Northeastern University

Trending YouTube Video Analysis

Group 6

Group Members:

Ashok Thiruvengadam

Megha Batra

Vidyun Akila Sundhara Raaman

Snehal Rajwar

DAMG 7290: Data Warehousing and Business Intelligence Spring 2022, Northeastern University

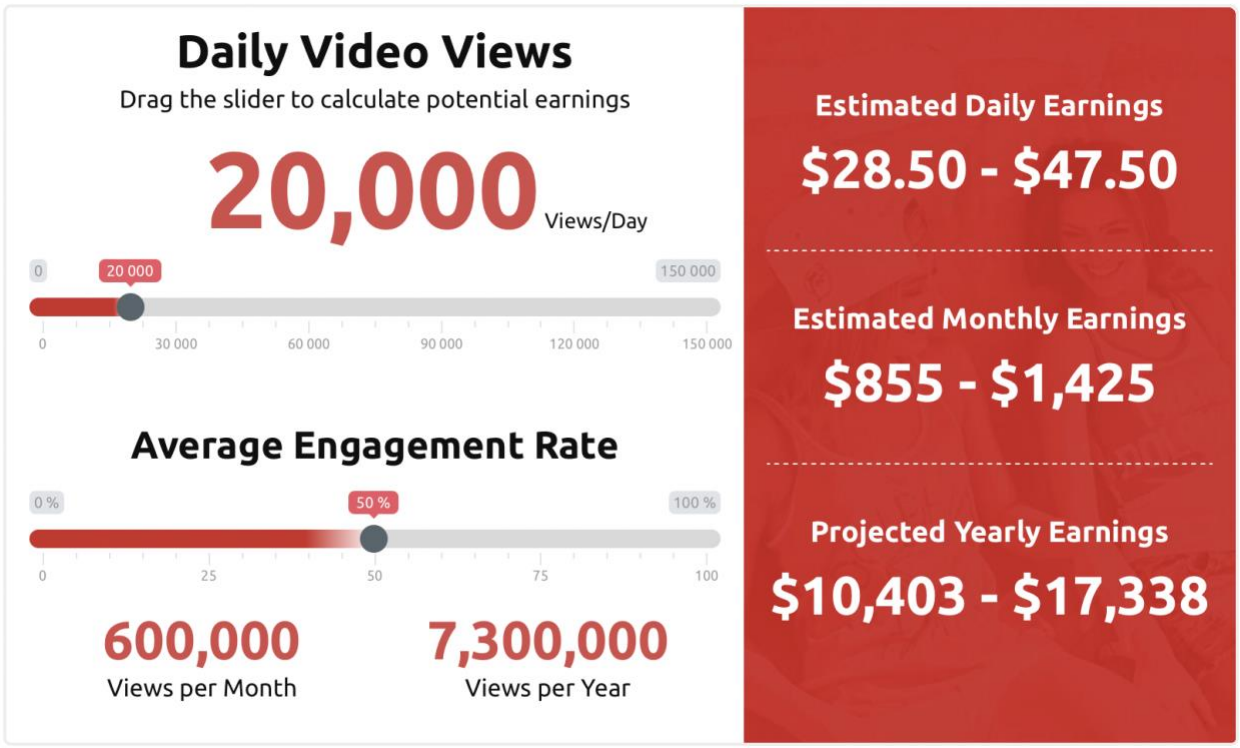
Instructor:

Prof. Vincent Lattuada

Table of Contents

Introduction:

The Project is on YouTube Trending Videos. YouTube is an American Online video sharing and social media platform launched in 2005, owned by Google. YouTube is the top second most-visited website after Google with 34.6 billion visits each month, followed by Meta. YouTube has 1B monthly users and as of 2019 videos were being uploaded at a rate of more than 500 hours of content per minute. YouTube reported revenue for 2020 was $19.8 Billion. YouTube and approved creators participate in the google AdSence program which seeks to generate more revenue for both parties. There are numerous creators who have made their fortune online through YouTube.



Business Problem:

Consider we have a client or company who wants to run some ad campaign online and they selected their main advertising channel as YouTube they want to understand some of the initial questions, they have such as:

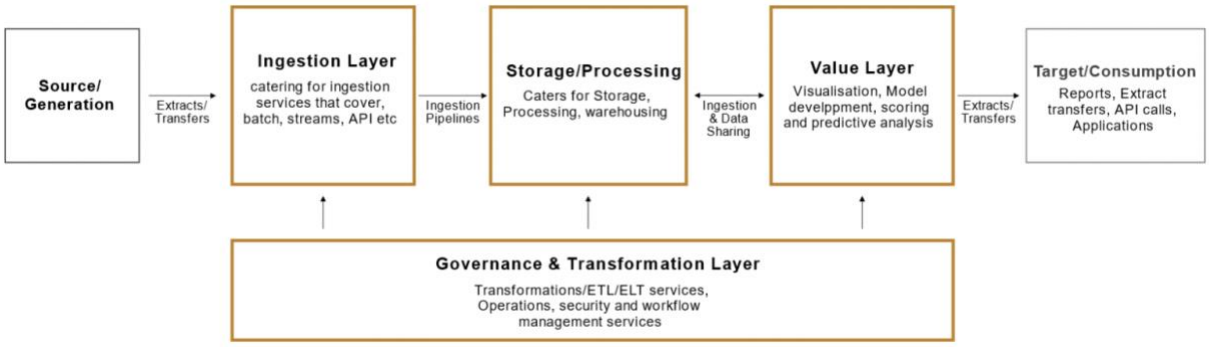
1. How to categorize videos based on their comments and statistics
2. What factors influence how popular a YouTube video will be.

These are some of the questions that a client might have before investing money in a YouTube advertising campaign.

Project Overview:

Goals and Success Criteria to help the customers how they might succeed in YouTube Ad Campaign using our analysis.

1. Data Ingestion: Ingest data, one-offs, and incrementally
2. Data Lake: Design and build a new Data Lake architecture
3. AWS Cloud: AWS as the cloud provider
4. ETL Design: Extract, Transform, and load data efficiently
5. Scalability: The data architecture should scale efficiently
6. Reporting: Build a Business Intelligence, Dashboard



Dataset Introduction:

Our Dataset from YouTube: https://www.kaggle.com/datasets/datasnaek/youtube-new

Context

YouTube (the world-famous video-sharing website) maintains a list of the top trending videos on the platform. According to Variety magazine, “To determine the year’s top-trending videos, YouTube uses a combination of factors including measuring users’ interactions (number of views, shares, comments, and likes). Note that they’re not the most-viewed videos overall for the calendar year”. Top performers on the YouTube trending list are music videos (such as the famously virile “Gangam Style”), celebrity and/or reality TV performances, and the random dude-with-a-camera viral videos that YouTube is well-known for.

This dataset is a daily record of the top trending YouTube videos.

Note that this dataset is a structurally improved version of this dataset.

Content

This dataset includes several months (and counting) of data on daily trending YouTube videos. Data is included for the US, GB, DE, CA, FR, RU, MX, KR, JP, and IN regions (USA, Great Britain, Germany, Canada, France, Russia, Mexico, South Korea, Japan, and India respectively), with up to 200 listed trending videos per day.

Each region's data is in a separate file. Data includes the video title, channel title, publish time, tags, views, likes, and dislikes, description, and comment count.

The data also includes a category_id field, which varies between regions. To retrieve the categories for a specific video, find it in the associated JSON. One such file is included for each of the 10 regions in the dataset.

Data Dictionary:

Our dataset contains 20 files, 10 csv & 10 json files. It has 160 columns (String: 60, Integer: 40, Boolean: 30, other: 30)

.CSV Files: We have 10 CSV files with us.

Column Name	Data Type	Description
video_id	string	Id of the video
trending_date	string	The when the video trended
title	string	Title of the video
channel_title	string	Name of the YouTube channel
category_id	bigint	Id of the video's category
publish_time	string	Time when the video was published
tags	string	The hastags of the video
views	bigint	Number of times the video was watched
likes	bigint	Number of times the video was liked
dislikes	bigint	Number of times the video was disliked
comment_count	bigint	Number of comments posted in the video
thumbnail_link	string	The link to the thumbnail of the video
comments_disable	boolean	To disable the comments in the video
rating_disable	boolean	To disable the comments in the video
video_error_or_removed	boolean	To notify whether the video has error or when removed
description	string	Description of the video
region	string	Region where the video was streamed

.Json Files

Column Name	Data Type	Description
Kind	string	The video category
Etag	String	
Id	array	

Data Flow:

Below is the overview of our data flow:

Software Requirements:

We implemented our entire project on Amazon Web Services.

Services used are IAM, S3, Glue Crawler, Lambda, Athena, Glue Data Studio, and Quick Sight.

Since we had certain limitations with free versions of Quick sight we implemented the Business Intelligence part with QlikView as well.

IAM: AWS Identity and Access Management (IAM) provides fine-grained access control across all of AWS. With IAM, you can specify who can access which services and resources, and under which conditions. With IAM policies, you manage permissions to your workforce and systems to ensure the least privilege permissions.

S3 Bucket: An Amazon S3 bucket is a public cloud storage resource available in Amazon Web Services (AWS) Simple Storage Service (S3), an object storage offering. Amazon S3 buckets, which are like file folders, store objects, which consist of data and its descriptive metadata.

Glue Crawler: AWS Glue crawler is used to connect to a data store, progresses done through a priority list of the classifiers used to extract the schema of the data and other statistics, and in turn populate the Glue Data Catalog with the help of the metadata.

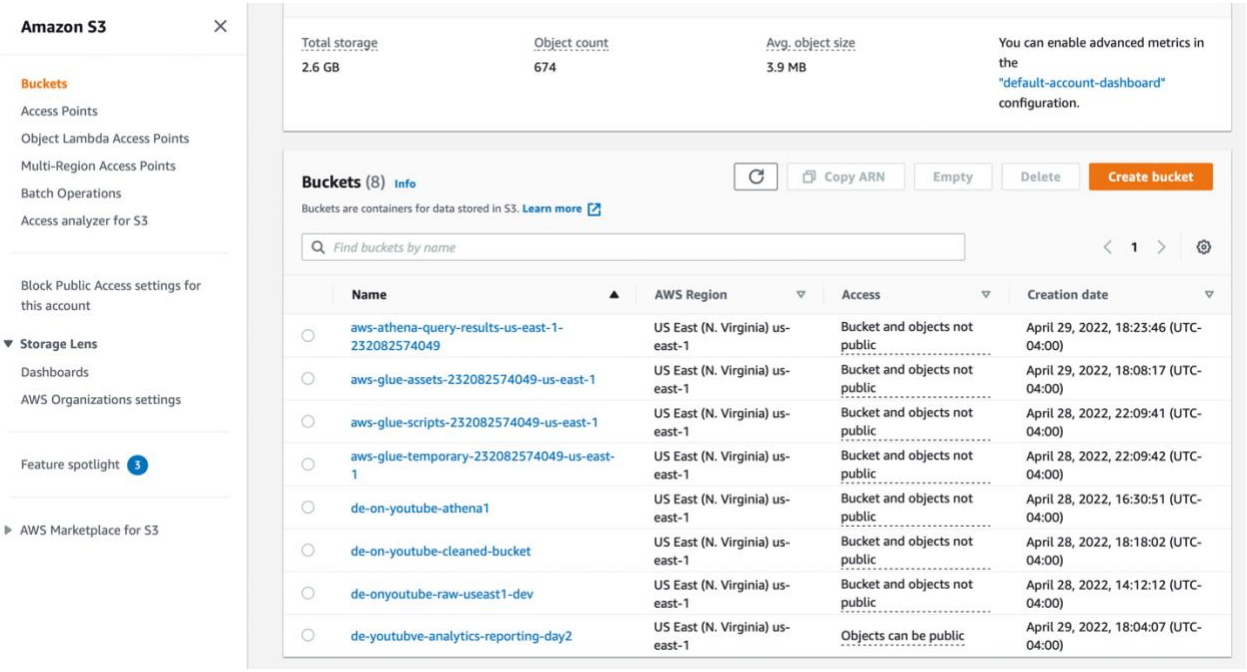
Lambda: Lambda is a compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, code monitoring, and logging.

Glue Data Studio: AWS Glue Studio is a new graphical interface that makes it easy to create, run, and monitor extract, transform, and load (ETL) jobs in AWS Glue. You can visually compose data transformation workflows and seamlessly run them on AWS Glue's Apache Spark-based serverless ETL engine. You can inspect the schema and data results in each step of the job.

Quick Sight: Amazon QuickSight is a cloud-scale business intelligence (BI) service that you can use to deliver easy-to-understand insights to the people whom you work with, wherever they are. Amazon QuickSight connects to your data in the cloud and combines data from many different sources.

Steps to implement the project

Step-1: Creating an S3 Bucket(data loading) : We will create an S3 Bucket (de-onyoutube-raw-useast1-dev) which will act as a staging area. We will load the raw data that is JSON and CSV into the S3 bucket country-wise. We are loading the source data into the AWS S3 bucket 1. This acts as the staging area. We are using the Extract, Loading, and Transformation process. S3 Bucket-1: Acts as a staging area where all the raw incoming data (CSV&Json) is being loaded using the Command Line Interface.



Step-2. Building Glue Crawler and Catalogue: -

Added the first crawler (de-utube-1-dev) in AWS Glue for that we needed to create an IAM role with permission to access the S3 bucket and AWSGlueServiceRole. We created de-on-youtube-glue-s3-role IAM role. Also created a database that is de-on-youtube-db1.

#Crawler -1 de-utube-1-dev

Crawlers > de-utube-1-dev

Run crawler | Edit

Name	de-utube-1-dev
Description	
Create a single schema for each S3 path	false
Security configuration	
Tags	-
State	Ready
Schedule	
Last updated	Thu Apr 28 16:27:18 GMT-400 2022
Date created	Thu Apr 28 16:27:18 GMT-400 2022
Database	de-on-youtube-db1
Table level	
Service role	de-on-youtube-glue-s3-role
Selected classifiers	
Data store	S3
Include path	s3://de-onyoutube-raw-useast1-dev/youtube/raw_statistics_reference_data
Connection	
Exclude patterns	

#1st IAM role

de-on-youtube-glue-s3-role

Allows Glue to call AWS services on your behalf.

Summary

Creation date
April 28, 2022, 15:36 (UTC-04:00)

ARN
arn:aws:iam::232082574049:role/de-on-youtube-glue-s3-role

Last activity
13 hours ago

Maximum session duration
1 hour

Permissions | Trust relationships | Tags | Access Advisor | Revoke sessions

Permissions policies (2)
You can attach up to 10 managed policies.

Simulate | Remove | Add per

Filter policies by property or policy name and press enter

Policy name	Type	Description
AmazonS3FullAccess	AWS managed	Provides full acces
AWSGlueServiceRole	AWS managed	Policy for AWS Glu

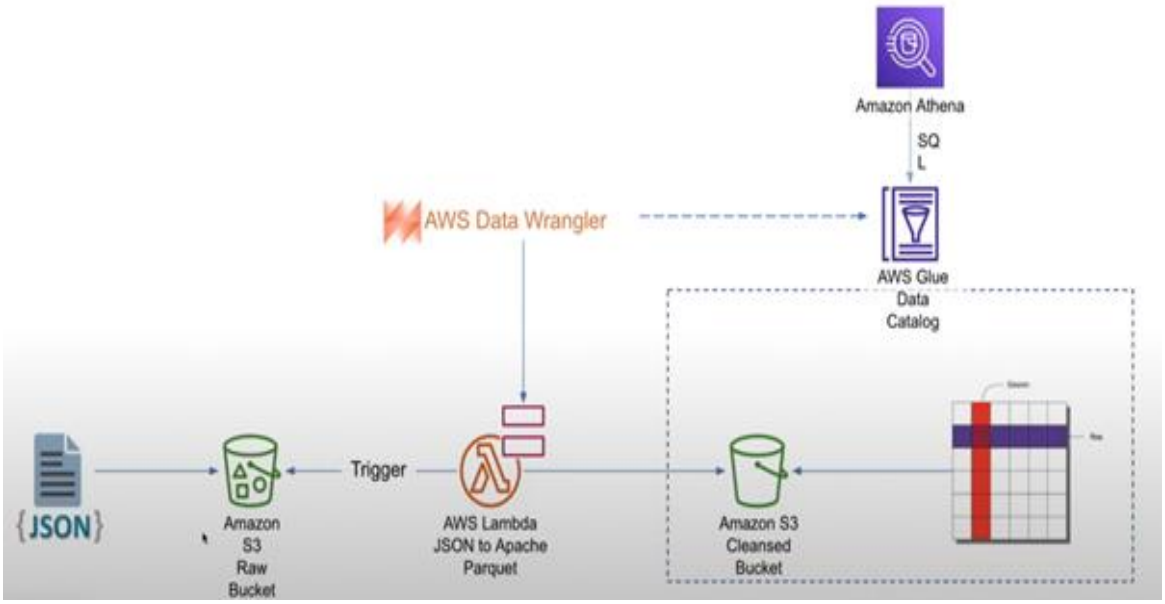
#Catalogue with the database name as de-on-youtube-db1 schema for the JSON data

Name	raw_statistics_reference_data
Description	
Database	de-on-youtube-db1
Classification	json
Location	s3://de-onyoutube-raw-useast1-dev/youtube/raw_statistics_reference_data/
Connection	
Deprecated	No
Last updated	Thu Apr 28 16:28:17 GMT-400 2022
Input format	org.apache.hadoop.mapred.TextInputFormat
Output format	org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib	org.openx.data.jsonserde.JsonSerDe
Serde parameters	paths **etag,items,kind**
Table properties	sizeKey **81579** objectCount **10** UPDATED_BY_CRAWLER **de-utube-1-dev**
	CrawlerSchemaSerializerVersion **1.0** recordCount **10** averageRecordSize **8156**
	CrawlerSchemaDeserializerVersion **1.0** compressionType **none** typeOfData **file**

Schema

	Column name	Data type	Partition key	Comment
1	kind	string		
2	etag	string		
3	items	array		

Step3- To query the tables in AWS Athena we need to transform the file from JSON to Parquet. These transformations can be done in AWS Lambda. For that we need to select the IAM role and add the database. Then we create a Lambda function for transforming a .Json file into parquet format as the Json data is not structured. Create a Python script in Lambda to transform the data.



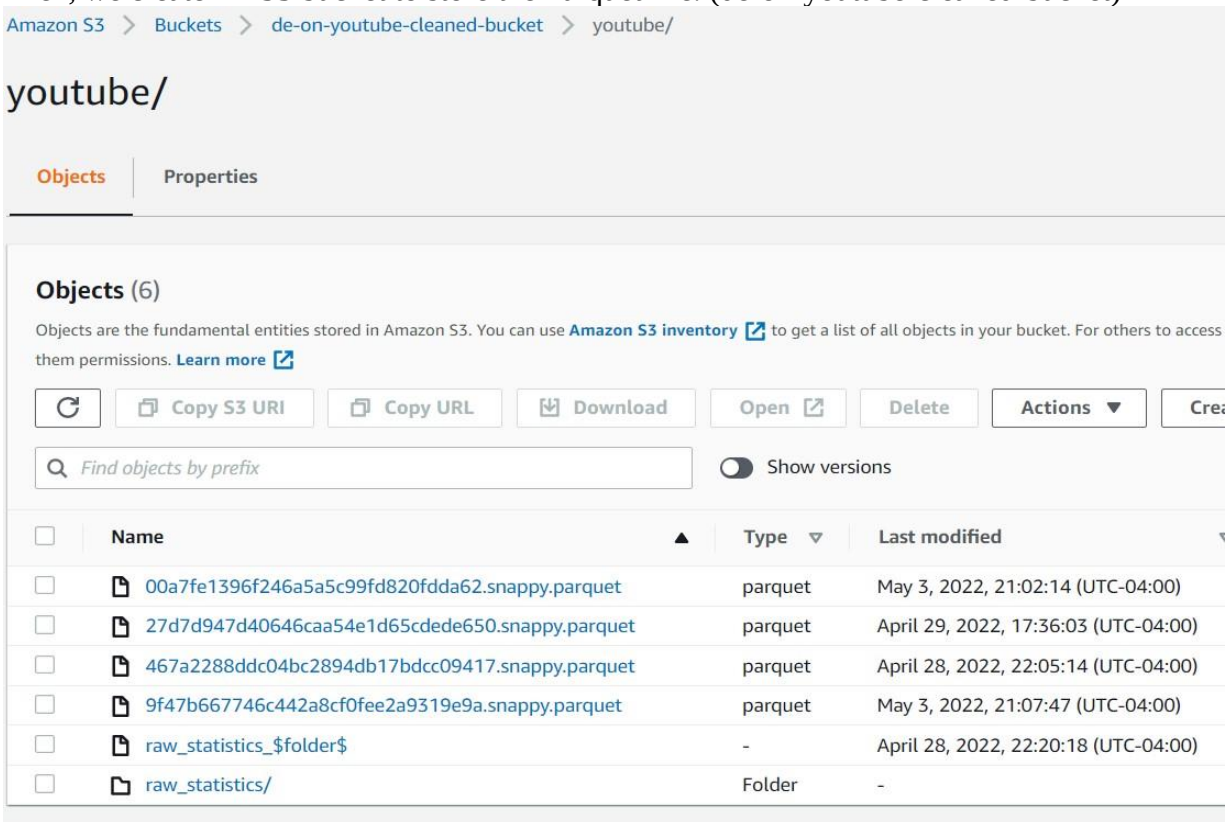
#Transformation in Lambda to convert JSON to Parquet format for querying in Athena

lambda_function

```python
import awswrangler as wr
import pandas as pd
import urllib.parse
import os

# Temporary hard-coded AWS Settings; i.e. to be set as OS variable in Lambda
os_input_s3_cleansed_layer = os.environ['s3_cleansed_layer']
os_input_glue_catalog_db_name = os.environ['glue_catalog_db_name']
os_input_glue_catalog_table_name = os.environ['glue_catalog_table_name']
os_input_write_data_operation = os.environ['write_data_operation']


def lambda_handler(event, context):
    # Get the object from the event and show its content type
    bucket = event['Records'][0]['s3']['bucket']['name']
    key = urllib.parse.unquote_plus(event['Records'][0]['s3']['object']['key'], encoding='utf-8')
    try:

        # Creating DF from content
        df_raw = wr.s3.read_json('s3://{}/{}'.format(bucket, key))

        # Extract required columns:
        df_step_1 = pd.json_normalize(df_raw['items'])

        # Write to S3
        wr_response = wr.s3.to_parquet(
            df=df_step_1,
            path=os_input_s3_cleansed_layer,
            dataset=True,
            database=os_input_glue_catalog_db_name,
            table=os_input_glue_catalog_table_name,
            mode=os_input_write_data_operation
        )

        return wr_response
    except Exception as e:
        print(e)
```

1:1 Python Sp

Then, we create 2nd S3 bucket to store the Parquet file. (de-on-youtube-cleaned-bucket)



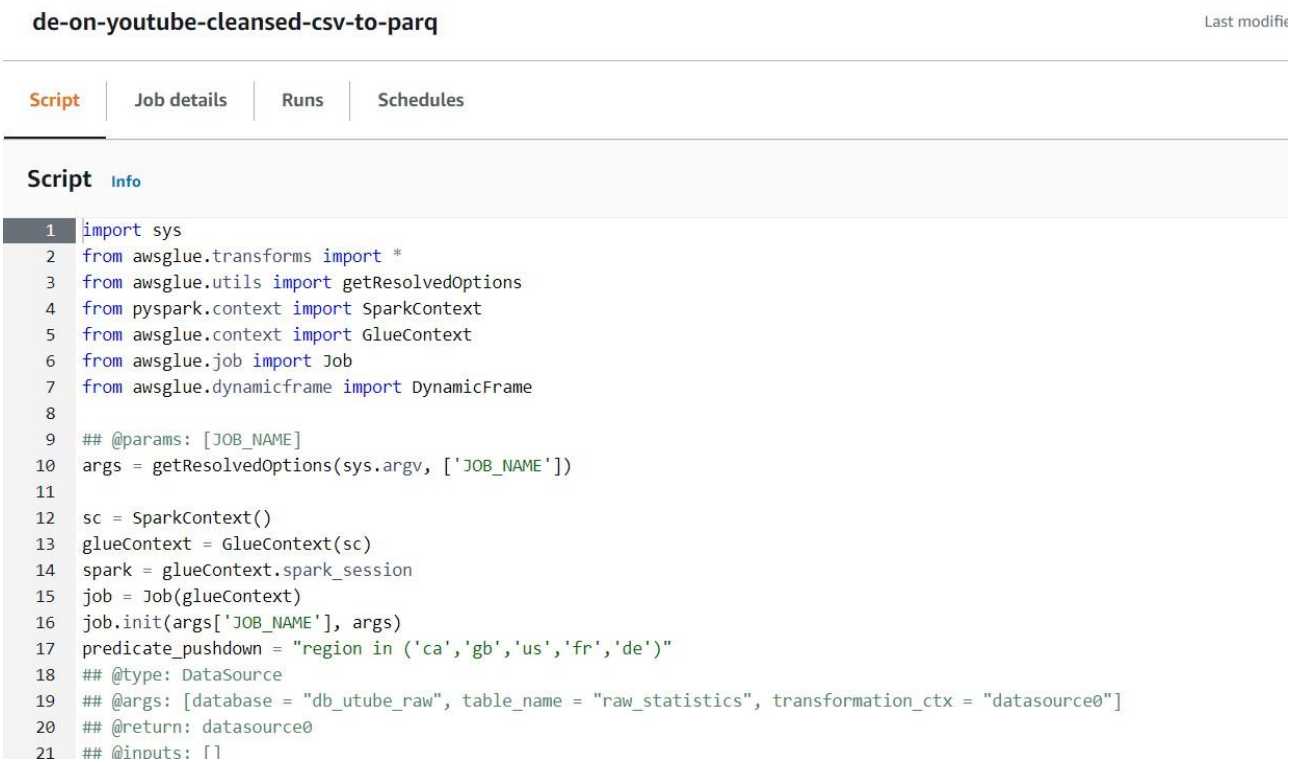
Step4- Converting csv to Parquet.

After JSON files are converted to Parquet, we start with the conversion of CSV files. We need to create new crawler(2nd) . We need to make sure the data types are converted properly before we start the conversion, for example, we had to make sure that long was converted to Big int everywhere

Steps followed for CSV- Parquet: -

1. Opened AWS Glue
2. Added a job
3. Filled the required information about source and destination properly
4. Checked the monitoring options
5. Selected a target path for S3(de-on-youtube-cleaned-bucket)
6. Made sure that data types are consistent in the schema. (Long- BigInt)
7. This job generated a Spark script for converting.
8. Edited the script to add a dynamic frame to make the code more efficient for working with data
9. Added partition key as a region to make sure each region's data is created in a different folder.
10. We added a filter because our data had different languages besides English which are not accepted by the UTF format.

#Job created in Glue



Database-

	Column name	Data type	Partition key	Comment
1	video_id	string		
2	trending_date	string		
3	title	string		
4	channel_title	string		
5	category_id	bigint		
6	publish_time	string		
7	tags	string		
8	views	bigint		
9	likes	bigint		
10	dislikes	bigint		
11	comment_count	bigint		
12	thumbnail_link	string		
13	comments_disabled	boolean		
14	ratings_disabled	boolean		
15	video_error_or_removed	boolean		
16	description	string		
17	region	string	Partition (0)	

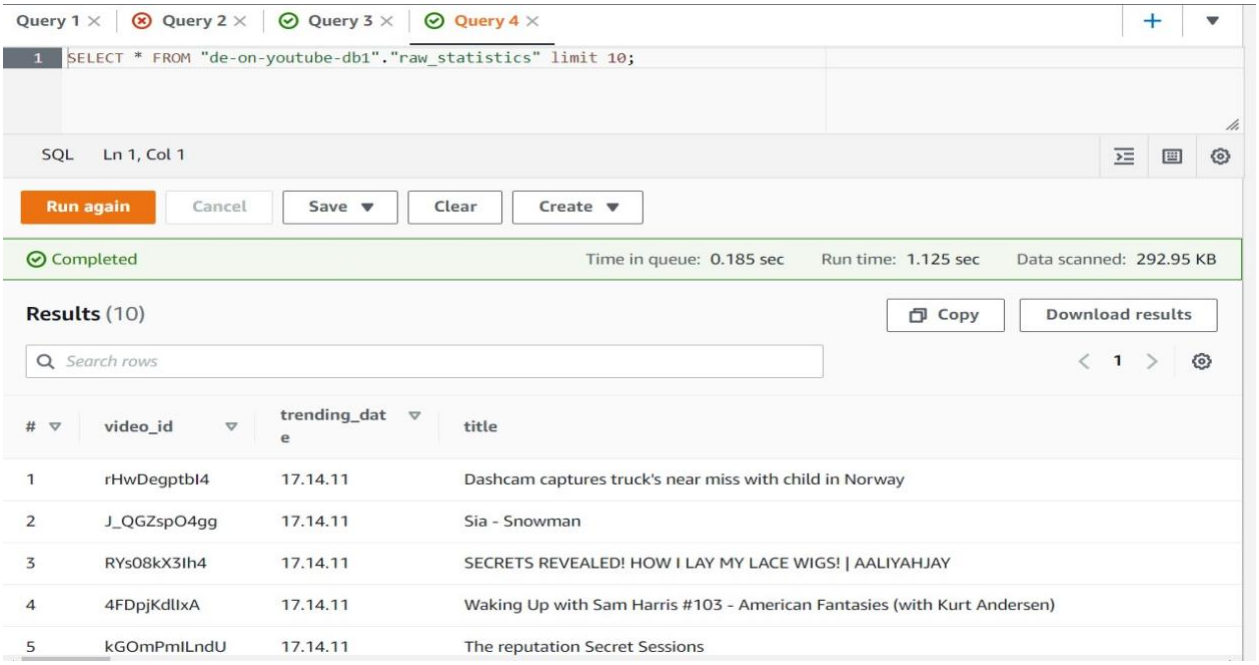
Crawlers > de-on-youtube-crawler-csvtopar2

Run crawler | Edit

Name	de-on-youtube-crawler-csvtopar2
Description	
Create a single schema for each S3 path	false
Security configuration	
Tags	-
State	Ready
Schedule	
Last updated	Thu Apr 28 22:56:34 GMT-400 2022
Date created	Thu Apr 28 22:56:34 GMT-400 2022
Database	de-on-youtube-db1
Table level	
Service role	de-on-youtube-glue-s3-role
Selected classifiers	
Data store	S3
Include path	s3://de-on-youtube-cleaned-bucket/youtube/raw_statistics/
Connection	
Exclude patterns	

Configuration options

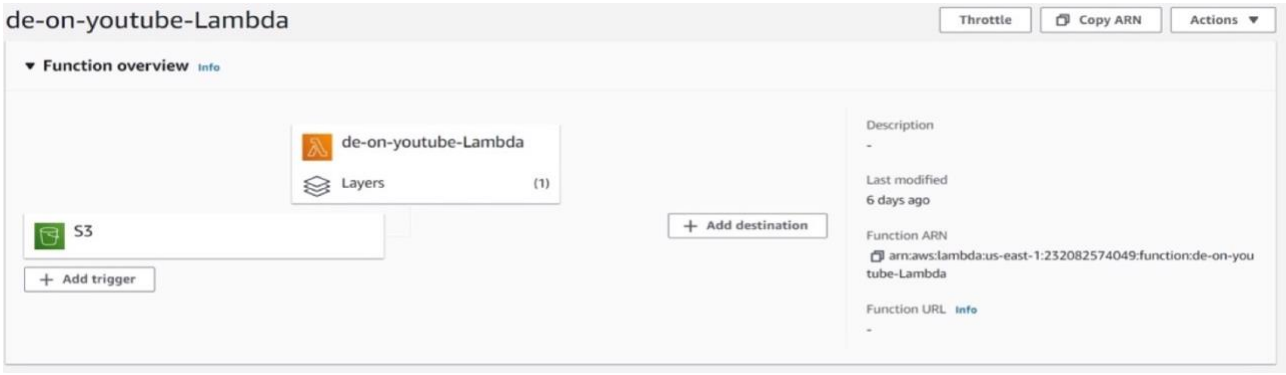
Schema updates in the data store	Update the table definition in the data catalog.
Object deletion in the data store	Mark the table as deprecated in the data catalog.



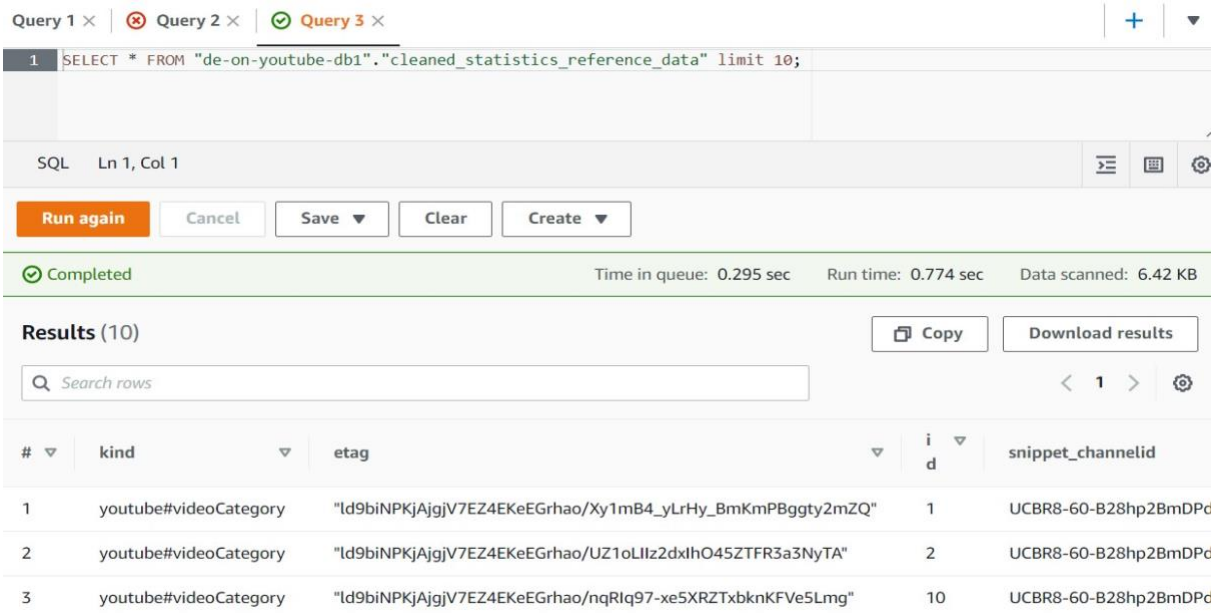
The data was transformed and moved into the cleaned bucket which had the cleaned JSON files.

Step 5:- we need to add a trigger on AWS Lambda,so that whenever a new file is added Lambda function can perform the transformations on it automatically.

1. Added trigger on the raw S3 bucket
2 Select the event that you want to trigger the lambda function.
3 Filled in the required information
4 Added the suffix according to the format of the file in our case JSON and CSV.



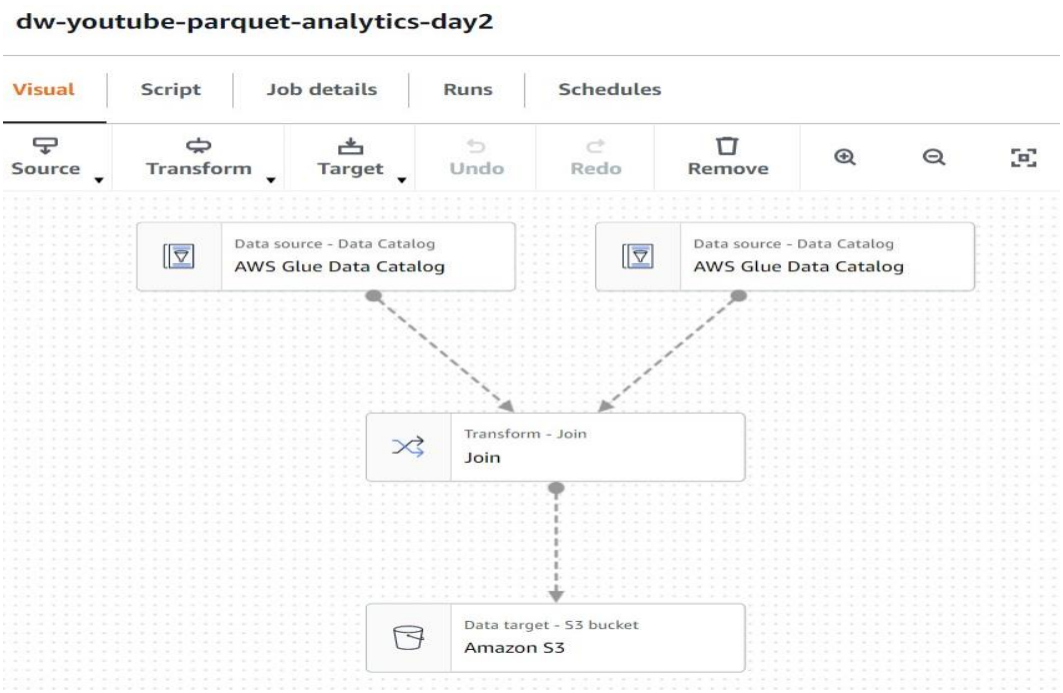
Ran the Lambda function and checked whether the files are being created in the above-mentioned S3 bucket. Open Amazon Athena to check whether the database is accessible.



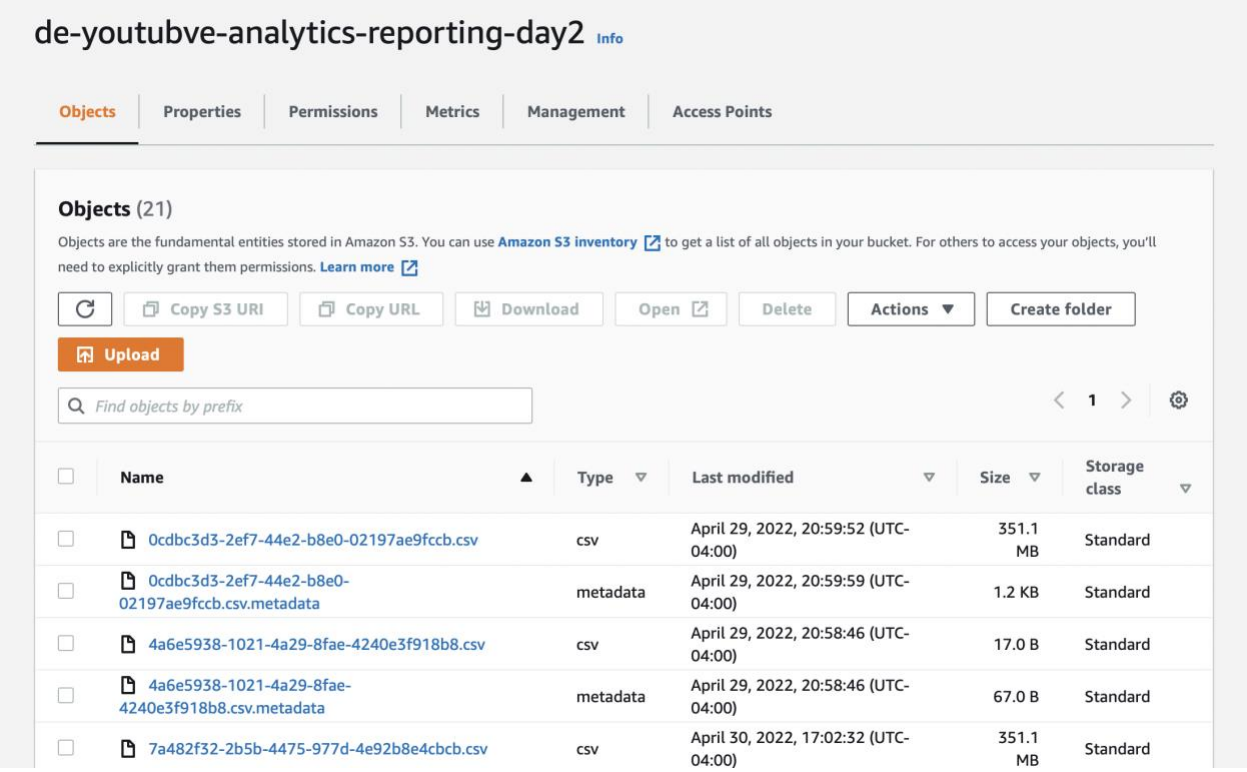
Step 6 -After that, we added to the analytical bucket for final reporting and visualization.

Steps to create an analytical bucket: -

1. Opened AWS Glue studio
2. Created an empty job
3. Added the sources (JSON and CSV)



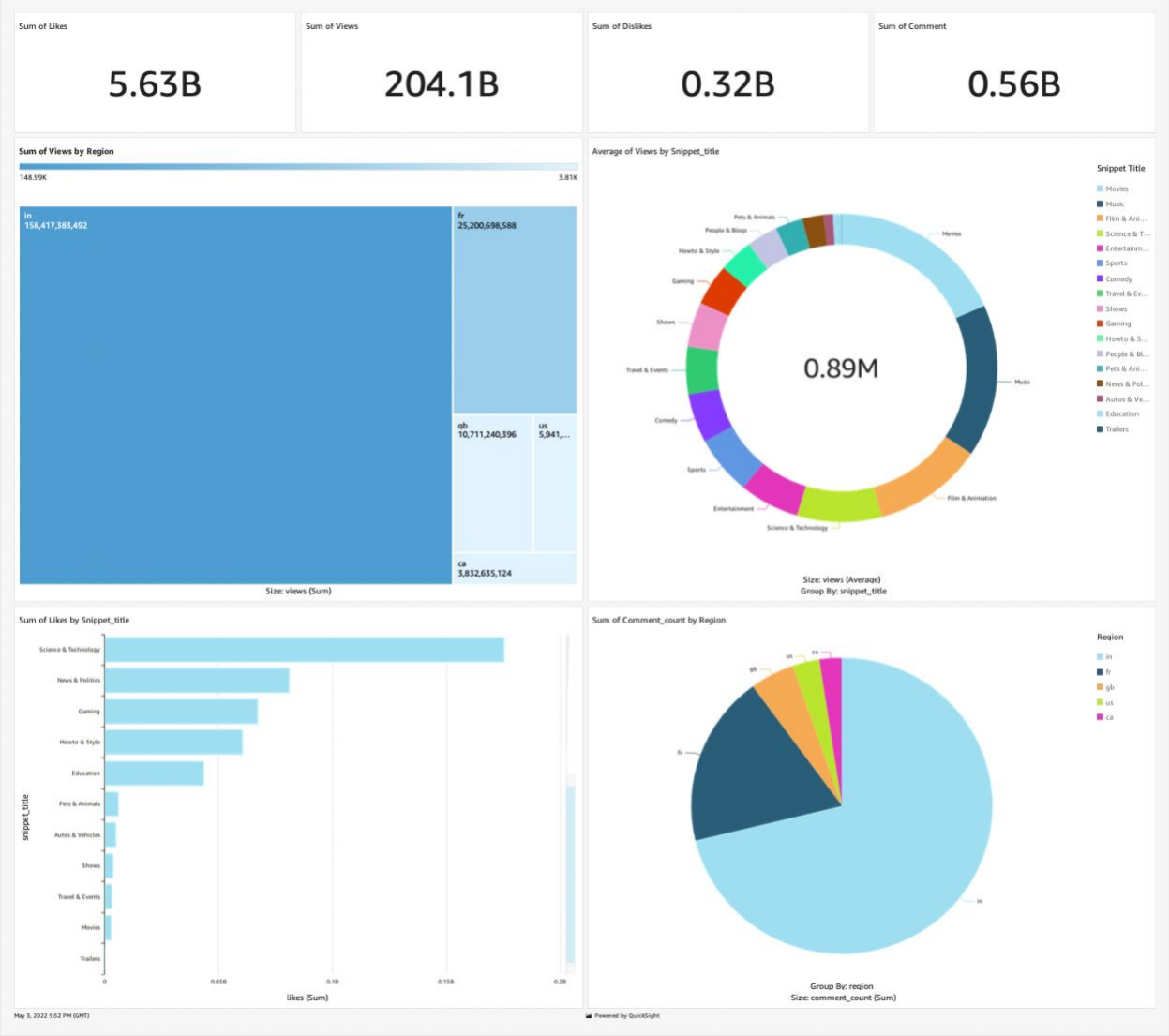
4. Did Inner join based on appropriate columns
5. Move the data into the analytical S3 bucket (3rd) and select the appropriate format
6. Select the appropriate format for saving the file and IAM role
7. Run the job
8. Check inside the analytical bucket to make sure the job has run successfully



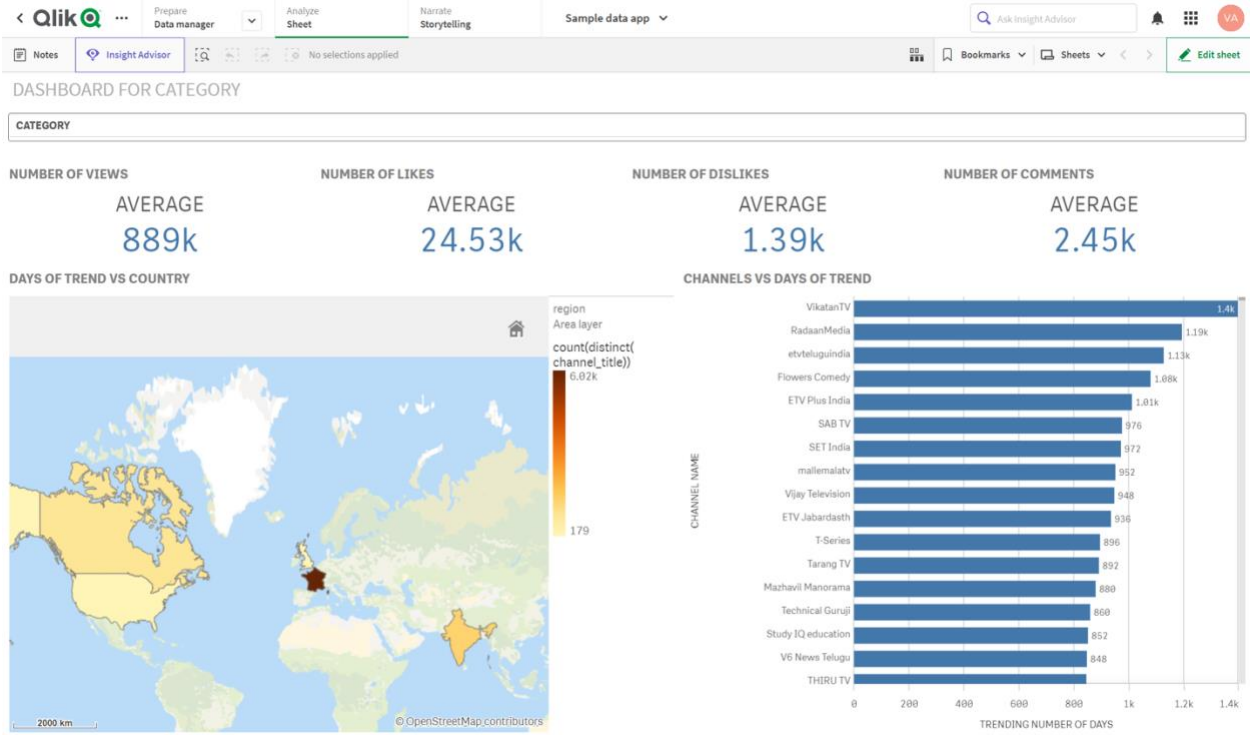
Business Intelligence

We created visualizations according to our business problem on two dashboards on two different platforms QuickSight and QlikSense

#Dashboard - 1



#Dashboard 2



Inferences and suggestions based on data: -

1. The greatest number of views are coming from the region of India, which might be due to the immense population but also means that any company looks to use YouTube as a medium promotional tool, can get the maximum coverage if it targets in and around that region or at least has some specifications according to it.
2. The greatest number of views is from the category movies, which makes it the best location for adding placements for maximum visibility.
3. Although the views are fewer, in terms of likes Science and technology have the most likes showing the most sort after the quality of content. This also means that there is a space to leverage in the Science and Technology sector channels.
4. If a company is planning to promote its product in its respective category. Then the visualization in the second dashboard can help them understand the channel that is trending more in that category to target more customers

Conclusion

So, we conclude that the company would look at the above inferences and will understand that the target country can be India as maximum views are coming from there also Science and technology videos have the most likes. We would suggest that the company should place their ad campaign in Entertainment since it has the maximum number of views and the second option can be Science and technology as it has the most number of likes.

References:

https://aws-dojo.com/workshoplists/

https://aws.amazon.com/blogs/big-data/build-a-lake-house-architecture-on-aws/

https://aws.plainenglish.io/making-sense-of-the-aws-lake-house-architecture-ef0d8bc47897

https://aws.amazon.com/blogs/architecture/benefits-of-modernizing-on-premise-analytics-with-an-aws-lake-house/

https://aws-dojo.com/workshoplists/

https://aws.amazon.com/glue/?whats-new-cards.sort-by=item.additionalFields.postDateTime&whats-new-cards.sort-order=desc