**AWS News Sentiment Analysis Dashboard**

A full-stack, real-time news sentiment analytics pipeline built with Python, Streamlit, PostgreSQL, Docker, and AWS services like ECS, Lambda, EventBridge, and S3. This project automatically fetches news, performs sentiment analysis, stores it in a database, and visualizes it in a beautiful interactive dashboard.

**🧠 Project Highlights**

* Real-time news ingestion with NewsAPI
* Sentiment analysis using TextBlob
* Data storage in AWS RDS (PostgreSQL)
* Visual analytics using Streamlit
* Dockerized deployment with AWS ECS Fargate
* Scheduled automation using Lambda and EventBridge

**🧰 Tools & Services Used**

* Python (3.9/3.10)
* Streamlit
* TextBlob
* PostgreSQL (AWS RDS)
* SQLAlchemy
* Docker
* AWS ECS Fargate
* AWS Lambda
* AWS EventBridge
* AWS S3
* AWS CloudWatch
* IAM, VPC

**📂 Project Files**

* app.py: Streamlit dashboard
* news\_fetcher.py: Fetches news and stores results
* sentiment.py: Sentiment scoring logic
* testrds.py: RDS connection test script
* requirements.txt: Python dependencies
* Dockerfile: Streamlit dashboard container
* fetcher.dockerfile: News fetcher container
* Dockerfile.dashboard: Alternate container for dashboard
* backup.sql: Optional schema backup
* news\_raw.json: Sample fetched news data

**🖥️ Local Development Setup**

1. Install Python 3.9+ and Docker
2. Install Python packages:  
   pip install -r requirements.txt
3. Run the dashboard locally:  
   streamlit run app.py
4. (Optional) Start local PostgreSQL via Docker:  
   docker run --name news-db -e POSTGRES\_PASSWORD=1234 -p 5432:5432 -d postgres
5. Create the table:

sql

CopyEdit

CREATE TABLE news\_analytics (

Id SERIAL PRIMARY KEY,

author VARCHAR(50),

description VARCHAR(700),

sentiment\_score DOUBLE PRECISION,

timestamp VARCHAR(20)

);

**🐳 Docker Setup**

**Build & Run Dashboard:**

bash

CopyEdit

docker build -f Dockerfile.dashboard -t news-dashboard .

docker run -p 8501:8501 news-dashboard

**Build & Run Fetcher:**

bash

CopyEdit

docker build -f fetcher.dockerfile -t news-fetcher .

docker run news-fetcher

**☁️ AWS Deployment Overview**

**1. Amazon RDS (PostgreSQL)**

* Launch DB instance
* Open port 5432 in Security Group
* Create news\_analytics table

**2. Amazon S3**

* Create bucket (e.g., news-raw-data-2025)
* Store raw fetched JSON files

**3. Amazon ECS (Fargate)**

* Create ECS cluster: news-dashboard-cluster
* Define task definitions:
  + news\_dashboard\_task (Streamlit dashboard)
  + fetcher\_task (news fetcher)
* Deploy services with ALB

**4. AWS Lambda (Trigger Fetch Task)**

python

CopyEdit

import boto3

def lambda\_handler(event, context):

client = boto3.client('ecs')

response = client.run\_task(

cluster='news-dashboard-cluster',

launchType='FARGATE',

taskDefinition='fetcher\_task',

networkConfiguration={

'awsvpcConfiguration': {

'subnets': ['subnet-xxxx'],

'securityGroups': ['sg-xxxx'],

'assignPublicIp': 'ENABLED'

}

}

)

print("Task started:", response)

**5. Amazon EventBridge**

* Schedule the Lambda function to run hourly (or as needed)

**📈 Dashboard Features**

* Histogram of sentiment scores
* Table view with color-coded sentiment
* Timeline-based trend charts
* Filtered analytics (optional)

**✅ Project Outcomes**

* Automated news ingestion
* Real-time sentiment analytics
* Interactive web dashboard
* Serverless AWS-based deployment
* Scalable and modular architecture