PURBANCHAL UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING
KHWOPA ENGINEERING COLLEGE
LIBALI-08, BHAKTAPUR

A PROJECT PROPOSAL ON

SpoTex

(Predictive Site Selection System)

Project work submitted in partial fulfillment of the requirements for the degree of Bachelor of Engineering in Computer Engineering (Eight Semester)

Submitted by

Anish Maka	(780304)
Roj Gosai	(780334)
Subekshya Kadel	(780341)
Sujal Koju	(780342)

Khwopa Engineering College
Libali-08, Bhaktapur
February 24, 2006

Acknowledgement

We take this opportunity to express our deepest and sincere gratitude for the chance to work on our project, **'Predictive Site Selection System**. This work is submitted as a partial fulfillment of the requirements for the seventh semester of our Bachelor of Computer Engineering degree under Purbanchal University.

We are sincerely thankful to our institution, **Khwopa Engineering College**, for providing us with the platform and resources necessary to undertake this project. We would like to extend our special and heartfelt gratitude to our Head of the Department of Computer Engineering, **Er. Bikash Chawal**, for his invaluable guidance and for providing us with this golden opportunity to apply our skills to a real-world challenge.

This project is not just an academic exercise; it is our endeavor to bridge the gap between complex data science and practical, everyday business decisions. We would also like to express our sincere gratitude to all those whose ideas, insights, and innovations have inspired this project.

Anish Maka	(780304)
Roj Gosai	(780334)
Subekshya Kadel	(780341)
Sujal Koju	(780342)

Abstract

Selecting the optimal location for a new retail business is a critical factor that significantly influences its success. Traditional methods for site selection often rely on manual surveys, intuition, and static demographic data, which fail to account for dynamic urban patterns and competitor influence. This project proposes an AI-powered tool that leverages geospatial data, competitor analysis, and crowd origin sources—such as schools, parks, and public places—to predict the success rate of a shop at a specific location. By integrating machine learning models with interactive mapping systems, the tool analyzes factors like monthly customer footfall, competitor popularity, and proximity to high-traffic areas to provide accurate predictions and actionable recommendations. The system also features a success heatmap visualization, enabling users to explore and compare potential locations effectively. Designed to support small and medium businesses, this smart site selection tool democratizes access to advanced location intelligence for informed retail planning. a reliable and effective solution.

Keywords:*Retail Site Selection, Geospatial Analysis, Crowd Origin Mapping, Interactive Map Visualization*

Contents

List of Figures

Chapter 1

Introduction

1.1 Background Introduction

Anyone who walks down a busy street in Kathmandu, Patan, or Bhaktapur can see the cycle of hope and hardship firsthand. A new, brightly lit cafe opens, full of promise and excitement. A few months later, that same spot might have a ”To-Let” sign hanging in its window. For every successful business that becomes a local landmark, many others quietly disappear, often taking an entrepreneur’s life savings with them. This isn’t just a business problem; it’s a deeply human one that affects families and communities across Nepal.

The age-old wisdom has always been that success in retail or hospitality comes down to three things: ”location, location, location.” But for most small business owners in our country, choosing a location has always been a matter of intuition. Decisions are often based on gut feeling, family advice, or simply what feels like a ”good area.” While this experience is valuable, it’s also a huge gamble. It’s a process filled with uncertainty, where a single wrong choice can be the difference between a thriving business and a failed dream.

As computer science students, we are taught about the power of data. We learn how global giants like Amazon or Starbucks use sophisticated data analysis, Geographic Information Systems (GIS), and Artificial Intelligence (AI) to make these exact decisions with incredible accuracy. They don’t guess; they forecast. They analyze dozens of factors—from foot traffic and competitor density to local income levels—to predict a location’s potential before investing a single rupee. This creates a huge analytical gap between them and the local entrepreneurs we see every day.

We saw this gap as the perfect challenge for our final year project. We asked ourselves: can we use the skills we’ve learned in our engineering course to build a tool that levels the playing field? Can we create an intelligent system that brings the power of predictive analytics to a local business owner in Bhaktapur, helping them make a smarter, safer investment? This project is our attempt to answer that question.

1.2 Motivation

Our motivation for this project is both personal and academic. Personally, as students living in Nepal, we see this problem all around us. We are driven by the desire to build something that has a real, tangible impact on our local community. The idea that our technical skills could help a fellow Nepali make a more confident business decision is a powerful motivator. We want to build a tool that we would be proud to show a friend or family member who is thinking of opening their own shop.

Academically, this project presents the perfect challenge. It forces us to go beyond textbook theory and tackle a complete, end-to-end data science problem. We are motivated by the opportunity to apply what we've learned about GIS, data engineering, and machine learning to a messy, real-world scenario. The process of building our own unique dataset from scratch, training a predictive model like XGBoost, and deploying it as a live web application is an incredible learning experience that encapsulates everything we've studied.

1.3 Statement of Problem

The core problem is this: opening a new business in Nepal is a huge gamble, primarily because there is no accessible, data-driven way for a local entrepreneur to assess the potential of a location. This reliance on intuition over analysis leads to a high risk of choosing the wrong spot, resulting in financial loss and business failure. Our project directly addresses this by tackling the lack of a quantitative decision-making tool. We aim to solve the problem of uncertainty in site selection by providing a system that delivers a clear, evidence-based forecast of a location's viability.

1.4 Main Objectives

The principal objective of this project is:

- To design, develop, and deploy an intelligent, data-driven system that accurately predicts the potential success of a new branch/store in a site.

1.5 Scopes and Application

1.5.1 Scopes

To make sure we can successfully complete this project within our academic timeline, we have defined a clear and focused scope:

- **Where?** Our project will focus exclusively on the core commercial area of **Bhaktapur**.
- **What kind of shop?** Our first model will be an expert on one thing: **cafes and coffee shops**.
- **What data?** We will only use data that is publicly available (like from OpenStreetMap) or that we can collect ourselves manually.
- **What it isn't?** This tool is for predicting a location's potential. It is not a real estate listing site or a business management software.

1.5.2 Application

The finished application is designed to be a practical tool for real people. Here's how it could be used:

- **Help an Entrepreneur Before They Invest:** A user can check a location's potential score *before* they sign an expensive rental agreement.
- **Compare Two or More Locations:** If someone is trying to decide between a spot in Durbar Square and another in Taumadhi, our tool can provide an objective, data-driven comparison.
- **Understand the Market:** Users can click around different neighborhoods to understand why certain areas are hotspots and others are not, helping them with their market research.
- **A Tool for Others:** Beyond entrepreneurs, this system could also be useful for urban planners or real estate agents trying to understand the commercial landscape of a city.

Chapter 2

Literature Review

2.1 The Huff Gravity Model of Retail Location

In his seminal paper, David Huff presented a probabilistic model for retail site selection that became a cornerstone of location analysis [3]. The Huff Gravity Model posits that the probability of a consumer patronizing a specific retail location is a function of two main factors: the ”attractiveness” of the store (e.g., its size, variety of goods) and the ”friction” of distance (the time and cost to travel to it). The model provides a mathematical formula to estimate the trade area of a potential store by calculating these probabilities for surrounding population centers. While foundational and still influential, the model is inherently limited. It relies on a relatively simple formula and struggles to incorporate the vast number of complex, non-linear variables that influence modern consumer behavior, such as competitor density, visibility, and local demographic nuances. Our project aims to overcome these limitations by using a data-driven approach rather than a predefined formula.

2.2 Applying Machine Learning to Retail Site Selection

Moving beyond static formulas, researchers began applying machine learning to the site selection problem to better capture its complexity. In a notable study, Kuo et. al. demonstrated the superiority of AI-based approaches over traditional methods [4]. The authors used a decision tree model to predict the sales volume of new convenience stores. They integrated a wide range of parameters, including demographic data (population, income) and location-specific variables (number of nearby competitors, proximity to schools and offices). Their findings showed that the machine learning model could uncover complex, interactive relationships between these variables that were invisible to traditional statistical models. The model achieved significantly higher predictive accuracy, proving that AI could provide a more robust and reliable framework for site selection. This paper validates our core methodology of using an ensemble of data points to train a predictive model.

2.3 XGBoost: A Gradient Boosting Framework

For projects involving predictive modeling on structured, tabular data—such as ours—the choice of algorithm is critical. Chen & Guestrin presented XGBoost, an optimized and scalable implementation of gradient boosted decision trees that has become a dominant force in applied machine learning [2]. Gradient boosting is an ensemble technique that builds a strong predictive model by sequentially adding weaker models, where each new model corrects the errors of its predecessor. The power of XGBoost lies in its significant optimizations. It employs advanced regularization techniques (L1 and L2) to prevent overfitting, is designed for parallel processing to drastically reduce training time, and can gracefully handle missing data. For our project, which will involve a dataset with many diverse features (demographic, geographic, competitive), XGBoost offers the high performance, accuracy, and robustness required to build a reliable predictive engine.

2.4 Industry Standard: Location Intelligence Platforms

The academic pursuit of site selection has materialized into a significant industry known as Location Intelligence. Commercial platforms like Placer.ai and CARTO represent the current state-of-the-art in this field [5]. These platforms have moved beyond census data and leverage large-scale, anonymized mobile location data to derive highly accurate metrics. They provide dashboards that visualize real-time foot traffic patterns, customer demographics (psychographics), trade area analysis, and competitor performance. For instance, Placer.ai can show how many people visit a competitor's cafe, where they came from, what time of day is busiest, and what other stores they frequent. While building a system with access to such proprietary mobile data is beyond our project's scope, studying these platforms provides a crucial benchmark. Our project aims to emulate the analytical power of these systems by creating our own unique dataset and using AI to derive similar predictive insights.

2.5 GIS-based Feature Engineering for Machine Learning

A critical challenge in any geospatial prediction task is converting raw map data into meaningful features that a machine learning model can understand. A study by Al-Ruzouq et. al. provides a clear framework for this process, which is often termed "geospatial feature engineering" [1]. The authors demonstrate the use of Geographic Information Systems (GIS) tools to systematically generate predictive variables. Their methodology involved using GIS to calculate buffer zones around potential sites, count

the number of competitors within those zones, measure the distance to the nearest major road or point of interest (POI), and determine the density of the surrounding road network. These engineered features were then used to train a machine learning model for predicting land suitability. This paper is directly relevant to our project, as it outlines the exact process we will follow: using a GIS tool (QGIS) to transform our raw geographic data into a structured set of features (competitor density, proximity to 'people magnets', etc.) that will form the input for our XGBoost model.

Chapter 3

PROJECT MANAGEMENT

3.1 Team Members

This project is the joint effort of:

1. Anish Maka - 780304
2. Roj Gosai - 780334
3. Subekshya Kadel - 780341
4. Sujal Koju - 780342

3.2 Software Requirements

The software requirements for the development and deployment of the predictive site selection system include:

1. **Collaboration & Planning:** Google Drive, Trello, Discord
2. **Backend Development:** Python 3.9+
3. **GIS & Data Analysis:** QGIS Desktop
4. **Database:** PostgreSQL with the PostGIS extension
5. **Frontend Development:** Node.js, React.js
6. **Version Control:** Git, GitHub
7. **Code Editor:** Visual Studio Code
8. **Documentation:** Overleaf

3.3 Hardware Requirements

The hardware requirements for the development machine and end-user system are as follows:

a. **Development Machine:**

- **RAM:** Minimum 8GB, recommended 16GB (for handling datasets and running the database server).
- **Processor:** Modern multi-core CPU (e.g., Intel Core i5 / AMD Ryzen 5 or better).
- **Operating System:** Windows, macOS, or Linux.

b. **End-User System:**

- A modern web browser (e.g., Chrome, Firefox, Safari) on any desktop or mobile device.

3.4 Functional Requirements

The functional requirements define the specific behaviors and capabilities of the proposed system:

a. **Geospatial Data Visualization:** The system must display an interactive map of the defined geographic area (e.g., Bhaktapur).

b. **Interactive Location Selection:** Users must be able to select any point on the map to analyze its suitability.

c. **Predictive Score Generation:** Upon selection, the system must process the location's features and return a quantitative "Success Score".

d. **Feature-based Insights:** The system must display a breakdown of the key factors (e.g., competitor density, proximity to POIs) that influenced the predicted score.

e. **Data Management Interface (Admin):** A secure interface must exist for the project team to update the underlying geographic data (e.g., add a new competitor).

3.5 Non-Functional Requirements

The non-functional requirements define the quality attributes and operational standards of the system:

a. **Accuracy:** The predictive model must achieve a satisfactory level of accuracy (measured by metrics like MAE and R^2) when evaluated against the test dataset.

b. **Performance / Latency:** The system must return a prediction within a few seconds (e.g., < 3 seconds) of a user selecting a location to ensure a responsive user experience.

c. **Reliability:** The web application should have high availability and consistently provide predictions without crashing or producing errors.

d. **Scalability:** The backend architecture must be able to handle multiple concurrent user requests without significant degradation in performance.

e. **Data Integrity:** The system must ensure that the geographic database is protected from corruption and unauthorized modifications.

Chapter 4

Methodology

4.1 Block Diagram

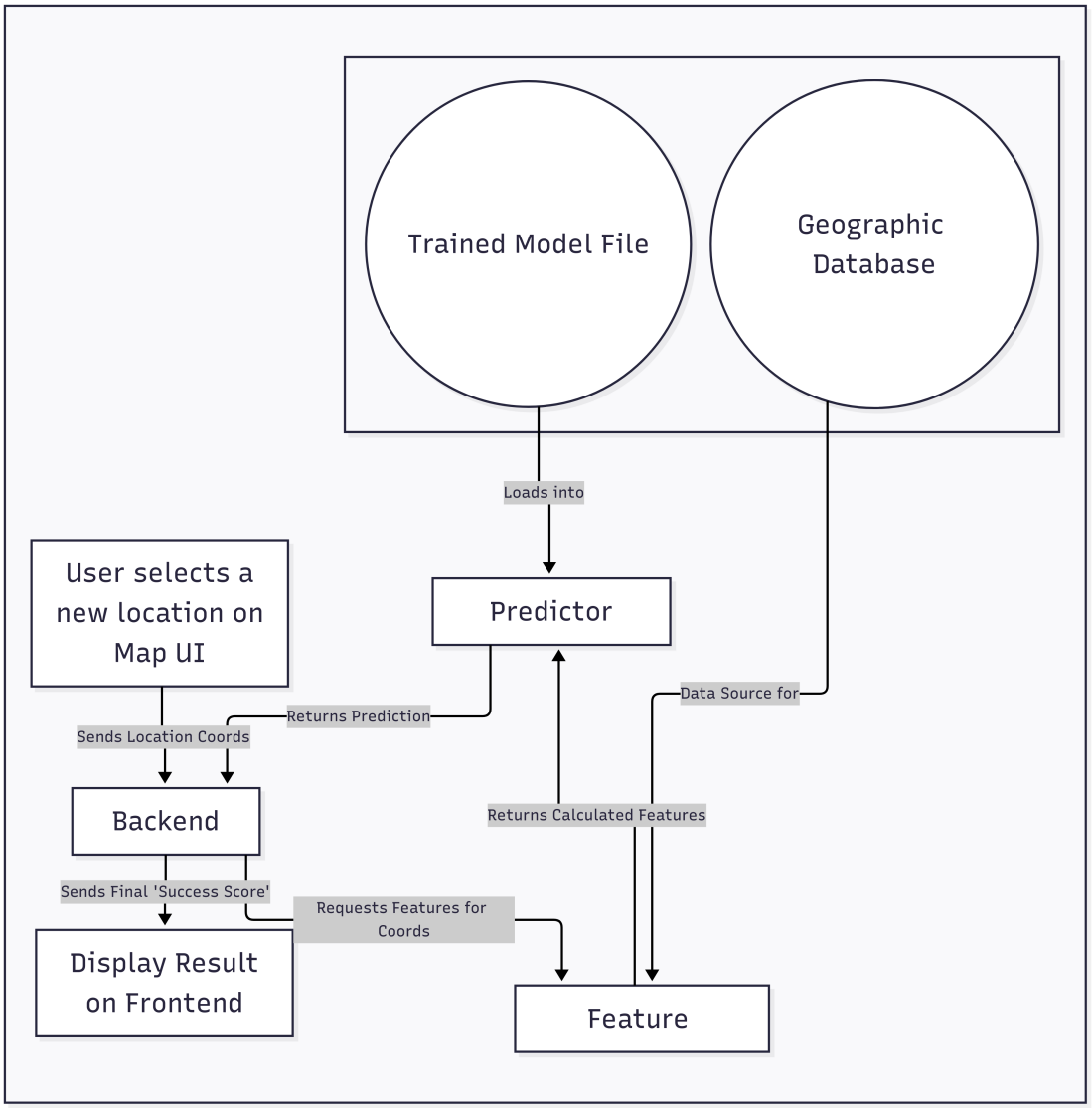

Figure 4.1: Block Diagram

The diagram illustrates the block diagram of the Predictive Site Selection System

- **User Interaction Components**: The process begins when the **User** selects a location on the **Map UI**. The **Backend** receives the coordinates and, after processing,

sends the final ”Success Score” to the **Frontend** for display.

- **Data Processing Engine**: The core logic is handled by two distinct services. The **Feature** calculator receives the coordinates from the backend and uses the **Geographic Database** to calculate a set of numerical features. These features are then sent to the **Predictor**, which uses the **Trained Model File** to generate a prediction score.
- **External Assets**: The system relies on two critical, pre-existing assets: the **Geographic Database**, which stores all location data, and the **Trained Model File** (`.pkl`), which contains the system's predictive intelligence. These are consumed by the live system but created offline.

4.2 Usecase Diagram

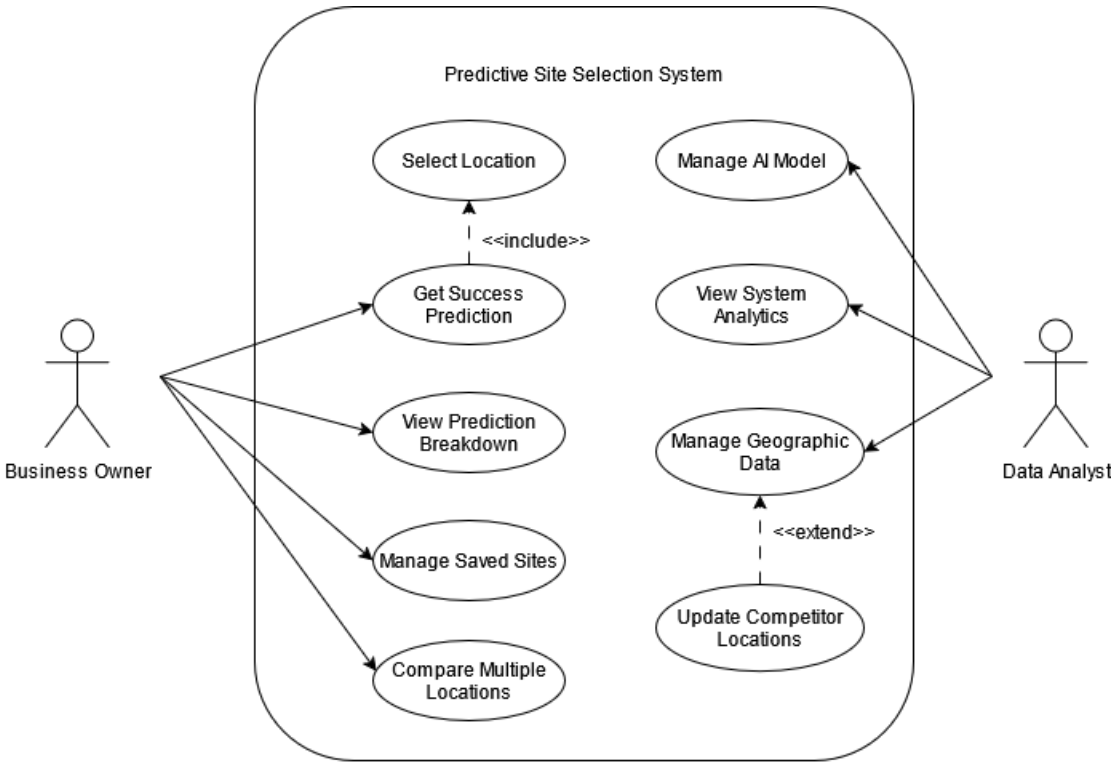

Figure 4.2: Usecase Diagram

The use case diagram for the Predictive Site Selection System illustrates interactions between the system and its users: Business Owner and Data Analyst. Key components include:

- **Business Owner**: Performs *Select Location*, *Get Success Prediction*, *View Prediction Breakdown*, *Manage Saved Sites*, and *Compare Multiple Locations*.

- **Data Analyst**: Manages *AI Model*, *View System Analytics*, *Manage Geographic Data*, and *Update Competitor Locations*.
- **System**: Central entity with use cases extending and including various functionalities.

4.3 Zero Context level Diagram

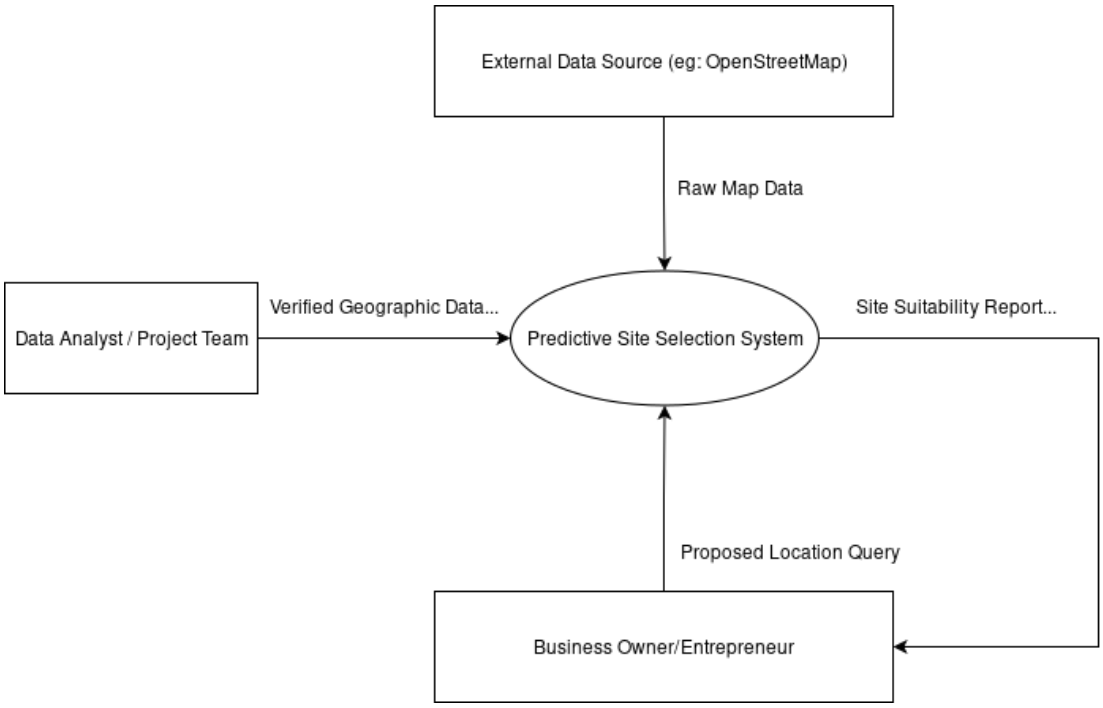

Figure 4.3: 0-level Context Diagram

This is a high-level view of the entire system.

External Entities:

1. **Data Analyst / Project Team**: verifies and inputs Geographic data of Relevant Competitors.
2. **Business Owner / Entrepreneur**: requests analysis of proposed location.

Process:

1. **Predictive Site Selection System**: runs prediction models to obtain the success rate of the proposed site location and provides the resulting score report and insights to the Business Owner / Entrepreneur.

Data Store

1. **External Data Storage**: stores raw map data obtained from various sources such as OpenStreetMap.

4.4 Workflow Diagram

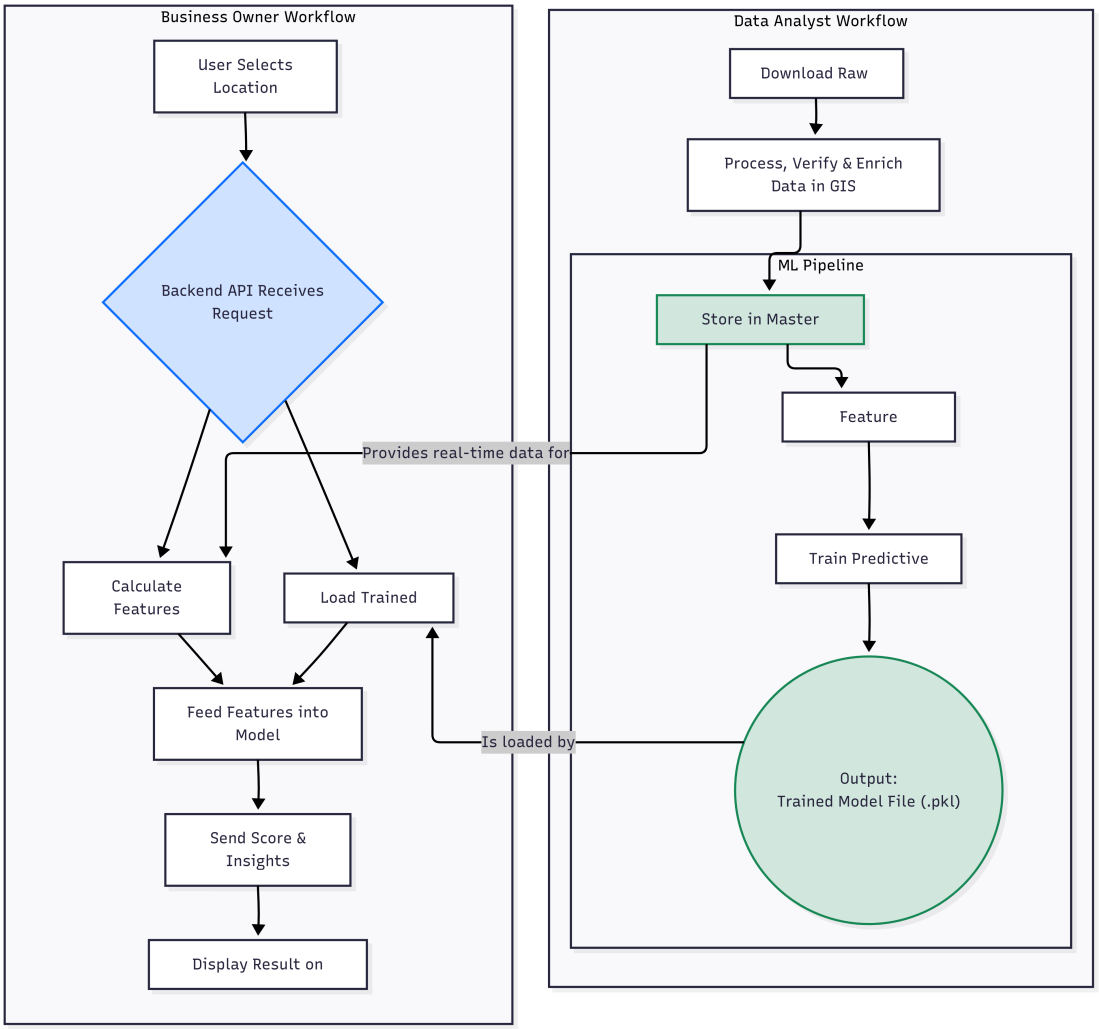

Figure 4.4: Workflow Diagram

This diagram illustrates the two primary and distinct workflows that define the entire project.

- **Data Analyst Workflow (Right Side)** This represents the offline, foundational process required to build the system's intelligence. It begins with downloading raw data, processing it in GIS, and storing it in a **Master Database**. This data is then used in an **ML Pipeline** to engineer features and train a model, resulting in the final **Trained Model File**.

- **Business Owner Workflow (Left Side)** This represents the real-time, online process. A **User** selects a location, which triggers the **Backend API**. The API orchestrates the calculation of features, loads the trained model, feeds the features into it, and sends the final score and insights to the user for display.
- **The Bridge** The diagram critically shows the connection between the two workflows. The outputs of the Analyst's work—the **Master Database** (used by the "Calculate Features" step) and the **Trained Model File** (used by the "Load Trained" step)—are the essential inputs that power the live Business Owner workflow.

4.4.1 Geospatial Data Collection and Aggregation

The project's foundation is a high-quality, multi-layered geographic dataset. This "Master Dataset" will be built by aggregating data from both public online sources and manual field verification within a defined geographic scope (e.g., Bhaktapur).

The primary tool for this phase will be the open-source Geographic Information System (GIS) software, `QGIS`. Data will be collected and organized into distinct layers:

- **Competitor Layer:** Locations of all existing businesses of the same type (e.g., cafes). This will be sourced initially from OpenStreetMap (OSM) and then verified and augmented through manual "ground-truthing" to ensure accuracy and completeness.
- **People Magnet Layer (Points of Interest - POIs):** Locations that attract significant foot traffic, such as schools, temples, major offices, and bus stops. This data will also be sourced from OSM and manually verified.
- **Road Network Layer:** The complete network of roads, paths, and alleys within the study area, sourced from OSM.

Each data point will be stored with its precise geographic coordinates (latitude and longitude) in a central database.

4.4.2 Feature Engineering and Data Structuring

This critical phase involves converting the raw geospatial data into a structured, numerical format suitable for machine learning. This process, known as feature engineering, will be performed using GIS analysis tools within QGIS and Python libraries like `GeoPandas`.

For each existing cafe in our dataset, we will calculate a set of predictive features:

- **Competitor Density:** The number of other cafes within a 100-meter and 300-meter radius, calculated using a buffer analysis.
- **Proximity to POIs:** The straight-line distance to the nearest "People Magnet" in each category (e.g., nearest school, nearest bus stop), calculated using a distance matrix analysis.
- **Road Type Classification:** A manually assigned score based on the road's classification (e.g., Main Commercial Route, Feeder Street, Local Alley).
- **Visibility Score:** A manually assigned score (1-5) based on the storefront's visibility from main pathways.

Alongside these features, we will define our target variable:

- **Popularity Score (Target Variable):** A composite score (1-10) calculated from publicly available data like Google Maps ratings, the number of reviews and Social Media Presence. This score acts as a proxy for business success.

Each data point will be structured in a JSON-like format for training:

{
"id": 25,
"cafe_name": "Himalayan Java",
"popularity_score": 8.7,
"competitor_density_100m": 3,
"dist_to_school_m": 152,
"road_type": "Main Commercial Route",
"visibility_score": 5
}

To prepare for model training and evaluation, the dataset will be partitioned using the standard train-test-validation split:

- 70% for training the model.
- 15% for validation during hyperparameter tuning.
- 15% for final testing on unseen data.

4.4.3 Predictive Modeling System

The core of the application is the system that provides real-time predictions for new, hypothetical locations. The user interaction is designed to be simple and intuitive.

- A user selects any point on the interactive map interface.
- The frontend application sends the selected coordinates to the backend API.
- The backend, in real-time, calculates all the engineered features for that specific point by querying the geographic database.
- These features are then fed into the pre-trained machine learning model.
- The model returns a single output: the predicted ”Success Score”.
- The system will also provide a ”Prediction Breakdown,” showing which features contributed most positively or negatively to the score, offering actionable insights.

4.4.4 Predictive Engine via Supervised Learning

Unlike adaptive systems that use Reinforcement Learning, our project will utilize a **Supervised Learning** model. This approach is ideal for our problem, as we are training a model to predict a specific, known target value (Popularity Score) based on a labeled dataset of existing examples.

The system will be modeled using the standard supervised regression framework:

ML Component	Description
Input Features (X):	A vector of numerical values representing the engineered features (competitor density, proximity, visibility, etc.).
Target Variable (Y):	The continuous ”Popularity Score” that the model learns to predict.
Model Architecture:	A gradient boosted decision tree model, specifically **XGBoost**, chosen for its high accuracy and robustness on tabular data.
Prediction Output:	A single, continuous value (e.g., 7.8) representing the predicted success score for a new location.

4.4.4.1 Example Input Feature Vector

An example input vector fed to the model for a new location might look like:

```
[ 2, 4, 250, 80, 3, 4 ]
// Represents [competitors_100m, competitors_300m, dist_to_school,
// dist_to_bus_stop, road_type_score, visibility_score]
```

4.4.5 Model Training and Evaluation Strategy

The training process is designed to ensure the model is both accurate and reliable.

- A baseline model (e.g., Linear Regression) will be established to measure the performance uplift provided by our chosen model.
- The primary model, **XGBoost**, will be trained on the 70% training dataset.
- Model performance will be assessed on the test set using standard regression metrics:
  - **Mean Absolute Error (MAE):** Measures the average absolute difference between predicted and actual scores.
  - **Root Mean Squared Error (RMSE):** Similar to MAE but penalizes larger errors more heavily.
  - **R-squared (R^2):** Indicates the proportion of the variance in the success score that is predictable from the input features.

4.4.6 Data Schema and Storage

To support the real-time feature calculation, all aggregated geographic data will be stored in a professional-grade database system.

- **Database Technology: PostgreSQL** with the **PostGIS** extension.
- **Why PostGIS?** This extension provides powerful functions to perform geographic queries directly in the database (e.g., `ST_DWithin`, `ST_Distance`), which is extremely efficient.

Example Schema for ‘locations‘ Table

{
”location_id”: ”pk_101”,
”name”: ”Himalayan Java”,
”type”: ”cafe”,
”geom”: ”POINT(85.4290 27.6715)”, – *PostGIS geometry type*
”google_rating”: 4.5,
”google_reviews_count”: 1800
}

Chapter 5

EXPECTED RESULT

5.1 Expected Result

When we finish our final year project, we don't just want a grade. We want to have built something that actually works and means something. Our main goal is to create a real, live web application that anyone can use. Think of a friend, a dai or didi, wanting to open a small cafe in Bhaktapur. We want them to be able to use our tool to move past just a 'gut feeling' and make a decision based on real data. It's about giving them a better, safer chance to succeed.

This tool won't just spit out a random 'success score.' We expect it to give a clear breakdown, explaining 'why' a spot is good. Maybe it's close to a college, or maybe there aren't too many other cafes nearby. The idea is to give a small business owner the kind of smart insights that big companies have, but for our own local streets. We want to level the playing field, so that starting a business feels less like a gamble.

On a personal level, this project is everything for us. It's our chance to take all the theory we've learned over seven semesters—all the GIS, the data engineering, the machine learning with XGBoost—and build something from scratch. We're creating our own dataset, training our own model, and deploying a full application. This is the real test. We expect to come out of this not just as students, but as engineers who have actually built a complete, end-to-end data science solution for a problem we genuinely care about, right here in Nepal.

REFERENCES

[1] Rami Alruzouq, Azzam Shanableh, and Basma Yaghi. Gis-based multicriteria evaluation for gas station site selection in dubai. *International Journal of Sustainable Transportation*, 13(3):193–205, 2019.

[2] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[3] David L. Huff. A probabilistic analysis of shopping center trade areas. *Land Economics*, 39(1):81–90, 1963.

[4] R. J. Kuo, S. C. Chi, and S. S. Kao. Application of grey relational analysis and grey decision-making to the location choice of convenience store. *International Journal of Management Science*, 30(5):447–455, 2002.

[5] Placer.ai. Placer.ai dataset, 2022. `https://www.placer.ai/data/`.