University *of* New Haven

TAGLIATELA COLLEGE OF ENGINEERING

Electrical & Computer Engineering and Computer Science

Electrical & Computer Engineering & Computer Science (ECECS)

TECHNICAL REPORT

Fall 22

TECHNICAL REPORT

Trending YouTube Video Analysis

Group 6
Appidi Moni (amoni3@unh.newhaven.edu)
Mohammad Feroz Ahmad Qureshi (fmoha7@unh.newhaven.edu)
Ravina Ingole Ringo (ringo1@unh.newhaven.edu)
Sai Gowtham Reddy Kallu (skall8@unh.newhaven.edu)
Sudip Adhikari (sadhi9@unh.newhaven.edu)

University of New Haven
Electrical & Computer Engineering & Computer Science (ECECS)
West Haven, CT

December 2022

TABLE OF CONTENTS

Abstract 2
Highlights of the Project 2
Introduction 2
Methodology 3
Machine Learning-YouTube Views Prediction 9
Result Section/Discussion 11
Conclusion 11
References 11

Abstract

YouTube (the world's most popular video sharing website) maintains track of the top trending videos on the platform. According to Variety magazine, "to pick the year's top-trending videos, YouTube applies a variety of factors, including tracking viewer interactions" (number of views, shares, comments and likes). This study provides a detailed investigation of Trending YouTube Video Statistics. It will help in addressing business questions such as which titles are most liked and disliked on YouTube. We used Kaggle datasets for this, followed by data cleaning, database generation, data visualization, and the development of a Machine Learning model for YouTube view forecasting.

Highlights of the Project

This project will help to answer below business questions:

a. Data Visualization using Power BI

- Region which has most liked and disliked titles.
- Ratio of Number of likes upon number of views.
- Sum of likes by region.
- Region with most disabled comments.

b. Efficiently implemented ETL process

- Created pipelines to store data in S3 buckets
- Created ETL pipeline to join tables in AWS Glue Studio
- Used AWS Athena to query databases.

c. YouTube View Predictions using Machine Learning

- Found correlation between views, likes, and dislikes.
- Data exploration, pre-processing, and modelelopment
- Test data Evaluation

Introduction

YouTube subscribers are growing by the day, making it difficult for artists to keep up with their demands. As a result, it is critical to employ statistical data to construct recommendation systems that can help answer what is trending depending on location and which artists are most popular among consumers. This project will assist in answering such queries based on areas, comments, likes and dislikes, and machine learning algorithms will assist in predicting YouTube views.

The full documentation of the project is available at the github-
https://github.com/iamsudipadhikari/trending-youtube-video-analysis

The Dashboard of the project can be accessed here:
https://app.powerbi.com/links/hq7X9RJNxl?ctid=3c71cbab-b5ed-4f3b-ac0d-95509d6c0e93&pbi_source=linkShare

Methodology

Understanding the Business

YouTube Analysis can give you with the most recent performance statistics for all of your videos. The several reports provide a variety of data, including likes, comments, and traffic sources. YouTube Analytics gives you valuable information on how your material is consumed, how long it is watched for, and if it is liked or disliked. Knowing your audience and their behaviors will enable you to tailor your videos, broaden your reach, and turn your channel into a viewing destination.

Data Understanding

The Kaggle website provided the dataset. Below is a link to the source.

https://www.kaggle.com/datasets/datasnaek/youtube-new

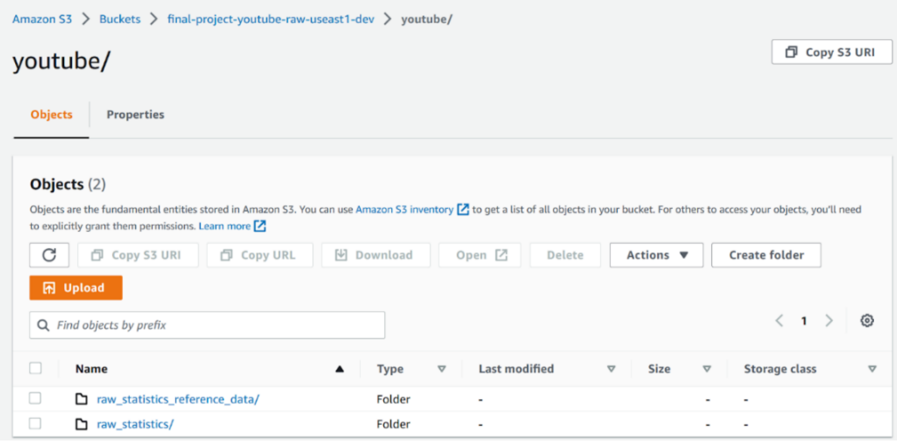
The daily YouTube video trends through time are detailed in this dataset (and counting). Data from the RU, MX, KR, JP, and IN regions are supplied for the same time period (respectively, Russia, Mexico, South Korea, Japan, and India). The data for each region is saved in a separate file. The information comprises the video's title, the channel name, the date and time it was published, tags, views, likes, dislikes, a description, and the number of comments.

A category id field, which varies by region, is also present in the data. Find the associated JSON to retrieve the categories for a certain video. These files are dispersed throughout the dataset's five regions.

Preparation of Data

Below steps were followed for Data Cleaning and data storing on AWS:

1. Fetched the data from Kaggle; data was in the form of json and csv file format.
   Link: - https://www.kaggle.com/datasets/datasnaek/youtube-new
2. Stored the data into S3 bucket in different directories; separate directory for csv files and json.



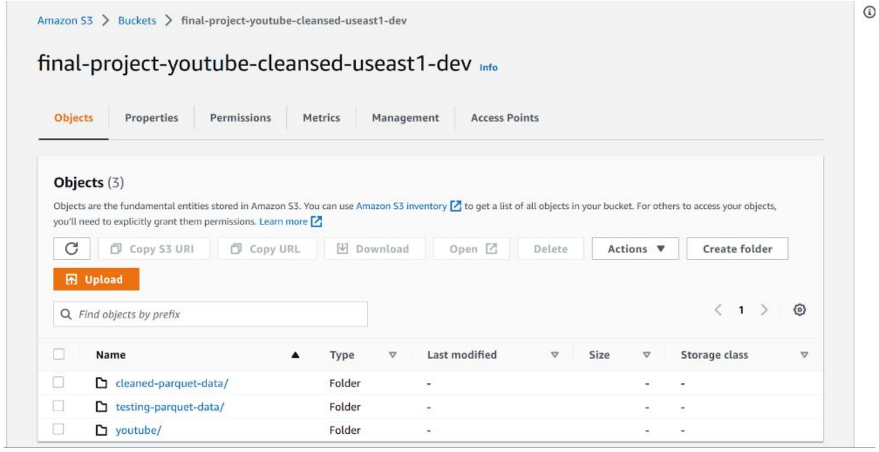
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Last modified	Size	Storage class
□	raw_statistics_reference_data/	Folder	-	-	-
□	raw_statistics/	Folder	-	-	-

Fig: Bit-Buckets file Store

All the CSV data is partitioned in terms of region and stored in respective buckets depending on region names.

3. Wrote a Lambda function to transform the json and csv data to parquet data (parquet is just another file format like pickle which stores data by serializing. By converting the data to Parquet, we had the homogeneous data of json and csv) . We have stored the parquet data into different buckets named as cleaned-data-bucket. The cleaned json and csv data is put into different directories.



Objects (3)

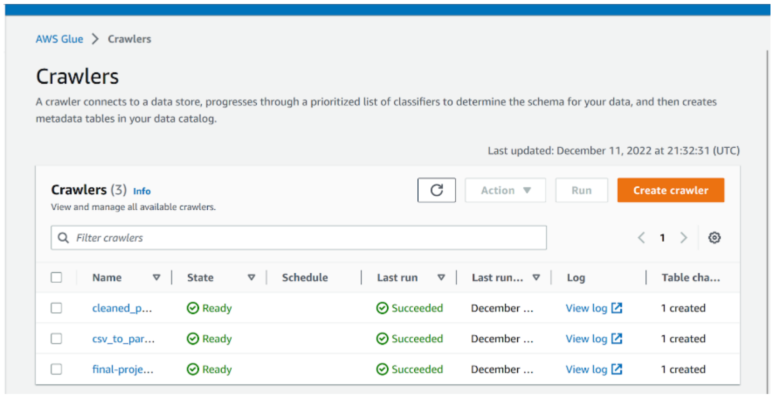
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	cleaned-parquet-data/	Folder	-	-	-
□	testing-parquet-data/	Folder	-	-	-
□	youtube/	Folder	-	-	-

Fig: Bit-Bucket-File-Store-1

Database creation:

Used AWS glue to create two separate crawlers for 2 different data sets (name: cleaned_parquet_data and youtube) located in S3 bucket (final-project-youtube-cleaned-useast1-dev).



A crawler connects to a data store, progresses through a prioritized list of classifiers to determine the schema for your data, and then creates metadata tables in your data catalog.

Last updated: December 11, 2022 at 21:32:31 (UTC)

Crawlers (3) Info

View and manage all available crawlers.

□	Name	State	Schedule	Last run	Last run...	Log	Table cha...
□	cleaned_p...	Ready		Succeeded	December ...	View log	1 created
□	csv_to_par...	Ready		Succeeded	December ...	View log	1 created
□	final-proje...	Ready		Succeeded	December ...	View log	1 created

Fig: Crawlers

Data Architecture:

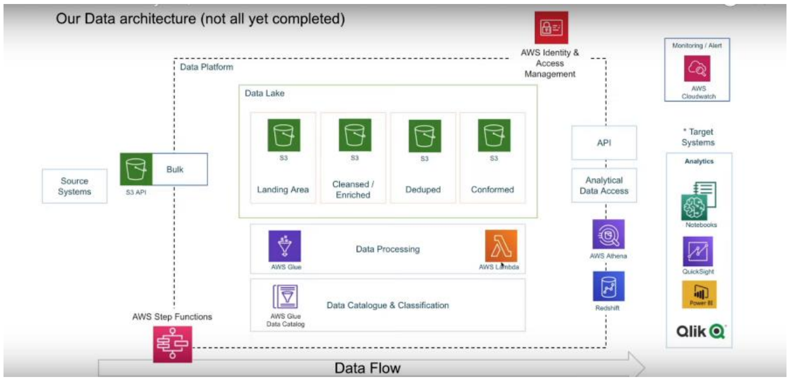


Fig: Data Architecture

Tools Used

S3

AWS glue Data Catalog

AWS glue Studio for ETL

AWS Athena

AWS Lambda

Pandas

Power BI

ODBS Connector

S3

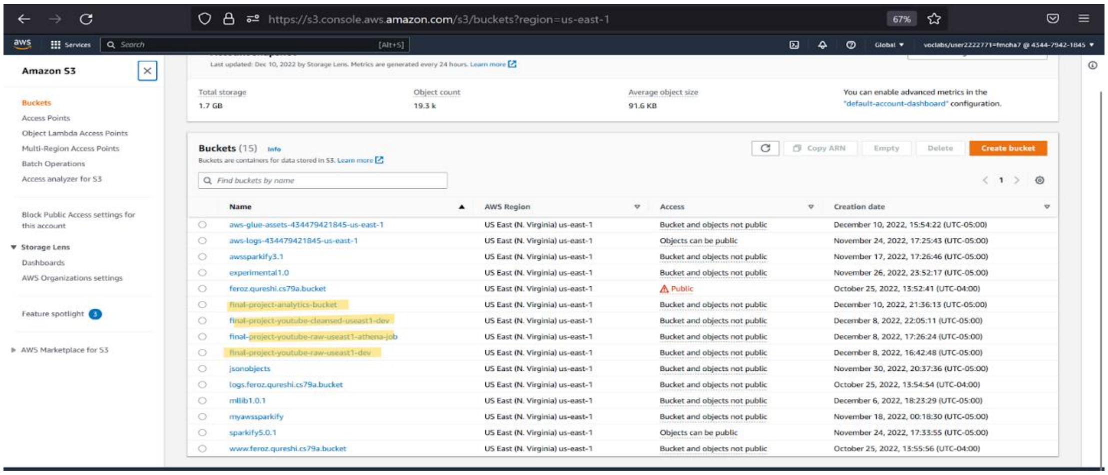
We've used S3 to store the data in the buckets. We've created 4 buckets for successful operation.

Firstly, we've used a raw bucket where we put all our uncleaned data which was directly downloaded from Kaggle. In Kaggle, the data was available in the form of json and csv for Youtube Data Analysis Data. We created separate directories for json and csv data respectively.

Going along, to store the cleaned data, we created another bucket. We've transformed the both csv and json files to parquet file format to make the data serialized and homogenized. Even here, we didn't give up our convention and created directories for json to parquet converted data and csv to parquet converted data. Here, we've used pandas to transform the id column in json data from string data type to integer data type which would be so helpful in further querying.

The joined data, the joined parquet data of csv and json, has been stored in the final reporting bucket (more information about joining the data will be explained in the Aws Glue Studio tool usage). Used reporting bucket to create PowerBI dashboard.

All the S3 buckets used for this project are highlighted in yellow. We've followed a naming convention for the final project; all the bucket names starting with "final-project" are related to this project.


https://s3.console.aws.amazon.com/s3/buckets?region=us-east-1
67%
aws
Services
Search
[Alt+S]
Global
voclabs/user2222771=fmoha7 @ 4344-7942-1845
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
AWS Marketplace for S3
Copy ARN
Empty
Delete
Create bucket
Find buckets by name

Last updated: Dec 10, 2022 by Storage Lens. Metrics are generated every 24 hours. Learn more

Total storage	Object count	Average object size
1.7 GB	19.3 k	91.6 KB

You can enable advanced metrics in the "default-account-dashboard" configuration.

Buckets (15) Info

Buckets are containers for data stored in S3. Learn more

Name	AWS Region	Access	Creation date
aws-glue-assets-434479421845-us-east-1	US East (N. Virginia) us-east-1	Bucket and objects not public	December 10, 2022, 15:54:22 (UTC-05:00)
aws-logs-434479421845-us-east-1	US East (N. Virginia) us-east-1	Objects can be public	November 24, 2022, 17:25:43 (UTC-05:00)
awssparkify3.1	US East (N. Virginia) us-east-1	Bucket and objects not public	November 17, 2022, 17:26:46 (UTC-05:00)
experimental1.0	US East (N. Virginia) us-east-1	Bucket and objects not public	November 26, 2022, 23:52:17 (UTC-05:00)
feroz.qureshi.cs79a.bucket	US East (N. Virginia) us-east-1	⚠ Public	October 25, 2022, 13:52:41 (UTC-04:00)
final-project-analytics-bucket	US East (N. Virginia) us-east-1	Bucket and objects not public	December 10, 2022, 21:36:13 (UTC-05:00)
final-project-youtube-cleansed-useast1-dev	US East (N. Virginia) us-east-1	Bucket and objects not public	December 8, 2022, 22:05:11 (UTC-05:00)
final-project-youtube-raw-useast1-athena-job	US East (N. Virginia) us-east-1	Bucket and objects not public	December 8, 2022, 17:26:24 (UTC-05:00)
final-project-youtube-raw-useast1-dev	US East (N. Virginia) us-east-1	Bucket and objects not public	December 8, 2022, 16:42:48 (UTC-05:00)
jsonobjects	US East (N. Virginia) us-east-1	Bucket and objects not public	November 30, 2022, 20:37:36 (UTC-05:00)
logs.feroz.qureshi.cs79a.bucket	US East (N. Virginia) us-east-1	Bucket and objects not public	October 25, 2022, 13:54:54 (UTC-04:00)
mllib1.0.1	US East (N. Virginia) us-east-1	Bucket and objects not public	December 6, 2022, 18:23:29 (UTC-05:00)
myawssparkify	US East (N. Virginia) us-east-1	Bucket and objects not public	November 18, 2022, 00:18:30 (UTC-05:00)
sparkify5.0.1	US East (N. Virginia) us-east-1	Objects can be public	November 24, 2022, 17:33:55 (UTC-05:00)
www.feroz.qureshi.cs79a.bucket	US East (N. Virginia) us-east-1	Bucket and objects not public	October 25, 2022, 13:55:56 (UTC-04:00)

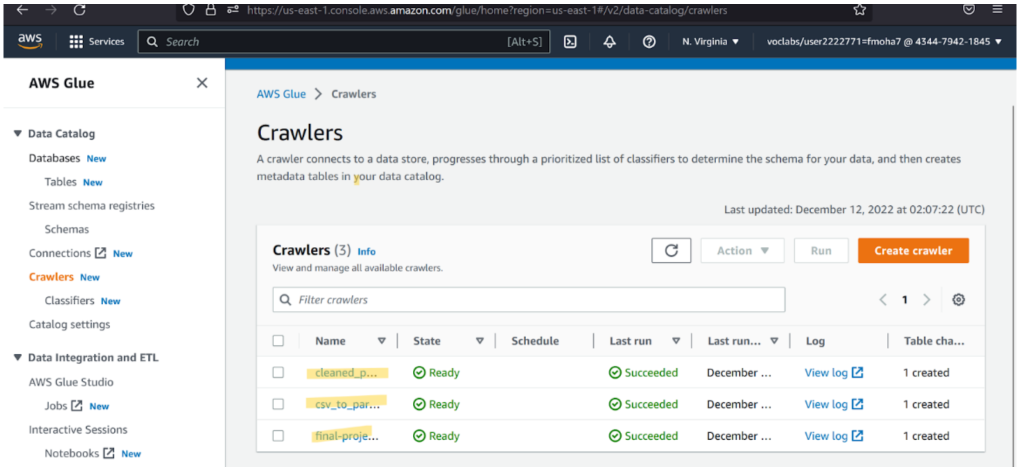
Fig: S3 Buckets

AWS Glue Data Catalog

We've created a Glue data catalog to create databases and then tables. To achieve this, we've created 2 crawlers. The first crawler would be responsible to crawl over all the json to parquet converted data and extract metadata from that; using that we created a table in the database. There is this second crawler which is responsible for extracting metadata from csv data which is converted in parquet data in the previous operations.

Finally, when these two tables are joined in AWS Glue ETL operation, (this would be discussed in the AWS Glue studio tool usage part) and stored in the analytics bucket, we run another crawler to get the metadata from this bucket which in turn creates the final database for direct usage.

All the crawlers created for this project have been highlighted in yellow.



Crawlers

A crawler connects to a data store, progresses through a prioritized list of classifiers to determine the schema for your data, and then creates metadata tables in your data catalog.

Last updated: December 12, 2022 at 02:07:22 (UTC)

Crawlers (3) Info

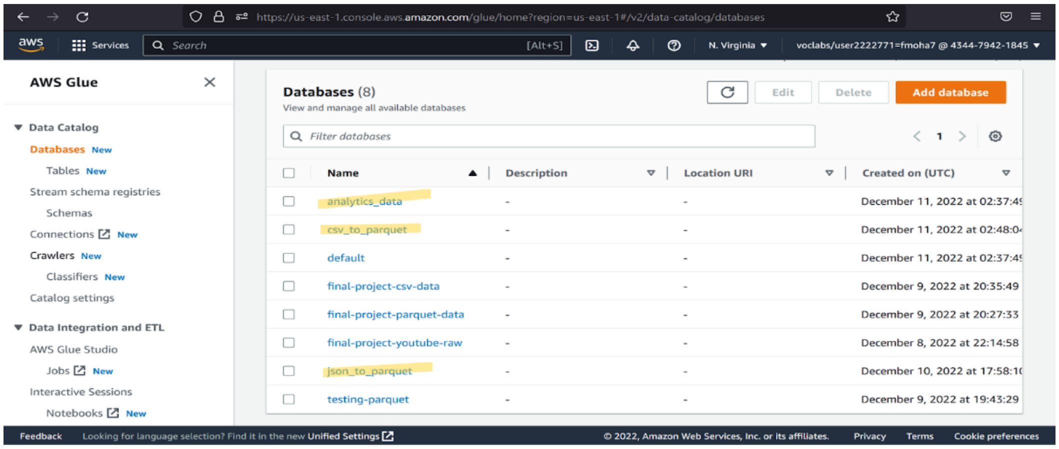
View and manage all available crawlers.

Name	State	Schedule	Last run	Last run...	Log	Table cha...
cleaned_p...	Ready		Succeeded	December ...	View log	1 created
csv_to_par...	Ready		Succeeded	December ...	View log	1 created
final-proje...	Ready		Succeeded	December ...	View log	1 created

Fig: Crawlers

AWS Glue for ETL

We’ve used AWS Glue visual Studio manager to join the two different tables from the database. The 3 databases can be seen highlighted in the picture.



Databases (8)

View and manage all available databases

□	Name	Description	Location URI	Created on (UTC)
□	analytics_data	-	-	December 11, 2022 at 02:37:4
□	csv_to_parquet	-	-	December 11, 2022 at 02:48:0
□	default	-	-	December 11, 2022 at 02:37:4
□	final-project-csv-data	-	-	December 9, 2022 at 20:35:49
□	final-project-parquet-data	-	-	December 9, 2022 at 20:27:33
□	final-project-youtube-raw	-	-	December 8, 2022 at 22:14:58
□	json_to_parquet	-	-	December 10, 2022 at 17:58:1
□	testing-parquet	-	-	December 9, 2022 at 19:43:29

Fig: Database

Glue studio is kind of visual, and we’ve written a sql join statement in the Glue studio and set the target path to the analytics bucket

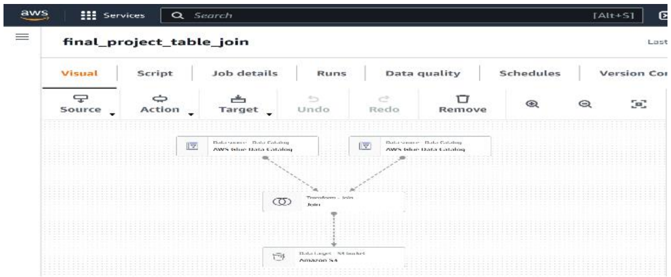


Fig: Table Join

AWS Athena

Aws Athena is generally a query editor. Here, We’ve created databases. We’ve done some querying before finalizing the database for reporting.

AWS Lambda

We’ve used AWS Lambda to write certain functions for data transformation.

Pandas

Pandas is where we’ve transformed the data from json to parquet in the first place. The code for that would be in the GitHub repo.

Power BI

More information about visuals would be discussed in the visualization module.

ODBC Connector

The ODBC connector is used for AWS Athena with Power BI. We just need to create a DSN in ODBC and access that DSN in Power BI. The DSN we created for this job is “group6”.

Visualization

Connected AWS Athena to Power BI and made a dashboard using data from the reporting bucket.

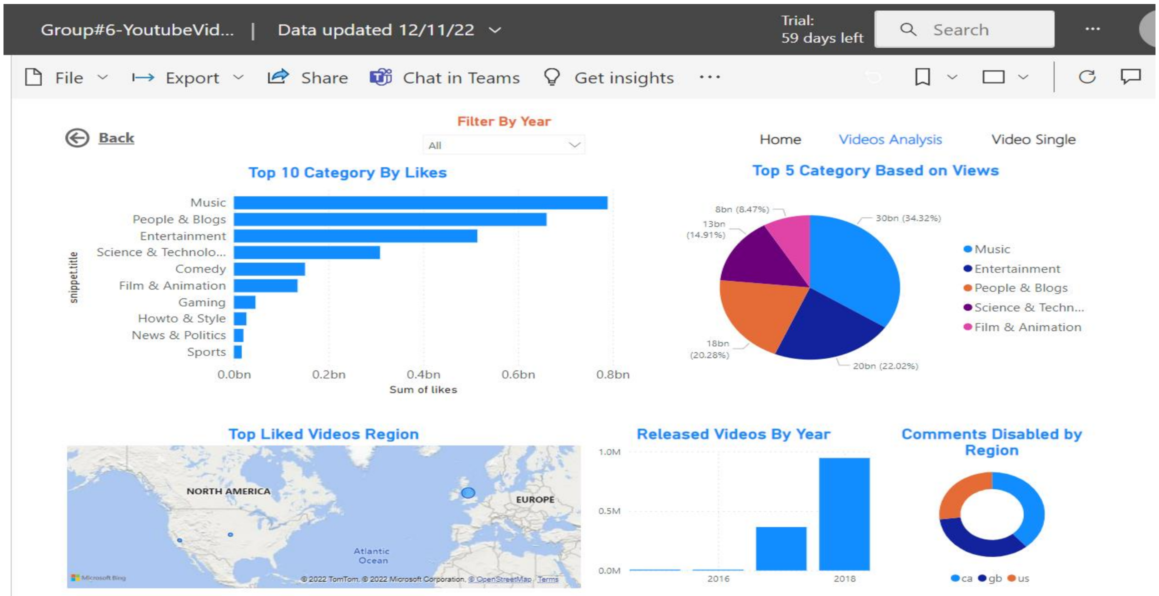
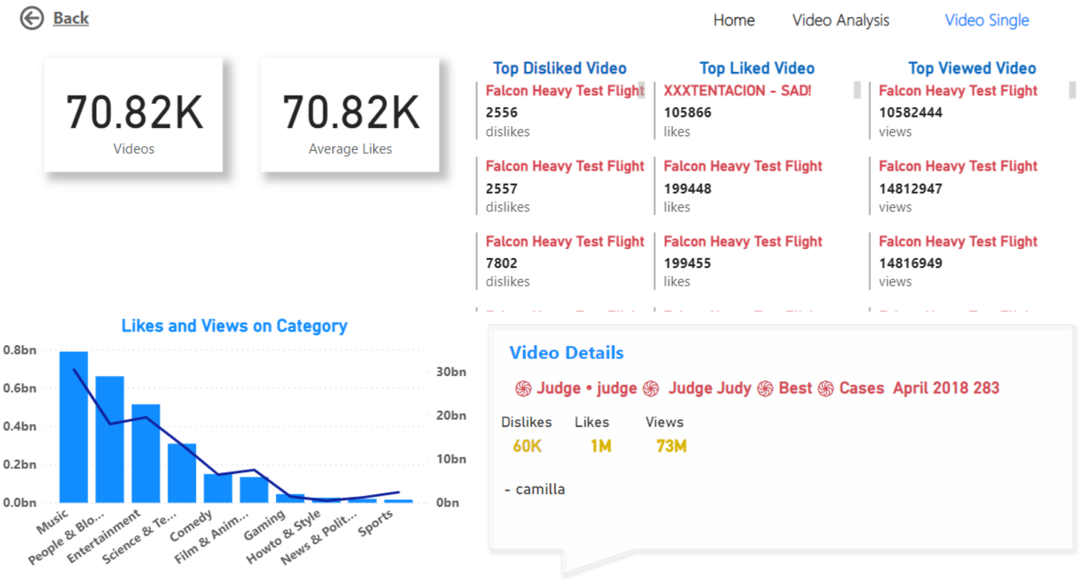


Fig: Visualization-1



70.82K
Videos

70.82K
Average Likes

Top Disliked Video	Top Liked Video	Top Viewed Video
Falcon Heavy Test Flight 2556 dislikes	XXXTENTACION - SAD! 105866 likes	Falcon Heavy Test Flight 10582444 views
Falcon Heavy Test Flight 2557 dislikes	Falcon Heavy Test Flight 199448 likes	Falcon Heavy Test Flight 14812947 views
Falcon Heavy Test Flight 7802 dislikes	Falcon Heavy Test Flight 199455 likes	Falcon Heavy Test Flight 14816949 views

Video Details

Judge • judge Judge Judy Best Cases April 2018 283

Dislikes	Likes	Views
60K	1M	73M

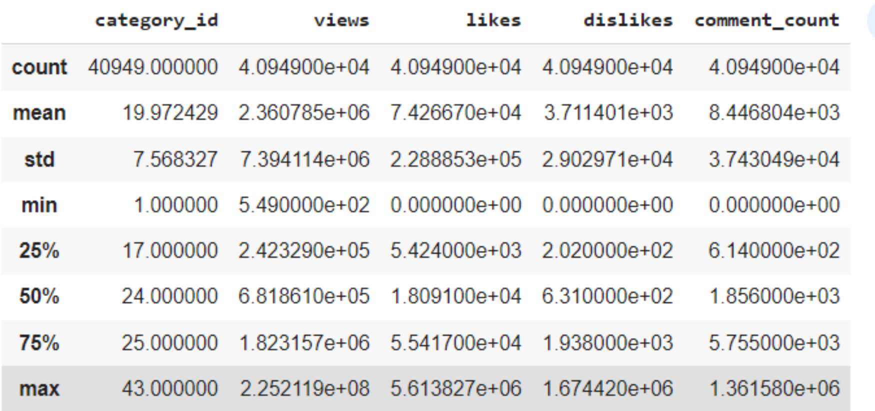
- camilla

Fig: Visualization-2

Machine Learning-YouTube Views Prediction

Exploratory Data Analysis

- The distribution for feature views, likes, dislikes, and comment count appear to be skewed (mean & median are not close enough)
- It can be seen in the boxplot graph that the view, likes, dislikes, comment_count features have many outliers so that logarithmic transformations are needed for these features.



	category_id	views	likes	dislikes	comment_count
count	40949.000000	4.094900e+04	4.094900e+04	4.094900e+04	4.094900e+04
mean	19.972429	2.360785e+06	7.426670e+04	3.711401e+03	8.446804e+03
std	7.568327	7.394114e+06	2.288853e+05	2.902971e+04	3.743049e+04
min	1.000000	5.490000e+02	0.000000e+00	0.000000e+00	0.000000e+00
25%	17.000000	2.423290e+05	5.424000e+03	2.020000e+02	6.140000e+02
50%	24.000000	6.818610e+05	1.809100e+04	6.310000e+02	1.856000e+03
75%	25.000000	1.823157e+06	5.541700e+04	1.938000e+03	5.755000e+03
max	43.000000	2.252119e+08	5.613827e+06	1.674420e+06	1.361580e+06

Fig: DataFrame View

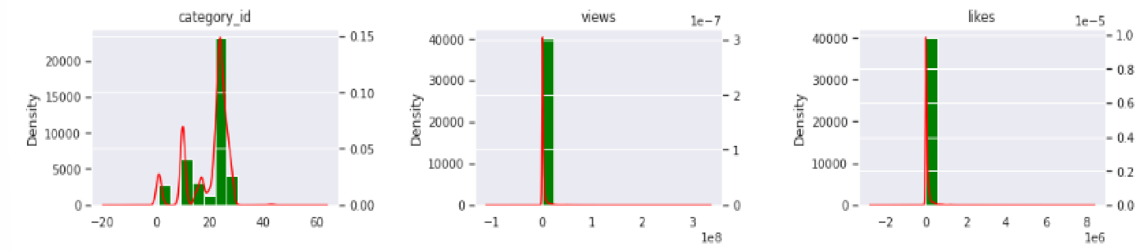


Fig: Visualization-3

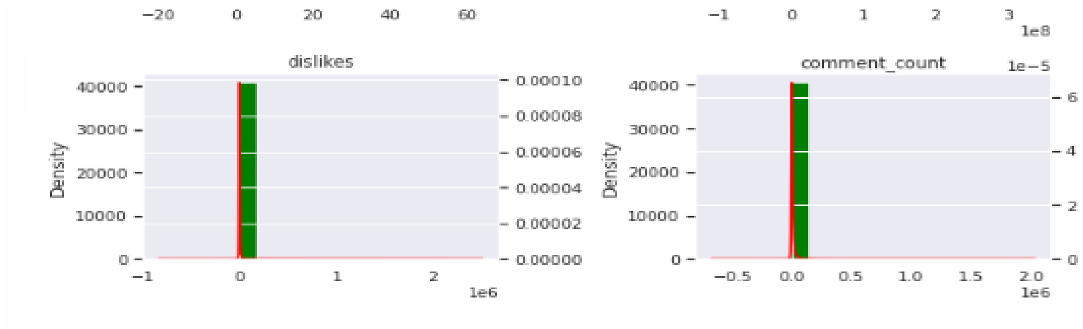


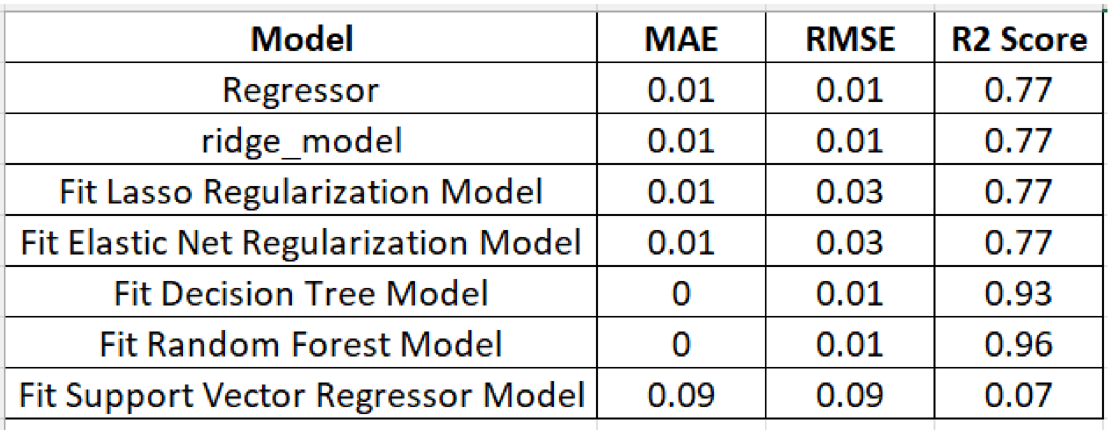
Fig: Visualization-4

Data Pre-processing

- Publish_time, publish_date, description, tags, title, channel_title features are removed because they have a very large number of unique values, so it is assumed that each row of data has a different value.
- Filled Missing Column with Median
- The transformation log is used to transform features with numerical data values into distributions that are normal or nearly normal, as there are many features that contain skewed data.
- Finally, we normalized so that the scale of each numeric feature has the same scale and it is hoped that it can simplify the process of learning the machine learning model data that we created.

Model Building with hyperparameter tuning

Used below models and it’s calculated MAE, RMSE and R2 score for each model:



Model	MAE	RMSE	R2 Score
Regressor	0.01	0.01	0.77
ridge_model	0.01	0.01	0.77
Fit Lasso Regularization Model	0.01	0.03	0.77
Fit Elastic Net Regularization Model	0.01	0.03	0.77
Fit Decision Tree Model	0	0.01	0.93
Fit Random Forest Model	0	0.01	0.96
Fit Support Vector Regressor Model	0.09	0.09	0.07

Fig: Error & Accuracy Table

The model we chose is the Random Forest Model, which has a low tendency to overfit, but we believe it can be tolerated within typical tolerable limitations, with a train accuracy of 99% and a test accuracy of 96%.

Conclusion

Based on the results of the experiments, there are several models that are useful for predicting the views of YouTube videos.

- Random Forest with R2 of 0.96 makes as the best model so far
- Decision Tree is the second-best sequence model after Random Forest with a slightly smaller R2 value of 0.92
- The next best model is Ridge Regularization with 0.01 MAE, RMSE values, and R2 of 0.77
- A very influential feature is the number of likes and dislikes of a video

Result Section/Discussion

1. Data Analysis results: -

- The most liked category is Music.
- The most viewed video is Falcon Heavy Test flight.
- The top disliked video is also Falcon Heavy Test flight.

2. YouTube views prediction modeling results: -

- Random Forest of 0.96 makes Random Forest the best model so far.
- The most influential feature is the number of likes and dislikes of a video.

Conclusion

In conclusion, we have built the data pipelines to efficiently carry out the ETL process. Additionally, we've created dashboards to visualize all the key insights. On the top of that, we've modeled a regression model and evaluated it.

References

The ETL process and other AWS tools reference is taken from the below video.
https://www.youtube.com/watch?v=yZKJFKu49Dk

Data Analysis reference's taken from the following blog.
https://medium.com/@sonaliknr/understand-data-analysis-process-step-by-step-27ed384c13bb

BI visuals have been referred from this link.
https://learn.microsoft.com/en-us/power-bi/developer/visuals/develop-circle-card

Other references:
https://www.dremio.com/resources/guides/intro-data-engineering/#:~:text=Data%20engineering%20is%20the%20process,businesses%20can%20use%20to%20thrive.