Geo-Matrix AI: A Multi-Modal Fusion Model for Hyper-Accurate IP Geolocation

Team Name: Protocol Pioneers

Table of Contents

1. Problem Description

The accurate geolocation of an IP address to a specific city is a critical function for a vast number of internet services, yet current methods are often unreliable.

- **Inaccurate Commercial Databases:** Most services rely on commercial geolocation databases that often map IPs based on company registration data rather than the physical location of the hardware. This leads to significant errors, such as a server in Nagpur being incorrectly located in Mumbai.

- **Impact on User Experience:** This inaccuracy directly harms user experience. Content Delivery Networks (CDNs) may serve content from a distant, slower server, causing buffering for video streaming. E-commerce sites might show incorrect currency or shipping options.
- **Security Risks:** Inaccurate geolocation can lead to false positives or negatives in fraud detection systems. A legitimate transaction might be flagged as fraudulent, or a real threat could be missed.
- **Lack of Real-Time Data:** Existing databases are static and updated infrequently. They cannot account for dynamic changes in network routing or the deployment of new infrastructure.

2. Solution Proposed

We propose

Geo-Matrix AI, a supervised machine learning system that determines an IP's physical location by fusing data from multiple real-time network and web-based sources.

Our solution moves beyond simple database lookups by learning the complex relationship between an IP's network behaviour and its physical location. The core idea is that while an IP's registration can be misleading, its physical properties—like the time it takes for data to travel to it from known locations (latency)—cannot be faked.

Geo-Matrix AI will collect a rich dataset for any target IP, including:

❖ **Active Network Measurements:** Using a network of distributed probes, we will measure network latency (RTT via ping) and the network path (traceroute). Low latency from a specific probe is a powerful indicator of proximity.

❖ **Passive & Web Data:** We will enrich this data with information from WHOIS records, reverse DNS (rDNS) lookups, and even by scraping location information from websites hosted on the IP address.

This multi-modal data is then fed into a machine learning model trained to predict the most probable city-level coordinates for the IP address.

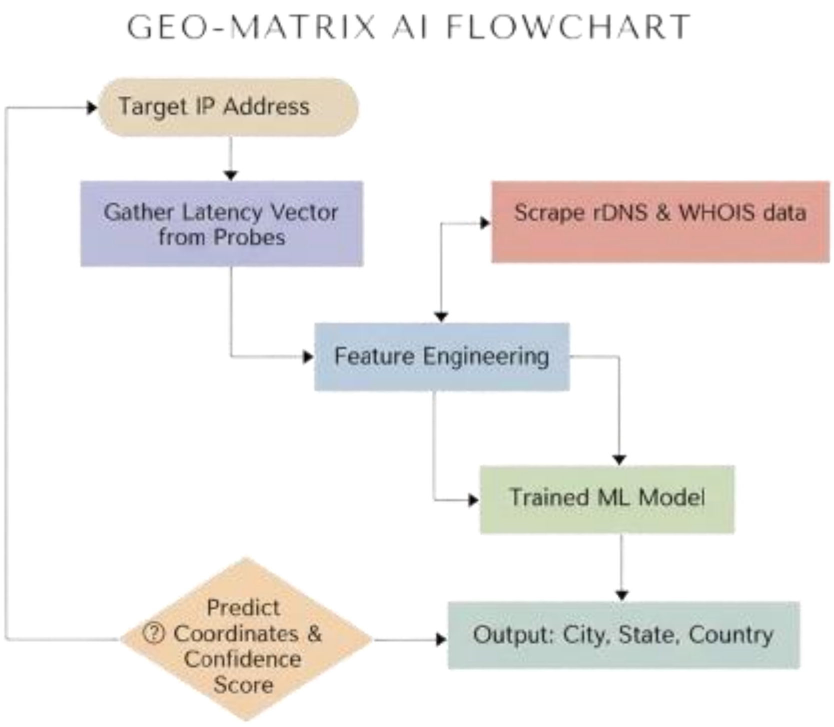
3. Optimization Proposed by the Team :

Our solution is a fundamental optimization over the current industry standard of relying on static, often-inaccurate geolocation databases.

Scenario	Before (Static Database)	After (Geo-Matrix AI)
Data Source	A single, static database file.	Multiple, real-time data sources (Latency, rDNS, WHOIS).
Accuracy	Low to medium. Prone to errors based on ISP registration data.	High. Based on physical network properties and corroborated sources.
Methodology	Simple key-value lookup.	Supervised Machine Learning model that learns complex patterns. [6]
Output	A single, often questionable, city location. [7]	A predicted location with a **confidence score** and evidence.

Scenario	Before (Static Database)	After (Geo-Matrix AI)
Adaptability	Slow to adapt to network changes (weeks/months). [9]	Adapts in near real-time as network routes change. [10]

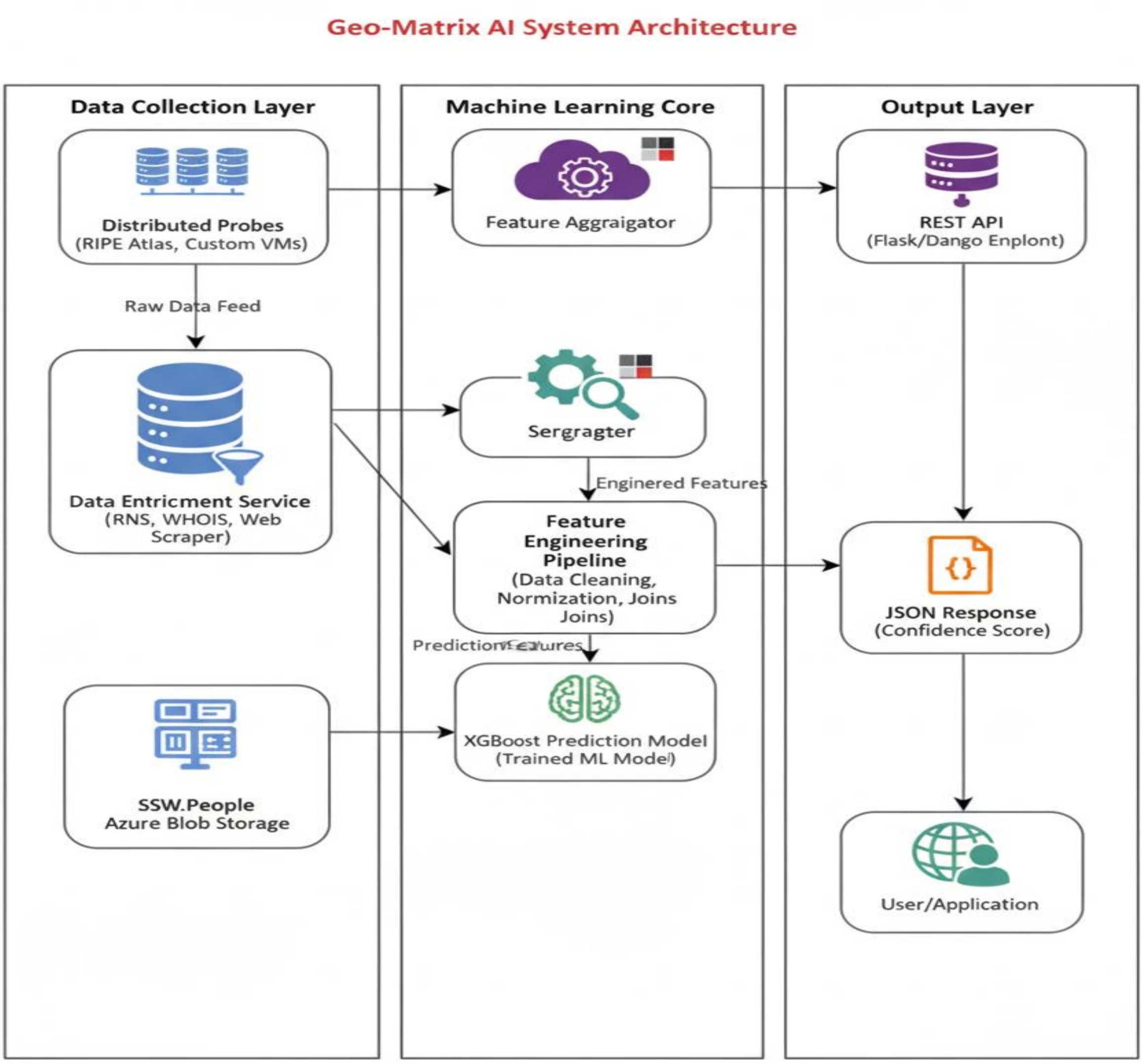
Optimization Process Flowchart:



4. Solution Architecture and Design :

The Geo-Matrix AI architecture is designed for scalability and accuracy, consisting of a Data Collection layer, a Machine Learning Core, and an API/Output layer.

Architecture Visual:



5. Timeline of Delivery :

The project will be delivered in four key milestones over the course of the hackathon.

- **Milestone 1 (Hours 0-6): Data Collection & Probes**
  - Set up at least three virtual probes in different cloud regions.
  - Write scripts to perform automated ping and traceroute measurements.
- **Milestone 2 (Hours 7-15): Feature Engineering & ML Model Training**
  - Gather a training dataset of IPs with known locations.

  - Train the first version of the XGBoost model.
- **Milestone 3 (Hours 16-20): API Development**
  - Build the Flask/Django REST API to serve the model.
  - Create the JSON output format, including location and confidence score.
- **Milestone 4 (Hours 21-24): Integration, Testing & Presentation**
  - Integrate all modules and perform end-to-end testing.
  - Prepare the final presentation and demo.

6. References :

- **Tools:** Python, Scikit-learn, XGBoost, Pandas, Flask, RIPE Atlas.
- **Concepts:** Supervised Learning, Feature Engineering, Network Latency (RTT), Traceroute, WHOIS, DNS.
- **Data Sources:** RIPE Atlas (for probe network), Public WHOIS servers.

7. Conclusion :

Geo-Matrix AI directly addresses a fundamental flaw in how internet services understand their users' locations. By replacing outdated, static databases with a dynamic machine learning model, we can provide a significantly more accurate and reliable IP geolocation service. This has immediate benefits for user experience in media streaming, e-commerce, and for the security of online financial transactions.