



(Final Presentation)

ADVANCED OPEN-SOURCE INTELLIGENCE (OSINT) PLATFORM TO COMBAT MISINFORMATION ON SOCIAL MEDIA

(based on Boeing's Project 2 - **Using OSINT for Social Good**)

Group - 3

Laxman Muthe

Ran Lyu

Shekhar Kumar

Sourav Mondal

Srikar Ummineni



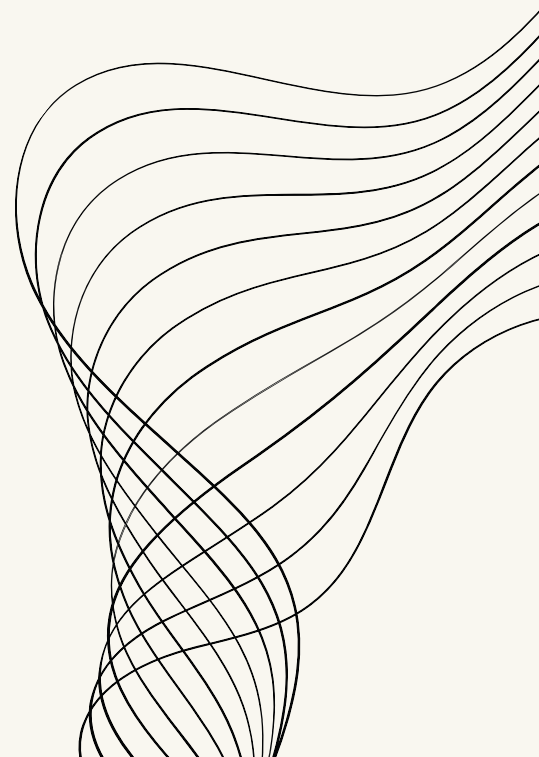
Background and Problem Statement

- Social media platforms have become the primary source of information for millions.
- Present day challenges:
 - spread of misinformation like fake news
 - hate speech
 - impact on public trust and sentiment
 - challenging for users to identify and differentiate from legitimate content.
- As platforms such as Twitter grow in influence, it becomes essential to develop tools that can:
 - Detect misinformation quickly and accurately.
 - Analyze its impact.
 - Mitigate its spread.



Proposed Solution

- **Development of an AI-powered Open-Source Intelligence (OSINT) platform**
 - Utilizes Machine Learning, Natural Language Processing (NLP)
 - Designed to detect and combat misinformation.
 - Designed to detect hate speech
- **Real-time analysis:**
 - Processes tweets and textual data from social media platforms like Twitter.
- **Ease of access:**
 - Browser Extension(Mozilla Firefox, Google Chrome, etc.)
 - Application Programming Interface (API)
- **Primary goal:**
 - Aims to contribute to a safer and more reliable social media ecosystem.



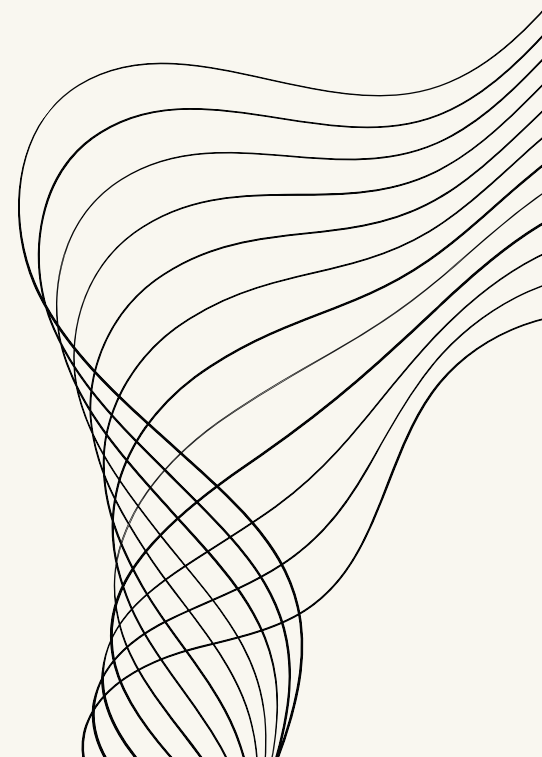
Target Users and Features

- **Target Users**

- social media users
- researchers and journalists
- digital content monitors
- news organizations and fact-checking institutions

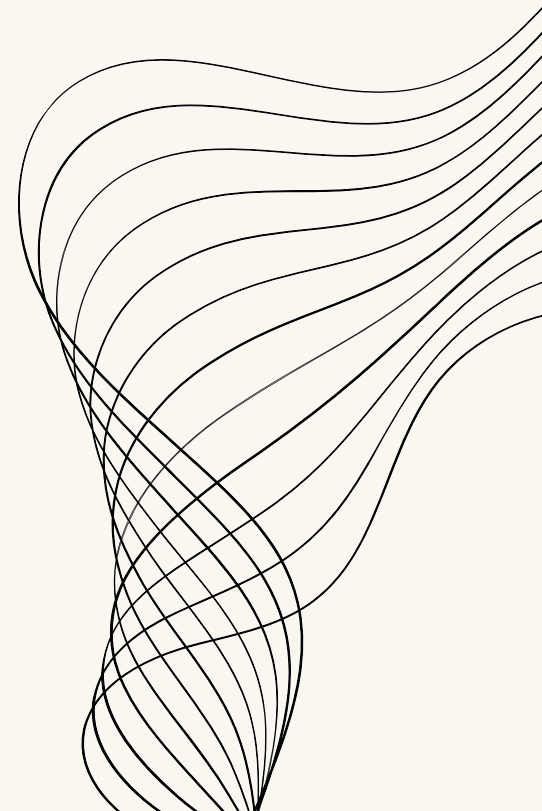
- **Unique Features:**

- real-time analysis of tweets (credibility and sentiment)
- ground truth verification and related article retrieval
- user feedback loop for reducing bias
- ease of access through APIs, UI and browser extensions






Work Accomplished

- **Engineering**
 - Browser Extension & Webpage UI - **Completed**
 - RestAPI Implementation & Integration - **Completed**
 - User Feedback Loop - **Completed**
 - Ground Truth Verification - **Completed**
- **Cloud Infrastructure Setup**
 - API Stack - **Completed**
 - ETL Pipeline - **Completed**
- **Model Building**
 - Fake News Classifier
 - Embedding and CLF Integration - **Completed**
 - Fine-Tuning - **Completed**
 - Sentiment Analysis
 - BERT - **Completed**






IDEs and Tools

- **IDEs**

-  Visual Studio Code
-  PyCharm
-  Google Colab



- **Libraries and Frameworks**

-  PyTorch and TensorFlow (for deep learning)
-  FastAPI (for API endpoints)
-  Reactjs






- **API Development and Testing**

-  Postman

- **Browser and Extensions**

-  Google Chrome
-  Mozilla Firefox

- **Cloud Services (AWS)**

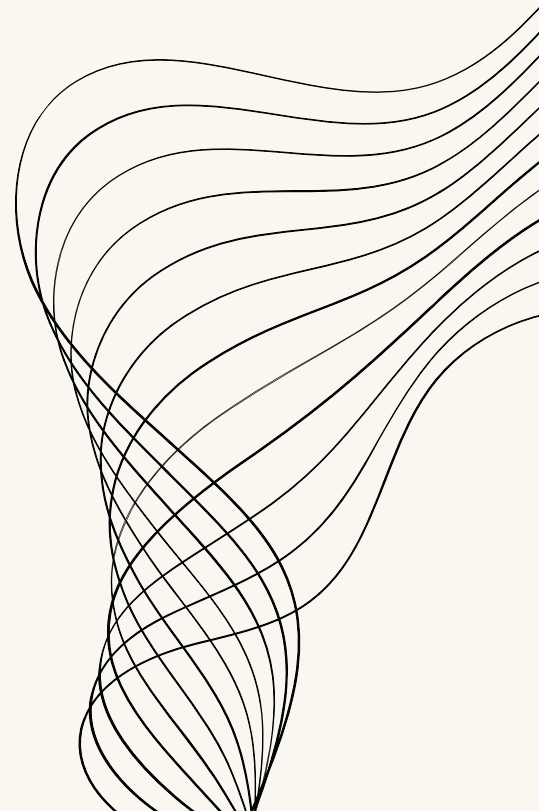
-  Glue
-  Amplify (continuous deployment of the React application)
-  Application Load Balancer(ALB)
-  Elastic Container Service
-  Fargate

- **Hardware Acceleration**

-  NVIDIA CUDA

- **Containerization**

-  Docker



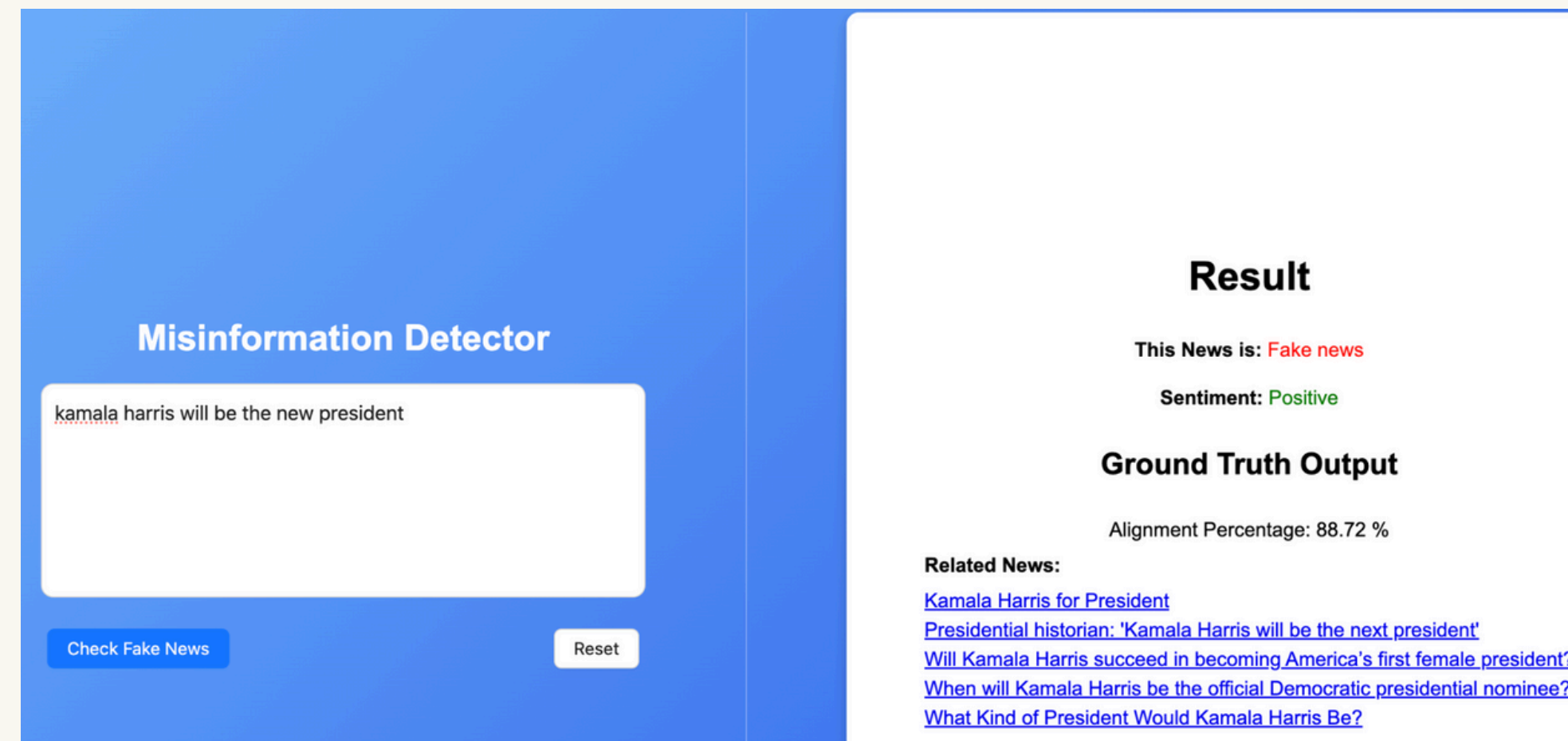
UI Design and API Integration

- **Browser Extension Integration**

- Status: Pending review
- Support domains: [X.com]
- Highlight: real time detection, feedback feature

- **Landing web page**

- URL
- The highlight of the webpage is its flexibility and Perplexity-like results, offering unique features that other fake news detector products lack.



Fake News Classifier

- **Dataset**

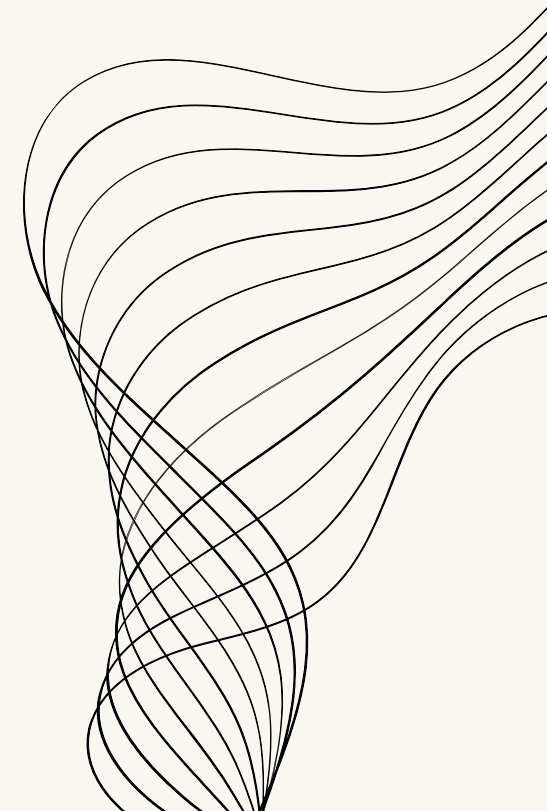
- Kaggle Link
- **180,000** labelled tweets.
- Binary classification: **True/False** labels.

- **Model Building**

- Fine-tuned **pre-trained BERT** (Bidirectional Encoder Representations from Transformers) for tweet classification.
 - understand contextual nuances in a short and noisy text.
 - pre-trained model can be fine-tuned on smaller datasets.
 - high accuracy even with limited data.
- Used **attention mask** for fine-tuning BERT.
- Train-test split: **80:20**
- The sequence length used is **128**.
- Padding is **enabled**.
- Truncation is **enabled**.

```
Epoch 1, Loss: 0.6574950767755509
Epoch 2, Loss: 0.6133579725027084
Epoch 3, Loss: 0.5892843220829964
Epoch 4, Loss: 0.5647664853334426
Epoch 5, Loss: 0.5550374411344529
Epoch 6, Loss: 0.5361743159294129
Epoch 7, Loss: 0.5191193963587284
Epoch 8, Loss: 0.5202245606780053
Epoch 9, Loss: 0.5126936365365982
Epoch 10, Loss: 0.5094907628297806
```

Epoch vs Loss

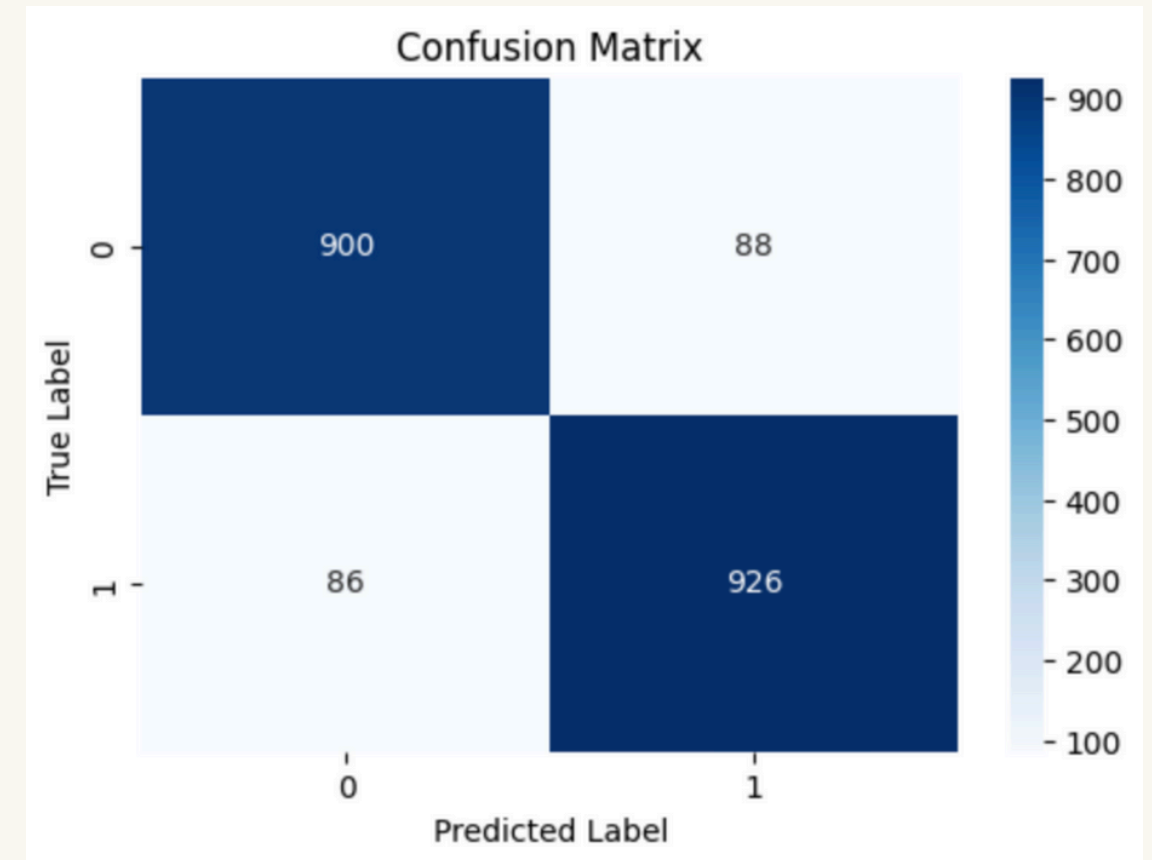


Fake News Classifier

- **Hyperparameters**

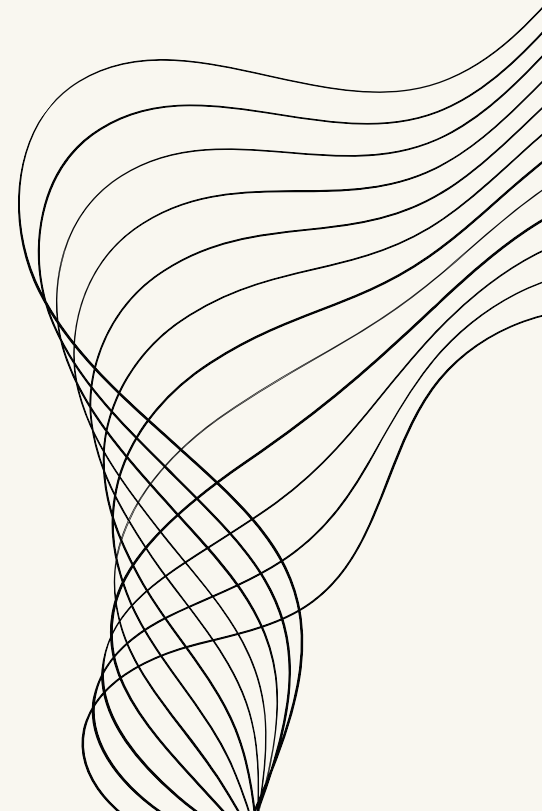
- Optimizer: AdamW
- Learning Rate: 0.01
- Weight Decay: 0.01
- Epochs: 20
- Batch Size: 32

- We have fine-tuned the BERT model, added an attention mask, and built a separate classifier layer on top of BERT.



Fake news classifier confusion matrix

Classification Report:				
	precision	recall	f1-score	support
0	0.91	0.91	0.91	988
1	0.91	0.92	0.91	1012
accuracy			0.91	2000
macro avg	0.91	0.91	0.91	2000
weighted avg	0.91	0.91	0.91	2000



Sentiment Analysis

Overview and Preprocessing

1. **Overview:** A method to evaluate text and determine its tone (positive or negative).
2. **Dataset Used:** Sentiment140 Dataset - 1.6 million tweets labeled for sentiment based on emoticons.
3. **Data Preprocessing:**
 - Data sourced from Twitter with labels for sentiment.
 - Steps: Removed URLs, mentions, and stop words; converted text to lowercase.
 - Balanced the dataset for equal representation of sentiments.

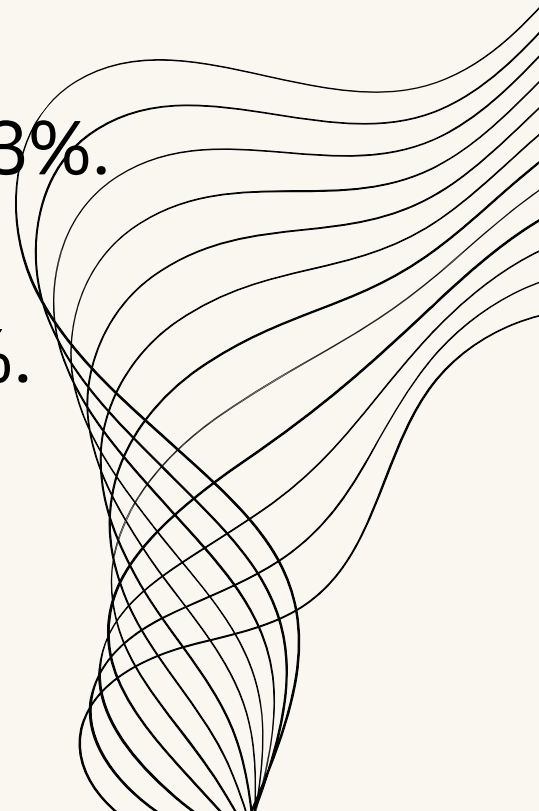
Model Development and Results

1. Models Used:

- Logistic Regression: Trained with TF-IDF features; achieved an accuracy of 71.83%.
- BERT Model: Transitioned to BERT, increasing accuracy to 75.47%.
- Expanded dataset to 100,000 rows, further boosting BERT accuracy to 80.03%.

2. Results:

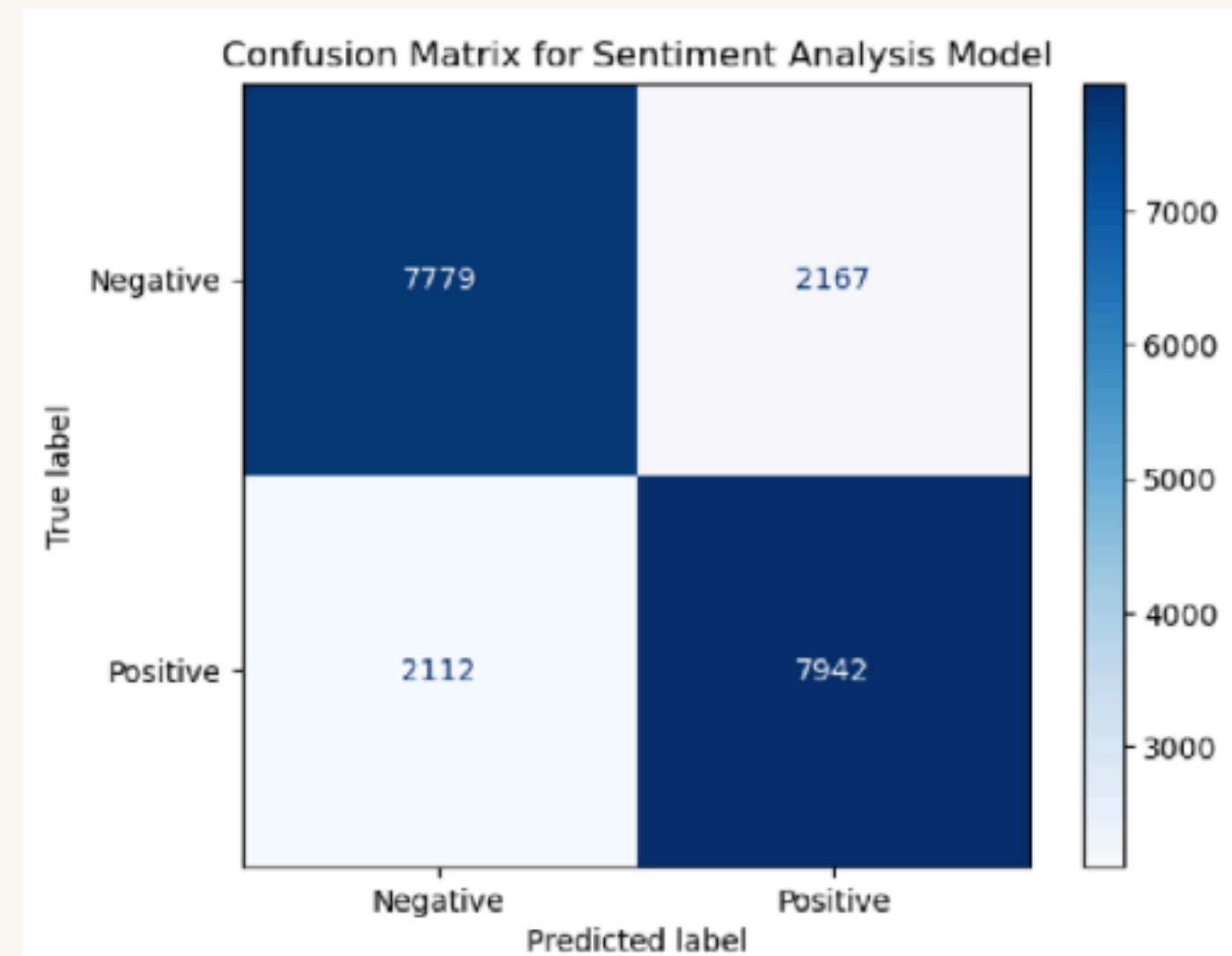
- Evaluation Metrics: Accuracy, precision, recall, F1 score.
- Confusion Matrix: Shows distribution of true vs. predicted sentiment classes.



- Additionally, through hyperparameter tuning, we optimized the model's performance, further increasing accuracy from 75% to 80%. These adjustments enhanced the model's ability to capture nuanced sentiment, solidifying its role as a robust tool in our sentiment analysis pipeline.

Accuracy	0.8003
Precision	0.8004

Recall	0.8003
F1	0.8004



ETL Pipeline

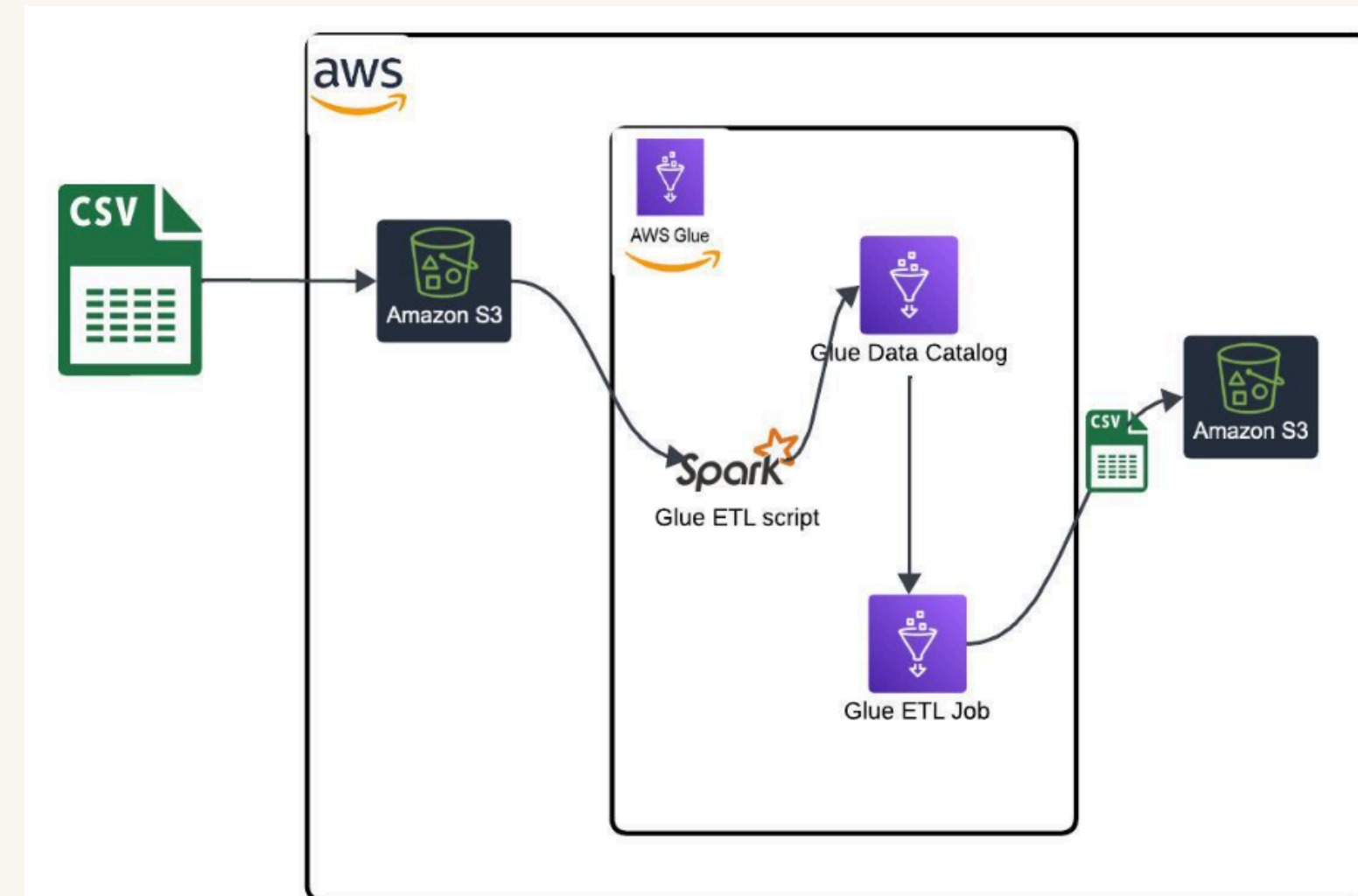
Overview : Integrated AWS Glue, a robust ETL tool, to streamline data preprocessing, reducing delays in preparing the dataset for model training.

Process Flow:

- Data Upload to S3
- Schema Discovery- Configured AWS Glue Crawler
- ETL Job Creation
 - Created an AWS Glue ETL job
 - Developed script to preprocess data
- Data Transformation and Storage to S3 bucket

Key Benefits:

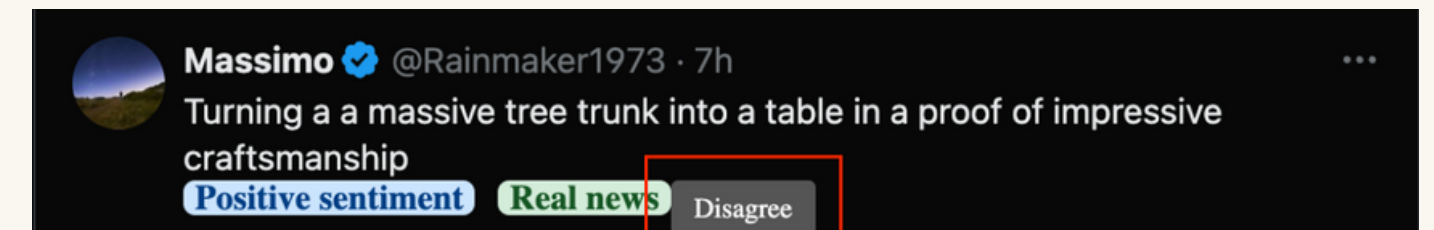
- Improved preprocessing speed.
- Automated schema discovery and data transformation
- Efficient and scalable data handling with AWS Glue and S3.



User Feedback and Ground Truth

- **User Feedback Loop**

- Users provide feedback on the model's predictions directly via browser extension.
- Feedback is stored in a PostgreSQL database for later use.
- A cron job triggers the retraining script when feedback entries exceed a predefined threshold and frequency limit ensures retraining is not triggered too often.
- Dynamically corrects systemic biases over time.



User feedback UI button

id	tweet_id	tweet_text	predicted_label	feedback_label	disagree_count
1	287365289	This tweet is part of feedback!	Positive	Negative	2

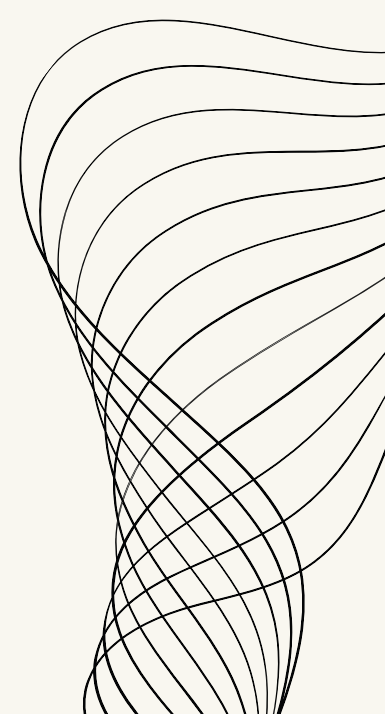
User feedback database view

- **Ground Truth Verification**

- Deep learning models often lack accuracy in verifying claims against real-world facts.
- Use SerpAPI to fetch relevant news articles from Google News based on the input text.
- Encode input text and article titles using SentenceTransformer (all-MiniLM-L6-v2).
- Determine validity by checking if any article exceeds the similarity threshold of 0.75.

A screenshot of a "Result" page. It shows "This News is: Fake news" in red, "Sentiment: Positive" in green, and "Ground Truth Output" in bold. Below that is "Alignment Percentage: 88.72 %". A red box highlights a "Related News:" section with five blue links: "Kamala Harris for President", "Presidential historian: 'Kamala Harris will be the next president'", "Will Kamala Harris succeed in becoming America's first female president?", "When will Kamala Harris be the official Democratic presidential nominee?", and "What Kind of President Would Kamala Harris Be?".

Ground truth verification response



Deployment on AWS

- **Deployment Tools**

- Utilized VPC, ECR, ECS, ALB, Route 53, and Amplify for deploying the backend and front-end.

- **Backend Setup**

- Developed FastAPI endpoints for predictions using sentiment analysis and fake news detection models.
- Packaged APIs and dependencies into a Docker container and pushed the image to ECR.

- **Container Orchestration**

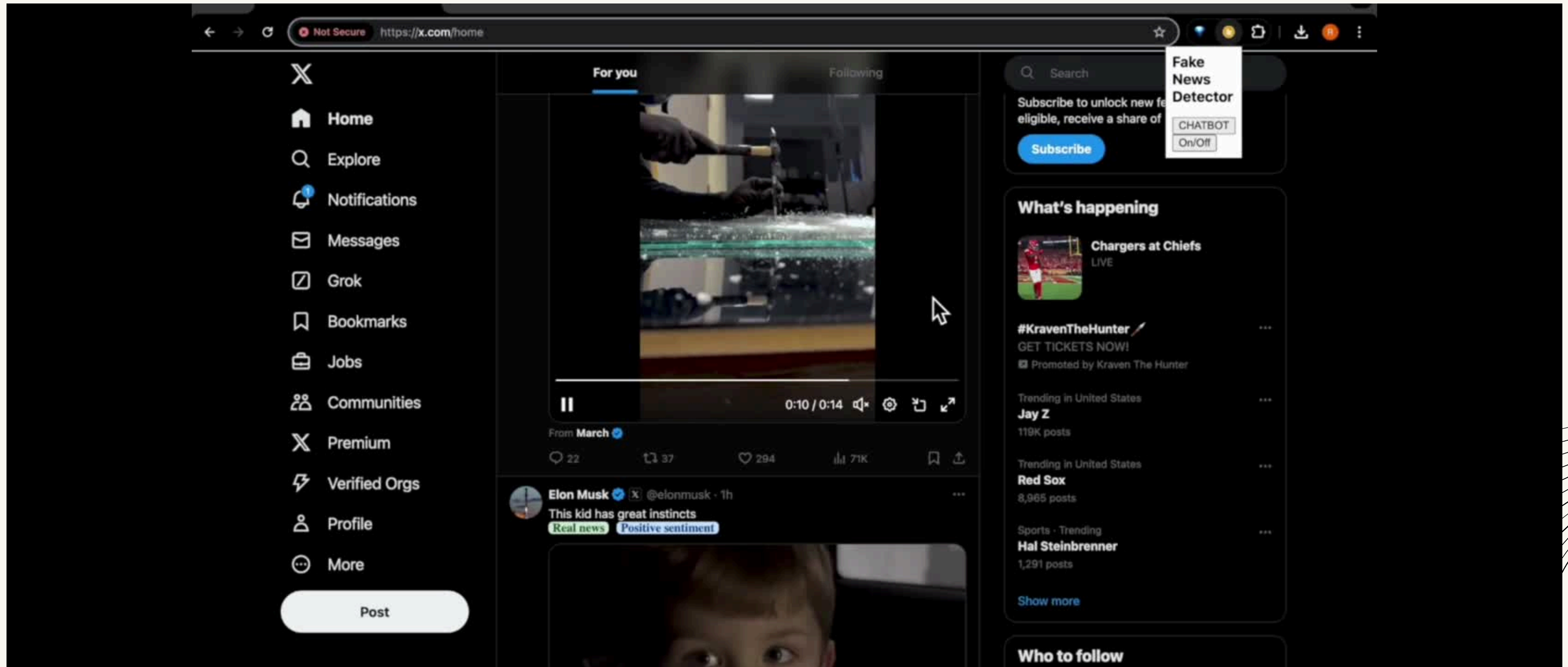
- Hosted the container on Fargate in a private subnet using ECS.
- Connected the container to an ALB within the same subnet.

- **DNS Configuration**

- Registered the ALB's DNS with a purchased domain using Route 53.



Product Demonstration



Challenges Faced

- **Resource Limitations**

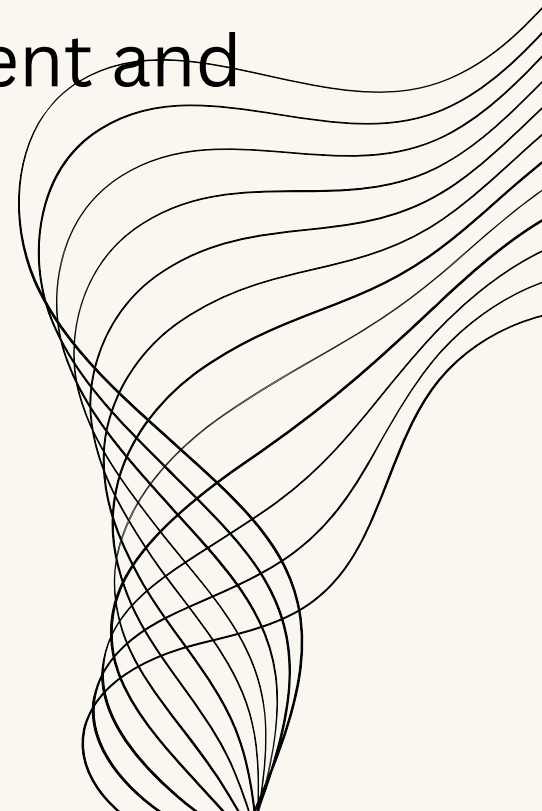
- Training models on larger datasets was computationally intensive, causing delays; while GPU acceleration (CUDA) helped, additional computational resources would enable faster and more effective training.

- **Data and API Challenges**

- Inefficient local preprocessing and issues with Twitter/X API access led to reliance on alternative datasets (e.g., Kaggle), limiting real-time data processing capabilities.

- **Delayed Feedback Integration & Costs**

- Backend integration for user feedback was delayed, restricting model refinement, while AWS service costs posed challenges, highlighting the need for resource management and AWS credits for scaling.



Reflection and Retrospection

- **Sprint Overview**

- The project aimed to develop a Fake News Detection and Sentiment Analysis system, complete with cloud infrastructure, data pipelines, and an integrated UI .

- **Key Achievements Across Sprints**

- **Sprint 1:** Cloud infrastructure setup, initial data collection, and UI design initiated.
- **Sprint 2:** Improved fake news model accuracy (~76%), sentiment analysis model accuracy (~78%), and implemented AWS Glue for data processing.
- **Sprint 3:** Finalized fake news model accuracy (~91%), sentiment analysis model (~80%), integrated APIs with UI, User Feedback and Ground Truth

- **Lessons Learned**

- **What Went Well:** Enhanced model accuracy, user feedback integration, and web extension.
- **What Could Be Improved:** Further improve accuracy with more records and training time.



Future Works

- Sentiment Analysis: enhance the performance of the BERT model by incorporating domain-specific data
- Data Pipeline: further improve the efficiency and scalability of our data pipeline
- Additional features:
 - a. User feedback data strategy
 - b. Extending browser extension support more news site
 - c. Image format data support



Thank you