Project Title: Problem Statement 15

“Supervised Learning for City-Level IP Geolocation”

Team Name: GEOSTHIRA

Team Members: Kavyashree K

Margaret Sheela C

Sneha Zolgikar (Internal Mentor)

College: Vemana Institute of Technology, Bengaluru

Table of Contents

Problem Description

- Current IP geolocation methods are mostly database-driven and often become outdated quickly, leading to poor city-level accuracy.
- IP addresses frequently shift across regions due to ISP reassignments, mobile networks, or routing changes, which causes incorrect mapping.
- Existing solutions generally do not provide confidence levels or error bounds, making it hard to assess how reliable a prediction is.
- Special cases such as VPNs, Carrier-Grade NAT (CGNAT), and Anycast introduce additional challenges, as the same IP can appear in multiple locations at the same time.
- Rule-based and static approaches fail to handle rare cities, class imbalance, and dynamic network conditions, limiting their generalization.
- There is a need for a supervised machine learning model that not only predicts the city-level location of an IP address but also outputs confidence scores and an estimated error radius (in kilometers).
- The proposed solution aims to combine IP→city datasets with auxiliary features (ASN, prefix, rDNS, RTTs, traceroute hints, time zone patterns, etc.) to improve accuracy and robustness.
- Such a system will be more adaptive, transparent, and reliable compared to traditional geolocation databases, making it suitable for real-world applications.

Key Issues

- Database-driven geolocation is outdated and often inaccurate at the city level.
- IP addresses can shift across regions due to ISP changes and mobility.
- No confidence score or error bound in existing solutions.
- VPNs, Anycast, and Carrier-Grade NAT make location prediction unreliable.
- Rule-based methods fail to generalize across ASNs, prefixes, and time.

Solution Proposed

The following points outline our structured plan for building a machine learning–based IP geolocation system that improves accuracy over traditional methods. Instead of relying on static databases, our approach combines multiple network signals with city-labeled datasets to deliver both predictions and confidence estimates.

- **Collecting Data:** We will gather baseline IP→city labels from public datasets such as MaxMind GeoLite2 and DB-IP Lite. To strengthen this, we will enhance the data with live traceroute and RTT measurements from the AIORI portal, combined with ASN and prefix metadata from RDAP/BGP lookups and reverse DNS hostname extraction. This mix of static and real-time data will give us a richer dataset for training.

- **Feature Engineering:** Each IP sample will be converted into feature vectors using techniques such as RTT summarization (mean, variance, quantiles), graph-based features from traceroute paths and IXP proximity, and ASN/prefix embeddings to represent routing structure. We will also apply NLP methods on reverse DNS hostnames for city/ISP hints and extract temporal activity patterns from usage data. To simplify complex signals, PCA or auto encoder embeddings will be used for dimensionality reduction. Additional diversity will be achieved by augmenting data with Aiori portal's traceroute/DNS outputs and synthetic zones.

- **Model Training:** Use machine learning models such as gradient-boosted trees (XGBoost/LightGBM) or softmax classifiers to predict the most likely city. MATLAB and ML Notebook will also be explored for experimenting with models and feature testing.

- **Confidence & Error Estimation:** Along with city prediction, the system will output a probability score and an estimated error radius (in km), helping users understand the reliability of each prediction.

- **Handling Special Cases:** IPs that belong to VPNs, Anycast, or Carrier-Grade NAT will be detected and flagged as low-confidence or mapped only to a broader region, rather than providing misleading city-level predictions.

- **API Serving:** Provide predictions through a lightweight REST API that outputs city, probability, coordinates, confidence radius, and top-k alternative cities in a structured format.

- **Validation Strategy:** Ensure the system is evaluated carefully by splitting the data by ASN, prefix, and time, to prevent data leakage and confirm the model generalizes well to unseen networks.

- **Tools & Environment:** We will use Python in Jupyter Notebook or Google Colab for training and experimentation, with libraries like scikit-learn, XGBoost/LightGBM, and imbalanced-learn (SMOTE-ENN). MATLAB and ML Notebook will be explored for experimenting with different modeling approaches. For deployment, Flask or FastAPI inside a Docker container will serve predictions efficiently.

Optimization Proposed by the Team

Our team identified the limitations of baseline IP geolocation methods and proposed a set of optimizations aimed at improving accuracy, reliability, and confidence in city-level predictions. These enhancements are designed to make the system more robust and adaptable for real-world deployment.

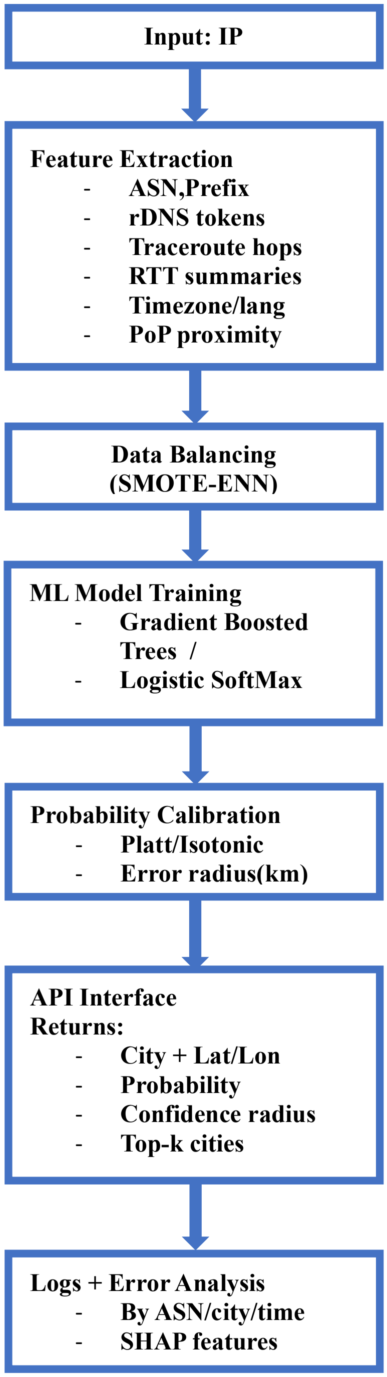
Key Optimizations and Enhancements

- **Enhanced Feature Set:** Use auxiliary features such as ASN, prefix length, reverse DNS tokens, RTT summaries, traceroute hints, and time zone patterns instead of relying only on static IP→city databases.
- **Data Balancing & Probability Calibration:** Apply SMOTE-ENN for class imbalance and Platt/Isotonic scaling to make model probabilities realistic and trustworthy.
- **Error & Special Case Handling:** Introduce a geo-centroid regressor for error radius estimation and detect VPNs, Anycast, or CGNAT IPs, marking them as low-confidence or region-level predictions.
- **Evaluation & API Improvements:** Split data by ASN, prefix, and time for robust validation and provide top-k predictions with confidence radius through the API.

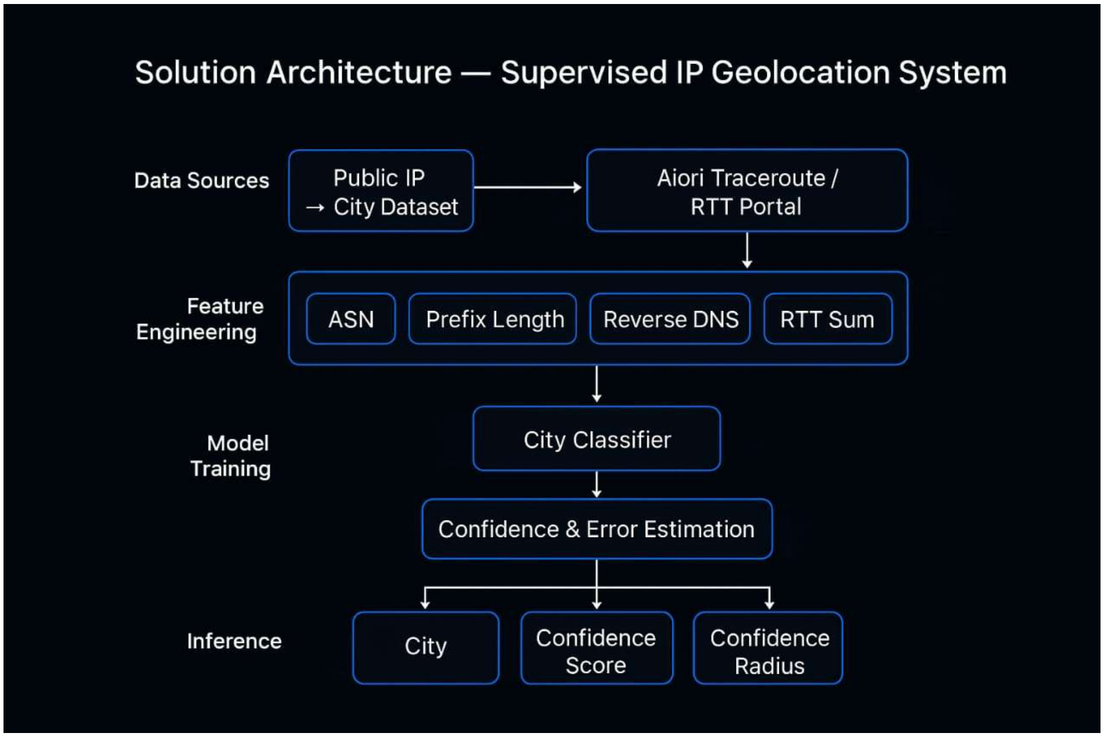
Comparison: Before vs. After Optimizations

Aspect	Before (Baseline Methods)	After(Proposed Optimization)
Accuracy (City-level)	Often outdated, low precision	Improved using supervised ML with richer features
Data Handling	Imbalanced (big cities dominate)	Balanced using SMOTE-ENN
Confidence Output	Not available	Calibrated probabilities + confidence radius
Error Bound	Absent	Predicted in kilometers using geo-regressor
Special Cases	Misleading for VPN/Anycast/CGNAT	Flagged as low-confidence/region-level prediction
Output	Single city guess	API returns {city, prob, lat, lon, radius, top-k}

Flow Chart of Optimization Process:



Solution Architecture and Design



Solution Architecture — Supervised IP Geolocation System

Data Sources

Public IP → City Dataset

Aiori Traceroute / RTT Portal

Feature Engineering

ASN

Prefix Length

Reverse DNS

RTT Sum

Model Training

City Classifier

Confidence & Error Estimation

Inference

City

Confidence Score

Confidence Radius

1.Instead of relying only on static IP databases, our system actively enriches IP addresses with live network signals such as traceroutes and latency data from the AIORI portal. This makes the model more aware of how the internet behaves in real time, not just how it is documented.

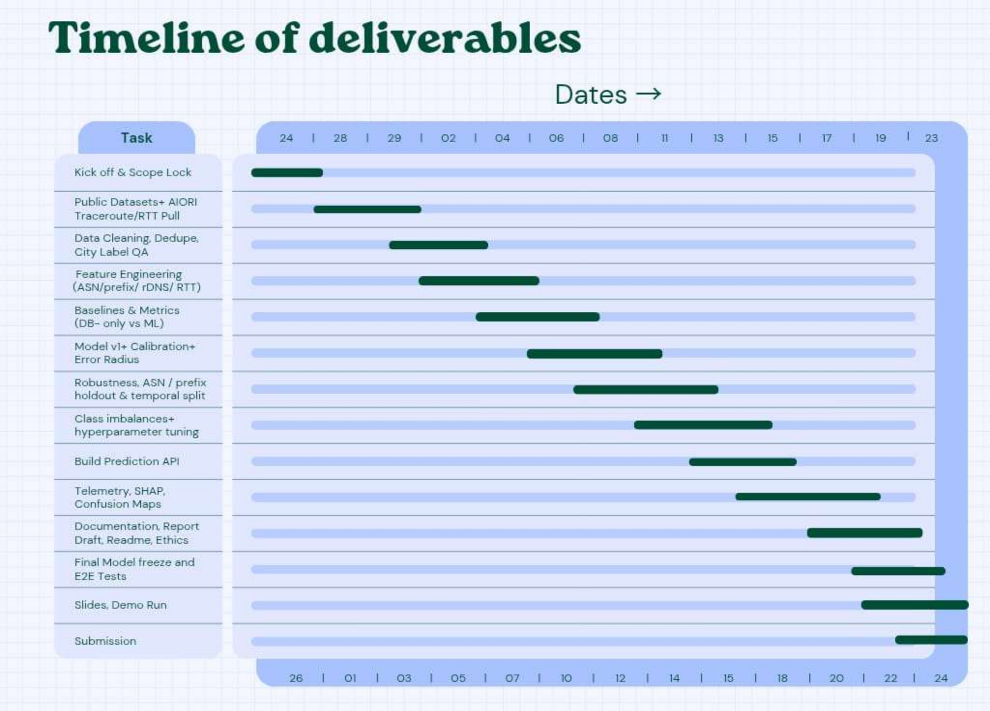
2. Each IP address is converted into a rich feature profile — including ASN, prefix structure, reverse DNS patterns, and RTT trends — so that the model understands how the IP is used, not just where it is registered.

3. The model is trained to predict both the most likely city and how confident it is in that prediction, which helps differentiate between stable residential IPs and ambiguous cases like VPNs or CGNAT.

4. Instead of guessing blindly, the system returns a calibrated confidence radius (for example, “within 25 km”) — making the output usable in practical applications like fraud detection or content delivery.

5. The architecture is modular — meaning even if better datasets or features become available later, they can be plugged into the same pipeline without rebuilding everything from scratch.

Timeline of Delivery:



References

1. IEEE Journals and Papers on IP Geolocation Techniques

   - Provides foundational knowledge and methods for accurate IP geolocation.

2. ACM SIGCOMM Publications on Internet Mapping, DNS, and Traceroute

3. AIORI Portal – Network and IP Data Access

   - Primary data source for IP addresses, zones, and network measurements in our project.

4. Google – IP Geolocation, DNS, and Networking Tools

   - Useful for verifying geolocation results and learning practical implementation methods.

5. Python Documentation – Data Analysis, Machine Learning, and Feature Engineering

6. MATLAB Documentation – Signal Processing, Machine Learning, and AI Toolboxes

   - Used for experimenting with algorithms, model training, and visualization of results.

7. Jupyter Notebook – Interactive Computing and Experimentation Environment

   - Provides an environment to organize code, run experiments, and document findings.

Conclusion

The “Supervised Learning for City-Level IP Geolocation” project demonstrates a robust and adaptive approach to improving IP geolocation accuracy at the city level. By combining publicly available IP→city datasets with auxiliary features such as ASN, prefix, RTT measurements, traceroute outputs, and time zone patterns, the system not only predicts city locations but also provides confidence scores and estimated error radii.

Optimizations such as feature enhancement, data balancing, probability calibration, and special-case handling (VPNs, Anycast, CGNAT) ensure that the predictions are reliable and generalizable to real-world networks. The proposed API allows structured, actionable output, making the system practical for deployment and integration.

This project highlights the potential of machine learning in enhancing network intelligence and provides a foundation for future improvements. Next steps could include integrating additional live data sources, refining the models for rare or underrepresented cities, and extending the platform for broader geospatial analytics.

Impact: By increasing the accuracy and reliability of city-level IP geolocation, this system can support better location-aware services, network diagnostics, cybersecurity applications, and data-driven decision-making.

Call to Action

We encourage further exploration of supervised geolocation models, integration with live network data sources, and development of user-friendly APIs to enable real-time, accurate, and reliable city-level IP location services for research, industry, and smart city applications.