Cloud-Based YouTube Trending Analytics Pipeline on AWS

CS 6705 Final Project

Kevin Bell Jacob Child

Fall 2025

Contents

1 Problem Statement and Scope

YouTube generates millions of daily interactions that indicate which videos are trending across regions. Stakeholders—content creators, advertisers, and platform operators—need timely insight into which videos will continue trending so they can optimize promotion, moderation, and infrastructure. Our project builds an end-to-end cloud pipeline that automatically ingests trending-video metadata and comments, curates reliable analytics datasets, applies sentiment analysis, and produces next-day trending predictions. The scope covers daily ingestion, scalable ETL, feature engineering, model training, and prediction publishing, with an emphasis on low-operations automation, cost control, and reproducibility.[1]

2 Background

Trending videos on YouTube are built on statistics of views, like, comments and contextual signals such as category, title language and audience sentiment. Industrial data platforms commonly use event-driven ingestion, data lakes for raw and curated layers, and distributed processing (e.g., AWS Glue/Spark) for feature computation. Sentiment-aware features can improve short-term popularity forecasting by capturing viewer reception. Our design follows the medallion architecture: raw JSON lands in Amazon S3, curated Parquet layers standardize schema, and downstream ML consumes labeled feature sets. Compared with single-node notebooks or cron-based scripts, the AWS EventBridge + Lambda + Glue stack provides resiliency, schema evolution through crawlers for both batch and near–real-time updates.

[1] Implementation details for the trending ETL job are in `Python/ETL/yt_trending_etl.py`.

3 System Architecture and Contributions

3.1 End-to-End Flow

A 6:00 AM EventBridge rule triggers two Lambda functions to fetch trending videos and comments via the YouTube Data API. The trending Lambda then launches a Glue Workflow that chains six ETL jobs, ensuring dependency ordering and eliminating manual coordination. Figure 1 summarizes the daily flow.

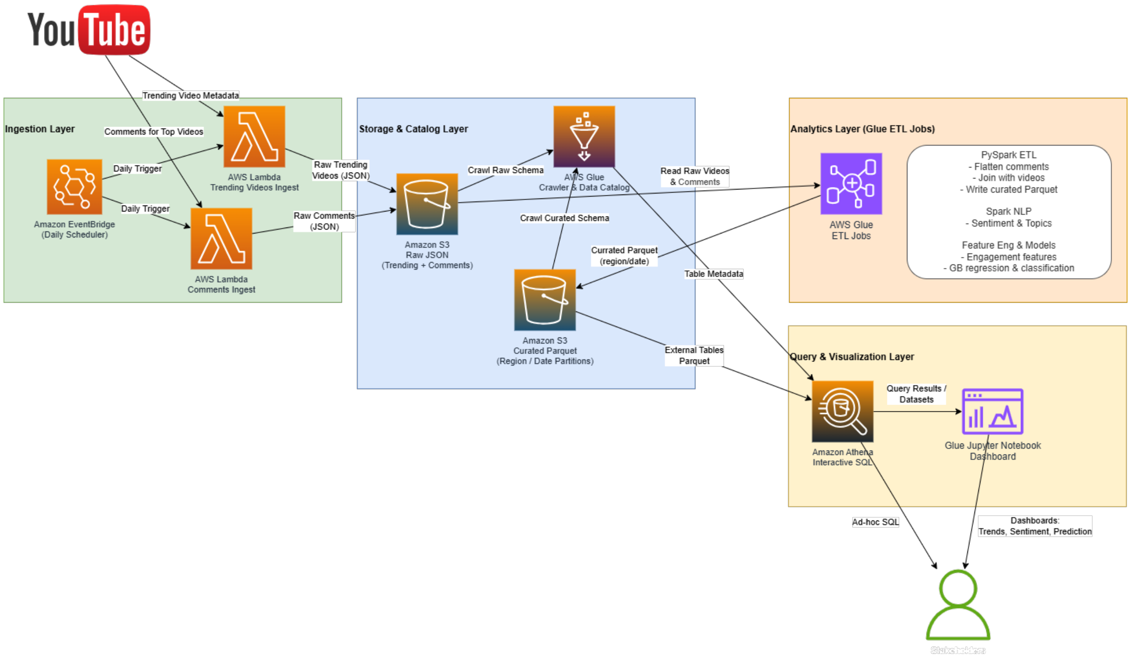


Figure 1: Daily ingestion and ETL orchestration triggered by EventBridge.

3.2 Trending ETL

The `yt_trending_etl.py` Glue job flattens nested API responses, extracts region and trending date from S3 paths, casts metrics, deduplicates by video ID (keeping the highest view count), and enriches timestamps with date, hour, week, and cleaned titles before writing curated Parquet partitioned by region and trending date.[2]

[2] See `Python/ETL/yt_trending_etl.py`, lines 12–168.

3.3 Comments ETL

The comments ETL normalizes one row per comment, keeps author, text, likes, and published timestamp, and derives region and ingest date from file names. Curated Parquet is appended and partitioned by region and ingest date, ready for sentiment analysis and engagement statistics.[3]

3.4 Sentiment and Feature Labeling

A subsequent Glue job scores each comment with AWS Comprehend to compute per-video averages of positive, negative, neutral, and mixed sentiment. The one-time backfill script joins curated trending data with sentiment, constructs time-window lags and ratios (view/like/comment velocities, growth ratios, engagement per view), derives labels for next-day view growth and continued trending presence, and writes labeled feature sets partitioned by region and ingest date.[4]

3.5 Modeling and Predictions

Within the Glue workflow, training and prediction steps retrain models on updated labeled features and emit next-day forecasts. Outputs include predicted view counts and probability a video will remain trending, stored as curated Parquet for dashboarding and ad hoc analytics using Amazon Athena.

3.6 Infrastructure and Security

A dedicated VPC isolates Glue resources, with public subnets for internet-facing components and private subnets using a NAT Gateway for outbound API and package access. Amazon S3 serves as the data lake for raw, curated, and model outputs. Secrets Manager supplies

[3] See `Python/ETL/yt_comments_etl.py`, lines 8–110.

[4] See `Python/ETL/yt-onetimebatch.py`, lines 1–209, and sentiment schema examples in `Notes.md`, lines 72–119.

API credentials to Lambda. Figure 2 depicts the network design, and the Glue workflow ordering mirrors the medallion progression from raw to curated to ML outputs.

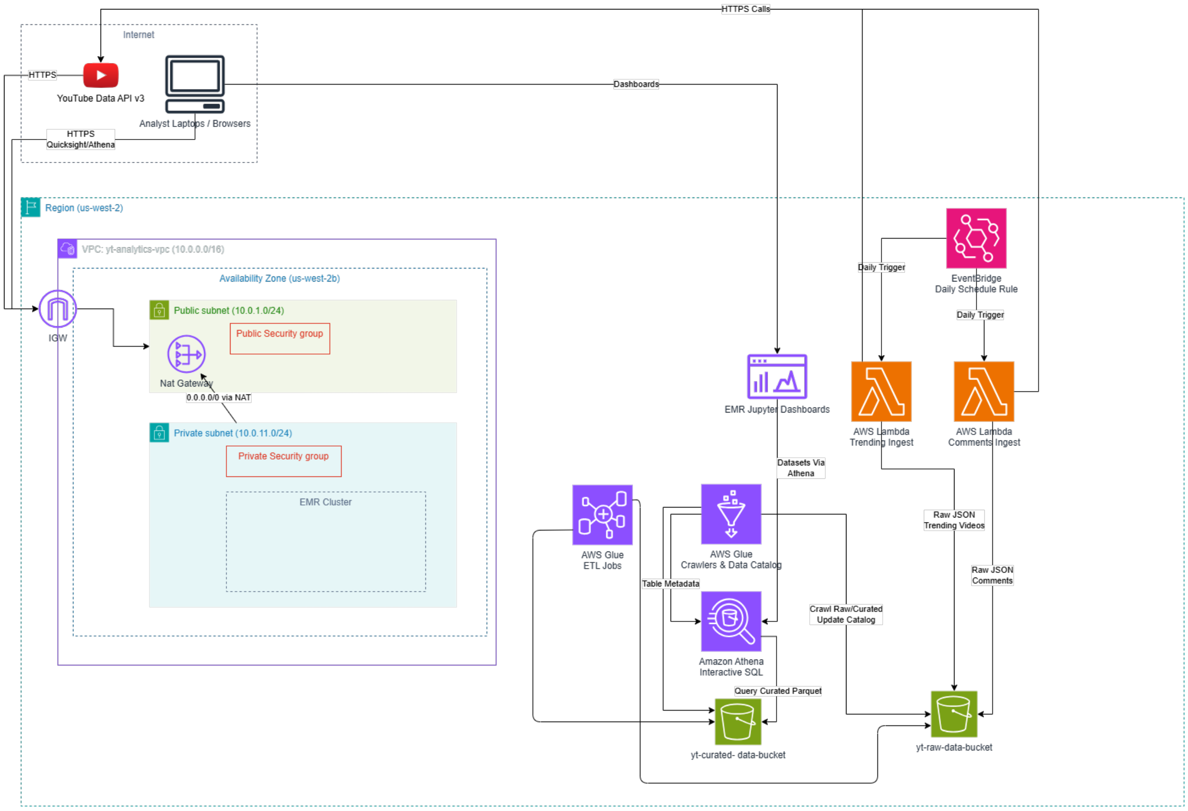


Figure 2: Network architecture: VPC isolation, NAT routing, and service boundaries.

4 Results and Analysis

4.1 Curated Schemas

The curated trending schema captures identifiers, metrics, time fields, and cleaned titles; the curated comments schema retains author, text, likes, and published timestamp. Example schemas are shown in Table 1.

Curated Trending (excerpt)	**Curated Comments (excerpt)**
video_id (string)	video_id (string)
channel_id (string)	author_display_name (string)
category_id (string)	comment_text (string)
title, title_clean (string)	like_count (long)
published_at (timestamp)	published_at (string)
view/like/comment_count (long)	region (string)
region (string)	ingest_date (date)
trending_date (date)	

Table 1: Curated schemas summarized from `Notes.md`.

4.2 Sentiment Insights

Aggregated sentiment features surface audience reception per video and date. Table 2 highlights example distributions (positive shares commonly above 0.45 with varying negative proportions), offering discriminative signals for trend persistence.

Video ID	**Neg.**	**Pos.**	**Neu.**	**Mix.**
f9cmVvoff_E	0.13	0.62	0.24	0.01
qyG8ECu6PLs	0.13	0.45	0.41	0.01
NED7nev2ywQ	0.03	0.59	0.37	0.00
5VYsnngkS_U	0.27	0.48	0.22	0.02
sNHHfHIewpU	0.21	0.14	0.62	0.02

Table 2: Sample aggregated comment sentiment scores.

4.3 Feature Engineering and Labels

Lag-based velocities and growth ratios quantify momentum, while engagement-per-view and log-count transforms stabilize variance across orders of magnitude. Labels track whether a video remains in the trending set on the next ingest date and estimate logarithmic view growth; this framing supports classification for stay-trending probability and regression for view-count prediction.

4.4 Model Outputs

Predicted outputs include next-day view counts and probabilities of staying in the trending set. Table 3 shows example rows with high-confidence predictions across diverse videos.

Video ID	**View Count**	**Predicted Next Views**	**Prob. Stay Trending**
0UI_Gc7OYWc	288,166	288,166	1.00
5VYsnngkS_U	7,946,614	7,946,614	1.00
LuJpdvb5bDk	4,239,644	4,239,644	1.00
QKYFfYLe5rs	5,120,928	5,120,928	1.00
ppp7dMhzrz8	653,684	653,684	1.00

Table 3: Sample model predictions for next-day trending likelihood.

4.5 Dashboard Visualizations

We built simple dashboards to contextualize the curated data and model outputs for stakeholders. The top-20 charts surface which creators and categories dominate each region on a given day, making it easy to verify the coverage and data quality of the ingestion pipeline. The prediction panel overlays probability scores on recent trending videos so product, ads, and moderation teams can spot content likely to remain popular.

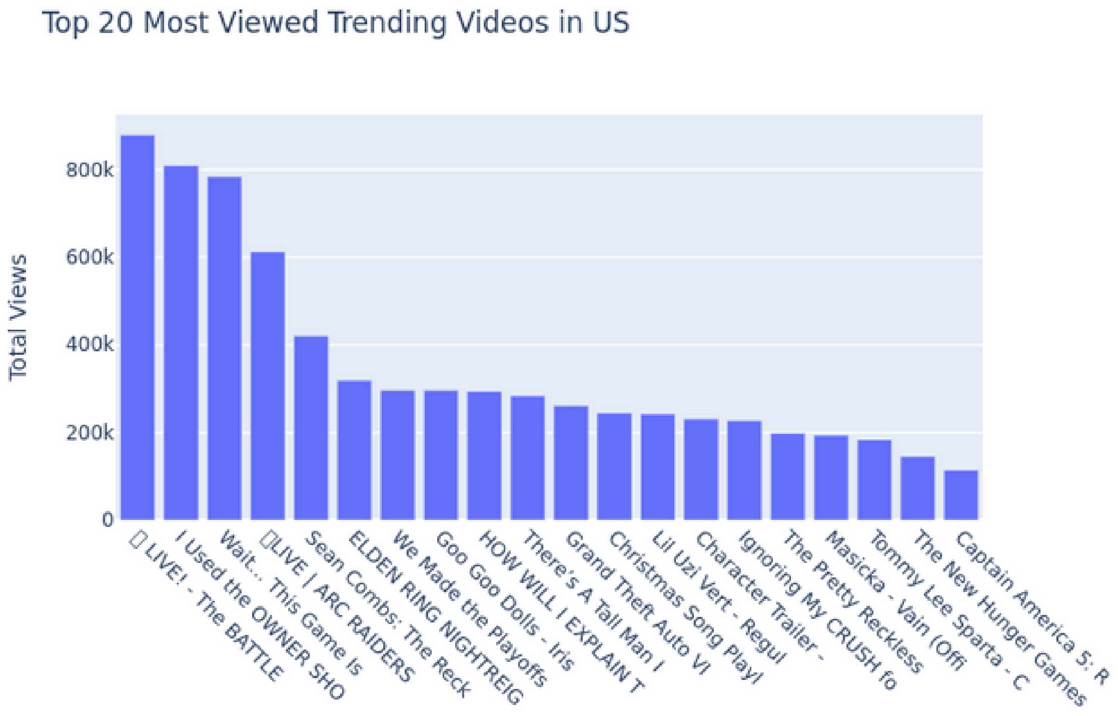


Figure 3: Top 20 most viewed trending videos in the United States, confirming ingestion coverage and regional partitioning.

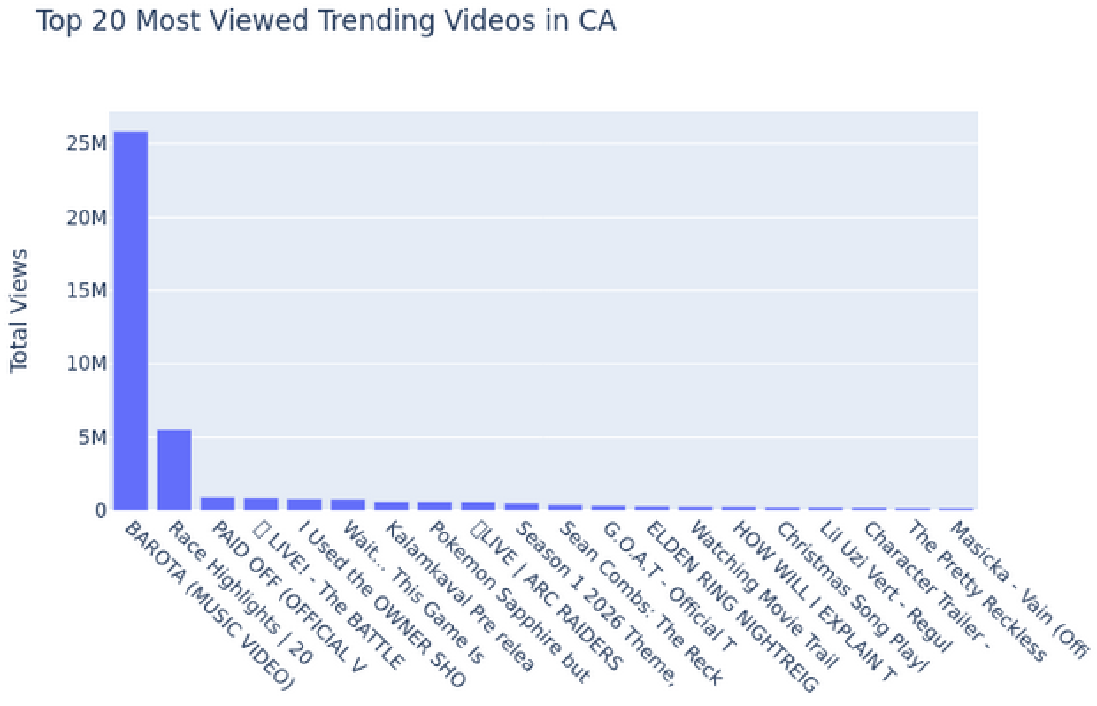


Figure 4: Top 20 most viewed trending videos in Canada, highlighting category distribution differences relative to the United States.

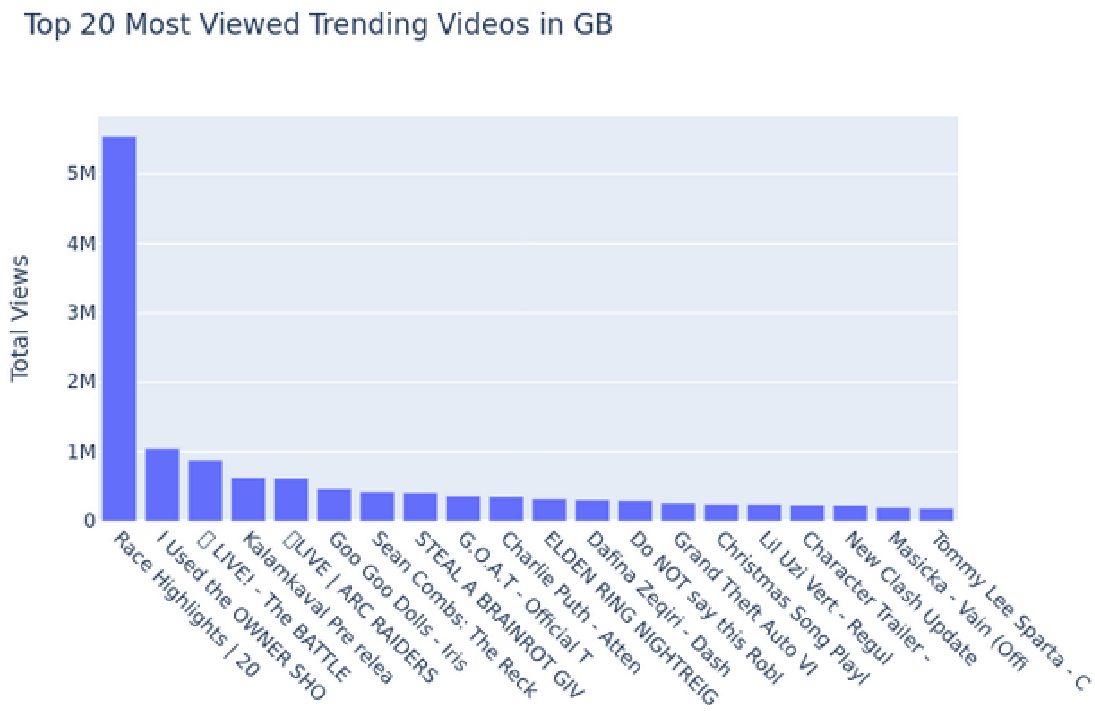
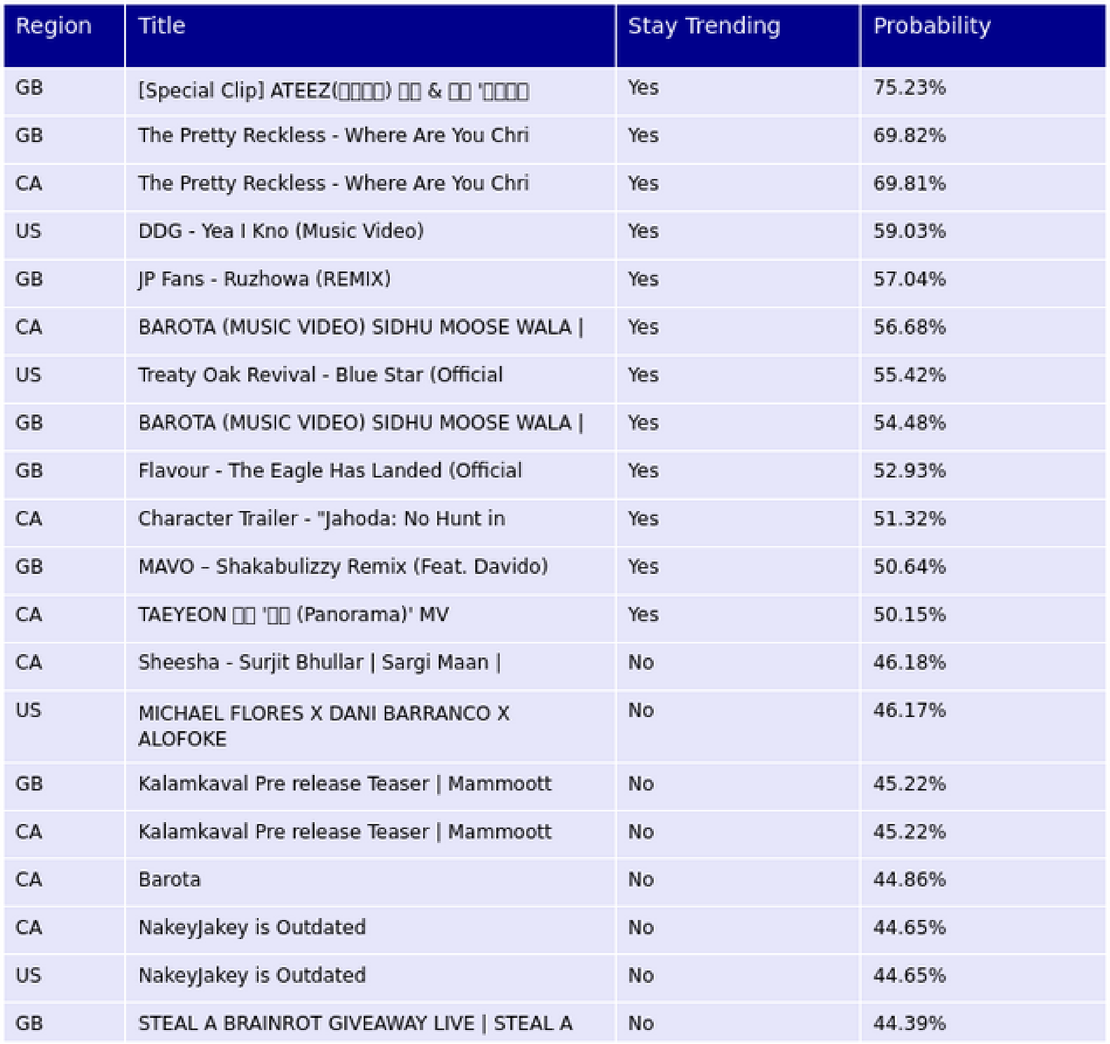


Figure 5: Top 20 most viewed trending videos in Great Britain, illustrating regional diversity and balanced coverage of music and entertainment.



Region	Title	Stay Trending	Probability
GB	[Special Clip] ATEEZ(□□□□) □□ & □□ '□□□□	Yes	75.23%
GB	The Pretty Reckless - Where Are You Chri	Yes	69.82%
CA	The Pretty Reckless - Where Are You Chri	Yes	69.81%
US	DDG - Yea I Kno (Music Video)	Yes	59.03%
GB	JP Fans - Ruzhowa (REMIX)	Yes	57.04%
CA	BAROTA (MUSIC VIDEO) SIDHU MOOSE WALA |	Yes	56.68%
US	Treaty Oak Revival - Blue Star (Official	Yes	55.42%
GB	BAROTA (MUSIC VIDEO) SIDHU MOOSE WALA |	Yes	54.48%
GB	Flavour - The Eagle Has Landed (Official	Yes	52.93%
CA	Character Trailer - "Jahoda: No Hunt in	Yes	51.32%
GB	MAVO – Shakabulizzy Remix (Feat. Davido)	Yes	50.64%
CA	TAEYEON □□ '□□ (Panorama)' MV	Yes	50.15%
CA	Sheesha - Surjit Bhullar | Sargi Maan |	No	46.18%
US	MICHAEL FLORES X DANI BARRANCO X ALOFOKE	No	46.17%
GB	Kalamkaval Pre release Teaser | Mammoott	No	45.22%
CA	Kalamkaval Pre release Teaser | Mammoott	No	45.22%
CA	Barota	No	44.86%
CA	NakeyJakey is Outdated	No	44.65%
US	NakeyJakey is Outdated	No	44.65%
GB	STEAL A BRAINROT GIVEAWAY LIVE | STEAL A	No	44.39%

Figure 6: Prediction dashboard showing next-day trending likelihood scores for currently trending videos.

4.6 Operational Considerations

Using serverless ingestion and managed Spark (Glue) minimized operational burden. Most spend comes from Glue job runtime and NAT egress during package downloads; partition pruning and Parquet compression reduce Athena costs. The workflow's dependency graph lowers failure domains by sequencing ETL stages.

5 Team Contributions

- **Kevin Bell**: VPC/network design and Lambda + EventBridge integration; implemented trending ETL, curated schema definitions, and presentation materials documenting the workflow and infrastructure.

- **Jacob Child**: Led architecture for comments ingestion, sentiment processing, and feature/label engineering; authored the backfill job that joins sentiment with trending metrics and produces ML-ready datasets; supported model training and prediction stages.

Workload was split evenly, with paired code reviews on each ETL milestone and shared notebook-based data validation.

6 Conclusion and Future Work

We delivered an automated, cloud-native YouTube analytics pipeline that ingests daily trending data, curates clean datasets, enriches them with sentiment, engineers predictive features, and outputs next-day trending predictions. The architecture balances scalability and cost, and the modular ETL jobs simplify maintenance. Future enhancements include incremental model retraining with drift detection, additional NLP features from titles/descriptions (e.g., embeddings) and toxicity detection on comments, near–real-time micro-batching for faster trend detection, and expanded dashboards for stakeholder-facing monitoring.