

Serverless Social Media Analytics using AWS Kinesis

*A Course Project Report Submitted in partial fulfillment of the course requirements
for the award of grades in the subject of*

CLOUD BASED AIML SPECIALITY (22SDCS07A)

by

Mikkilineni Namitha

2210030074

Under the esteemed guidance of

Ms. P. Sree Lakshmi

Assistant Professor,

Department of Computer Science and Engineering



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

K L Deemed to be UNIVERSITY

*Aziznagar, Moinabad, Hyderabad,
Telangana, Pincode: 500075*

April 2025

K L Deemed to be UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



Certificate

This is Certified that the project entitled “**Serverless Social Media Analytics using AWS Kinesis**” which is a **Experimental** work carried out by **Mikkilineni Namitha (2210030074)**, in partial fulfillment of the course requirements for the award of grades in the subject of **CLOUD BASED AIML SPECIALITY**, during the year **2024-2025**. The project has been approved as it satisfies the academic requirements.

Ms.P.Sree Lakshmi

Course Coordinator

Dr. Arpita Gupta

Head of the Department

Ms. P. Sree Lakshmi

Course Instructor

CONTENTS

Page No.

1. Introduction	1
2. AWS Services Used as part of the project	2
3. Steps involved in solving project problem statement	5
4. Stepwise Screenshots with brief description	6
5. Learning Outcomes	12
6. Conclusion	13
7. References	14

1. Introduction

The Social media platforms like Twitter, Facebook, and Instagram are constantly generating large volumes of data every second. This data can be very valuable, it can show what topics are trending, how people feel about certain events, or how brands are being discussed online. But analyzing such a huge amount of fast-moving data in real time can be challenging, especially with traditional systems that require a lot of setup and server management.

To solve this problem, this project uses a serverless architecture offered by Amazon Web Services (AWS). Serverless means there is no need to manage physical or virtual servers—everything runs in the cloud and scales automatically depending on the workload. This makes it easier, faster, and cheaper to build applications.

The main component used for real-time data collection is Amazon Kinesis. It is a service designed to handle large streams of data and pass them along for processing. When new data flows in, AWS Lambda functions are triggered automatically to process that data. Lambda allows code to run in response to events, and it handles all the compute resources in the background without any manual setup.

Once the data is processed, it needs to be stored in a way that can be easily accessed for analysis or visualization. For this, Amazon DynamoDB is used—a fully managed NoSQL database that can quickly store and retrieve data. It's a great fit for serverless applications because it scales on its own and handles large amounts of data smoothly.

Security and access control are also important. This is handled using AWS Identity and Access Management (IAM), which helps set rules and permissions to control who can access or change the system's resources.

To set everything up and manage the services more efficiently, the AWS Command Line Interface (CLI) is used. It allows users to configure settings, run commands, and interact with AWS services directly from a terminal window.

By combining these AWS services, the system can collect, process, and analyze real-time social media data without the need to manage any servers. It's highly scalable, cost-effective, and ideal for gaining insights from social media streams as they happen.

2. Aws Services Used As Part Of The Project

The project utilizes several AWS services to build a serverless architecture:

1. Amazon Kinesis

- Collects and ingests real-time social media data streams.
- Acts as the backbone for handling continuous and high-volume data input [2].
- Integrates with Lambda for automatic data processing upon arrival.



2. AWS Lambda

- Processes each record received from Kinesis in real-time [1].
- Extracts keywords, hashtags, and sentiment from social media posts.
- Automatically scales with the volume of incoming data, requiring no manual server management.



3. Amazon S3

- Stores raw or processed social media data streamed via Amazon Kinesis for further analysis or archival.
- Serves as a durable and scalable sink for logs, sentiment results, and media files generated during analytics.
- Enables integration with downstream analytics services like Amazon Athena for querying stored insights.



4. AWS Identity and Access Management (IAM)

- Grants secure permissions for Lambda to read from Kinesis and write to DynamoDB [4].
- Defines roles and policies to manage access between AWS services securely.



5. AWS Command Line Interface (CLI)

- Used to set up and manage all AWS resources quickly via terminal [5].
- Helps in deploying Lambda functions, configuring Kinesis streams, and testing the setup.



6. Amazon QuickSight

- For visualizing analytics results like top hashtags, user activity heatmaps, sentiment trends, etc.
- Connects to S3 and generates dashboards without much setup.



These services together form a scalable, event-driven, and fully serverless architecture capable of analyzing real-time social media data efficiently and securely.

3. Steps Involved in Solving the Project Problem Statement

- **Create a Kinesis Data Stream**

Create a Kinesis Data Stream named social-media-stream in On-Demand mode, which enables automatic scaling based on the volume of incoming data.

- **Send Messages from dataset to Kinesis**

Write a Python script using boto3 to send sample social media messages to the stream using the put_record() method for testing real-time ingestion.

- **Create First Lambda Function (Stream Processing)**

Set up a Lambda function triggered by the Kinesis stream. In this function, extract the message content from each record and store it into an S3 bucket under the prefix tweets/ for further processing[3].

- **Attach IAM Role to Lambda**

Attach an IAM role to the Lambda function with the policies: AmazonKinesisReadOnlyAccess, AmazonS3FullAccess, ComprehendReadOnlyAccess, and CloudWatchLogsFullAccess to allow access to required services.

- **Create Second Lambda Function (Hashtag & Sentiment Analysis)**

Create another Lambda function that is triggered when new objects are added to the tweets/ folder in the S3 bucket. This function should:

- Read each message,
- Extract hashtags from the text,
- Perform sentiment analysis using Amazon Comprehend,
- Store the analysis results (including hashtag, sentiment, and date) into the processed/ folder in the same S3 bucket.

- **Connect Amazon QuickSight to S3**

Open Amazon QuickSight, create a new dataset using the S3 bucket as the data source, and point it to the processed/ folder where the sentiment analysis results are stored.

- **Build Dashboards in QuickSight**

Design interactive dashboards in QuickSight to visualize:

- Sentiment trends over time,
- Popular hashtags and their corresponding sentiments,
- Additional insights such as volume of posts or daily sentiment shifts.

4. Stepwise Screenshots with Brief Description

Step 1: Creating the Kinesis Stream

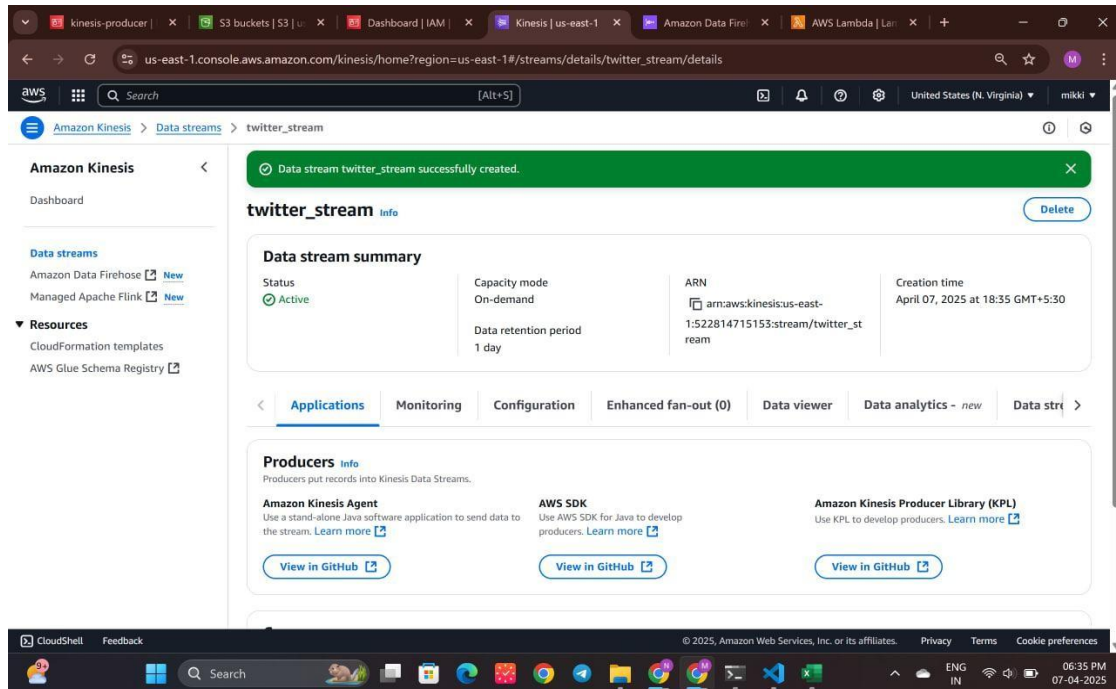


Fig. 4.1: Creating Kinesis Data Stream

This screenshot shows the creation of a Kinesis data stream named social-media-stream in on-demand mode. It enables automatic scaling to handle variable real-time data flow.

Step 2: Sending Data to the Stream

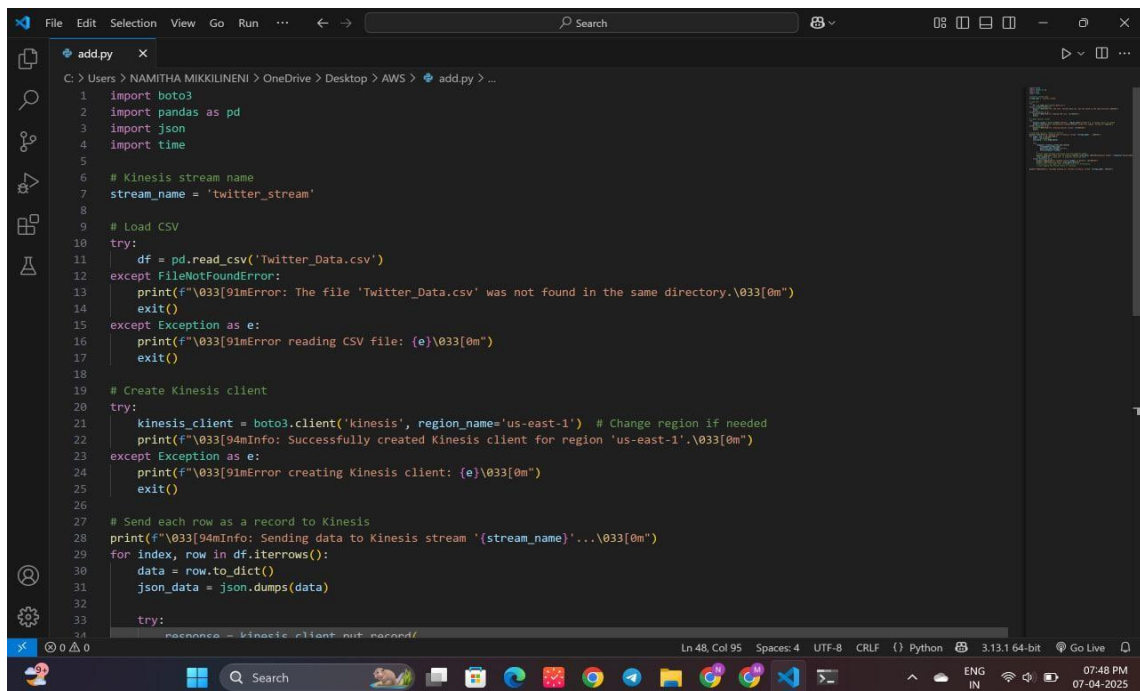


Fig. 4.2: Writing Python Script to Push Data

```
Command Prompt - python t x + v
Microsoft Windows [Version 10.0.26100.3624]
(c) Microsoft Corporation. All rights reserved.

C:\Users\NAMITHA MIKKILINENI>cd OneDrive
C:\Users\NAMITHA MIKKILINENI\OneDrive>cd Desktop
C:\Users\NAMITHA MIKKILINENI\OneDrive\Desktop>cd AWS
C:\Users\NAMITHA MIKKILINENI\OneDrive\Desktop\AWS>aws configure
AWS Access Key ID [*****YIS7]:
AWS Secret Access Key [*****z4y]:
Default region name [us-east-1]:
Default output format [json]:

C:\Users\NAMITHA MIKKILINENI\OneDrive\Desktop\AWS>python add.py
Info: Successfully created Kinesis client for region 'us-east-1'.
Info: Sending data to Kinesis stream 'twitter_stream'...
✓Tweet sent to Kinesis: {"clean_text": "when modi promised \u201cminimum government maximum governance\u201d expected him begin th
e difficult job reforming the state why does take years get justice state should and not business and should exit psus and temples",
"category": -1.0} (Sequence Number: 49662148407209441156277971776036930088947885671750565938)
✓Tweet sent to Kinesis: {"clean_text": "talk all the nonsense and continue all the drama will vote for modi ", "category": 0.0} (S
equence Number: 49662148407209441156277971776151778041811275512067129394)
✓Tweet sent to Kinesis: {"clean_text": "what did just say vote for modi welcome bjp told you rahul the main campaigner for modi t
hink modi should just relax", "category": 1.0} (Sequence Number: 49662148407209441156277971776414114944667650111697846322)
✓Tweet sent to Kinesis: {"clean_text": "asking his supporters prefix chowkidar their names modi did great service now there confus
ion what read what not now crustal clear what will crass filthy nonsensical see how most abuses are coming from chowkidars", "categ
ory": 1.0} (Sequence Number: 49662148407209441156277971776813060465140477945509314610)
✓Tweet sent to Kinesis: {"clean_text": "answer who among these the most powerful world leader today trump putin modi may ", "categ
ory": 1.0} (Sequence Number: 49662148407187140411079441153949577685379194362905231394)
✓Tweet sent to Kinesis: {"clean_text": "kiya tho refresh maarkefir comment karo ", "category": 0.0} (Sequence Number: 496621484072
09441156277971777300257570445173571635380274)
✓Tweet sent to Kinesis: {"clean_text": "surat women perform yagna seeks divine grace for narendra modi become again\n", "category"
: 0.0} (Sequence Number: 49662148407142538920682379908076332101683256998838140930)
✓Tweet sent to Kinesis: {"clean_text": "this comes from cabinet which has scholars like modi smriti and hema time introspect", "ca
tegory": 0.0} (Sequence Number: 49662148407187140411079441154601188702151479694230290466)
✓Tweet sent to Kinesis: {"clean_text": "with upcoming election india saga going important pair look current modi leads govt electe
d with deal brexit combination this weekly looks juicy bears imho ", "category": 1.0} (Sequence Number: 496621484072094411562779717
```

Fig. 4.3: Running Python File

A Python script is written using Boto3 to push social media messages into the Kinesis stream using the `put_record()` method. This simulates real-time data ingestion.

Step 3: Creating the First Lambda Function

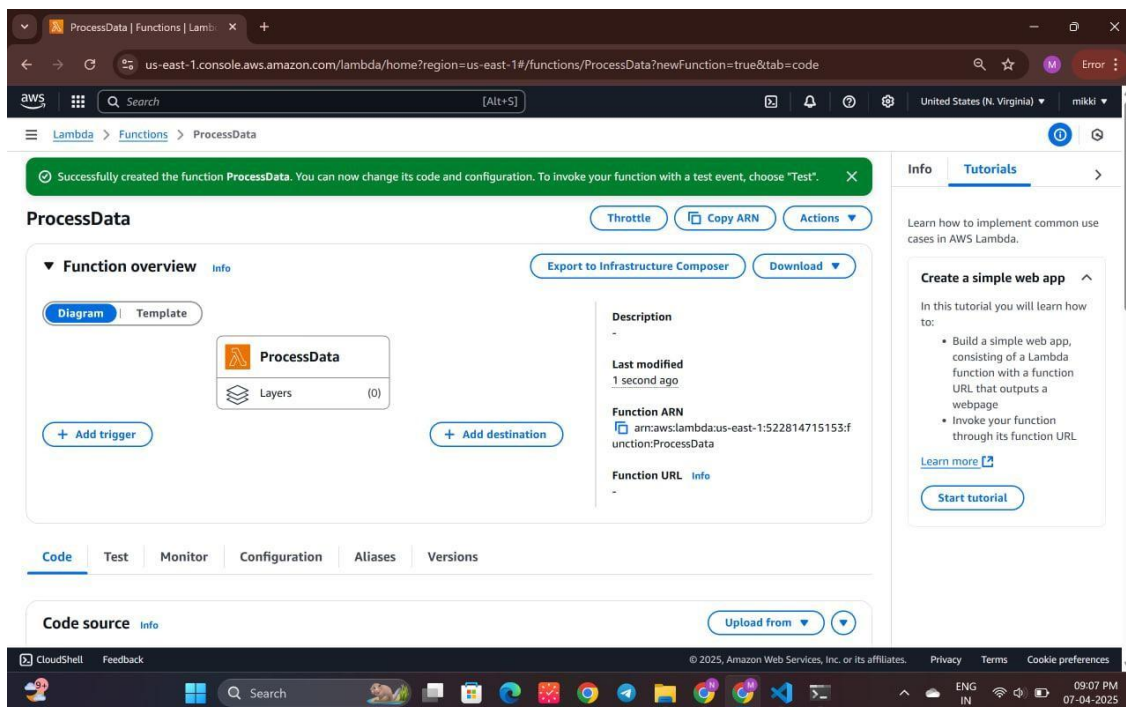


Fig. 4.4: Creating Lambda Function "Process Data"

A Lambda function named "Process Data" is created to consume and process the records arriving from the Kinesis stream.

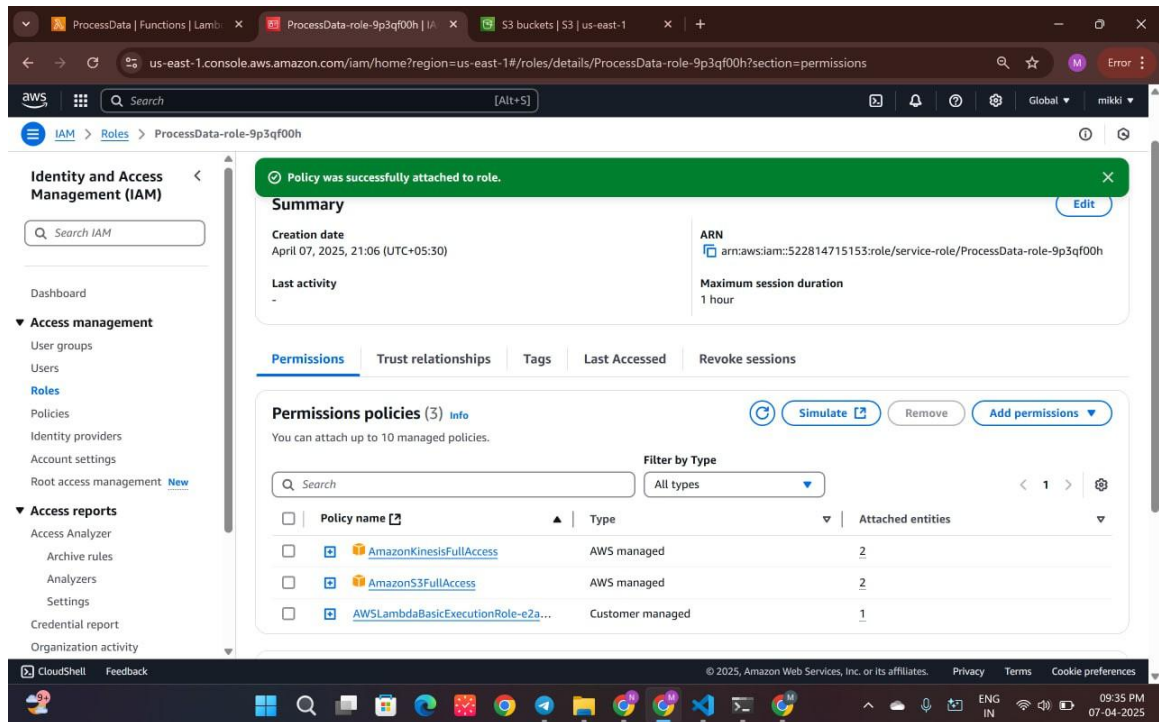


Fig: 4.5: Attaching IAM Policies to Lambda Role

IAM policies like AmazonKinesisFullAccess and AmazonS3FullAccess are attached to the Lambda role to grant necessary permissions.

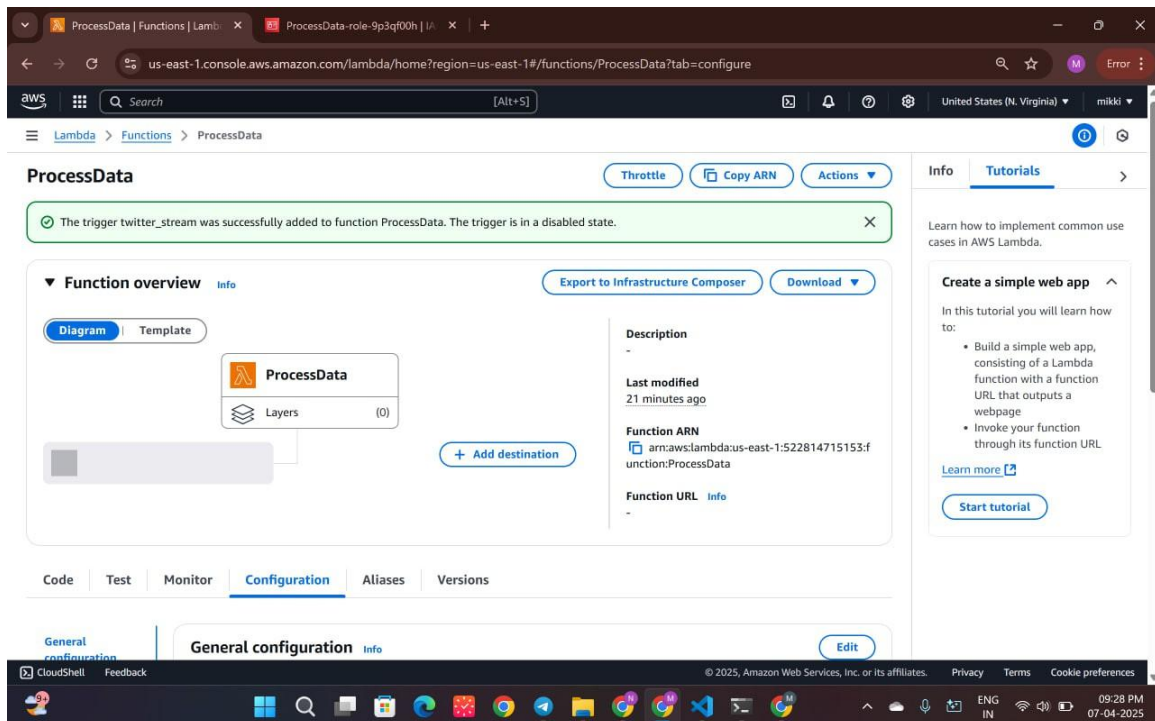


Fig: 4.6: Adding Trigger to Lambda Function

The Lambda function is configured with a trigger to automatically respond to events in the social-media-stream.

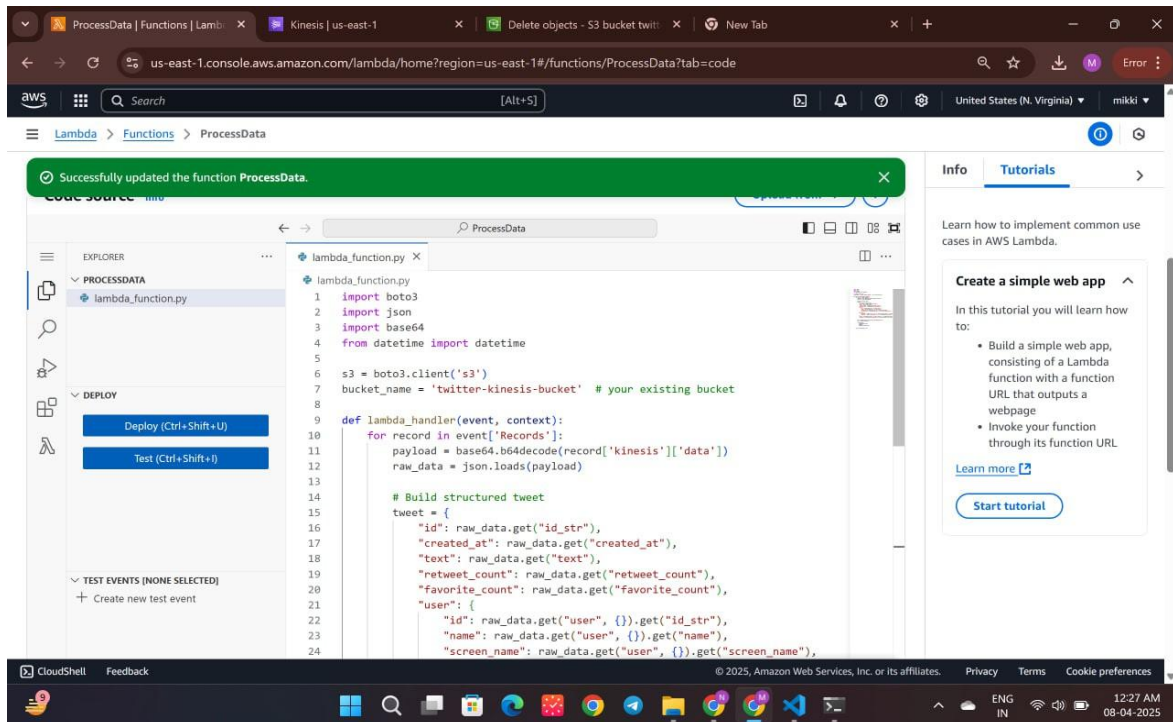


Fig: 4.7: Updating the Lambda Function

This image displays the Lambda function code being updated to include logic for data processing and storage.

Step 4: Verifying Data in S3

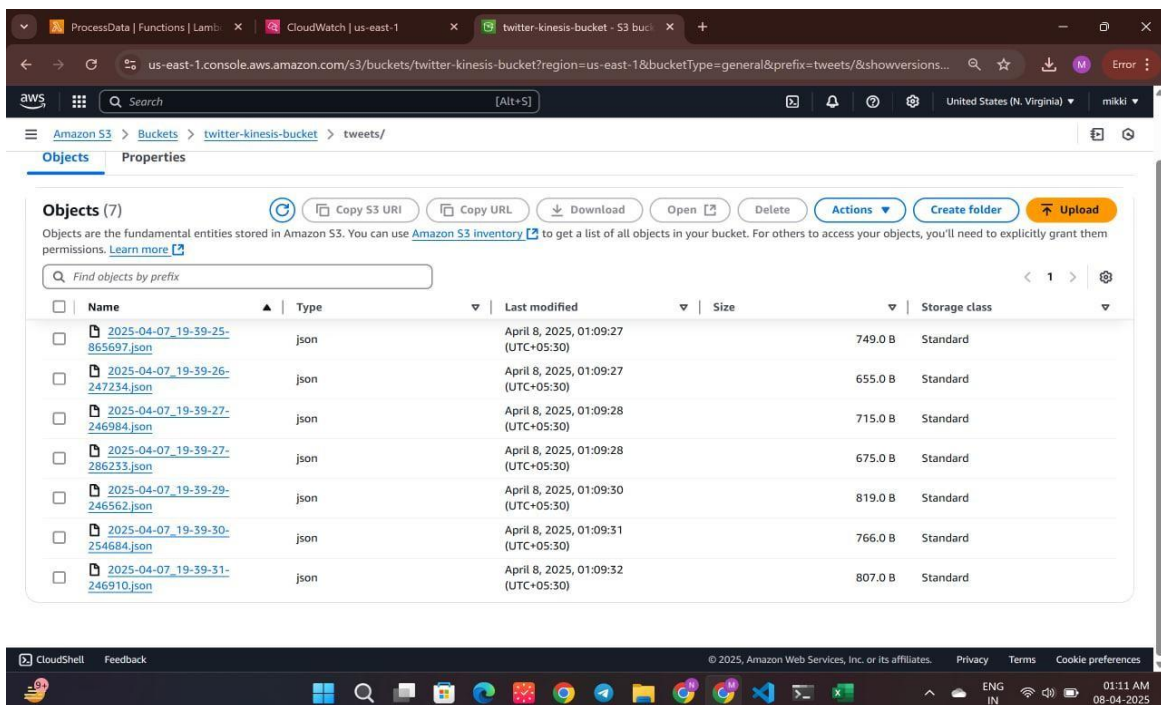


Fig: 4.8: Data Stored in S3 Bucket "twitter-kinesis-bucket"

The screenshot shows that the incoming tweet data is successfully being saved in the S3 bucket under the tweets/ prefix.

Step 5: Visualizing Data with Amazon QuickSight

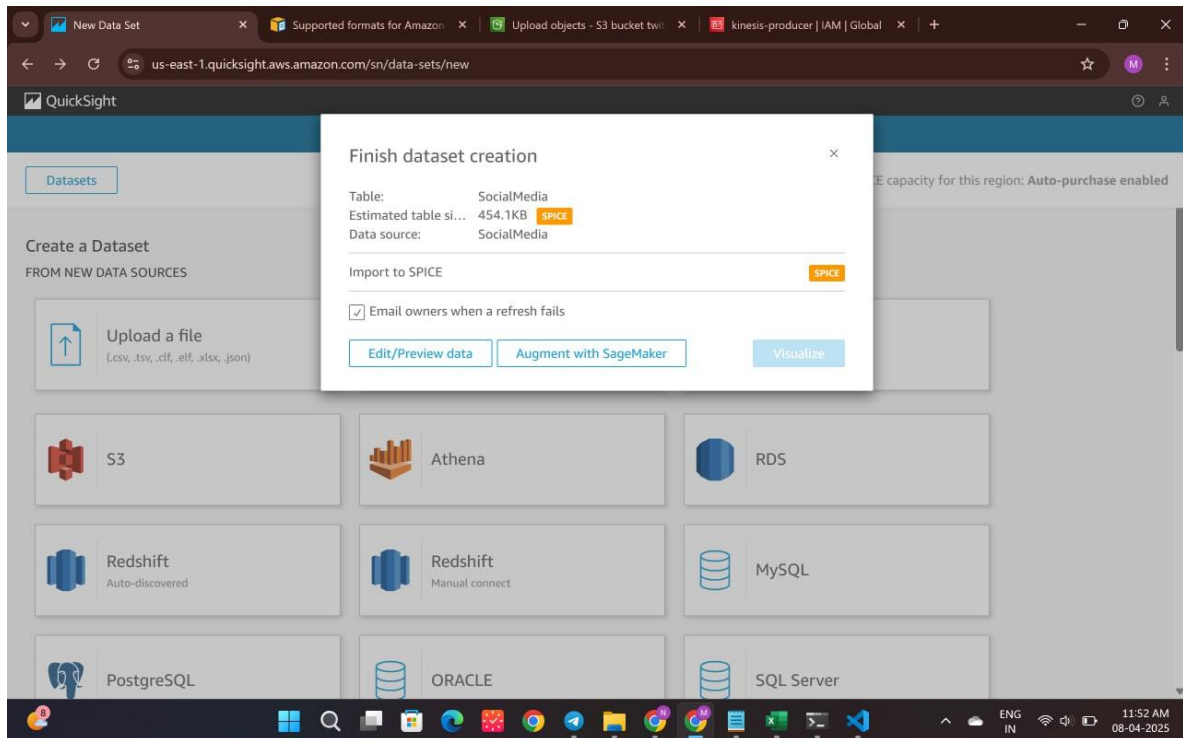


Fig: 4.9: Connecting S3 Bucket to QuickSight

Amazon QuickSight is connected to the S3 bucket as a data source for visualization of analytics results.

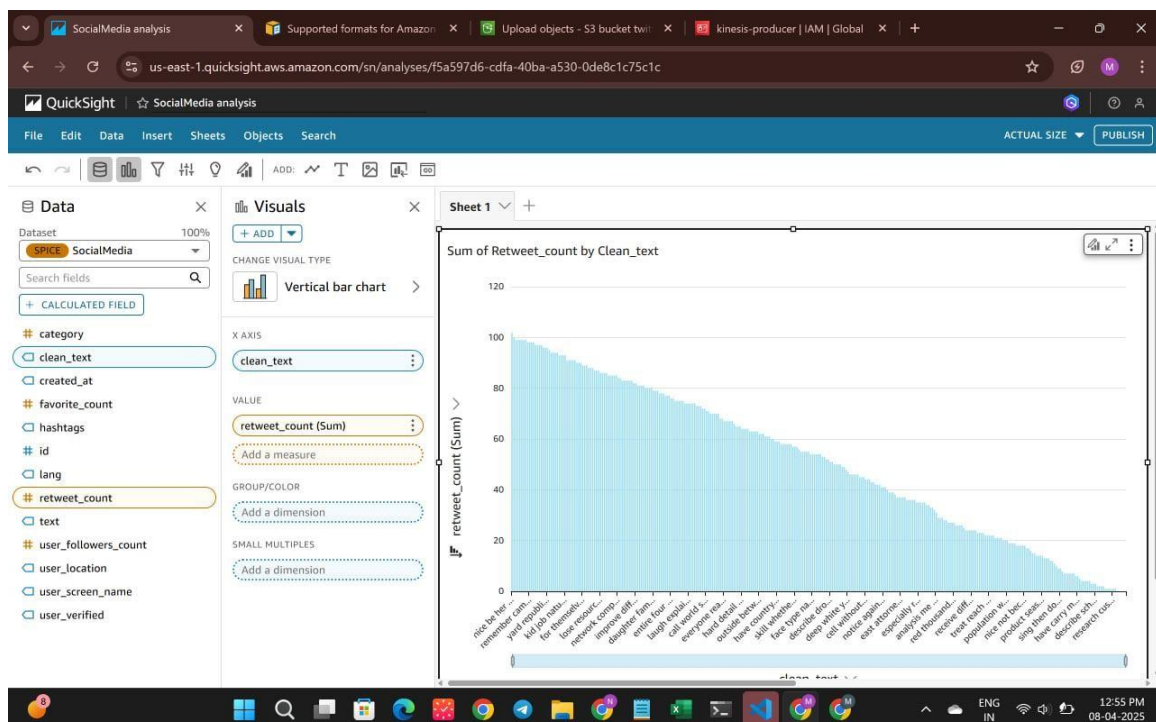


Fig: 4.10: Trending Tweets Based on Retweet Count

This dashboard in QuickSight displays the most trending tweets, ranked by their retweet count.

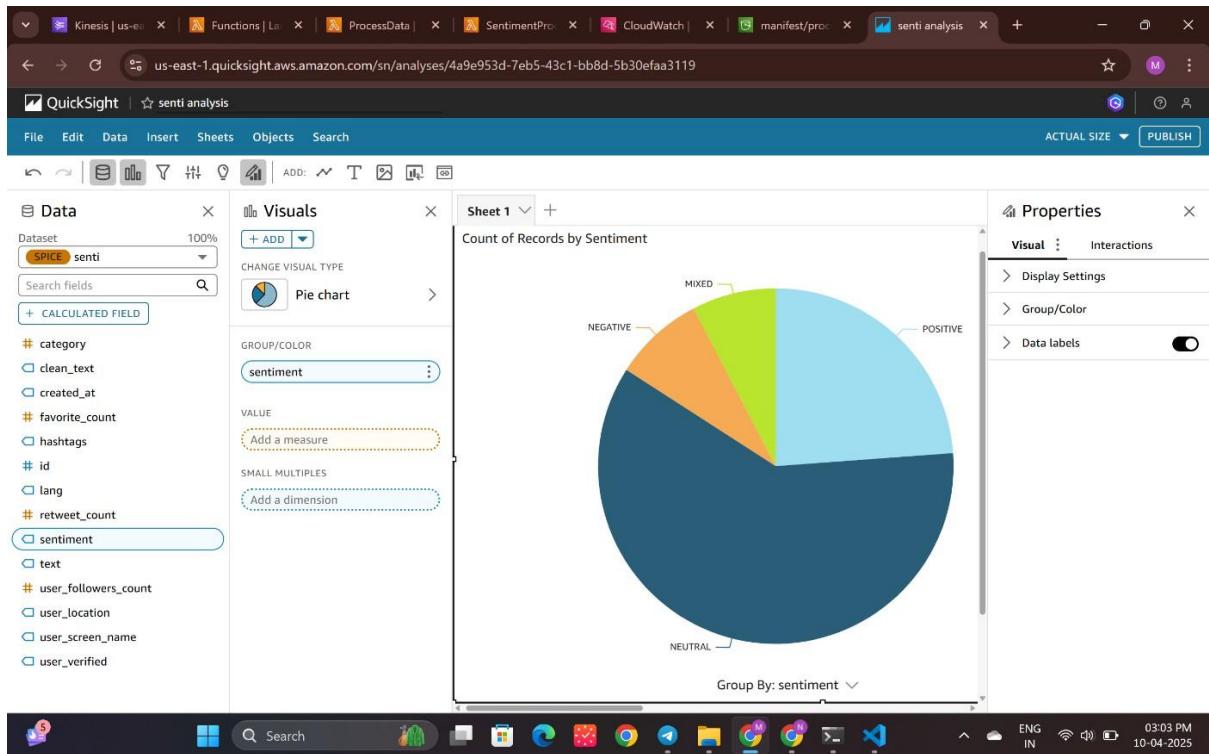


Fig: 4.11: Sentiment Analysis Summary

The sentiment analysis results are visualized here, showing the distribution of positive, negative, and neutral tweets.

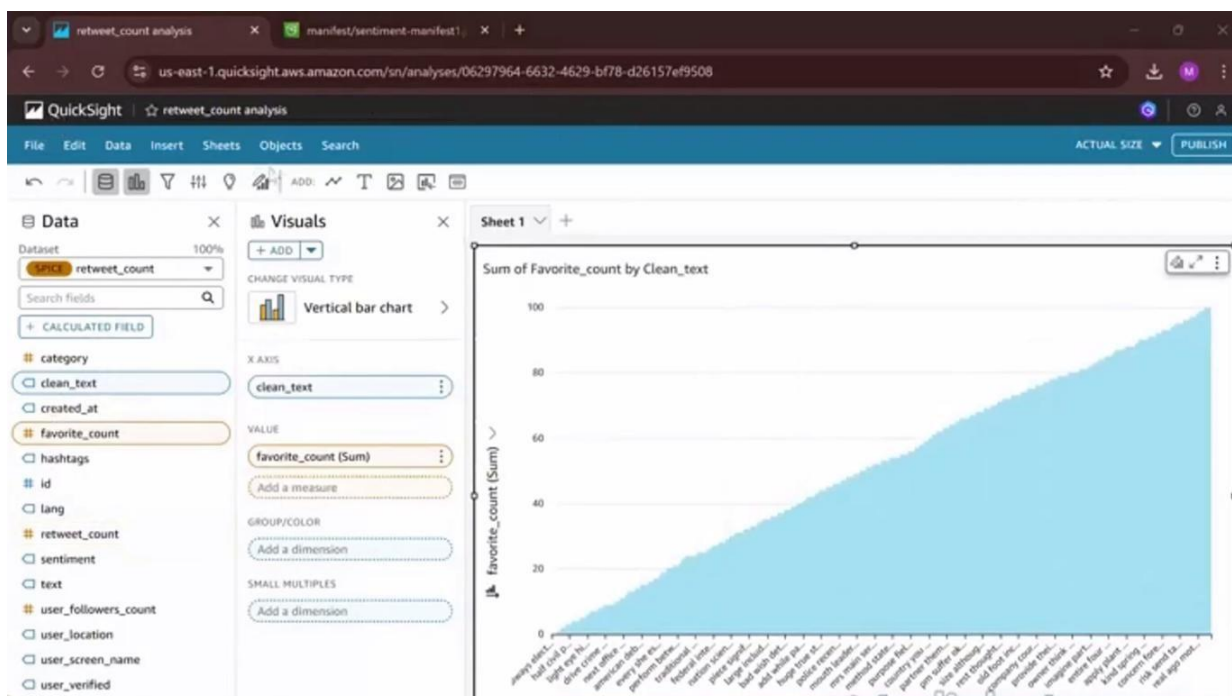


Fig: 4.12: Most Liked Tweets by User

This visualization shows the most liked tweets categorized by individual users, offering insight into user engagement.

5. Learning Outcomes

This project provided hands-on experience in building a serverless social media analysis pipeline using multiple AWS services. The following key outcomes were achieved:

- Gained practical knowledge of AWS Lambda for event-driven processing and deploying scalable functions.
- Learned to configure Amazon Kinesis Data Streams for real-time social media data ingestion.
- Understood how to manage and trigger actions using Amazon S3 bucket structures.
- Applied IAM roles and permissions to securely manage service-to-service communication.
- Developed skills in using the AWS CLI for service configuration and automation.
- Explored Amazon Comprehend for sentiment analysis and integrated the results into QuickSight dashboards [6].
- Understood how to design a real-time, serverless data pipeline without traditional infrastructure.
- Implemented automated event-based triggers using S3 and Lambda.
- Learned to extract hashtags and analyze sentiments from unstructured text.
- Monitored and debugged workflows using CloudWatch Logs and metrics.
- Organized data effectively using S3 prefixes (tweets/, processed/) for clean separation.
- Integrated AI services like Comprehend without manual model deployment.
- Traced the complete data journey from ingestion to final visualization.
- Strengthened understanding of cloud-native data engineering concepts.
- Recognized the value of real-time sentiment tracking for social media insights.

These outcomes reflect a solid grasp of serverless architecture, real-time data integration, and the use of AWS tools for scalable and intelligent automation.

6. Conclusion

This project used AWS services like Kinesis, Lambda, S3, and Comprehend to build a serverless system that can analyze social media data in real time. It collects data, processes it automatically, and stores it without needing to manage any servers. The results, like trending hashtags and tweet sentiments, are shown using Amazon QuickSight. The system is fast, easy to scale, and secure. It also saves time and cost by handling everything in the cloud. Through this project, I learned how to connect different AWS services and create a working data pipeline from start to finish. This kind of setup can be very useful for companies to understand what people are saying online and make better decisions. I also understood how event-based automation works using Lambda triggers. Overall, the project gave me hands-on experience in cloud computing and real-time analytics.

7. References

[1] AWS Lambda Developer Guide:

<https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>

[2] Amazon Kinesis Documentation

https://docs.aws.amazon.com/ru_ru/kinesis/

[3] Amazon Simple Storage Service Documentation

https://docs.aws.amazon.com/s3/?icmpid=docs_homepage_featuredsvcs

[4] AWS IAM Documentation:

<https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html>

[5] Configuring settings for the AWS CLI

<https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-configure.html>

[6] Amazon QuickSight Documentation

<https://docs.aws.amazon.com/quicksight/>