Project Report: AI Sentiment Analysis on Twitter/X

1. Introduction

This report provides a comprehensive overview of the **AI Sentiment Analysis on Twitter/X** project. The primary objective was to build an automated data pipeline to collect, process, and analyse public sentiment related to the topic of Artificial Intelligence (AI) on the social media platform Twitter (now X). The project culminates in an interactive dashboard that provides real-time insights into public opinion. This report documents the project's methodology, tools, and key findings.

2. Abstract

The **AI Sentiment Analysis** project is a multi-stage data processing and visualization system. The pipeline begins by extracting tweets containing specific keywords using the Twitter API. These raw tweets are then subjected to a robust cleaning and sentiment analysis process, which includes removing noise, translating non-English text, and assigning sentiment scores. The final, processed dataset is then fed into a **Streamlit** dashboard, which displays key metrics and interactive charts for sentiment distribution and trends over time. This project successfully demonstrates a full-cycle data science workflow, from data acquisition to interactive visualization.

3. Tools and Technologies

The project was developed using a range of Python-based tools and libraries to handle different stages of the data pipeline:

- **Python:** The core programming language for the entire project.
- **Tweepy:** A library used to connect to the Twitter/X API for fetching tweets.
- **Pandas:** The essential library for data manipulation and analysis.
- **Streamlit:** An open-source app framework used to create the interactive web dashboard.
- **Altair:** A powerful declarative library for creating beautiful and interactive statistical visualizations.
- **NLTK (Natural Language Toolkit):** Used specifically for the **VADER** (Valence Aware Dictionary and Sentiment Reasoner) sentiment analyser to score tweets.
- **langdetect & deep_translator:** Libraries used for language detection and translation of tweets, ensuring accurate sentiment analysis regardless of the original language.

4. Steps Involved in Building the Project

The project was executed in three main stages, each handled by a dedicated Python script.

Stage 1: Tweet Extraction (stream_tweets.py)

This script serves as the data acquisition layer of the pipeline. It connects to the Twitter/X API using a bearer token and queries for recent tweets based on predefined keywords (in this case, related to "AI"). The script fetches up to 50 recent tweets and prints the timestamp and text of each tweet to a text file named tweets.txt. This step is crucial for creating the raw dataset for subsequent analysis.

Stage 2: Data Cleaning and Sentiment Analysis (cleaning_and_sentiment.py)

This script is the core processing engine. It reads the raw tweets from tweets.txt and performs several key operations:

- **Text Cleaning:** It removes noise such as retweets (RT), user mentions (@), hashtags (#), and URLs. It also cleans up non-ASCII characters and extra white spaces.
- **Language Translation:** Using langdetect and deep_translator, the script identifies tweets not in English and automatically translates them. This ensures that the sentiment analysis model works accurately on all data.
- **Sentiment Scoring:** The **NLTK VADER** analyser is used to calculate sentiment scores (compound, positive, negative, and neutral). Based on the **compound score**, a final sentiment label ('Positive', 'Negative', or 'Neutral') is assigned to each tweet.
- **Data Export:** The script saves the cleaned and labelled data as a CSV file (cleaned_sentiment_tweets.csv), which is ready for visualization.

Stage 3: Dashboard Visualization (streamlit_sentiment_dashboard.py)

This final stage brings the project to life as an interactive dashboard. The **Streamlit** script reads the processed data from the CSV file and presents it through a user-friendly interface.

- **Summary Metrics:** The dashboard displays key performance indicators at the top, including the total number of tweets and the average sentiment score.
- **Sentiment Distribution:** A bar chart, built with **Altair**, visually represents the count of positive, negative, and neutral tweets, giving an immediate overview of public opinion.
- **Sentiment Trend:** An hourly line chart tracks the average sentiment over time, allowing users to identify trends and shifts in public sentiment throughout the day.

5. Conclusion

The successful completion of this project demonstrates a comprehensive understanding of a modern data science workflow, from data collection and processing to visualization. While the project uses a static dataset due to the limitations of a free API account, the pipeline is fully functional and scalable. The dashboard effectively translates complex textual data into simple, actionable insights, making it a valuable tool for anyone interested in tracking public opinion on AI or any other topic. This project serves as a strong foundation for future enhancements, such as integrating real-time data streaming or deploying the dashboard to a live server.