Introduction

The rapid rise of online reviews and customer feedback in the digital landscape has made it essential for businesses to analyze sentiments and understand their customers' opinions. Sentiment analysis helps classify these opinions as positive, neutral, or negative, enabling businesses to respond appropriately. This project develops a serverless sentiment analysis system utilizing AWS services such as Lambda, Comprehend, and S3. The system automatically processes user reviews uploaded to S3, conducts sentiment analysis with AWS Comprehend, and saves the results back into S3 for further processing or analysis.

Field and Background of Invention

Sentiment analysis is a part of positive language processing (NLP) that focuses on identifying and analyzing the emotions expressed in a document. Businesses use sentiment analysis to understand the sentiment behind customer feedback and adjust their strategies accordingly. The routine process of customer verification is usually manual and time-consuming. The system is designed to use cloud-based serverless computing to work on the process, ensuring efficiency, speed, and cost-effectiveness.

Objective

The aim of this project is to analyze customer reviews using AWS services. Users can upload review data to an S3 bucket, which then automatically activates an AWS Lambda function to process the reviews. This enables users to understand customer sentiments through AWS Comprehend. The processed results are saved in the S3 bucket in CSV format.

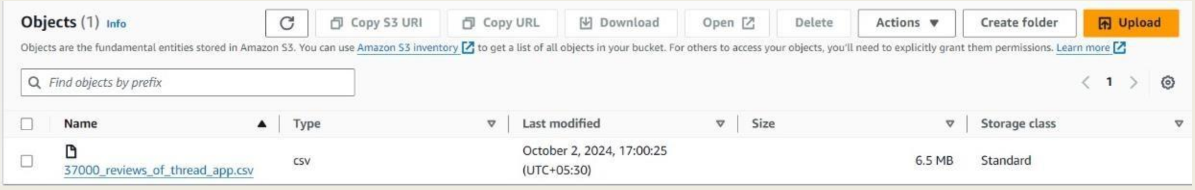
Brief Description of the System

A CSV file with user comments is uploaded to S3. This upload triggers a Lambda function. The Lambda function reads the data, sends each message to AWS Comprehend for sentiment analysis, and compiles the results.

The analysis results are saved as a CSV file in S3, with each analysis categorized as positive, neutral, or negative. AWS Lambda is utilized to initiate the processing of the uploaded files.

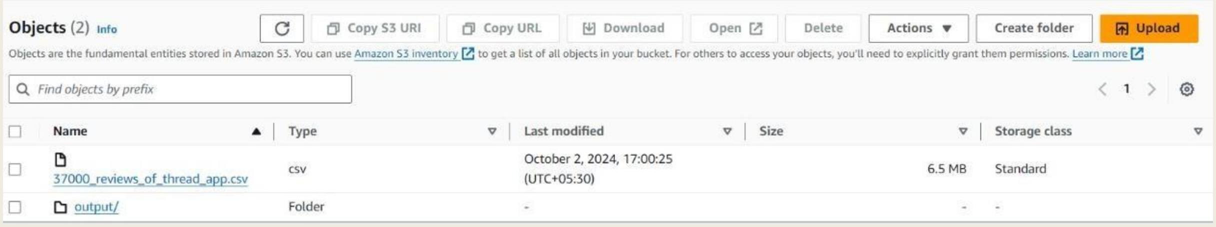
- AWS S3 to store the input and output files.
- AWS Lambda to trigger the processing of the uploaded file.
- AWS Comprehend for performing the sentiment analysis.

System Features



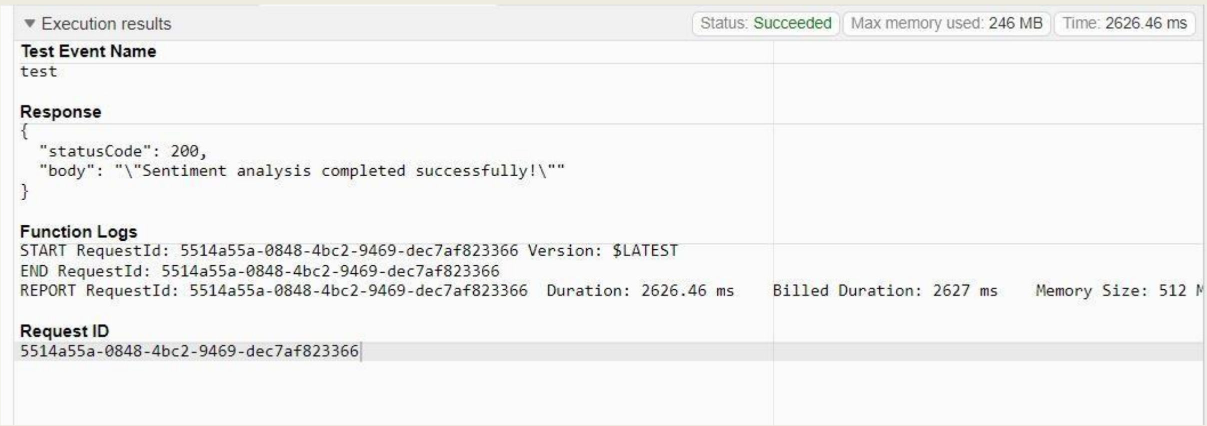
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Last modified	Size	Storage class
□	37000_reviews_of_thread_app.csv	csv	October 2, 2024, 17:00:25 (UTC+05:30)	6.5 MB	Standard



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
□	37000_reviews_of_thread_app.csv	csv	October 2, 2024, 17:00:25 (UTC+05:30)	6.5 MB	Standard
□	output/	Folder	-	-	-



Test Event Name

```
test
```

Response

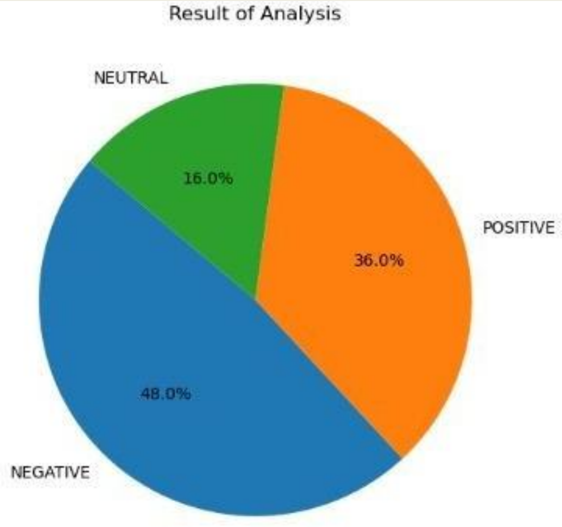
```
{
  "statusCode": 200,
  "body": "\"Sentiment analysis completed successfully!\""
}
```

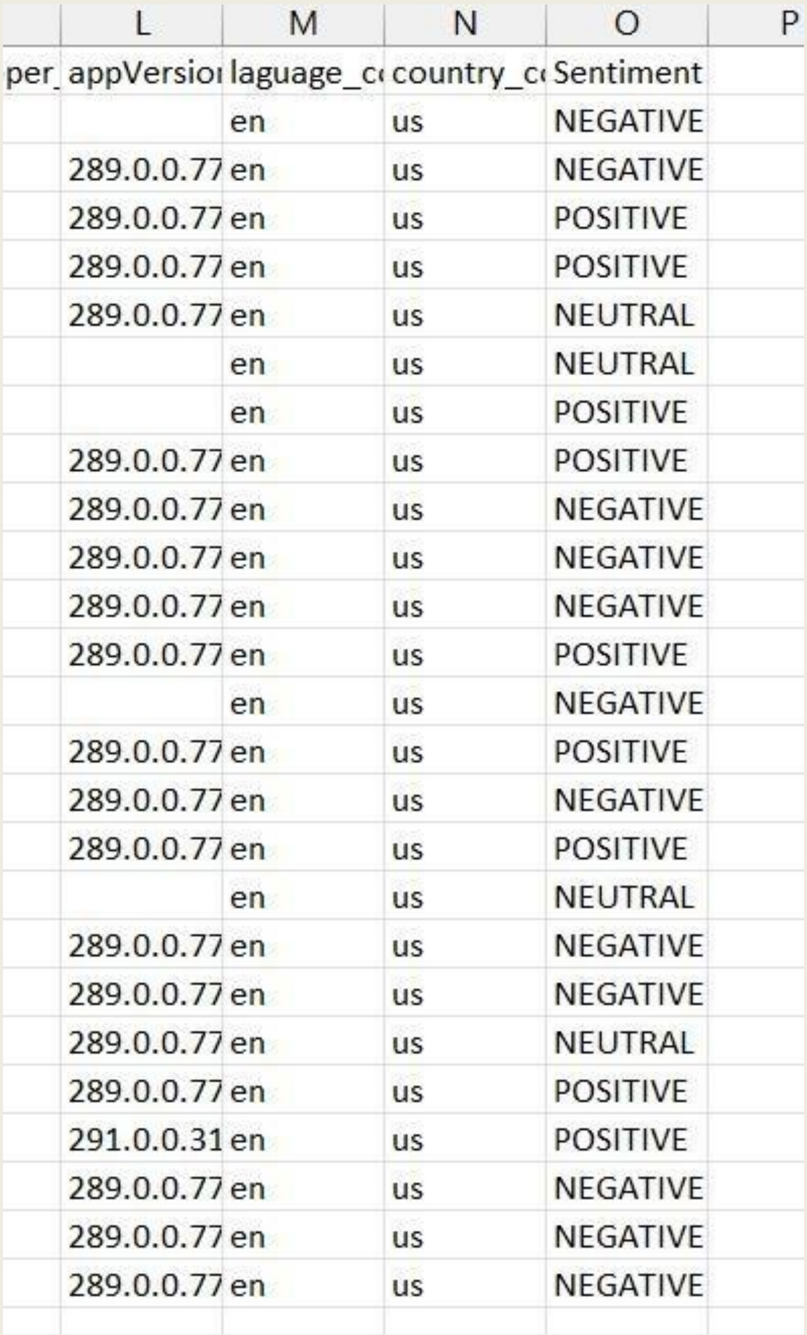
Function Logs

```
START RequestId: 5514a55a-0848-4bc2-9469-dec7af823366 Version: $LATEST
END RequestId: 5514a55a-0848-4bc2-9469-dec7af823366
REPORT RequestId: 5514a55a-0848-4bc2-9469-dec7af823366  Duration: 2626.46 ms    Billed Duration: 2627 ms    Memory Size: 512 M
```

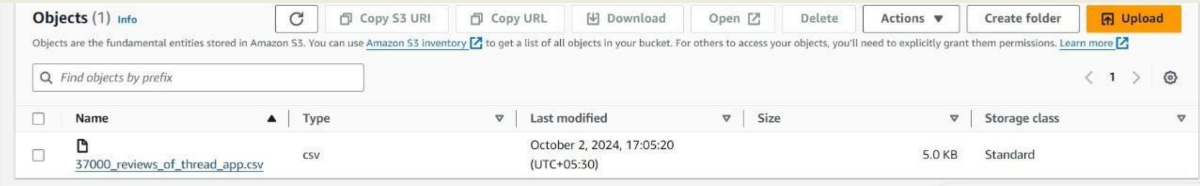
Request ID

```
5514a55a-0848-4bc2-9469-dec7af823366
```





	L	M	N	O	P
per_	appVersioı	laguage_c	country_c	Sentiment	
		en	us	NEGATIVE	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	POSITIVE	
	289.0.0.77	en	us	POSITIVE	
	289.0.0.77	en	us	NEUTRAL	
		en	us	NEUTRAL	
		en	us	POSITIVE	
	289.0.0.77	en	us	POSITIVE	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	POSITIVE	
		en	us	NEGATIVE	
	289.0.0.77	en	us	POSITIVE	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	POSITIVE	
		en	us	NEUTRAL	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	NEUTRAL	
	289.0.0.77	en	us	POSITIVE	
	291.0.0.31	en	us	POSITIVE	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	NEGATIVE	
	289.0.0.77	en	us	NEGATIVE	



Objects (1) Info

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
□	37000_reviews_of_thread_app.csv	csv	October 2, 2024, 17:05:20 (UTC+05:30)	5.0 KB	Standard

Example code:

```
import json
import boto3
import pandas as pd
from io import StringIO
import matplotlib.pyplot as plt

def lambda_handler(event, context):
    s3 = boto3.client('s3')
    comprehend = boto3.client('comprehend')

    try:
        bucket = event['Records'][0]['s3']['bucket']['name']
        key = event['Records'][0]['s3']['object']['key']

        response = s3.get_object(Bucket=bucket, Key=key)
        data = response['Body'].read().decode('utf-8')

        df = pd.read_csv(StringIO(data))

        if 'review_description' not in df.columns:
            return {
                'statusCode': 400,
                'body': json.dumps("Column 'review_description' not
found.")
```

```
        }

    sample_df = df.sample(n=25, random_state=1)  # Use
random_state for reproducibility

    sentiments = []
    for review in sample_df['review_description']:
        sentiment = comprehend.detect_sentiment(Text=review,
LanguageCode='en')
        sentiments.append(sentiment['Sentiment'])

    sample_df['Sentiment'] = sentiments

    sentiment_counts = sample_df['Sentiment'].value_counts()
    plt.figure(figsize=(6, 6))
    plt.pie(sentiment_counts, labels=sentiment_counts.index,
autopct='%1.1f%%', startangle=140)
    plt.title('Sentiment Distribution')

    pie_chart_path = '/tmp/pie_chart.png'
    plt.savefig(pie_chart_path)

    pie_chart_key = 'output/pie_chart_' + key.split('/')[-
1].replace('.csv', '.png')
    with open(pie_chart_path, 'rb') as pie_image:
        s3.put_object(Bucket=bucket, Key=pie_chart_key, 
Body=pie_image)

    output_key = 'output/' + key.split('/')[-1]  # Using only the 
filename for the output
    output_csv = sample_df.to_csv(index=False)

    s3.put_object(Bucket=bucket, Key=output_key, 
Body=output_csv)

    return {
        'statusCode': 200,
        'body': json.dumps('Sentiment analysis and pie chart creation 
completed successfully!')
    }

except Exception as e:
    print(f"Error: {str(e)}")  # Log the error
    return {
        'statusCode': 500,
        'body': json.dumps(f"Error processing file: {str(e)}")
    }
```

Functionality

1. Input: CSV file uploaded to an S3 bucket. The file must include a column named review_description containing the text of customer reviews.

2. Trigger: The upload of the CSV file triggers an AWS Lambda function automatically.

3. Process: The Lambda function reads the CSV file from S3. It checks for the presence of the review_description column. A random sample of 25 reviews is selected from the file for analysis. Each review is analyzed by AWS Comprehend, and the sentiment (Positive, Neutral, Negative) is appended as a new column to the selected reviews.

4. Output: The processed data, including the sentiment labels, is saved back to an S3 bucket in CSV format.

Newness

The system's innovation lies in its serverless architecture, enabling automatic measurements and reducing maintenance costs. Unlike traditional systems that depend on dedicated servers and manual management, this project leverages AWS Lambda for data entry. Furthermore, the integration of AWS Comprehend delivers precise and measurable outcomes without the necessity of creating custom NLP models. All systems can be utilized with minimal setup and operational expenses.

Claims

Automation: The system automates the process of sentiment analysis, minimizing manual work and human error.

Scalability: Using AWS Lambda, the system scales automatically based on the volume of uploaded reviews.

Cost-Efficiency: Being serverless, the system only incurs costs for actual usage, eliminating the need for maintaining dedicated servers.

Accurate Analysis: AWS Comprehend ensures accurate and real-time sentiment classification of customer reviews.

Data Handling: The system securely processes and stores review data using S3, ensuring data integrity and accessibility.

Conclusion

This project showcases the effectiveness of serverless architecture and cloud-based machine learning in automating sentiment analysis. By utilizing AWS services, the system offers a streamlined, scalable, and budget-friendly solution for processing and auditing. This method minimizes the reliance on manual oversight, accelerates the workflow, and equips businesses with valuable insights for making informed decisions.