RAGHU INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Accredited by NAAC A+ and NBA, Affiliated to JNTU-GV(Vizianagaram)
Dakamarri (V), Bheemunipatnam (M),
Visakhapatnam

AWS-SERVERLESS FEEDBACK SYSTEM: AUTOMATING ANALYSIS AND VISUALIZATION

Presented By – BATCH 18 (CSE-D)

1. CH.Jahnavi(213J1A0532)
2. Ch.Pranay kumar (213J1A0526)
3. G.sai teja (213J1A0553)
1. G.Maruthi sai (213J1A0547)

Under the estimated guidance of,
MR. B.S.V REDDY ,
Associate Professor

ABSTRACT

Serverless application for feedback storage and sentiment analysis using AWS is a comprehensive project designed to automate the collection, analysis, and visualization of user feedback using cutting-edge AWS services. The application leverages AWS Lambda for serverless computing, ensuring that the system automatically scales based on demand without requiring manual intervention. AWS API Gateway enables secure and efficient handling of RESTful API methods, allowing users to submit feedback and retrieve processed data in real time. The feedback is analyzed using AWS Comprehend, which performs sentiment analysis, categorizing the feedback as positive, negative, or neutral based on natural language processing (NLP) techniques. The results are then stored in AWS DynamoDB, a NoSQL database that provides high availability and scalability, ensuring seamless handling of large data volumes.

One of the key features of this system is real-time data visualization, achieved through Google Charts, which displays user sentiment trends in the form of stacked bar charts. This allows administrators to easily track shifts in customer feedback and make informed decisions based on real-time insights. The entire architecture is serverless, significantly reducing operational costs using AWS's pay-as-you-go model. AWS Lambda ensures that compute resources are only used when necessary, while AWS DynamoDB provides fast and reliable storage without the need for manual scaling.

The use of AWS Comprehend's sentiment analysis adds an intelligent layer of data processing, turning raw feedback into actionable insights. Additionally, the system is designed to be highly secure, reliable, and efficient, providing businesses with a scalable solution to collect, analyze, and act on user feedback in real-time

Existing System

Traditional feedback systems rely on server-based architectures that require constant maintenance, manual scaling, and intervention, making them inefficient. Many of these systems lack real-time sentiment analysis and effective data visualization, limiting their ability to respond promptly to user feedback. Existing research has primarily focused on either data collection automation or sentiment analysis separately, rather than a fully integrated solution. Additionally, these systems struggle to handle large-scale, dynamic datasets, leading to performance bottlenecks. This gap presents an opportunity to develop a scalable, serverless feedback system with automated analysis and real-time visualization. High latency in processing and storing feedback data has been a significant challenge, preventing organizations from gaining instant insights. Traditional models require frequent retraining, consuming valuable time and resources. Security and compliance issues also arise due to the lack of robust encryption and access control mechanisms. Deploying sentiment analysis models in production often requires extensive DevOps support, increasing operational costs. These challenges highlight the need for an automated, scalable, and real-time sentiment analysis system that enhances efficiency while reducing computational complexity.

Proposed System

The proposed system is a fully serverless architecture that automates feedback collection, sentiment analysis, and real-time visualization using AWS services. AWS Lambda ensures automatic feedback processing without requiring dedicated servers, reducing infrastructure costs. Sentiment analysis is performed in real time using AWS Comprehend, classifying feedback as Positive, Negative, or Neutral. The analyzed data is stored in AWS DynamoDB, ensuring scalability and quick retrieval. Real-time visualization is achieved through Google Charts, allowing administrators to track sentiment trends instantly. AWS AppSync provides live data updates, ensuring that decision-makers always have the latest feedback insights. This system improves efficiency, scalability, and security while minimizing operational overhead.

LITERATURE SURVEY-1

TITLE:"Using Machine Learning and Thematic Analysis Methods to Evaluate Mental Health Apps Based on User Reviews"
Author: Oladope Oyebode
Date of publication: May 11, 2020

1. **Algorithms and Methods Used:**

Sentiment Analysis Using Machine Learning :Five supervised machine learning classifiers : Support Vector Machine (SVM),Multinomial Naïve Bayes (MNB),Stochastic Gradient Descent (SGD),Logistic Regression (LR),Random Forest (RF).To predict user reviews are positive or negative.
Natural Language Processing (NLP):removing punctuation,slangs, and stopwords,as well as lemmatizing words to their root forms.The Term Frequency-Inverse Document Frequency (TF-IDF) method was used to vectorize the data for machine learning.
Thematic Analysis:Themes were categorized into areas like usability, content quality, customer support, and cost.

2. Result and Conclusion: We performed sentiment analysis using machine learning (ML) approach to understand users' opinion regarding mental health apps with the aim of uncovering positive and negative sentiments. Specifically, we implemented and compared the performance of five ML classifiers, and then used the best performing classifier, with F1-score of 89.42%, to predict the sentiment polarity of user reviews.

As part of future work, we plan to extend our approach to apps in other domains to uncover their strengths and weaknesses, and then offer design suggestions that app developers and researchers can adopt to improve the quality and effectiveness of their apps.

3.**Cons**:

Algorithmic Limitations:Some algorithms like Logistic Regression and Random Forest performed marginally lower than SGD, which could mean that certain nuances of the data were not captured equally across all classifiers.

Data Limitations:The dataset consists only of user reviews from Google Play and App Store, limiting the analysis to the opinions of those willing to leave reviews. This introduces a potential bias in the analysis.

Subjectivity in Thematic Analysis:Thematic analysis involves subjective interpretation, and although automated tools like NVivo are used, there can still be biases in how themes are categorized or interpreted.

Imbalanced Data:The training data needed to be balanced due to a higher number of positive reviews compared to negative ones, which might have impacted the classification model's accuracy on real-world unbalanced data.These methods help in determining the effectiveness and user satisfaction of mental health apps, but the mentioned limitations can impact the generalizability and depth of the findings.

MERITS AND DEMERITS

Merits	Demerits
1. Comprehensive analysis using multiple classifiers.	1. Limited to only user reviews, may not capture entire user experience.
2. High accuracy in sentiment prediction.	2. Does not consider non-textual app features like usability and design.
3. Provides actionable insights with thematic analysis.	3. Potentially biased dataset depending on the user base of the apps reviewed.

LITERATURE SURVEY-2

TITLE:” Understanding User Perspectives on Data Visualization in mHealth Apps”
Author: **“** Yasmeen Anjeer Alshehhi”
Date of publication: 12 July 2023

1. **Algorithms and Methods Used:**

Data Visualization Techniques: The study emphasizes **data visualization** in mHealth apps and evaluates how different types of visual representations (bar charts, pie charts, line charts, etc.) are received by users. The visualizations are not based on advanced machine learning algorithms but instead focus on user interaction with simple chart types like bar and pie charts.
Survey-based Analysis: The project uses a survey methodology to gather user feedback. No complex algorithms like sentiment analysis or machine learning classifiers are used here. Instead, the research relies on user surveys and thematic analysis to understand preferences and challenges related to data visualization in mHealth apps.

2. Result and Conclusion : This paper presents a survey aimed at exploring user needs, challenges, and goals related to data visualisations in mHealth apps. The study employed the well-known value proposition canvas framework, which consists of three components that cover user tasks, pains, and gains. We have collected 56 complete responses, and our results show that bar and pie charts are more favoured as they are simple and easy to use.

3.**Cons :**

Sample Size: The survey had only **56 participants**, which may limit the generalizability of the findings. A larger sample size could provide a more representative view of user needs and preferences.

Subjective Bias: Since the study relies heavily on self-reported data from users, there could be **subjective biases**. Users' perceptions and personal experiences might vary significantly, which can affect the accuracy of the conclusions .

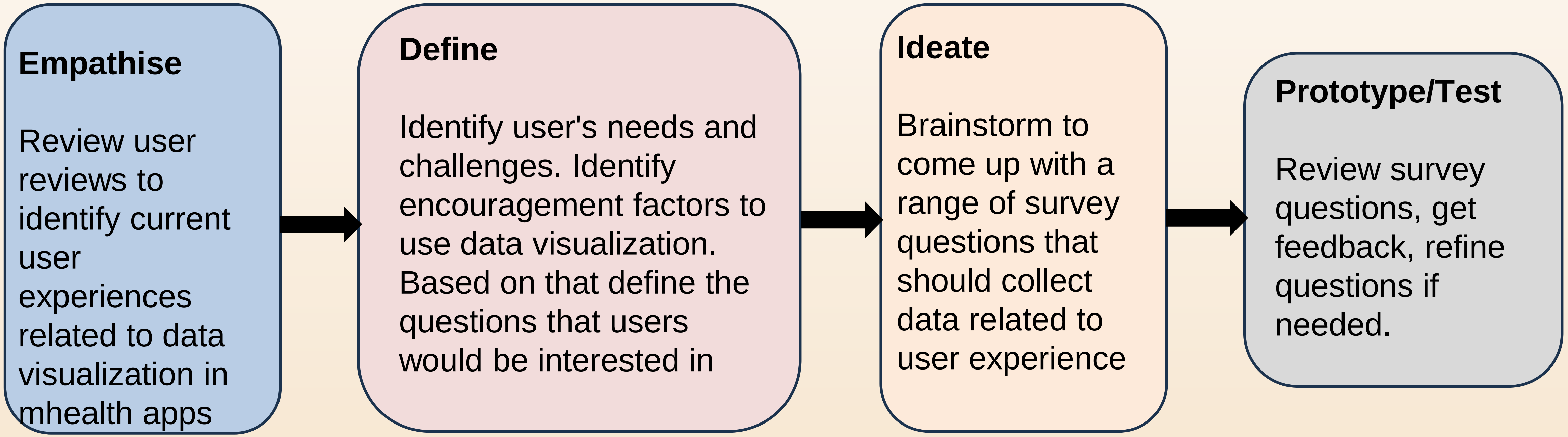
Limited Focus on Algorithms: Unlike more complex studies that use advanced algorithms (like machine learning or AI for prediction and analysis), this project focuses primarily on **user experience** and **data presentation**, with fewer insights into algorithmic improvements in mHealth apps.

Design Gaps: There are still **gaps in understanding** how non-expert users interpret different types of visual data, especially on small mobile screens. The study acknowledges this as a challenge in designing mHealth apps that cater to a diverse audience with varying levels of data literacy.

MERITS AND DEMERITS

Merits	Demerits
1. Addresses user interaction with visual data.	1. Survey-based results might not be generalizable to all user groups.
2. Identifies preferred visual formats.	2. Limited by the number of participants (56), which may affect the validity of results.
3. Highlights specific user challenges in real-time use.	3. Focuses only on specific chart types, potentially overlooking emerging or less common visualizations.

DIAGRAM



Empathise

Review user reviews to identify current user experiences related to data visualization in mhealth apps

Define

Identify user's needs and challenges. Identify encouragement factors to use data visualization. Based on that define the questions that users would be interested in

Ideate

Brainstorm to come up with a range of survey questions that should collect data related to user experience

Prototype/Test

Review survey questions, get feedback, refine questions if needed.

LITERATURE SURVEY-3

TITLE:” “Harnessing Customer Feedback for Product Recommendations: An Aspect-Level Sentiment Analysis Framework”
Author: Nimesh Bali Yadav
Date of publication: 27 March 2023

1. **Algorithms and Methods Used:**

Aspect-Based Sentiment Analysis (ABSA):ABSA identifies specific features (aspects) of a product (e.g., price, battery, screen) from customer reviews and determines the sentiment associated with each aspect.
WordNet is used for identifying relevant aspects and their synonyms, helping categorize user comments based on these aspects.
Sentiment analysis is then performed on the categorized comments, classifying them as positive, negative, or neutral.
Rule-Based Sentiment Analysis: A rule-based model is used for sentiment classification, which processes the polarity of comments based on predefined human rules. This method classifies comments by counting positive and negative words and then assigns an overall sentiment to the comment.
Recommendation System: weighted score-based recommendation system and preference-based recommendation method is employed.

2.Result and conclusion: In this paper, the most cared aspects that has been considered are price, battery, colour, and screen for the cell-phonebased dataset. This dataset is an Amazon dataset, and it’s available to use. We have grouped the comments of each brand and product based on the most cared aspects. Then sentiment analysis is performed on those aspect-based comments and found whether the comments are positive, negative, and neutral. We then calculated the polarities for the same. Finally, for all brands, we have calculated the scores for each aspect. This helped to predict what products to be recommended to the customer based on their most cared aspects.

3. Cons:

Single Comment Limitation: he recommendation system has limitations when there is only a single review for a product. It assigns a 100% recommendation if the sentiment is positive, which may not be accurate without more context or data points.

Bias in Aspect Selection: The selection of aspects (price, battery, screen) is predefined, meaning that other important aspects that users care about may not be considered unless explicitly identified in the system. This may lead to incomplete sentiment analysis for certain products.

MERITS AND DEMERITS

Merits	Demerits
1. Targeted approach using LSTM for specific feedback.	1. Limited to textual feedback, ignoring potential non-verbal feedback elements.
2. High accuracy in aspect and sentiment analysis.	2. Complexity of model requires substantial computational resources.
3. Useful in academic settings for performance evaluation.	3. Customized to an academic setting, might require adjustments for other domains.

LITERATURE SURVEY-4

TITLE:" Aspect-Based Opinion Mining on Student's Feedback for Faculty Teaching Performance Evaluation"
Author: Irum Sindhu
Date of publication: July 15, 2019

1. **Algorithms and Methods Used:**

Aspect-Based Sentiment Analysis (ABSA): Extract aspects (like teaching, behavior, etc.) from text and determine the sentiment (positive, negative, neutral) associated with each aspect.
LSTM (Long Short-Term Memory) Neural Network: Sequence modeling for aspect extraction and sentiment classification.
Skip-Gram Model (Word2Vec): To create word embeddings (vector representations) that capture the semantic meaning of words in the domain of student feedback.
Bi-Directional LSTM (BiLSTM): Improves accuracy in aspect extraction and sentiment classification by looking at input sequences from both directions (past and future contexts).
Softmax Layer (Classification): Used for classification tasks to predict the probability of different classes (e.g., positive, negative, neutral sentiments.
Preprocessing Techniques: To clean and structure the input text data before feeding it into the model.

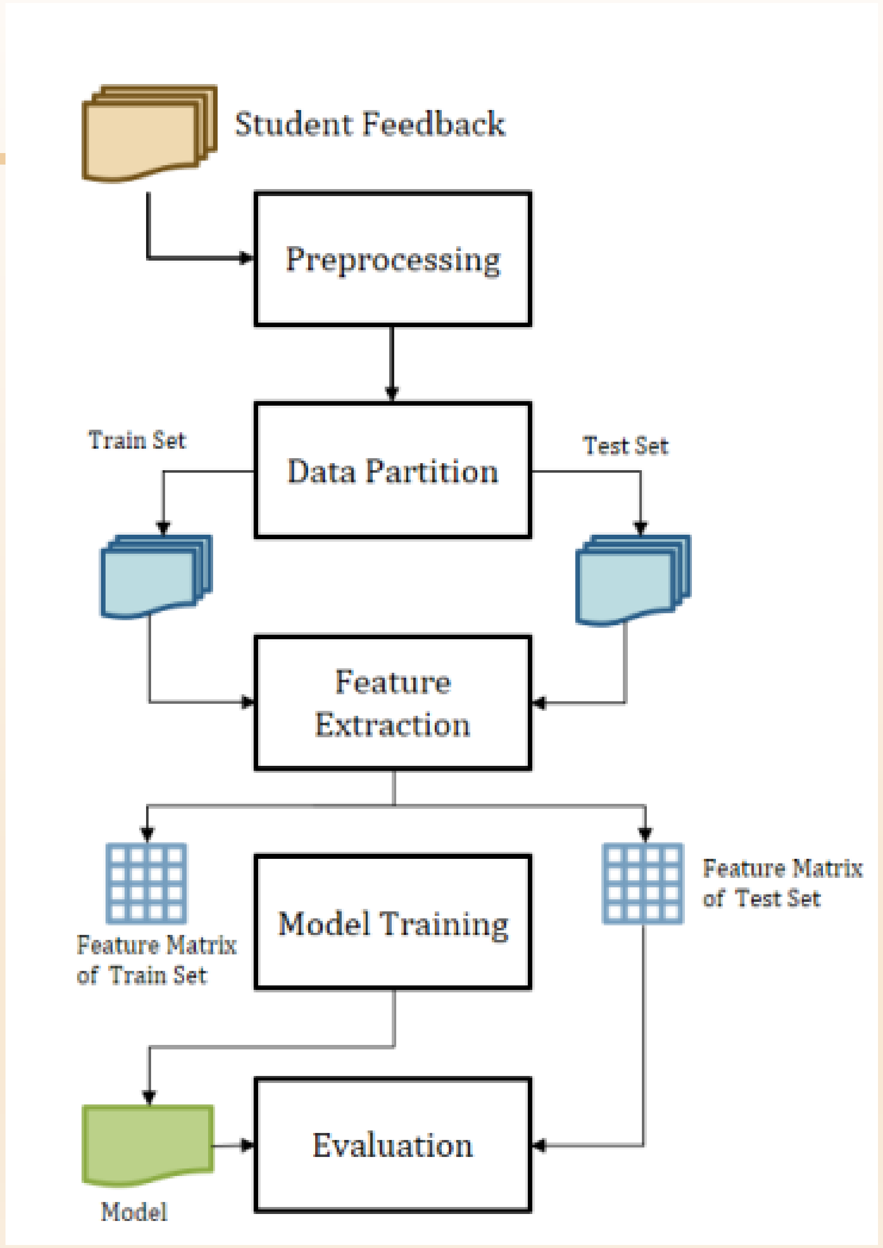
2. Result and conclusion: Students while performing teacher’s evaluation discuss several aspects of a teacher in their comments. Manual processing of these comments and reviewing every mentioned aspect polarity is a laborious job. Though, their thorough understanding acts as a metric for evaluating faculty teaching performance.

3.Cons: evaluate student feedback on teaching performance. It employs a two-layered LSTM model for extracting aspects and classifying sentiment, leveraging Skip-Gram word embeddings for domain-specific word representations. The model preprocesses feedback data by segmenting sentences and removing stopwords, achieving high accuracy in aspect extraction and sentiment classification. Evaluations were conducted using a custom dataset and a standard benchmark, showing promising results. Future work includes enhancing the model to handle multiple aspects and different languages. such as teaching methods and educator behavior while identifying sentiment orientations as positive, negative, or neutral. Additionally, the project highlights the importance of using domain-specific embeddings to improve understanding of the unique terminologies found in academic feedback.

MERITS AND DEMERITS

Merits	Demerits
1. Combines ML and lexicon approaches for robust analysis.	1. May not fully capture the nuances of student sentiment due to the limitations of the lexicon.
2. Effective at categorizing sentiments.	2. Dependency on pre-defined lexicons and training data can affect adaptability.
3. Can be integrated into educational tools for real-time feedback analysis.	3. Performance largely depends on the quality and scope of the training data.

DIAGRAM


Student Feedback
Preprocessing
Train Set
Data Partition
Test Set
Feature Extraction
Feature Matrix of Train Set
Model Training
Feature Matrix of Test Set
Model
Evaluation

LITERATURE SURVEY-5

TITLE:" Sentiment analysis of student feedback using machine learning and lexicon based approaches"
Author: Zarmeen Nasim
Date of publication: 03 March 2021

1. **Algorithms and Methods Used:**

Sentiment Analysis:
Machine Learning:Supervised and unsupervised learning models are employed for text analysis. Algorithms such as Naive Bayes, Support Vector Machines (SVM), and Random Forests are used for sentiment classification.
Lexicon-based Approaches:Lexicons or dictionaries of sentiment-laden words are utilized to gauge the polarity of text (positive, negative, neutral).
Hybrid Approaches:This combines both machine learning techniques and lexicon-based methods for more accurate sentiment detection and classification.
TF-IDF and N-Grams:These are used to extract features from textual data for sentiment classification.

2.Result and conclusion: The paper described a hybrid approach for performing sentiment analysis on student feedbacks. The presented approach employed machine learning methods along with sentiment lexicons. The paper also investigated other APIs available for sentiment analysis and compared the results with the presented hybrid approach.

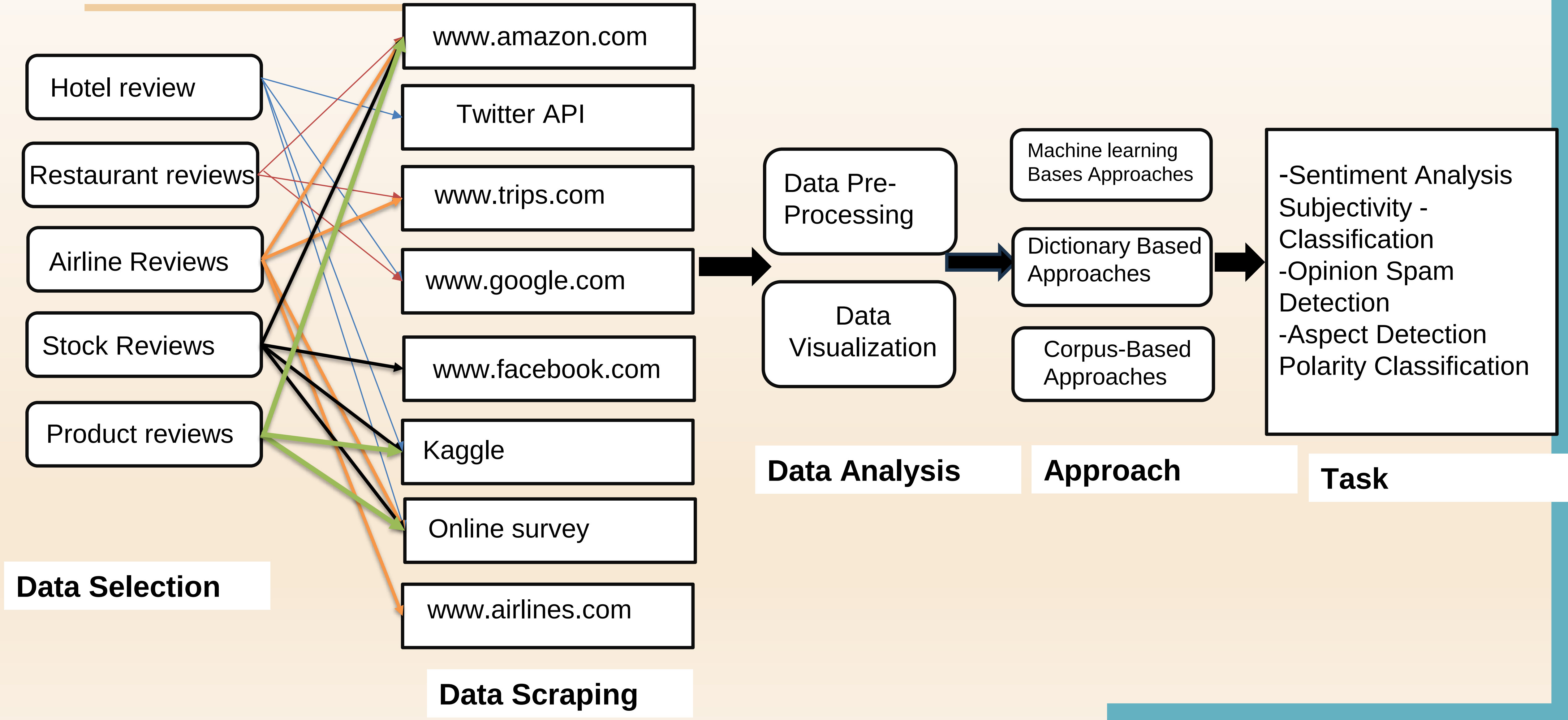
It was found that the best performing model was achieved using TF-IDF and domain-specific sentiment lexicon. In contrast to lexicon-based approach, the proposed approach of combining the use of sentiment lexicon with machine learning techniques was capable of predicting the sentiment of the textual content even if the opinion words do not exist in the lexicon.

3.Cons: The integration of **sentiment analysis** and **social network analysis** is highlighted as crucial for detecting and managing the spread of hateful content.
Tools and technologies like TwitterMate prove effective in crisis data collection, enabling authorities to analyze real-time information during terrorist attacks.
The study concludes that developing advanced algorithms and enhancing sentiment detection methods, especially those tailored for the context of terrorism, can improve early detection, response, and mitigation efforts.

MERITS AND DEMERITS

Merits	Demerits
1. Comprehensive review of sentiment analysis methods.	1. More a theoretical review than practical application, which may not translate directly to implementation.
2. Highlights a wide range of applications across different fields.	2. Does not provide specific case studies or examples of successful sentiment analysis implementation.
3. Discusses current challenges and future directions.	3. Lacks depth in discussing solutions to the challenges mentioned.

DIAGRAM


Hotel review
Restaurant reviews
Airline Reviews
Stock Reviews
Product reviews
Data Selection
www.amazon.com
Twitter API
www.trips.com
www.google.com
www.facebook.com
Kaggle
Online survey
www.airlines.com
Data Scraping
Data Pre-Processing
Data Visualization
Data Analysis
Machine learning Bases Approaches
Dictionary Based Approaches
Corpus-Based Approaches
Approach
-Sentiment Analysis
Subjectivity -Classification
-Opinion Spam Detection
-Aspect Detection
Polarity Classification
Task

LITERATURE SURVEY-6

TITLE:”A survey on sentiment analysis methods, applications, and challenges”
Author: Mayur Wankhade
Date of publication:7 February 2022

1. **Algorithms and Methods Used:**

Naïve Bayes (NB):A probabilistic classifier based on Bayes' Theorem. It works well for text classification when the assumption of independence between features is reasonable.
Support Vector Machine (SVM):A supervised learning model that uses hyperplanes to separate data into different classes. Often used for binary or multi-class text classification tasks.
Random Forest (RF):An ensemble learning method that uses multiple decision trees to improve classification accuracy.
Logistic Regression (LR):A probabilistic model used for binary classification that calculates the probability of a class label.
Decision Tree (DT):A non-linear model that splits the dataset into subsets based on feature values.
Long Short-Term Memory (LSTM):A type of recurrent neural network (RNN) that is particularly suited for sequential data such as text.
Convolutional Neural Network (CNN):Although primarily used in image recognition, CNNs can also be used for text classification tasks by treating text as a sequence of words.

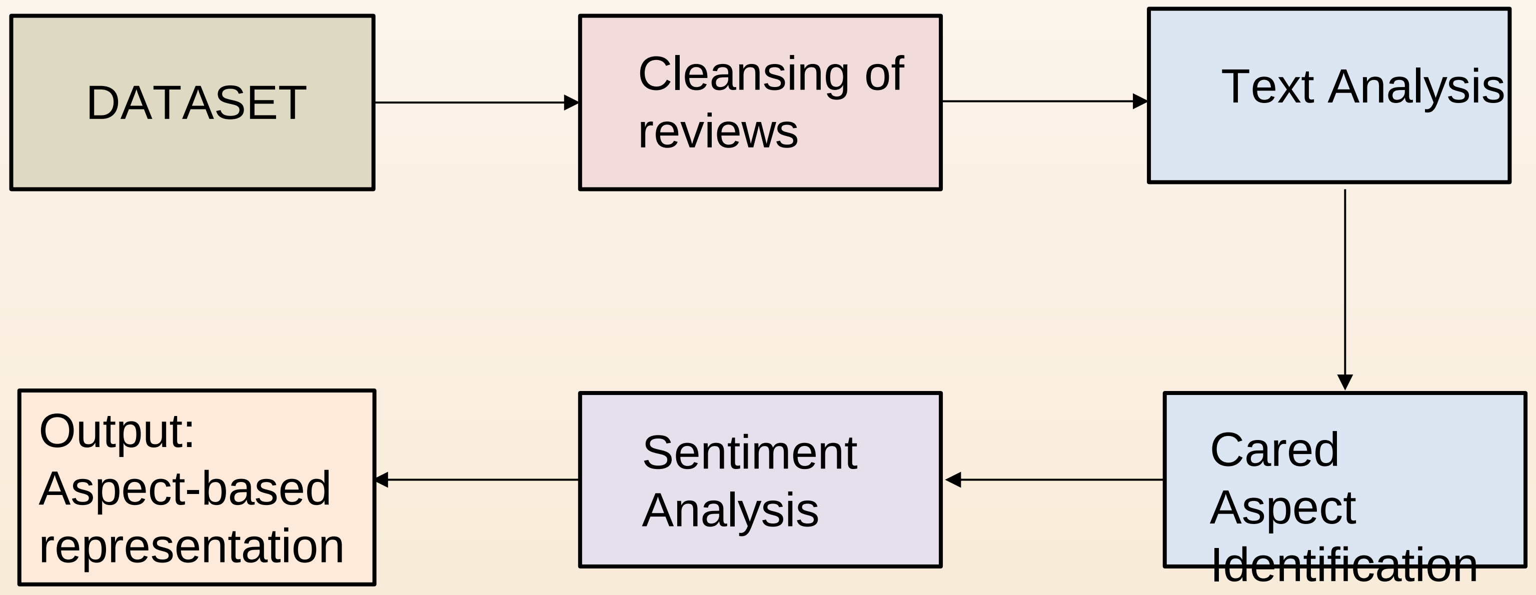
2.Result and conclusion:This article discussed sentiment analysis and associated techniques. The primary objective of this work is to investigate and complete classifcation methods with their advantage and disadvantages in sentiment analysis. To begin, several levels of sentiment analysis were discussed, followed by a quick overview of necessary procedures such as data collection and feature selection. Next, methods of sentiment categorization systems were classifed and compared in terms of their advantages and disadvantages. Due to their simplicity and excellent accuracy, supervised machine learning methods are often the widely utilized technique in this discipline.

3. concs: The paper highlights the growing importance of sentiment analysis for various domains such as e-commerce, healthcare, and social media. The conclusion emphasizes that machine learning algorithms, lexicon-based approaches, and hybrid models have all contributed to advancements in sentiment analysis. However, challenges such as sarcasm detection, context dependency, and multi-lingual sentiment analysis continue to pose difficulties. The document calls for more robust models and techniques to address these challenges and improve the accuracy of sentiment classification.

MERITS AND DEMERITS

Merits	Demerits
1. Innovative use of aspect-level analysis for product recommendations.	1. Relies on the accuracy of WordNet for aspect identification, which might not capture all relevant aspects.
2. Tailored recommendations based on user preferences.	2. Focus on positive comments may ignore critical feedback that could be useful.
3. Demonstrates practical application in e-commerce.	3. Dataset limited to Amazon reviews, which may not be representative of all types of customer feedback.

DIAGRAM


DATASET
Cleansing of reviews
Text Analysis
Cared Aspect Identification
Sentiment Analysis
Output: Aspect-based representation

PROBLEM STATEMENT

Problem Statement: Traditional machine learning techniques for sentiment analysis often require significant manual intervention, infrastructure setup, and maintenance, leading to increased costs and complexity. These systems typically lack the ability to scale automatically in response to fluctuating demand, which can result in performance bottlenecks and inefficiencies during peak usage. Additionally, integrating real-time feedback processing and data visualization requires multiple third-party services, further complicating the system architecture.

EXISTING SYSTEM AND DRAWBACKS

Existing system: Implementation of machine learning Random forest, Naive-bayes algorithms, Support Vector machine is bit complex to build due to the lack of information about the data visualization. Mathematical calculations are used in existing system for model building this may takes the lot of time and complexity.Additionally, integrating real-time feedback processing and data visualization requires multiple third-party services, further complicating the system architecture.

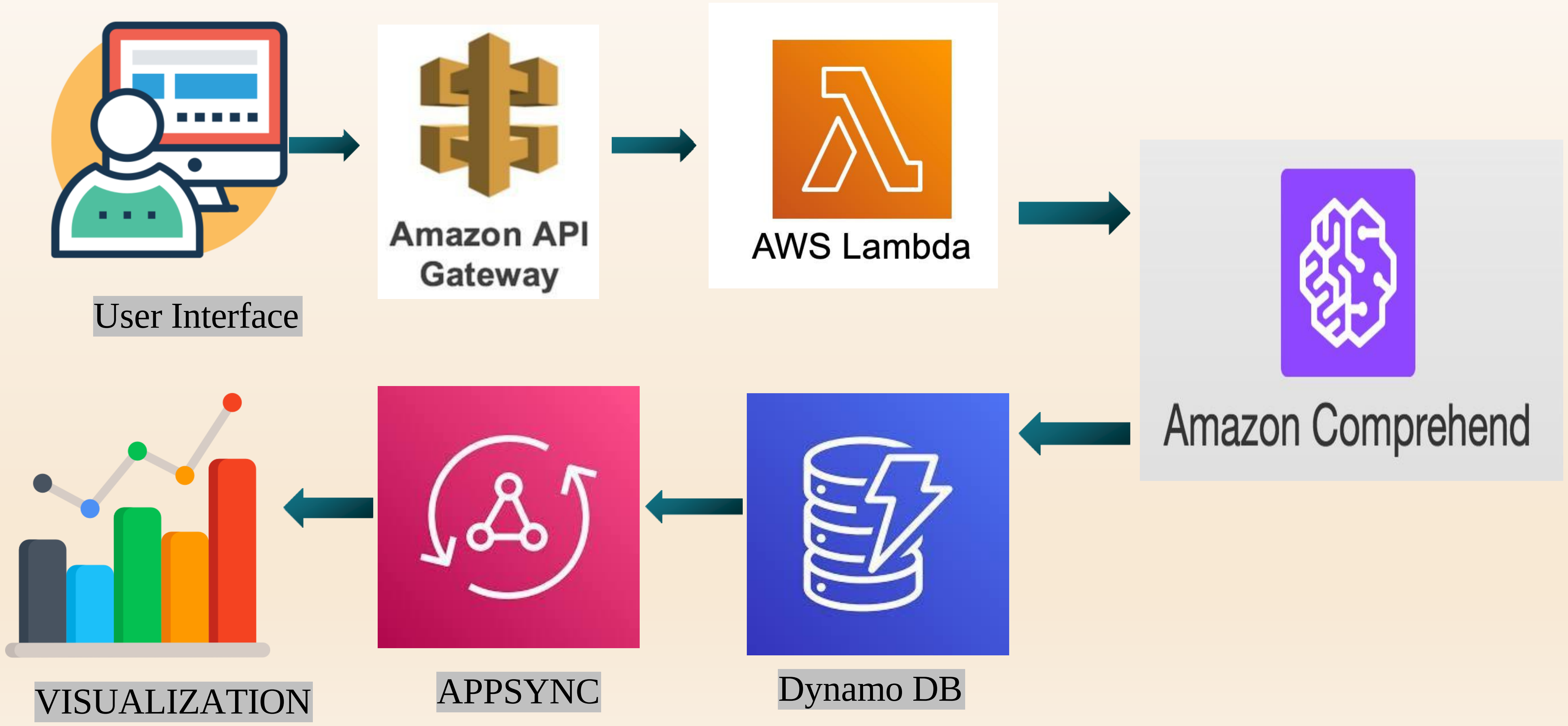
DRAWBACKS:
Sentiment Intensity and Aspect Identification: Models may struggle to identify the intensity of sentiment and the specific aspects or features being praised or criticized.
Handling Out-of-Vocabulary (OOV) Words: Models struggle with words or phrases not seen during training, impacting accuracy.
Handling Imbalanced Datasets: Models can be biased towards the majority class in imbalanced datasets.
Explainability and Interpretability: Traditional models lack transparency, making it difficult to understand the decision-making process.
Ambiguity and Sarcasm: Models find it challenging to identify sarcasm, irony, and ambiguous expressions, resulting in misclassification.

PROPOSED SYSTEM AND SOLUTION

Proposed system is AWS Serverless Feedback System: Automated Analysis And Visualization. Multiple third party services are not required as AWS itself completes the entire process with its own resources. AWS services, particularly AWS Comprehend for sentiment analysis, AWS Lambda for serverless computing, and DynamoDB for scalable storage, offer a more efficient and scalable solution. By leveraging AWS's fully managed services, businesses can reduce operational overhead, achieve real-time insights, and ensure seamless scalability without the need for constant manual adjustments.

ADVANTAGES:
Scalability with AWS Lambda
High Availability and Durability with DynamoDB
Real-Time Data Processing with AWS AppSync
Machine Learning with Amazon Comprehend
Cost Efficiency with Free Tier and Pay-as-You-Go

ARCHITECTURE

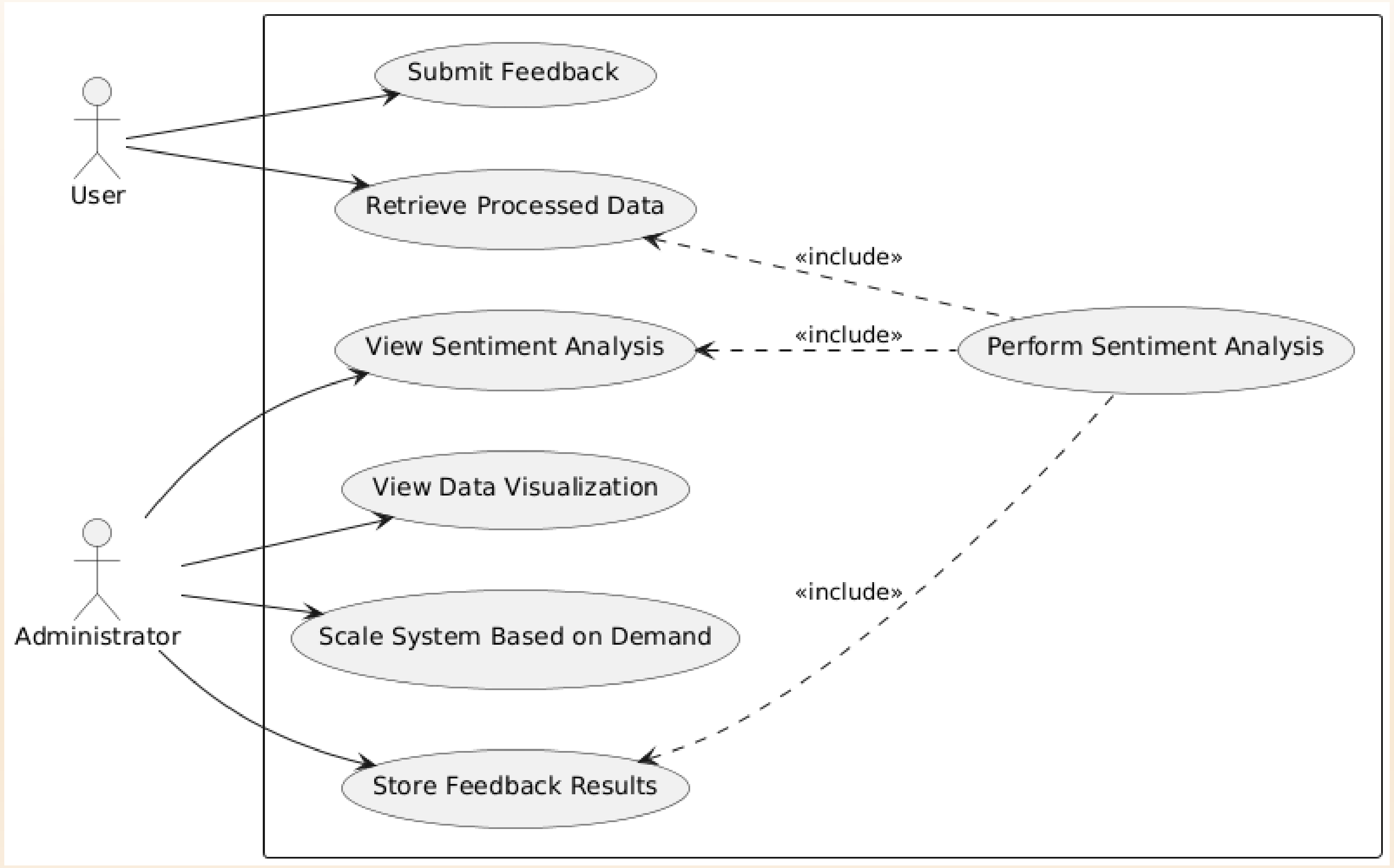


IMPLEMENTATION

Key Technologies Used in the Project:

Python (Backend Development): Used in AWS Lambda for handling API requests, processing feedback, and integrating AWS services and Supports data handling, sentiment analysis, and automation with libraries like boto3.
AWS Lambda (Serverless Processing): Executes functions in response to feedback submissions. Handles feedback processing, sentiment analysis, and database storage without requiring dedicated servers.
AWS API Gateway (Request Handling): Manages HTTP requests from users and routes them to AWS Lambda.Provides secure and scalable API endpoints for feedback submission.
AWS Comprehend (Sentiment Analysis): Uses Natural Language Processing (NLP) to classify feedback as Positive, Negative, or Neutral and Eliminates the need for manually implementing machine learning models.
AWS DynamoDB (Database for Storing Feedback): Stores processed sentiment analysis results for future retrieval and Ensures fast, scalable, and highly available storage without traditional database management.
AWS AppSync (Real-Time Data Retrieval): Fetches real-time sentiment data from DynamoDB and Ensures instant updates for administrators viewing feedback reports.
Google Charts (Data Visualization): Generates real-time graphical reports based on sentiment analysis results. Provides a clear and interactive representation of feedback trends for decision-making.
Web-Based Feedback System: Allows users to submit feedback through a simple web interface.Designed to be lightweight, responsive, and user-friendly using HTML, CSS, JavaScript, and Bootstrap.

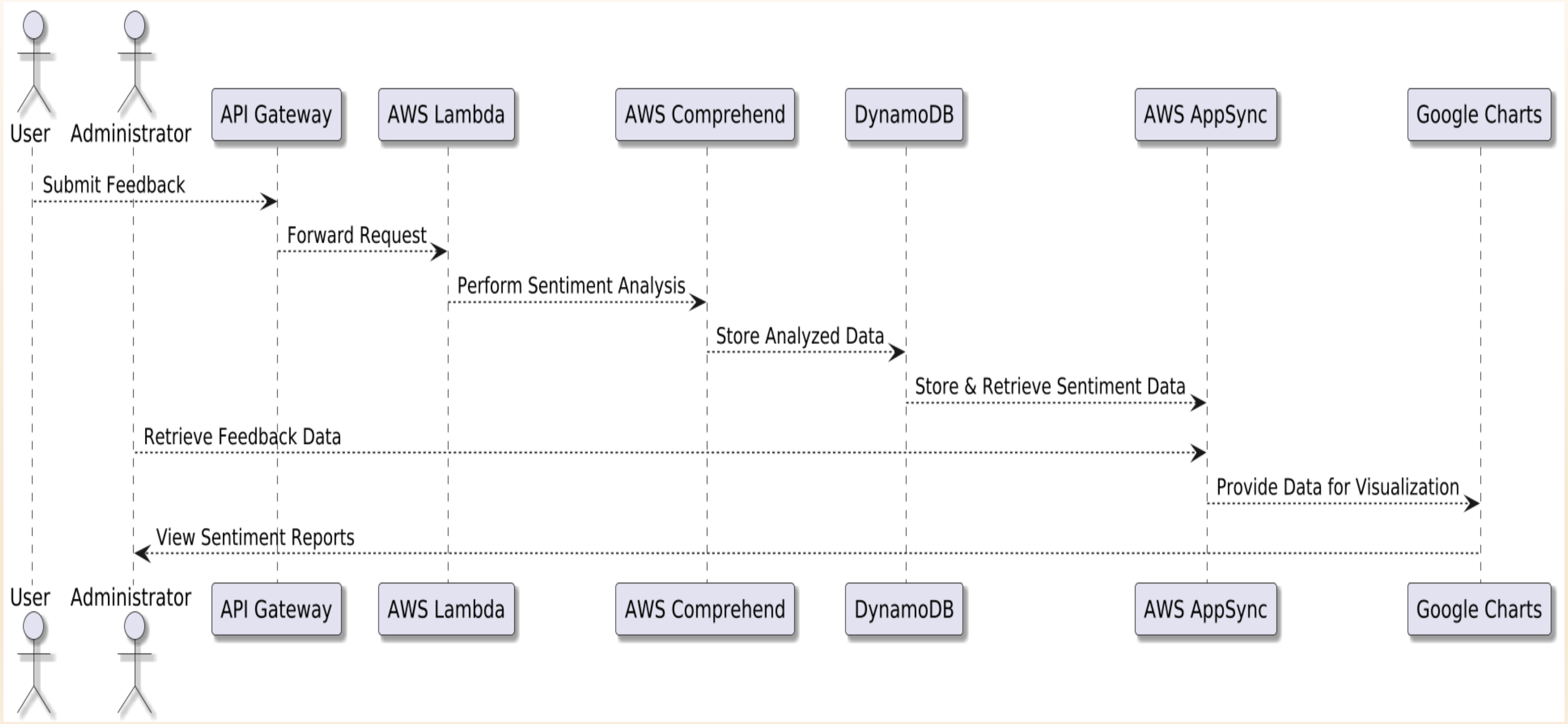
USE CASE DIAGRAM



EXPLANATION

•**Submit Feedback**: Users submit their feedback through the system.
•**Retrieve Processed Data**: Users can retrieve data that has been processed and analyzed.
•**View Sentiment Analysis**: Administrators can view the results of the sentiment analysis.
•**View Data Visualization**: Administrators can view the sentiment trends via data visualizations (e.g., Google Charts).
•**Scale System Based on Demand**: The system automatically scales without manual intervention.
•**Store Feedback Results**: The system stores the analyzed feedback results in a database (AWS DynamoDB).
•**Perform Sentiment Analysis**: This is an internal use case where the system processes the feedback using AWS Comprehend to determine sentiment.

SEQUENCE DIAGRAM



EXPLANATION

•**User Submits Feedback**: The user submits feedback through AWS API Gateway, which acts as the entry point for handling RESTful API methods securely and efficiently.

•**Trigger Processing**: AWS API Gateway triggers an AWS Lambda function, which manages the serverless computing aspect of the application.

•**Sentiment Analysis**: AWS Lambda forwards the feedback to AWS Comprehend for sentiment analysis, categorizing it into positive, negative, or neutral based on NLP techniques.

•**Store Processed Data**: After receiving the sentiment analysis results from AWS Comprehend, AWS Lambda stores these results in AWS DynamoDB, which handles large data volumes with high availability and scalability.

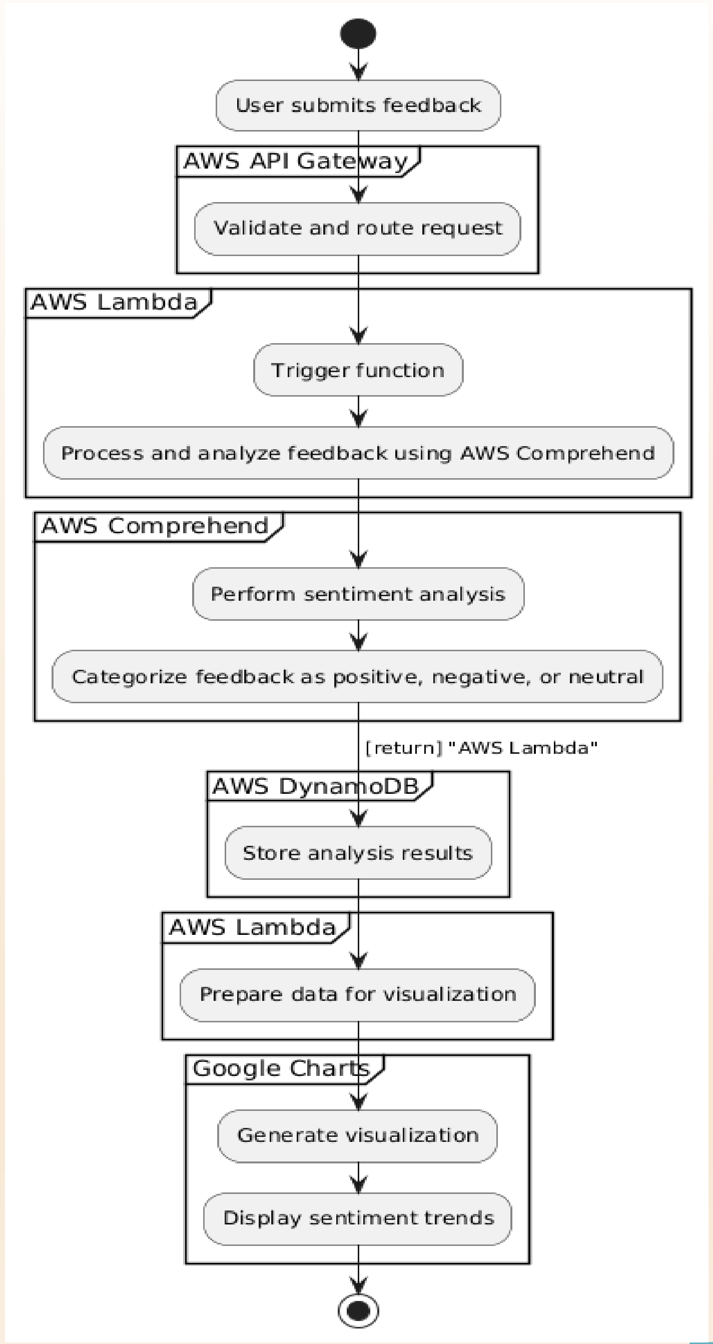
•**Confirm Submission**: AWS Lambda sends a confirmation back to the user through AWS API Gateway, indicating successful submission and processing.

•**Request Data Visualization**: An administrator requests data visualization to review user sentiment trends.

•**Fetch and Retrieve Processed Data**: AWS Lambda fetches and retrieves the processed data from AWS DynamoDB upon request.

•**Display Data via Google Charts**: Finally, the data is sent back to the administrator through AWS APPSYNC Gateway and displayed via Google Charts, showing user sentiment trends in stacked bar charts.

ACTIVITY DIAGRAM

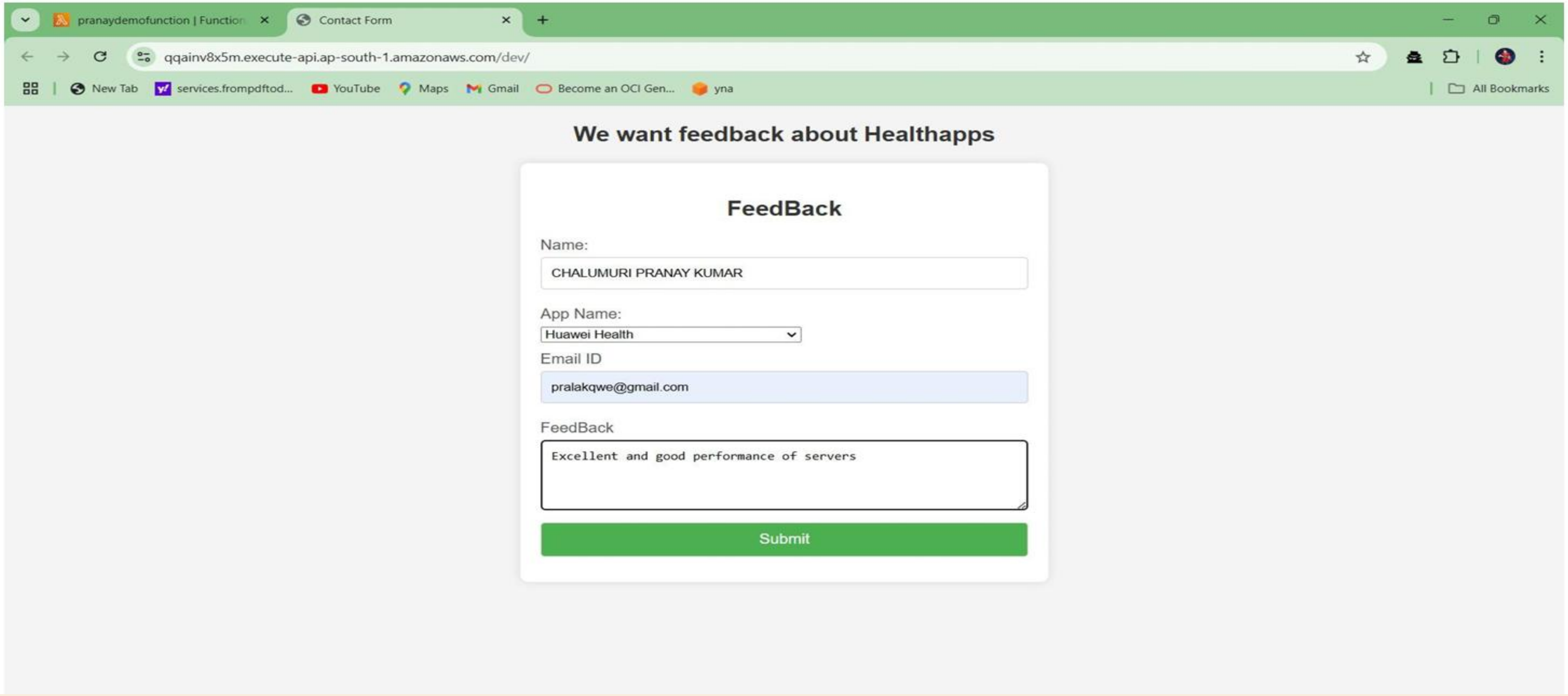


EXPLANATION

- **User Submits Feedback**: This is the initial action where the user submits feedback through the system.
- **AWS API Gateway**: This partitions handles the initial reception of the user's feedback. It validates
- and routes the request appropriately to trigger the necessary services.
- **AWS Lambda**: This AWS service is triggered next. It processes the feedback and utilizes AWS
- Comprehend for detailed sentiment analysis.
- **AWS Comprehend**: In this partition, the feedback is analyzed for sentiment, categorizing it into
- positive, negative, or neutral based on natural language processing techniques.
- **AWS DynamoDB**: The results from the sentiment analysis are stored in AWS DynamoDB,
- which ensures high availability and scalability for handling large data volumes.
- **AWS Lambda**: After storage, AWS Lambda prepares the analyzed data for visualization.
- **Google Charts**: Finally, the sentiment trends are visualized using Google Charts. This step allows
- administrators and users to view the analyzed sentiment trends in the form of stacked bar charts.

IMPLEMENTATION

1. USER INTERFACE (Feedback form)



We want feedback about Healthapps

FeedBack

Name: CHALUMURI PRANAY KUMAR

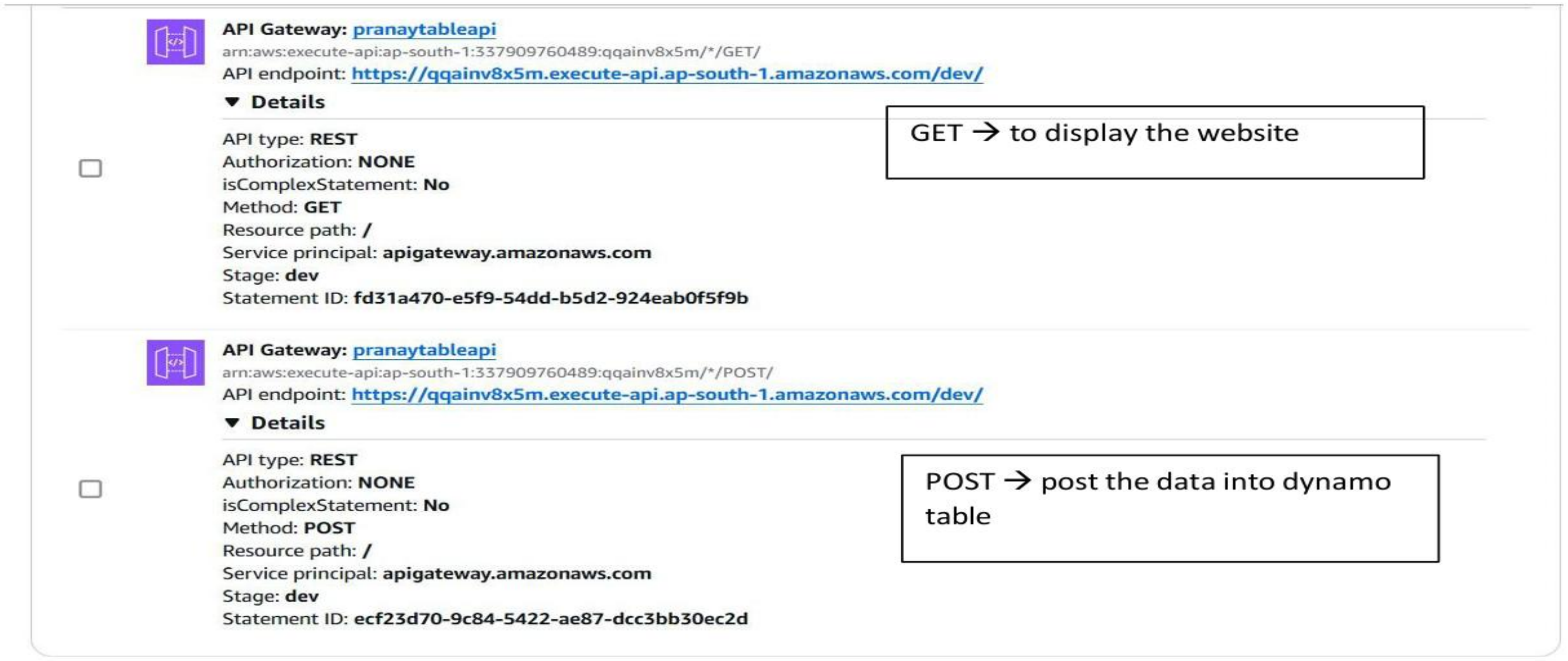
App Name: Huawei Health

Email ID: pralakqwe@gmail.com

FeedBack: Excellent and good performance of servers

Submit

2. API GATE WAYS :



API Gateway: pranaytableapi
arn:aws:execute-api:ap-south-1:337909760489:qqainv8x5m/*/GET/
API endpoint: https://qqainv8x5m.execute-api.ap-south-1.amazonaws.com/dev/
▼ Details

API type: **REST**
Authorization: **NONE**
isComplexStatement: **No**
Method: **GET**
Resource path: **/**
Service principal: **apigateway.amazonaws.com**
Stage: **dev**
Statement ID: **fd31a470-e5f9-54dd-b5d2-924eab0f5f9b**

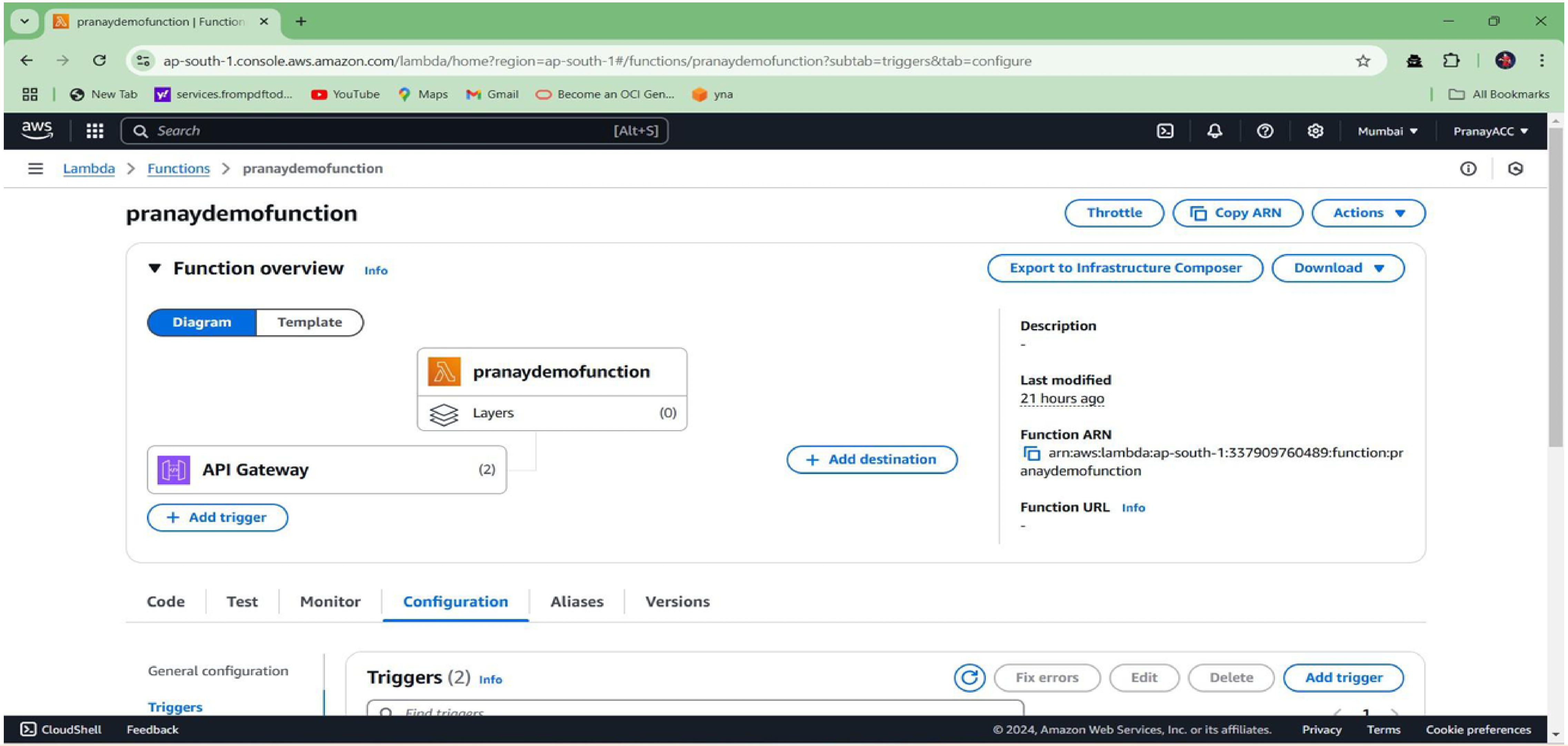
GET → to display the website

API Gateway: pranaytableapi
arn:aws:execute-api:ap-south-1:337909760489:qqainv8x5m/*/POST/
API endpoint: https://qqainv8x5m.execute-api.ap-south-1.amazonaws.com/dev/
▼ Details

API type: **REST**
Authorization: **NONE**
isComplexStatement: **No**
Method: **POST**
Resource path: **/**
Service principal: **apigateway.amazonaws.com**
Stage: **dev**
Statement ID: **ecf23d70-9c84-5422-ae87-dcc3bb30ec2d**

POST → post the data into dynamo table

3. LAMDA FUNCTION



pranaydemofunction | Function

ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions/pranaydemofunction?subtab=triggers&tab=configure

Lambda > Functions > pranaydemofunction

pranaydemofunction

Throttle | Copy ARN | Actions

Function overview Info

Export to Infrastructure Composer | Download

Diagram | Template

pranaydemofunction

Layers (0)

API Gateway (2)

+ Add trigger

+ Add destination

Description
-

Last modified
21 hours ago

Function ARN
arn:aws:lambda:ap-south-1:337909760489:function:pranaydemofunction

Function URL Info
-

Code | Test | Monitor | Configuration | Aliases | Versions

General configuration

Triggers

Triggers (2) Info

Fix errors | Edit | Delete | Add trigger

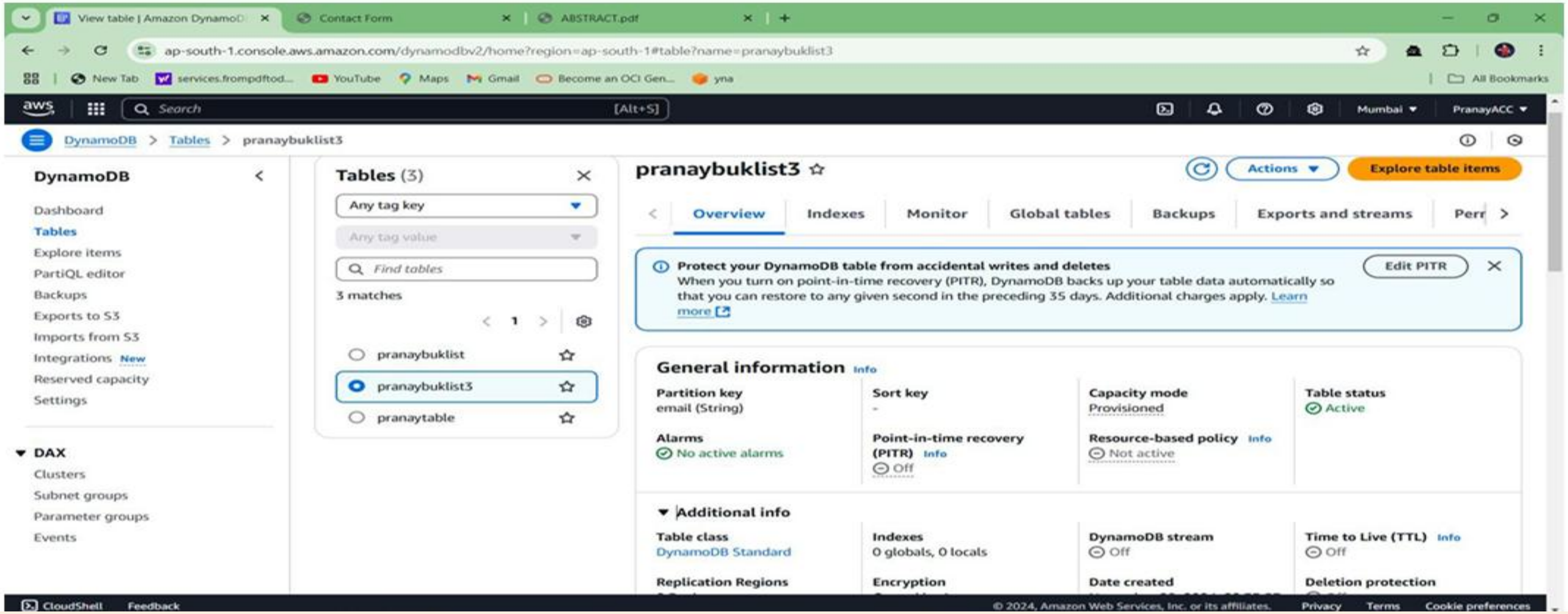
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

4. COMPREHEND

→I used a function directly inside the lamda_function.py (not manually used) and check the code I mentioned .

5. DYNAMO DB

Table name: pranaybuklist3



DynamoDB > Tables > pranaybuklist3

Tables (3)

Any tag key

Any tag value

Find tables

3 matches

- pranaybuklist
- pranaybuklist3
- pranaytable

pranaybuklist3

Protect your DynamoDB table from accidental writes and deletes
When you turn on point-in-time recovery (PITR), DynamoDB backs up your table data automatically so that you can restore to any given second in the preceding 35 days. Additional charges apply. Learn more

Edit PITR

General information Info

Partition key	Sort key	Capacity mode	Table status
email (String)	-	Provisioned	Active
Alarms	**Point-in-time recovery (PITR)** Info	**Resource-based policy** Info	
No active alarms	Off	Not active	

Additional info

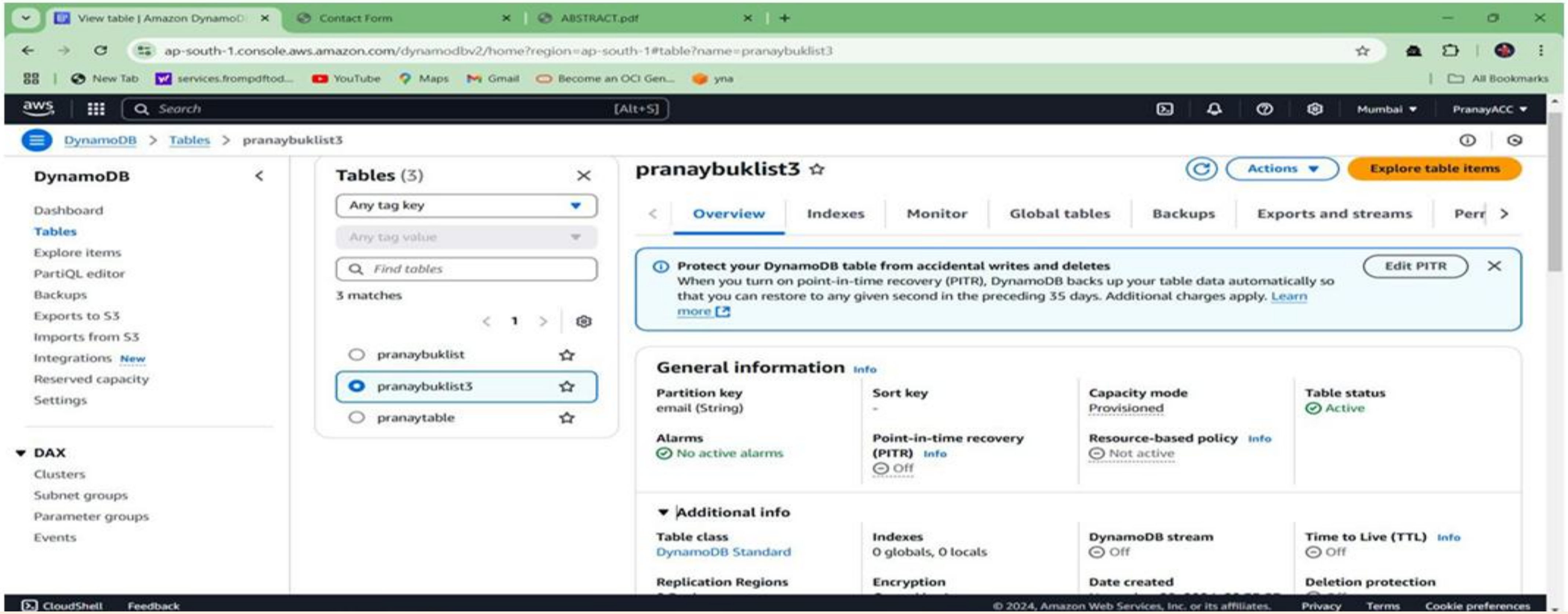
Table class	Indexes	DynamoDB stream	Time to Live (TTL) Info
DynamoDB Standard	0 globals, 0 locals	Off	Off
Replication Regions	**Encryption**	**Date created**	**Deletion protection**

4. COMPREHEND

→I used a function directly inside the lamda_function.py (not manually used) and check the code I mentioned .

5. DYNAMO DB

Table name: pranaybuklist3



pranaybuklist3

Protect your DynamoDB table from accidental writes and deletes
When you turn on point-in-time recovery (PITR), DynamoDB backs up your table data automatically so that you can restore to any given second in the preceding 35 days. Additional charges apply. Learn more

General information Info

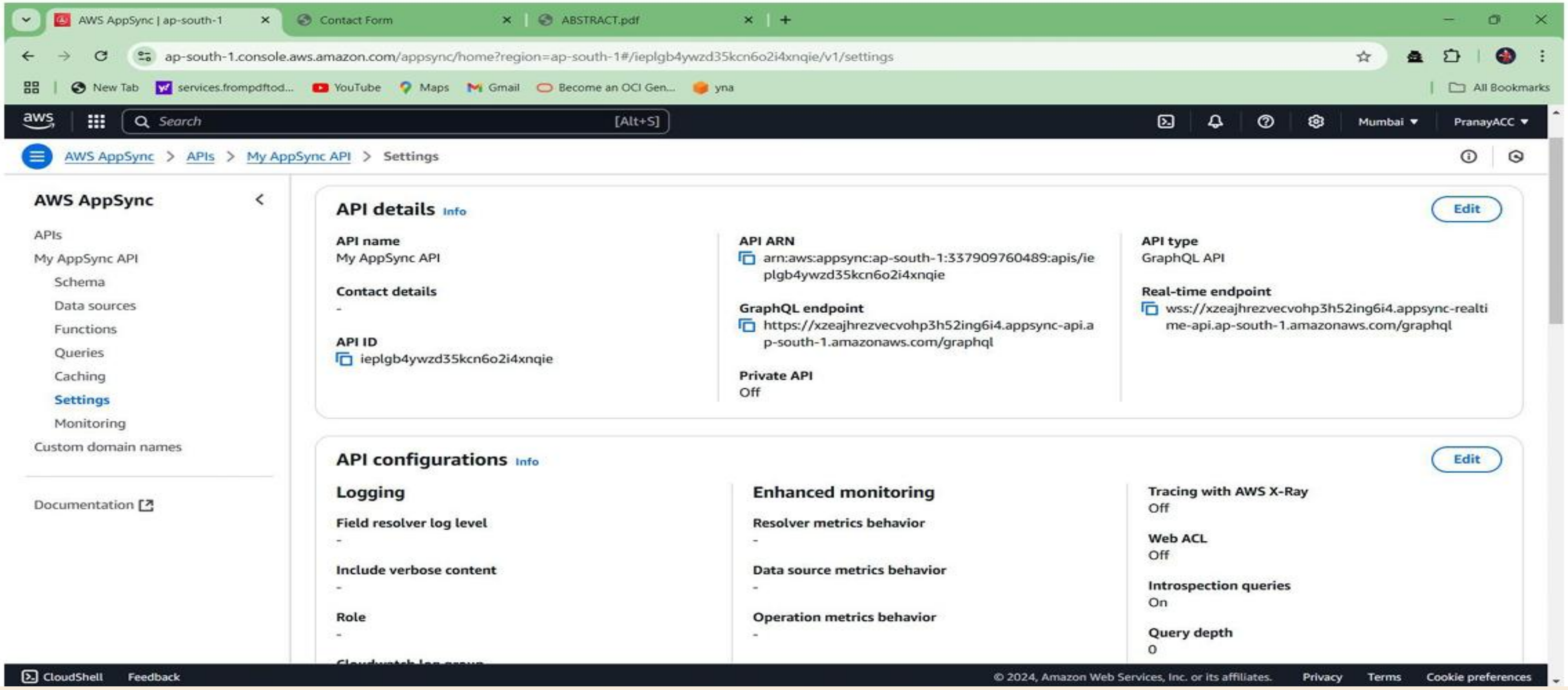
Partition key email (String)	Sort key -	Capacity mode Provisioned	Table status Active
Alarms No active alarms	Point-in-time recovery (PITR) Info Off	Resource-based policy Info Not active	

Additional info

Table class DynamoDB Standard	Indexes 0 globals, 0 locals	DynamoDB stream Off	Time to Live (TTL) Info Off
Replication Regions	Encryption	Date created	Deletion protection

6. APP SYNC

1. APP SYNC API DETAILS



ap-south-1.console.aws.amazon.com/appsync/home?region=ap-south-1#/ieplgb4ywzd35kcn6o2i4xnqie/v1/settings

AWS AppSync > APIs > My AppSync API > Settings

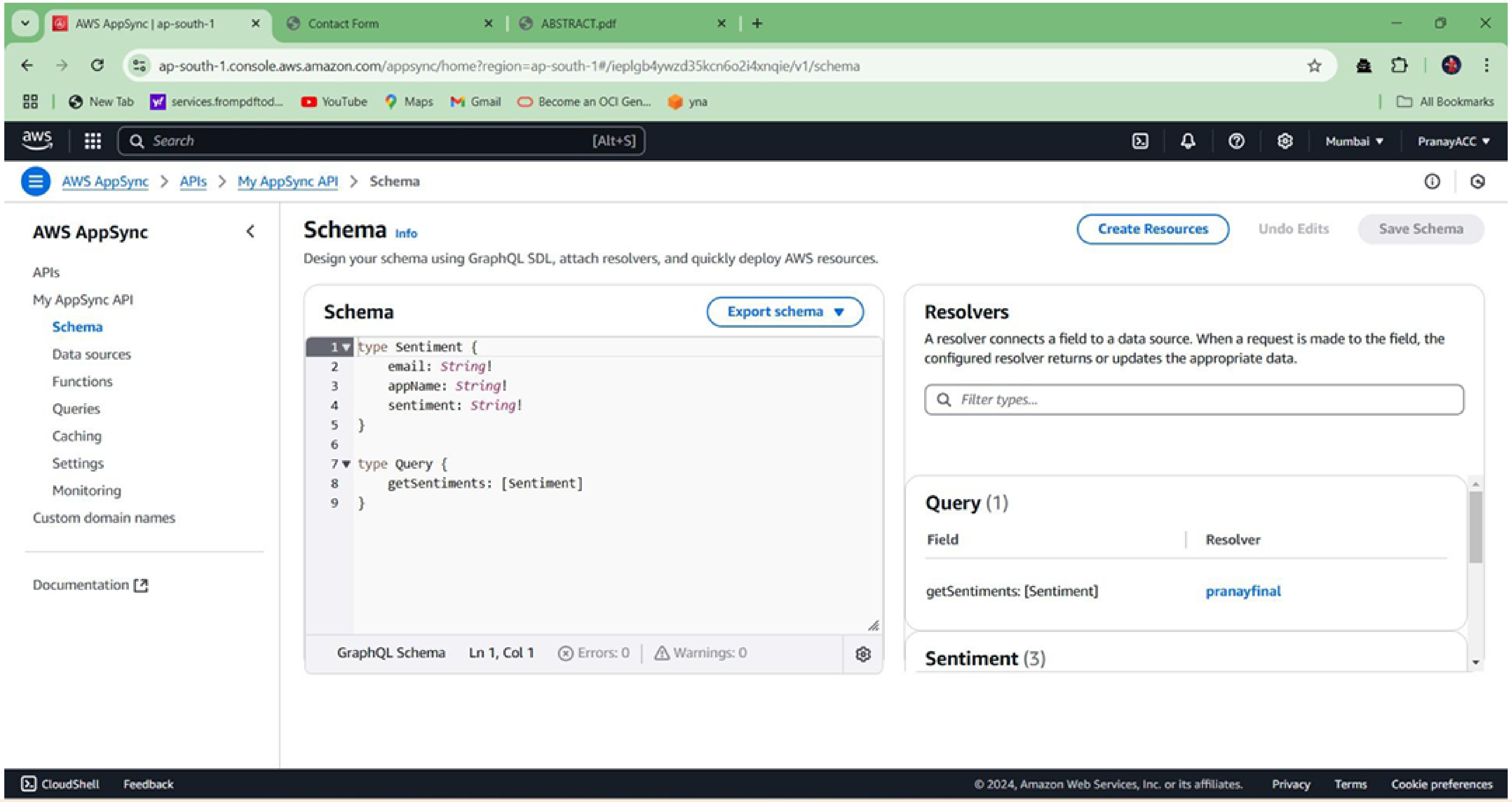
API details

API name: My AppSync API	API ARN: arn:aws:appsync:ap-south-1:337909760489:apis/ieplgb4ywzd35kcn6o2i4xnqie	API type: GraphQL API
Contact details: -	GraphQL endpoint: https://xzeajhrezvecvohp3h52ing6i4.appsync-api.ap-south-1.amazonaws.com/graphql	Real-time endpoint: wss://xzeajhrezvecvohp3h52ing6i4.appsync-realtime-api.ap-south-1.amazonaws.com/graphql
API ID: ieplgb4ywzd35kcn6o2i4xnqie	Private API: Off	

API configurations

Logging	Enhanced monitoring	
Field resolver log level: -	Resolver metrics behavior: -	Tracing with AWS X-Ray: Off
Include verbose content: -	Data source metrics behavior: -	Web ACL: Off
Role: -	Operation metrics behavior: -	Introspection queries: On
		Query depth: 0

2. SCHEMA



AWS AppSync > APIs > My AppSync API > Schema

Schema Info

Design your schema using GraphQL SDL, attach resolvers, and quickly deploy AWS resources.

```
type Sentiment {
    email: String!
    appName: String!
    sentiment: String!
}

type Query {
    getSentiments: [Sentiment]
}
```

GraphQL Schema Ln 1, Col 1 Errors: 0 Warnings: 0

Resolvers

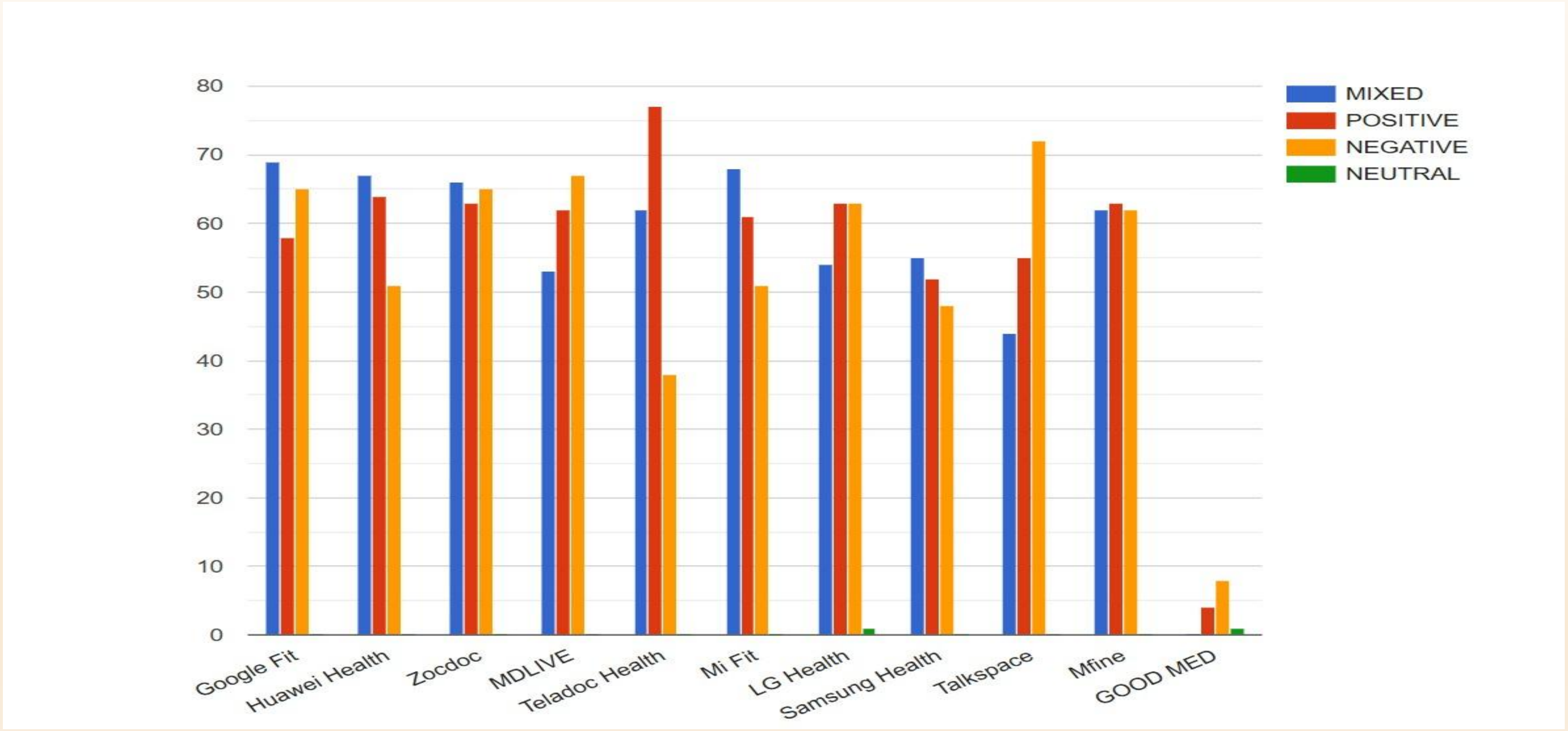
A resolver connects a field to a data source. When a request is made to the field, the configured resolver returns or updates the appropriate data.

Query (1)

Field	Resolver
getSentiments: [Sentiment]	pranayfinal

Sentiment (3)

RESULTS



80
70
60
50
40
30
20
10
0
MIXED
POSITIVE
NEGATIVE
NEUTRAL
Google Fit
Huawei Health
Zocdoc
MDLIVE
Teladoc Health
Mi Fit
LG Health
Samsung Health
Talkspace
Mfine
GOOD MED

CONCLUSION

In conclusion, the serverless application for feedback storage and sentiment analysis using AWS is a robust and scalable solution that automates the entire feedback lifecycle, from collection to actionable insights. By leveraging AWS Lambda and API Gateway, the system ensures high efficiency, scalability, and secure real-time data handling. AWS Comprehend adds an intelligent layer to analyze feedback, categorizing sentiments effectively using advanced NLP techniques. Storing the processed data in AWS DynamoDB guarantees reliability and high availability, even with large data volumes. Real-time visualization with Google Charts enables administrators to monitor sentiment trends effortlessly, fostering informed decision-making. This serverless architecture significantly reduces operational costs with AWS's pay-as-you-go model, making it cost-effective and resource-efficient. The solution is designed to be secure, resilient, and adaptable, empowering businesses to enhance customer engagement and improve services based on real-time feedback insights. It exemplifies a modern approach to handling user feedback in the digital age.

THANK YOU