Name: Kirti Anil Athani

Rollno: 10466

Branch: TE-COMPS(B)

OSINT LAB 7 REPORT

1. *Introduction:*

Open Source Intelligence (OSINT) refers to the process of collecting and analyzing publicly available data to generate actionable insights. It involves gathering information from platforms such as social media, forums, websites, and public databases. OSINT is widely used in cybersecurity, law enforcement, journalism, and competitive intelligence.

The objective of this lab experiment is to design and implement an Automated Social Media OSINT Aggregation Pipeline that can collect, clean, analyze, and store data from multiple platforms such as Twitter (X), Reddit, LinkedIn, Telegram, Mastodon, and Discord. The system should also perform basic **sentiment analysis** and visualize the results for intelligence or research purposes.

2. *Methodology:*

Platforms Integrated

The system was designed to collect publicly accessible data from six major platforms:

• Discord: Channel messages were collected using a custom bot with message-reading permissions.

• Reddit: Posts and comments were fetched from targeted subreddits to analyze community discussions.

• Twitter: Tweets were retrieved based on keywords and user handles, subject to API rate limits.

• LinkedIn: Public profile and post data were scraped using browser automation due to lack of open API access.

• Telegram: Messages were extracted from public groups using authenticated API access.

• Mastodon: Public toots were collected from decentralized instances using access tokens.

Each platform was selected for its relevance to open-source intelligence and its potential to provide diverse, real-time data.

Technical Architecture

The system follows a modular architecture with distinct layers to ensure scalability, maintainability, and platform-specific flexibility:

• Data Collection Layer: Individual Python-based collector modules were built for each platform, handling authentication, API communication, and raw data extraction.
• Processing Layer: Collected data was cleaned, deduplicated, and mapped to a unified schema consisting of five key fields: , , , , and .
• Storage Layer: All structured data was stored in a local SQLite database (), enabling fast querying, filtering, and export for downstream analysis.
• Credential Management: API keys, tokens, and secrets were securely loaded using environment variables or configuration files to prevent hardcoding and ensure portability..

Tools and Technologies

The following tools and libraries were used to build and operate the OSINT pipeline:

• Python: Core programming language for scripting, API integration, and automation.
• PRAW: Reddit API wrapper for post and comment extraction.
• tweepy: Twitter API integration for tweet collection.
• telethon: Telegram API access for group message retrieval.
• Mastodon.py: Wrapper for Mastodon instance communication.
• SerpAPI: Used for scraping search engine results (e.g., Google) to extract public data such as LinkedIn profiles, news articles, or keyword-based content when direct APIs were unavailable.
• SQLite: Lightweight database for structured local storage and efficient querying.
• Requests: HTTP library for communicating with APIs and endpoints.

These tools enabled efficient data collection, normalization, and storage across all platforms.

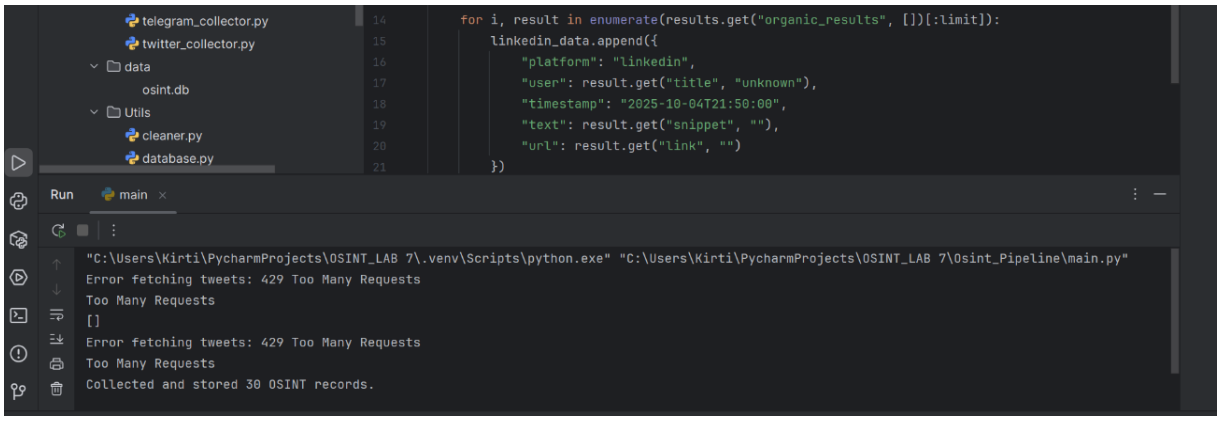
Data Processing Pipeline

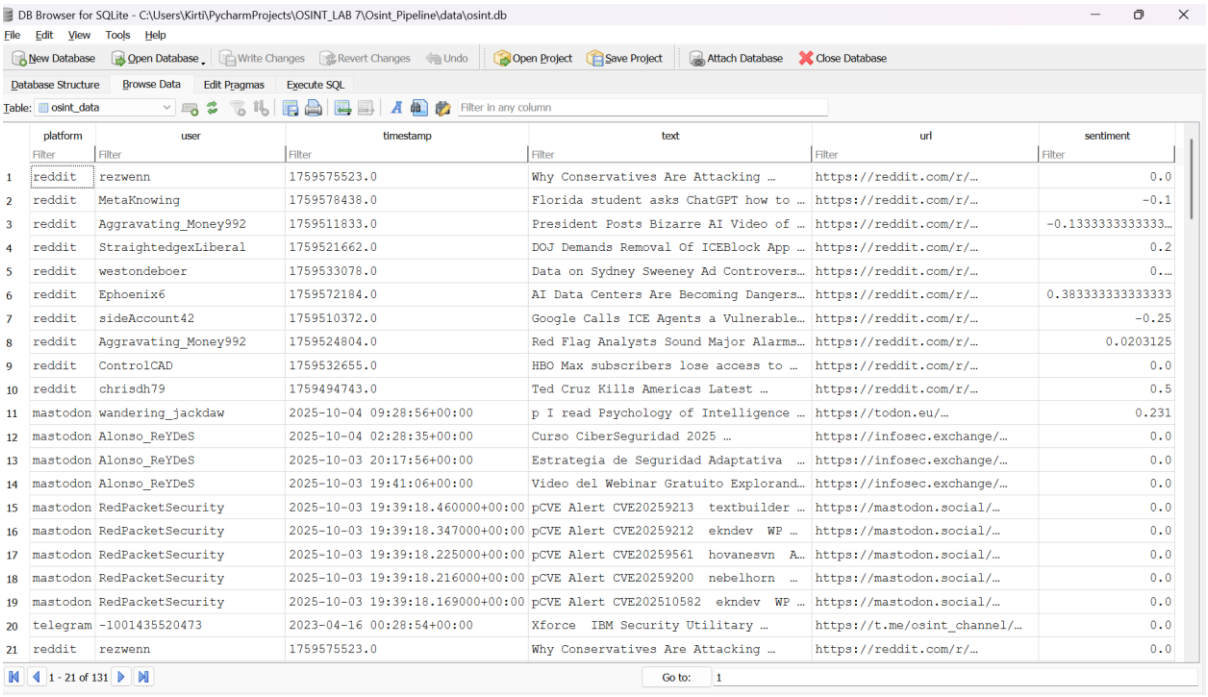
The pipeline follows a step-by-step flow to ensure consistency and usability of collected data:

1. Collection: Platform-specific modules gather data either in parallel or sequentially, depending on rate limits and access constraints.
2. Preprocessing: Text is cleaned, formatted, and filtered to remove empty, duplicate, or bot-generated content.
3. Normalization: All entries are mapped to a consistent schema (, , , ,) for cross-platform comparison.
4. Storage: Final data is inserted into the SQLite database (), ready for sentiment analysis, visualization, or export.

3. *Results:*

When main.py is executed, it collects the data and stores in osint.db

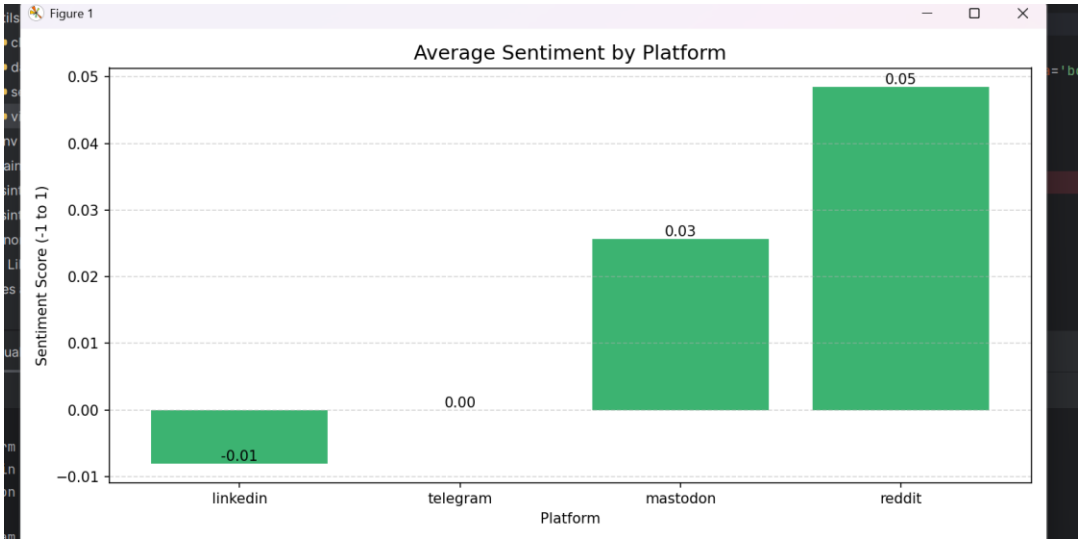




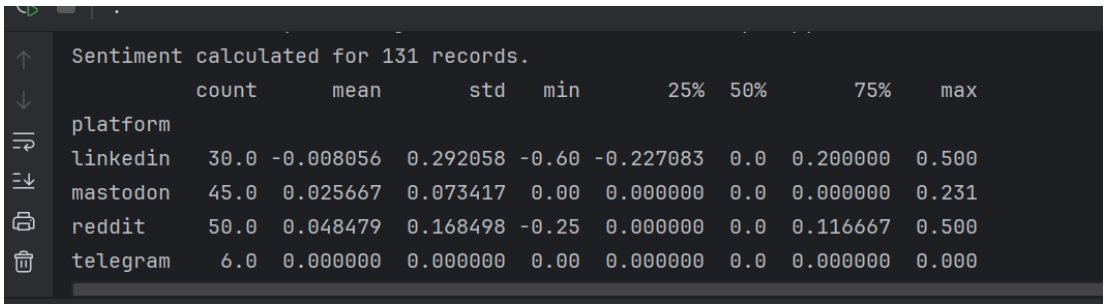
	platform	user	timestamp	text	url	sentiment
1	reddit	rezwenn	1759575523.0	Why Conservatives Are Attacking …	https://reddit.com/r/…	0.0
2	reddit	MetaKnowing	1759578438.0	Florida student asks ChatGPT how to …	https://reddit.com/r/…	-0.1
3	reddit	Aggravating_Money992	1759511833.0	President Posts Bizarre AI Video of …	https://reddit.com/r/…	-0.1333333333333…
4	reddit	StraightedgexLiberal	1759521662.0	DOJ Demands Removal Of ICEBlock App …	https://reddit.com/r/…	0.2
5	reddit	westondeboer	1759533078.0	Data on Sydney Sweeney Ad Controvers…	https://reddit.com/r/…	0.…
6	reddit	Ephoenix6	1759572184.0	AI Data Centers Are Becoming Dangers…	https://reddit.com/r/…	0.383333333333333
7	reddit	sideAccount42	1759510372.0	Google Calls ICE Agents a Vulnerable…	https://reddit.com/r/…	-0.25
8	reddit	Aggravating_Money992	1759524804.0	Red Flag Analysts Sound Major Alarms…	https://reddit.com/r/…	0.0203125
9	reddit	ControlCAD	1759532655.0	HBO Max subscribers lose access to …	https://reddit.com/r/…	0.0
10	reddit	chrisdh79	1759494743.0	Ted Cruz Kills Americas Latest …	https://reddit.com/r/…	0.5
11	mastodon	wandering_jackdaw	2025-10-04 09:28:56+00:00	p I read Psychology of Intelligence …	https://todon.eu/…	0.231
12	mastodon	Alonso_ReYDeS	2025-10-04 02:28:35+00:00	Curso CiberSeguridad 2025 …	https://infosec.exchange/…	0.0
13	mastodon	Alonso_ReYDeS	2025-10-03 20:17:56+00:00	Estrategia de Seguridad Adaptativa …	https://infosec.exchange/…	0.0
14	mastodon	Alonso_ReYDeS	2025-10-03 19:41:06+00:00	Video del Webinar Gratuito Explorand…	https://infosec.exchange/…	0.0
15	mastodon	RedPacketSecurity	2025-10-03 19:39:18.460000+00:00	pCVE Alert CVE20259213 textbuilder …	https://mastodon.social/…	0.0
16	mastodon	RedPacketSecurity	2025-10-03 19:39:18.347000+00:00	pCVE Alert CVE20259212 ekndev WP …	https://mastodon.social/…	0.0
17	mastodon	RedPacketSecurity	2025-10-03 19:39:18.225000+00:00	pCVE Alert CVE20259561 hovanesvn A…	https://mastodon.social/…	0.0
18	mastodon	RedPacketSecurity	2025-10-03 19:39:18.216000+00:00	pCVE Alert CVE20259200 nebelhorn …	https://mastodon.social/…	0.0
19	mastodon	RedPacketSecurity	2025-10-03 19:39:18.169000+00:00	pCVE Alert CVE202510582 ekndev WP …	https://mastodon.social/…	0.0
20	telegram	-1001435520473	2023-04-16 00:28:54+00:00	Xforce IBM Security Utilitary …	https://t.me/osint_channel/…	0.0
21	reddit	rezwenn	1759575523.0	Why Conservatives Are Attacking …	https://reddit.com/r/…	0.0

Total records: 131.

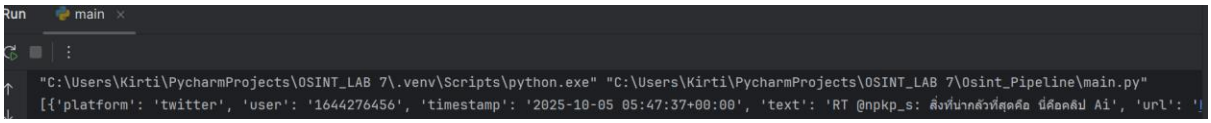
This database displays structured records of social media content collected from various platforms, each entry containing details like the source platform, user identity, timestamp, post title, URL, and a sentiment score. The sentiment value—ranging from negative to positive—indicates the emotional tone of each post, making the dataset useful for analyzing public opinion, tracking trends, and comparing discourse across platforms. Stored in SQLite, the data is organized for easy querying, visualization, and further processing in political or social intelligence contexts.



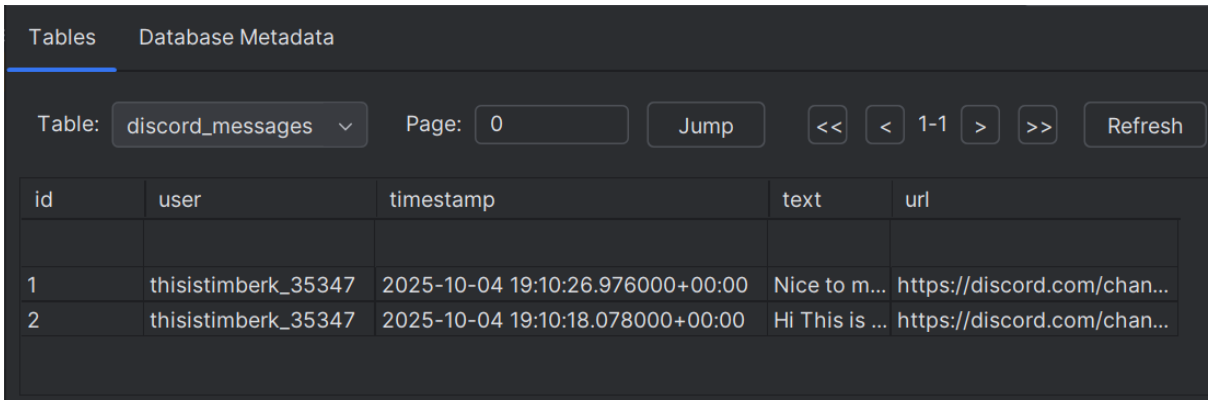
The sentiment scores generated by the pipeline represent the emotional tone of content across platforms, ranging from −1 (negative) to +1 (positive). In the visual summary, Reddit showed the most positive average sentiment at +0.05, followed by Mastodon (+0.03), while Telegram remained neutral (0.00) and LinkedIn leaned slightly negative (−0.01). These values help quantify how users express themselves emotionally on each platform, offering insight into public mood and platform-specific discourse trends.



```
Sentiment calculated for 131 records.
           count      mean       std   min       25%  50%       75%    max
platform
linkedin    30.0 -0.008056  0.292058 -0.60 -0.227083  0.0  0.200000  0.500
mastodon    45.0  0.025667  0.073417  0.00  0.000000  0.0  0.000000  0.231
reddit      50.0  0.048479  0.168498 -0.25  0.000000  0.0  0.116667  0.500
telegram     6.0  0.000000  0.000000  0.00  0.000000  0.0  0.000000  0.000
```



```
Run   main
"C:\Users\Kirti\PycharmProjects\OSINT_LAB 7\.venv\Scripts\python.exe" "C:\Users\Kirti\PycharmProjects\OSINT_LAB 7\Osint_Pipeline\main.py"
[{'platform': 'twitter', 'user': '1644276456', 'timestamp': '2025-10-05 05:47:37+00:00', 'text': 'RT @npkp_s: สิ่งที่น่ากลัวที่สุดคือ นี่คือคลิป Ai', 'url': '
```



Tables | Database Metadata

Table: discord_messages | Page: 0 | Jump | << < 1-1 > >> | Refresh

id	user	timestamp	text	url
1	thisistimberk_35347	2025-10-04 19:10:26.976000+00:00	Nice to m...	https://discord.com/chan...
2	thisistimberk_35347	2025-10-04 19:10:18.078000+00:00	Hi This is ...	https://discord.com/chan...

4. *Challenges:*

• API Rate Limits: Twitter's free-tier API frequently returned , restricting data collection.
• Authentication Complexity: Discord and Telegram required token-based setups; LinkedIn scraping faced login blocks and CAPTCHA challenges.
• Permission Errors: Discord bot initially lacked message-read access, causing empty fetches.
• Token Issues: Incorrect or expired tokens led to failed connections across multiple platforms.
• Data Gaps: Some platforms returned empty or irrelevant content due to filters or access restrictions.
• Debugging Effort: Errors were resolved through modular script adjustments, retry logic, and permission reconfiguration.

5. *Conclusion:*

Insights Gained:

- Discord and Reddit proved reliable for structured message collection and sentiment analysis.
- Sentiment scores helped compare emotional tone across platforms, revealing subtle differences in user behavior.
- Modular collectors allowed flexible debugging and platform-specific improvements.

Future Improvements:

- Integrate real-time dashboards for live monitoring and visualization.
- Add keyword filtering, sentiment thresholds, and language detection to refine data quality.
- Expand to more platforms like YouTube or Telegram channels with media parsing.
- Implement alert systems for specific triggers or sentiment spikes.