Name:Serene Dmello
Roll no:10181
TE Comps- A

OSINT LAB 7: AUTOMATED SOCIAL MEDIA OSINT AGGREGATION PIPELINE

FINAL PROJECT REPORT

1. Introduction

Open Source Intelligence (OSINT) refers to the collection and analysis of publicly available information for intelligence purposes. In today's digital age, social media platforms are rich sources of real-time public data. This project aimed to design and implement an automated OSINT pipeline capable of collecting, cleaning, enriching, and storing data from multiple social media platforms for intelligence analysis.

Objective:

To build a modular Python-based pipeline that aggregates data from Twitter, Reddit, and GitHub, performs sentiment analysis, stores results in a structured database, and supports automated scheduled collection.

2. Methodology

2.1 Tools and Technologies Used

Programming Language: Python 3.9+

Libraries:

tweepy – Twitter API integration

praw – Reddit API integration

PyGithub – GitHub API integration

textblob – Sentiment analysis

sqlite3 – Database management

pandas, matplotlib – Data processing and visualization

python-dotenv – Environment variable management

schedule – Automation scheduling

2.2 Pipeline Architecture

The pipeline consists of four main modules:

Data Collection: Platform-specific collectors using official APIs (Twitter, Reddit, GitHub).

Data Cleaning: Removal of URLs, special characters, and non-English text.

Data Enrichment: Sentiment analysis using TextBlob.

Data Storage: SQLite database with unified schema.

2.3 Platforms Integrated

Twitter: Used Tweepy with API v2 Bearer Token.

Reddit: Used PRAW with OAuth2 credentials.

GitHub: Used PyGithub with personal access token.

3. Results

3.1 Data Collected

Total Records: 118

Platform Distribution:

Reddit: 68 records

GitHub: 50 records

Sentiment Analysis:

Average Sentiment Score: 0.095

Positive Records: 46

Negative Records: 12

Neutral Records: 60

3.2 Sample Records

Platform	USer	Timestamp	Text	URL	Sentiment
Reddit	Valinaut	2023-11-05	Cleaned text from post	https://www.reddit.com/	0.00
GitHub	tenserflow	2023-11-04	Tenserflow repository	https://github.com/	0.15

3.3 Visualizations

Sentiment distribution histogram

Platform-wise record count bar chart

(Screenshots attached in repository)

4. Challenges Faced

4.1 API Rate Limiting

Twitter: Strict rate limits (450 requests/15 min) and authentication barriers.

Instagram: Blocked requests due to anonymous scraping; required login and still faced 429 errors.

Solution: Implemented retry logic, delays, and fallback scraping (snscrape).

4.2 Database Schema Mismatch

Initially, the sentiment column was missing, causing insert failures.

Solution: Modified database.py to auto-detect and fix schema issues.

4.3 Authentication Issues

Twitter API v2 requires Bearer Token and elevated access.

Reddit API requires user-agent and OAuth2 setup.

5. Conclusion

5.1 Key Achievements

Successfully collected 118 records from Reddit and GitHub.

Implemented end-to-end pipeline: collection → cleaning → enrichment → storage.

Automated the process using a scheduler.

Performed sentiment analysis and visualizations.

5.2 Future Improvements

Integrate more platforms: Telegram, LinkedIn, Facebook.

Add geolocation and entity extraction (names, locations, orgs).

Implement real-time dashboard with Streamlit or Flask.

Use proxies and rotating user agents to avoid rate limits.