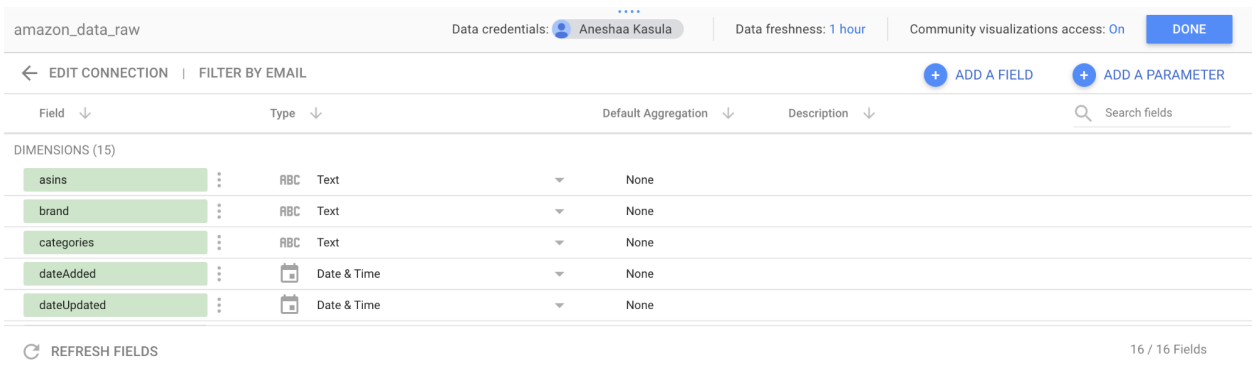
Data Exploration/Visualization

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data. Data visualization also makes it easier for the human brain to understand and pull insights from. In the world of Big Data, data visualization tools and technologies are essential to analyze massive amounts of information and make data-driven decisions.

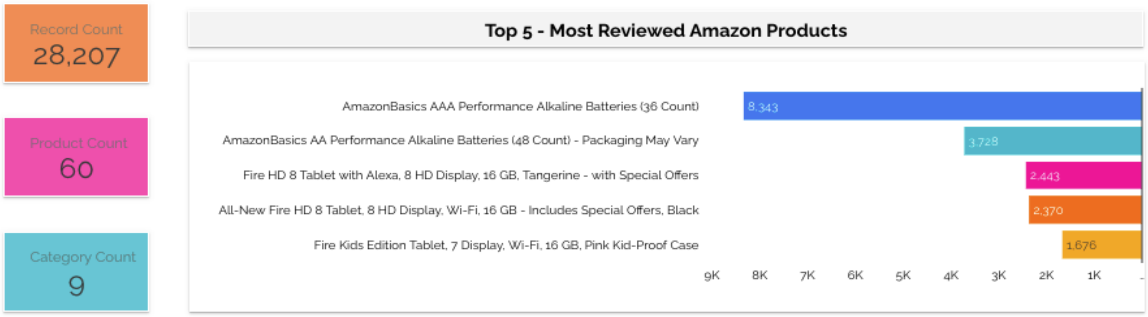
We used the data studio to visualize the amazon dataset which is stored in BigQuery. As there are multiple plugins built into data studio to import data from various sources, that includes BigQuery. We also used the community visualization tool “Simple Word Cloud” to

display word clouds for overall review text. It is the graphical representation of the most frequent words in a text. The higher the font size higher the frequency of that particular word has been used in Reviews.

We created a dashboard for the objectives by importing views we created earlier. We have set a 1-hour refresh cycle for the dashboard source data. So that, after 1 hour it will automatically refresh the data and update the dashboard accordingly as shown below:



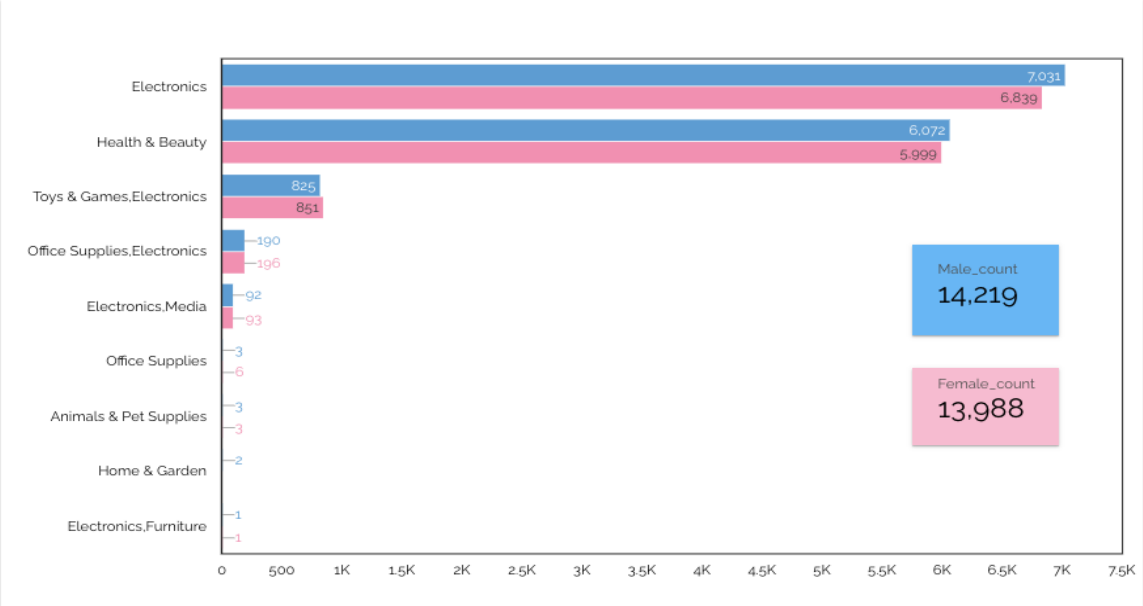
The screenshots below are each section of the dashboard as per our objectives:



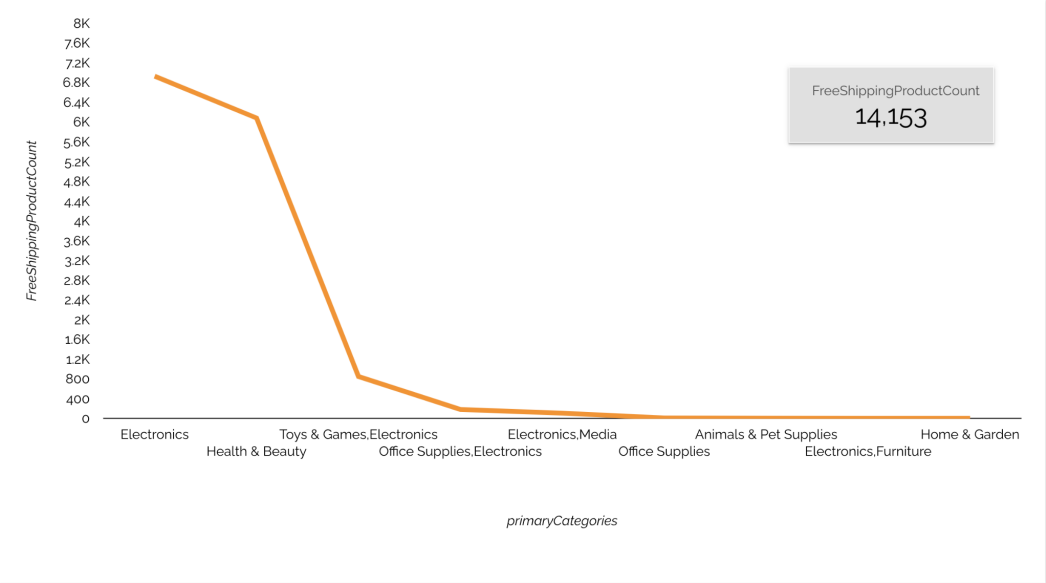
5 Ratings - Products with Categories

	ProductName	Category
1.	Two Door Top Load Pet Kennel Travel Crate Dog Cat Pet Cage Carrier Box Tray 23"	Animals & Pet Supplies
2.	Fire TV Stick Streaming Media Player Pair Kit	Electronics
3.	Expanding Accordion File Folder Plastic Portable Document Organizer Letter Size	Office Supplies
4.	Certified Refurbished Amazon Echo	Electronics
5.	Cat Litter Box Covered Tray Kitten Extra Large Enclosed Hooded Hidden Toilet	Animals & Pet Supplies
6.	AmazonBasics Single-Door Folding Metal Dog Crate - Large (42x28x30 Inches)	Animals & Pet Supplies
7.	AmazonBasics Nylon CD/DVD Binder (400 Capacity)	Electronics
8.	AmazonBasics Nespresso Pod Storage Drawer - 50 Capsule Capacity	Home & Garden
9.	AmazonBasics 16-Gauge Speaker Wire - 100 Feet	Electronics
10.	Amazon Kindle Charger Power Adapter Wall Charger And Usb Cable Micro Usb Cord	Electronics
11.	Amazon Echo ÄövÑv" White	Electronics
12.	Amazon Echo Show - Black	Electronics
13.	All-new Echo (2nd Generation) with improved sound, powered by Dolby, and a new design Walnut Finish	Electronics,Furniture

Popular Category based on Gender



Popularity of free shipping products based on Category



Most commonly used Words in Reviews given users

Summary & Conclusion

We have built a complete data analytics pipeline for consumer reviews of Amazon products. From this pipeline, buyers can analyze their product sales, and also they will be able to make informed decisions about their product. The dashboard we created can be useful for Amazon Internal teams such as marketing, inventory team, etc to strategize their future investments.

Future Work

- Based on fixed windows, get the trending products and trending countries.
- The word cloud visualization will be implemented in BQ rather than using 3rd party visualization tools directly.
- Review sentiment analysis using BQ and ML modules.
- Change the streaming ingestion from python program to API's.