

Social Media Analyzer

Introduction

The **Social Media Analyzer** is an AI-powered platform designed to analyze large-scale social media interactions, focusing on Reddit datasets. This system extracts insights, trends, and user behavior patterns using **Graph RAG**, **Neo4j**, and **NLP models** to generate meaningful insights from vast amounts of data.

Key Features

- **Reddit Dataset Analysis** – Processes and extracts insights from jsonl Reddit data.
- **Graph RAG-powered Insights** – Uses a combination of *Neo4j* and *NetworkX* for relationship-based analysis.
- **Neo4j-Based Social Network Visualization** – Generates interactive network graphs.
- **Time Series Analysis** – Tracks user engagement over time.
- **Community Distribution** – Identifies the most engaged posts in different communities.
- **Topic Trends Analysis** – Detects trending topics based on user queries.
- **Sentiment & AI Analysis** – Extracts key themes, sentiment, and notable patterns.
- **Chatbot-Powered Query Refinement** – Uses *Llama3-8B-8192* to improve user queries for better results.
- **Dynamic & Interactive Dashboard** – Built using *Next.js* for seamless user interaction.

System Architecture

The system is structured into two main components:

1. Client (Frontend - Next.js)

- Interactive **dashboard** for visualization.
- Query **filters** for refining search results.
- AI-generated **insights panel**.
- **Network graphs** for relationship analysis.

2. AI Server (Backend - FastAPI, Neo4j, NetworkX, Llama 3-8B)

- AI chatbot **refines user queries** for improved results.
- **Graph RAG module** fetches and analyzes relationships from **Neo4j AuraDB**.
- **Time series processing** tracks engagement trends over time.
- **Community distribution** identifies the most engaged posts.
- **Topic trend detection** provides insights into trending discussions.
- **Network Graph Visualization** generates **relationship-based insights** from Neo4j.

- AI-driven **summary generation** extracts **key themes, sentiment, and patterns**.

3. Database & Data Processing

- **Neo4j AuraDB** – Stores social media relationships and interaction data.
- **JSONL & Processed Data** – Stores structured Reddit datasets for further processing.
- **Backend services** handle requests, process data, and generate AI-driven insights.

Folder Structure

```
├── client/                # Frontend application
│   ├── app/              # Application components and pages
│   ├── components/      # Social media analyzer components
│   ├── utils/           # API connection and helper functions
│   └── package.json      # Frontend dependencies
├── ai-server/           # Backend AI server
│   ├── services/        # Core processing and AI model services
│   ├── data/            # JSONL and processed Reddit data
│   ├── scripts/         # Neo4j script for inserting data into Neo4j AuraDB
│   └── main.py           # FastAPI backend server
├── images/              # System architecture and UI screenshots
└── README.md            # Project documentation
```

Installation

Prerequisites

Ensure you have the following installed:

- **Node.js** (for frontend & API calls)
- **Python 3.11.11** (for AI backend)
- **Neo4j AuraDB** (for Graph RAG processing)
- **FastAPI** (for backend REST API)

API Endpoints

Endpoint	Method	Description
/api/init-database	POST	Initializes the database by processing the user query, inserting data into Neo4j, and generating insights.
/api/time-series	GET	Retrieves time-series data representing engagement trends over time, useful for understanding user activity patterns.
/api/community-distribution	GET	Analyzes and returns the distribution of communities discussing a particular topic, helping to identify the most engaged groups.
/api/topic-trends	GET	Fetches the most trending topics based on processed Reddit data, highlighting popular discussions over time.
/api/network-graph	GET	Provides a visualized Neo4j-based network graph, showing connections between users, topics, and communities.
/api/ai-analysis	POST	Performs AI-driven analysis on the given query, extracting key themes, notable patterns, sentiment, and main discussion points.
/api/chatbot	POST	Chatbot powered by Llama 3-8B, refines user queries and provides contextual insights based on processed social media data.

Technologies Used

- **Frontend:** Next.js, React.js, ShadCN, TailwindCSS
 - **Backend:** FastAPI, Python
 - **Database:** Neo4j AuraDB
 - **AI Models:** Llama 3-8B, NetworkX
 - **Data Processing:** JSONL, Graph RAG
-