**LOCOFIND: A GPS-ENABLED PLATFORM FOR LOCAL FOOD, CULTURE, SAFETY, AND HERITAGE EXPLORATION**

# A SOCIAL RELEVANT MINI PROJECT REPORT

***Submitted by***

**R. CHANDRIA [REGISTER NO:211422104099] ARUNA G [REGISTER NO:211422104056]**

***in partial fulfillment for the award of the degree of***

# BACHELOR OF ENGINEERING IN

**COMPUTER SCIENCE AND ENGINEERING**

****

**PANIMALAR ENGINEERING COLLEGE CHENNAI – 600123**

## (An Autonomous Institution Affiliated to Anna University, Chennai) OCTOBER 2025

**PANIMALAR ENGINEERING COLLEGE**

**CHENNAI – 600123**

## (An Autonomous Institution Affiliated to Anna University, Chennai)

**BONAFIDE CERTIFICATE**

Certified that this socially relevant mini project report **“LOCOFIND: A GPS ENABLED PLATFORM FOR LOCAL FOOD, CULTURE, SAFETY, AND HERITAGE EXPLORATION”** is the bonafide work of R.CHANDRIA(211422104099), ARUNA G (211422104056), who carried out the mini project work under my supervision.

**SIGNATURE SIGNATURE**

## Dr.L.JABASHEELA, M.E., Ph.D., ALIMA BEEVI A M.E.,

## HEAD OF THE DEPARTMENT ASSISTANT PROFESSOR

DEPARTMENT OF CSE DEPARTMENT OF CSE

PANIMALAR ENGINEERING COLLEGE PANIMALAR ENGINEERING COLLEGE

NASARATHPETTAI, NASARATHPETTAI,

POONAMALLEE, POONAMALLEE,

CHENNAI-600 123 CHENNAI-600 123

Submitted for the 23CS1512- Socially Relevant Mini Project Viva-Voce Examination

held on...........................

**INTERNAL EXAMINER EXTERNAL EXAMINER**

# DECLARATION BY THE STUDENT

We, R.CHANDRIA(211422104099), ARUNA G(211422104056), hereby declare that this project report is titled **LOCOFIND: A GPS-ENABLED PLATFORM FOR LOCAL FOOD, CULTURE, SAFETY, AND HERITAGE EXPLORATION**, under the guidance

of A L I M A B E E V I A M . E . , A S S I S T A N T P R O F E S S O R . T he original work was done by us, and we have not plagiarized or submitted to any other degree in any university by us.

1. **R. CHANDRIA**
2. **ARUNA G**

**ACKNOWLEDGEMENT**

We would like to express our deep gratitude to our respected Secretary and Correspondent, **Dr.P.CHINNADURAI, M.A., Ph.D.,** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We express our sincere and hearty thanks to our Directors, **Tmt.C.VIJAYARAJESWARI, Dr.C.SAKTHIKUMAR, M.E., Ph.D,** and  **Dr.SARANYASREE SAKTHI KUMAR B.E., M.B.A., Ph.D.,** for providing us with the necessary facilities to undertake this project. We also express our gratitude to our Principal, **Dr. K. MANI, M.E., Ph.D.,** who facilitated us in completion of the project.

We thank the Head of the CSE Department, **Dr.L.JABASHEELA, M.E.,Ph.D.,** for the support extended throughout the project.We would like to thank my Project Guide, **ALIMA BEEVI A M.E.,ASSISTANT** **PROFESSOR,** and all the faculty members of the Department of CSE for their advice and encouragement for the successful completion of the project.

**R. CHANDRIA ARUNA G**

**ABSTRACT**

Tourism in India is a growing sector, but travelers often face challenges such as identifying the best travel season, exploring authentic cuisines, handling transport and weather uncertainties, ensuring safety, and overcoming language barriers. Current applications address these issues individually but lack an integrated solution. To bridge this gap, we present LocoFind, a GPS-enabled smart tourism platform powered by Machine Learning (ML). The system integrates five modules: Travel Season Predictor (Gradient Boosting), Cuisine Recommender (Naïve Bayes & KNN), Transport & Weather Analyzer (Random Forest), Safety & Feedback Classifier (Logistic Regression), and Regional Language Module (Decision Trees). Each module was trained using official datasets from government portals, FSSAI, railway and accident records, and census data.

The models achieved strong predictive performance with accuracies of 98.2% (Season Predictor), 92.5% (Cuisine Recommender), 96.9% (Transport Analyzer), 99.8% (Safety Classifier), and 100% (Language Module), resulting in an overall accuracy of 97.5%**.** By integrating predictive analytics, classification, and recommendation techniques within a Flutter-based mobile app, LocoFind demonstrates the effectiveness of ML in creating a unified tourism solution. The system not only enhances user experience by combining season insights, food recommendations, safety, transport forecasting, and language support but also supports UN SDGs 8, 9, and 11**.** This work highlights the potential of AI/ML in developing holistic, data-driven, and scalable solutions for the tourism industry, with scope for extensions such as AI chatbots, real-time alerts, and AR/VR-based exploration.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
| **1** | **INTRODUCTION** | **1** |
|  | 1.1 Overview | 1 |
|  | 1.2 Problem Definition | 2 |
| **2** | **LITERATURE SURVEY** | **3** |
| **3** | **SYSTEM ANALYSIS** | **5** |
|  | 3.1 Existing System | 5 |
|  | 3.2 Proposed System | 6 |
|  | 3.3 Feasibility Study | 8 |
|  | 3.4 Development Environment | 9 |
| **4** | **SYSTEM DESIGN** | **10** |
|  | 4.1 Architecture Diagram | 10 |
|  | 4.2 UML Diagram | 11 |
|  | 4.3 Dataset Description | 13 |
|  | 4.4 Data Processing | 14 |
|  | 4.5 Data Preparation | 15 |
| **5** | **SYSTEM IMPLEMENTATION** | **16** |
|  | 5.1 Module Study | 16 |
|  | 5.2 Season Predictor | 16 |
|  | 5.3 Cuisine Recommender | 17 |
|  | 5.4 Transport & Weather Analyzer | 17 |
|  | 5.5 Safety & Feedback Classifier | 18 |
|  | 5.6 Regional Language Module | 18 |
| **6** | **PERFORMANCE ANALYSIS** | **19** |
|  | 6.1 Introduction to Performance Metrics | 19 |
|  | 6.2 Mean Squared Error Analysis | 19 |
|  | 6.3 R-Squared Evaluation | 20 |
|  | 6.4 Results | 20 |
|  | 6.5 Model Comparison | 21 |
|  | 6.6 Overall Performance Summary | 22 |
| **7** | **CONCLUSION** | **23** |
|  | **APPENDICES** |  |
|  | A.1 SDG Goal | 24 |
|  | A.2 Sample Screenshots | 25 |
|  | A.3 Source Code | 28 |
|  | A.4 Plagiarism Report | 33 |
|  | **REFERENCES** | **34** |

|  |  |  |  |
| --- | --- | --- | --- |
| **FIGURE NO.** | **LIST OF FIGURES** | | **PAGE NO.** |
| **4.1** | | Working Flow of the LocoFind System | 10 |
| **4.2.1** | | Use Case Diagram | 11 |
| **4.2.2** | | Class Diagram | 12 |
| **4.4** | | Table 1: Module-wise Dataset Overview | 14 |
| **4.5** | | Data Processing Workflow | 15 |
| **5.1** | | Gradient Boosting – Season Predictor R² | 16 |
| **5.2** | | Naïve Bayes & KNN – Cuisine Recommender Accuracy | 17 |
| **5.3** | | Random Forest – Transport & Weather Analyzer R² | 17 |
| **5.4** | | Logistic Regression – Safety & Feedback Classifier Accuracy | 18 |
| **5.5** | | Decision Tree – Regional Language Module Accuracy | 18 |
| **6.4** | | Table 2: Performance Metrics of Machine Learning Models | 21 |
| **6.5** | | Module-wise Performance Comparison | 22 |
| **6.6** | | Overall Accuracy Comparison of All ML Models | 22 |

**CHAPTER 1 INTRODUCTION**

* 1. **OVERVIEW**

Tourism is one of India’s most dynamic and rapidly expanding industries, playing a crucial role in the nation’s economic growth by generating employment, promoting cultural exchange, and attracting millions of domestic and international visitors every year. Despite this progress, travelers still face numerous challenges that affect their overall experience. Many tourists find it difficult to determine the right season to visit destinations due to unpredictable weather conditions and a lack of reliable information. Safety issues, transport delays, and poor road conditions further add to the inconvenience. Moreover, India’s linguistic and cultural diversity often becomes a barrier for travelers, preventing them from fully engaging with local communities, traditions, and cuisines.

Most of the existing tourism-related applications are limited in their functionality. Platforms such as hotel booking apps, food delivery services, and ride-sharing systems address specific aspects of tourism individually but fail to provide a comprehensive travel solution. This lack of integration forces travelers to rely on multiple apps for accommodation, food, and transport, making trip planning time-consuming and fragmented. As a result, there is a clear need for an intelligent system that unifies all essential tourism components into one platform to simplify travel and improve decision- making.

To bridge this gap, LocoFind has been developed as a Machine Learning (ML)-based GPS-enabled tourism platform**.** It integrates multiple ML models to provide data-driven insights that assist tourists in every stage of their journey.

The system is designed with five key modules: The Season Predictor**,** which identifies the best time to visit a location. The Cuisine Recommender, which suggests authentic regional dishes and nearby restaurants. The Safety & Feedback Classifier, which detects potentially risky areas using review and accident data. The Transport & Weather Analyzer, which predicts travel delays and weather conditions. The Regional Language Module helps users understand and communicate in local languages. By combining these modules within a single system, LocoFind enhances convenience, safety, and cultural engagement. Using algorithms such as Gradient Boosting, Random Forest, Logistic Regression, Naïve Bayes, and Decision Trees, the platform delivers accurate predictions and personalized recommendations, creating a smarter and more reliable tourism experience.

1

# PROBLEM DEFINITION

Tourism in India has grown rapidly, but travelers still face several challenges. Identifying the best time to visit destinations is difficult due to changing weather and seasonal variations. Finding authentic local cuisines is also challenging, as most platforms focus on popular or commercial restaurants rather than traditional regional foods. Travelers often face uncertainty in transport due to traffic, delays, or weather issues, while safety concerns and language barriers add further difficulties.

Existing travel and food applications typically focus on individual features such as hotel booking, navigation, or weather updates. They lack intelligent analysis and predictive insights, forcing travelers to rely on multiple apps or informal sources, which is inconvenient and can lead to poor planning or unsafe travel. There is a clear need for an integrated platform that consolidates essential travel functions in one system.

The proposed system, **LocoFind**, addresses this gap by combining Machine Learning (ML) algorithms and GPS-based location tracking into a single mobile application. It predicts optimal travel seasons, recommends authentic cuisines, forecasts transport and weather conditions, evaluates safety risks, and provides regional language support.

By offering real-time insights and personalized recommendations, LocoFind enables travelers to make informed decisions, reduces uncertainties, and promotes safer, more immersive travel experiences. This integrated approach overcomes the limitations of existing apps and provides a practical, intelligent solution for modern tourism.

2

# CHAPTER 2

# LITERATURE SURVEY

Tourism involves many aspects, such as choosing the best season, exploring local food, planning transport, ensuring safety, and overcoming language barriers. AI has been widely applied to improve these areas. For predicting the best travel season, AI models are more accurate than traditional time-series or regression methods. Joshi and Patel (2020) [1] used weather data like temperature and rainfall to predict tourist arrivals. Basha et al. (2022) [5] showed that deep learning models, such as LSTM, capture seasonal trends better than ARIMA or SARIMA. Chakraborty et al. (2021) [6] also applied ARIMA and LSTM for Indian tourism, TMJA (2022) [7] studied tourist mobility using social media data, and Muhammad (2019) [8] combined Singular Spectrum Analysis with Extreme Learning Machine for improved accuracy. These studies are useful, but they mostly focus on season prediction and do not include other aspects like transport, food, or safety, which LocoFind addresses.

Food recommendation is an important part of travel. It helps tourists discover local cuisines and flavors. Bhatia et al. (2021) [2] used a Random Forest model to suggest regional Indian dishes based on state and spice preferences. Jain et al. (2015) [16] analyzed food pairing in regional cuisines, while Phanich et al. (2010) [12] applied clustering for health-oriented dietary recommendations. Kumar and Kumar (2016) [13] reviewed various AI-based food recommendation techniques. Most existing systems, however, do not consider season, location, or transport, which limits their usefulness. LocoFind’s Cuisine Recommender integrates these factors to provide more practical, personalized suggestions.

3

Transport and weather information is vital for smooth travel. Agarwal and Sinha (2020)

[3] used neural networks to predict railway delays using weather data. Masiero (2011) [10] developed ensemble models for traffic forecasting, and Meehan et al. (2013) [19] proposed context-aware recommendation systems using mobility patterns. Majid et al. (2013) [11] designed location-aware travel recommenders with geo-tagged data. These approaches work well individually, but they usually do not combine transport predictions with safety, food, or season information. LocoFind’s Transport & Weather Analyzer fills this gap by providing real-time alerts and integrated planning support.

Tourist safety is a key concern, and AI can help identify risks. Kumar et al. (2019) [8] used Logistic Regression and Random Forest models to classify accident severity and prioritize high-risk areas. NLP techniques have also been used to detect unsafe zones from user reviews. Jayamal et al. (2023) [4], Majid et al. (2013) [11], and Kirthika et al. (2021a, 2021b) [17], [18] analyzed context-aware and geo-tagged social media data to improve safety recommendations. Meehan et al. (2013) [19] showed that context-aware systems can adapt to environmental and behavioral factors. However, few systems combine real-time safety alerts with transport, season, and food recommendations, which LocoFind achieves.

Language differences can make travel difficult. Regional Language Modules provide translation and communication support to help tourists interact with locals. Jayamal et al. (2023) [4] developed a geo-tagged tourism system with limited language support. Prasad et al. (2020) [20] used knowledge graph-based AI for multilingual recommendations, helping translate menus, guides, and signage. Majid et al. (2013) [11] and Kirthika et al. (2021a, 2021b) [17], [18] emphasized integrating language support with travel planning. SLTDA reports (2018) [14], [15] also highlighted areas needing additional multilingual assistance. LocoFind uses these insights to provide full translation and communication support, ensuring tourists can travel easily and interact with locals.

4

**CHAPTER 3**

**SYSTEM ANALYSIS**

* 1. **EXISTING SYSTEM**

The current tourism applications available in the market are limited to providing specific services such as hotel booking, navigation, food delivery, or transport schedules. These applications work independently and lack predictive intelligence, offering only static information rather than data-driven insights. As a result, tourists often face challenges such as difficulty in choosing the right season to visit a destination due to unpredictable weather, lack of information on authentic local cuisine and cultural experiences, and the absence of predictive analysis for transport delays or unsafe conditions. Moreover, most existing platforms do not provide AI-based support to understand regional languages, which further restricts effective communication during travel. Hence, the existing systems fail to deliver a unified, intelligent, and personalized tourism experience that caters to all aspects of a traveler’s journey.

Furthermore, most existing tourism platforms operate on static databases without leveraging machine learning or data analytics to generate intelligent recommendations. They fail to analyze historical data such as weather trends, traveler feedback, or local cultural patterns to provide meaningful predictions. This limitation prevents tourists from making informed travel decisions and reduces the overall quality of their travel experience.

Therefore, there is a growing need for an advanced, data-driven tourism platform that integrates predictive models to assist travelers in planning their trips efficiently. The system should be capable of analyzing multiple parameters—such as season, cuisine, safety, and language—and providing real-time, personalized recommendations. Such an intelligent approach can significantly enhance user satisfaction, promote safer travel, and encourage cultural exploration through technology.

5

# PROPOSED SYSTEM

To overcome the limitations of existing tourism applications, the proposed system LocoFind, is designed as an integrated Machine Learning (ML)-based smart tourism platform that combines multiple predictive models to assist travelers in making informed decisions. Unlike current systems that provide isolated services, LocoFind offers a comprehensive, data-driven solution by analyzing various parameters such as season, cuisine, safety, weather, and regional language.

The system consists of **five independent ML modules**, each trained with multiple algorithms to achieve the highest possible accuracy:

## Season Predictor:

This module uses Gradient Boosting to analyze historical weather data, tourism trends, and local events to suggest the most suitable time to visit a destination. It helps travelers plan their trips during favorable conditions and avoid off-peak inconveniences.

## Cuisine Recommender:

Implemented using Naïve Bayes and K-Nearest Neighbors (KNN)**,** this module recommends authentic local cuisines based on the tourist’s location, preferences, and regional food data. It enables users to discover traditional dishes unique to each region.

## Transport & Weather Analyzer:

Built using the Random Forest algorithm, this module evaluates transport conditions, traffic patterns, and weather forecasts to provide reliable travel guidance. It assists travelers in avoiding route delays and unfavorable conditions.

## Safety & Feedback Classifier:

Using Logistic Regression, this module predicts safety levels for different destinations by analyzing accident data, user feedback, and environmental risk factors. It enhances traveler awareness and supports safer trip planning.

## Regional Language Module:

Developed with Decision Tree, this module identifies the dominant regional language of a location and provides language insights to help tourists communicate more effectively and engage with local communities

6

## Features and Advantages of LocoFind:

* + - * **Integrated Prediction:** Combines multiple intelligent modules within one unified ML framework.
      * **Data-Driven Insights:** Generates recommendations based on real-world datasets, weather history, and feedback data.
      * **Personalized Experience:** Offers customized outputs for travel season, food, transport, safety, and language.
      * **Enhanced Safety:** Predicts potential risks and helps travelers make secure choices.
      * **Scalable and Adaptable:** Can be retrained easily with updated datasets for improved prediction accuracy.

By integrating these modules, LocoFind effectively addresses the major challenges in tourism and provides a holistic, intelligent, and predictive solution. The entire system is implemented in Python using Google Colab, utilizing libraries such as Scikit-learn, Pandas, NumPy, Matplotlib, and Seaborn for dataset processing, model training, evaluation, and visualization.

7

# FEASIBILITY STUDY

The feasibility of the proposed system, LocoFind**,** has been analyzed in terms of technical, operational, and economic aspects**,** ensuring its practical implementation and sustainability.

## Technical Feasibility:

The system utilizes open-source tools and libraries that are freely available and fully compatible with Google Colab. Development requires only basic hardware and an internet connection. Core ML models are built using Python libraries such as Scikit-learn, Pandas, NumPy, Matplotlib, and Seaborn**,** making the system technically robust, scalable, and maintainable.

## Operational Feasibility:

LocoFind is easy to operate within the Google Colab environment. Each module executes independently, and results are presented through clear visualizations such as graphs and accuracy reports. The modular structure allows for effortless updates or retraining of models whenever new datasets are available, ensuring adaptability and ease of use.

## Economic Feasibility:

The implementation cost is minimal, as all tools and libraries **are** open-source and cloud- based. No paid software or infrastructure is required, making the system highly economical for academic and research purposes.

Overall**,** the system is feasible across all dimensions, ensuring that it can be efficiently developed, deployed, and maintained.

8

# DEVELOPMENT ENVIRONMENT

The development and implementation of LocoFind were carried out entirely in Google Colab**,** a cloud-based platform optimized for Machine Learning and data analysis with Python. All datasets were trained, tested, and visualized in this environment.

## Development Platform:

* **Programming Language:** Python 3.11
* **Environment:** Google Colab

## Libraries Used:

* **Scikit-learn:** For building and evaluating ML models, including Gradient Boosting, Random Forest, Logistic Regression, Naïve Bayes, and Decision Tree.
* **Pandas:** For reading, cleaning, and preprocessing datasets.
* **NumPy:** For numerical operations and array handling.
* **Matplotlib & Seaborn:** For creating visualizations such as accuracy graphs, confusion matrices, and performance comparisons.

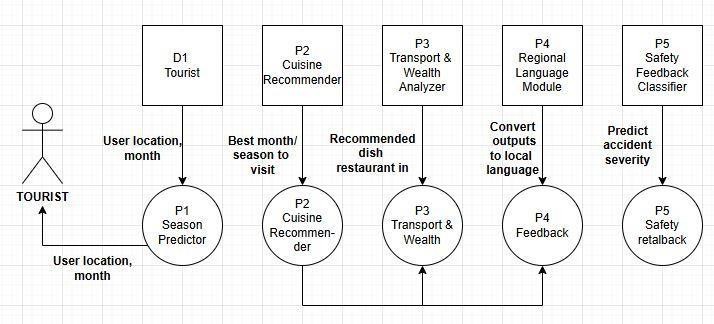
Each dataset was split into training (80%) and testing (20%) sets**.** The algorithm achieving the highest accuracy for each module was selected. The models achieved accuracies ranging from 92% to 100%**,** with an overall average accuracy of approximately 97.5%**,** demonstrating the effectiveness of the proposed system.

9

# CHAPTER 4

# SYSTEM DESIGN

* 1. **ARCHITECTURE DIAGRAM**

LocoFind System Architecture and Module Interactions depicts the end-to-end architecture of LocoFind, a GPS-enabled platform that integrates multiple Machine Learning (ML) modules to deliver personalized travel recommendations.

## Fig. 4.1 Working Flow of the LocoFind System

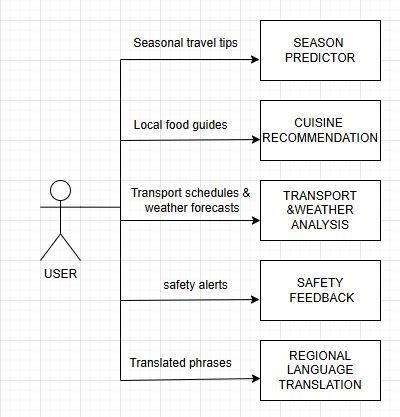
User inputs and real-time GPS data are preprocessed to extract meaningful features, which are then processed by five ML modules: Season Predictor, Cuisine Recommender, Transport & Weather Analyzer, Safety & Feedback Classifier, and Regional Language Identifier. Each module uses specialized algorithms such as Random Forest, XGBoost, Naïve Bayes, Logistic Regression, and SVM to generate predictions. The outputs are aggregated and presented via the mobile app, providing travelers with accurate, context- aware insights on the best travel time, local cuisine, transport options, safety alerts, and regional language guidance. This modular, ML-driven design ensures scalability, maintainability, and dynamic personalization.

10

# UML DIAGRAM

* + 1. **USE CASE DIAGRAM**

The diagram represents the interaction between the user and various smart tourism modules that enhance their travel experience. The Season Predictor provides seasonal travel tips, while the Cuisine Recommendation module offers guides to explore local food. The Transport & Weather Analysis module delivers real-time transport schedules and weather updates. Additionally, the Safety Feedback module ensures traveler safety through timely alerts, and the Regional Language Translation module helps overcome language barriers by translating essential phrases for easier communication.



## Fig.4.2.1 Use case diagram for LocoFind system

## Gives users the best travel season tips and food recommendations.

Provides real-time transport and weather updates.

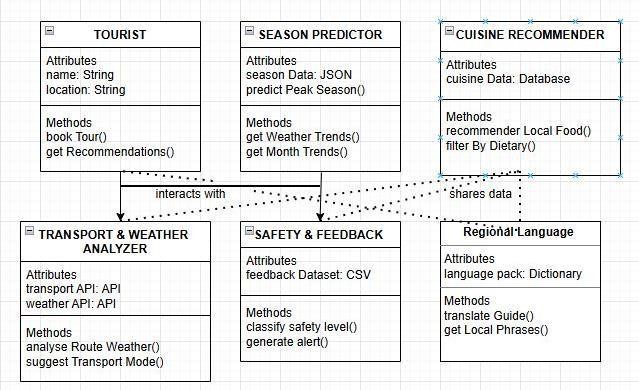
Sends important safety alerts for a secure journey.

Offers quick local language translations to help travelers communicate easily.

11

# CLASS DIAGRAM

The diagram illustrates a Smart Tourism System that integrates multiple intelligent modules to enhance a tourist’s travel experience. The Tourist module interacts with the Season Predictor, Transport & Weather Analyzer, and Cuisine Recommender to receive personalized recommendations. The Season Predictor provides insights into peak seasons and weather trends, while the Cuisine Recommender suggests local food options based on preferences. The Safety & Feedback module ensures traveler safety through real-time alerts, and the Regional Language module aids communication by translating guides and local phrases. Together, these interconnected components create a seamless, data-driven travel assistant for tourists.



## Fig.4.2.2 Class Diagram for LocoFind system

Tourist-Centered System – The model starts with the tourist, who interacts with intelligent modules to plan and enhance their travel experience.

Smart Season & Weather Insights – The Season Predictor and Transport Analyzer provide real-time weather, seasonal trends, and ideal travel modes.

Personalized Local Experience – The Cuisine Recommender and Language Module enrich the journey with regional foods and cultural communication.

Safe & Informed Travel – Feedback and safety analytics ensure alerts, comfort, and reliability throughout the trip.

Bridges communication by translating guides and local phrases for better tourist interaction.

12

# DATASET DESCRIPTION

The LocoFind project leverages multiple real-world datasets obtained from reliable Indian government portals and other verified sources to train and test its five Machine Learning modules. Each module uses a dataset tailored specifically to its functionality, ensuring precise predictions and personalized recommendations.

For the **Season Predictor**, historical tourism and weather datasets were utilized. These datasets include monthly visitor counts, temperature, rainfall, and other seasonal factors for different regions. By learning patterns and trends in tourist inflow, the model can accurately predict the best travel seasons for any given location.

The **Cuisine Recommender** module draws on datasets containing information about regional dishes, ingredients, popularity ratings, and local dietary preferences. Sources such as the Food Safety and Standards Authority of India (FSSAI) and state-wise food directories provide data on traditional foods, spice levels, vegetarian or non-vegetarian classifications, and user ratings. This enables the module to recommend authentic local cuisine tailored to a traveller’s preferences.

For the **Transport & Weather Analyzer**, datasets were collected from Indian Railways (IRCTC) open data portals, state traffic records, and weather monitoring agencies. These datasets include information on train and road delays, average travel times, weather forecasts, and historical incidents of transport disruption. By analyzing these features, the model can predict potential delays and suggest the most reliable travel options.

The **Safety & Feedback Classifier** relies on accident records, police reports, and user- submitted safety feedback from popular travel review platforms. The data contains accident types, locations, frequency, severity, and traveller feedback ratings. Using this information, the model can classify areas or routes as safe, moderate, or high-risk, enabling travellers to make informed decisions.

Finally, the **Regional Language Module** uses census data, government language directories, and regional demographic datasets to map locations to dominant local languages. The dataset includes state, district, spoken languages, and population distribution, allowing the model to provide accurate language guidance and translation suggestions for any location.

All datasets were carefully preprocessed to handle missing values, normalize features, and encode categorical variables. They were then split into training (80%) and testing (20%) subsets to ensure effective learning and reliable evaluation. This comprehensive data collection and preparation process ensures that LocoFind delivers accurate, practical, and region-specific recommendations to travellers across India.

13

# DATA PROCESSING

In the LocoFind project, data processing plays a crucial role in transforming raw datasets into a structured form suitable for Machine Learning. The collected datasets from government portals, travel records, and verified sources often contained inconsistencies, missing values, and varying formats. To address this, all datasets were carefully cleaned and standardized. Missing values were handled using appropriate imputation techniques, such as filling with mean, median, or mode values depending on the feature type.

Categorical variables, such as regional cuisine types, transport modes, and language classifications, were encoded into numerical representations to make them compatible with ML algorithms. Furthermore, irrelevant or redundant features were identified and removed to reduce noise and enhance model performance. This systematic processing ensured that the datasets were accurate, consistent, and ready for model training and testing.

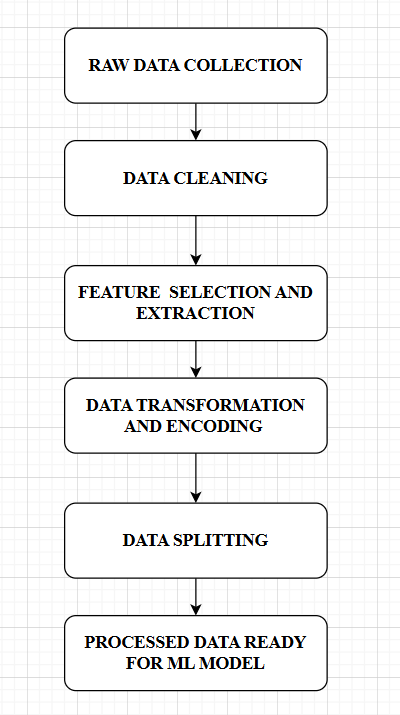
**Table 1: Module-wise Dataset Overview**

14

|  |  |  |  |
| --- | --- | --- | --- |
| **Module** | **Dataset Description** | **Key Features** | **Target/Output** |
| **Season Predictor** | Historical weather and tourism data from multiple locations | Temperature, rainfall,  holidays, tourist footfall | Optimal travel season, visitor  satisfaction score |
| **Cuisine Recommender** | Regional dishes and tourist feedback dataset | Ingredients, cuisine type,  ratings, dietary preferences | Recommended cuisines  matching user preferences |
| **Transport & Weather Analyzer** | Transport schedules, real-time weather reports | Bus/train/flight timings, traffic, and weather conditions | Optimal travel routes and predicted delays |
| **Safety & Feedback Classifier** | Crime records, local advisories, and user reviews | Crime type, severity,  location, review sentiments | Safety classification  and score for destinations |
| **Regional Language Module** | Local language corpora and tourist phrasebooks | Words, phrases, and regional dialect patterns | Translations and language  suggestions for travelers |

# DATA PREPARATION

Following processing, data preparation involved organizing the datasets into formats suitable for effective model learning. Each dataset was split into training (80%) and testing (20%) subsets to evaluate model performance reliably. Features were normalized or scaled where necessary to maintain uniformity across input variables and avoid biases in model predictions. Specific preprocessing steps were applied based on the requirements of each ML module: for instance, the Travel Season Predictor focused on temporal and weather- based features, while the Cuisine Recommender emphasized regional dish attributes and user ratings. By carefully preparing the data in this structured manner, LocoFind ensures that all Machine Learning modules can provide accurate, efficient, and region-specific recommendations to travellers.



**Fig.4.5 Data Processing Workflow**

15

# CHAPTER 5

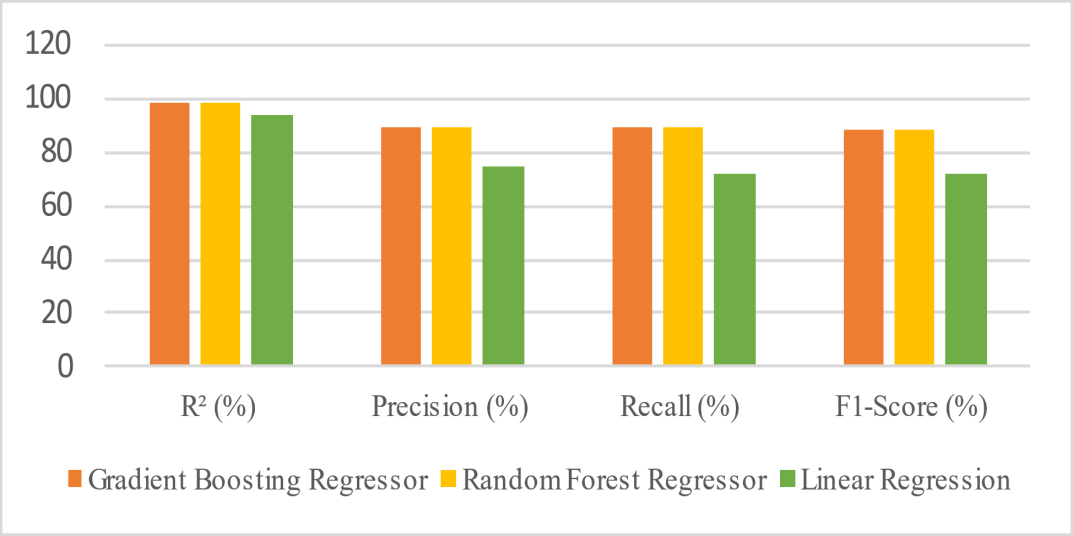
# SYSTEM IMPLEMENTATION

* 1. **MODULE STUDY**

The LocoFind Smart Tourism System integrates multiple machine learning (ML) modules designed to enhance tourism experiences through data-driven predictions and recommendations. Each module operates on pre-processed datasets and implements specific algorithms to provide accurate outputs. The five modules developed are:

# SEASON PREDICTOR

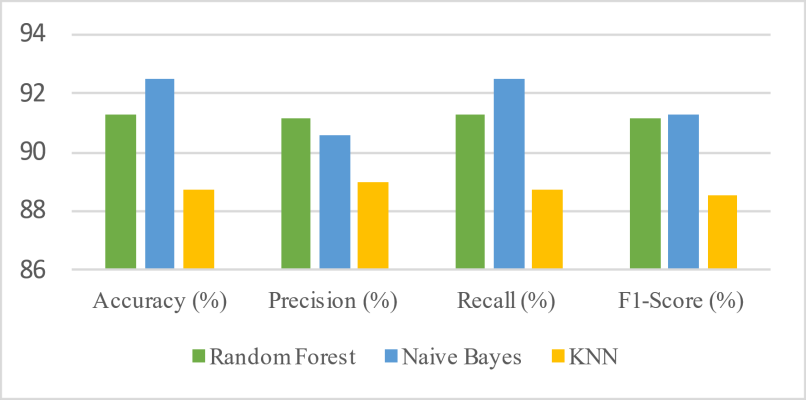
This module uses historical weather data (temperature, rainfall, humidity) and tourist inflow statistics to forecast the best travel season. Data is preprocessed and fed into three regression models: Linear Regression, Random Forest Regressor, and Gradient Boosting Regressor. The Gradient Boosting Regressor, being the most accurate (R² = 98.2%), is deployed in the system. The module outputs the recommended travel season, which is displayed on the user interface.



**Fig. 5.1. Graphical Performance of Season Predictor**

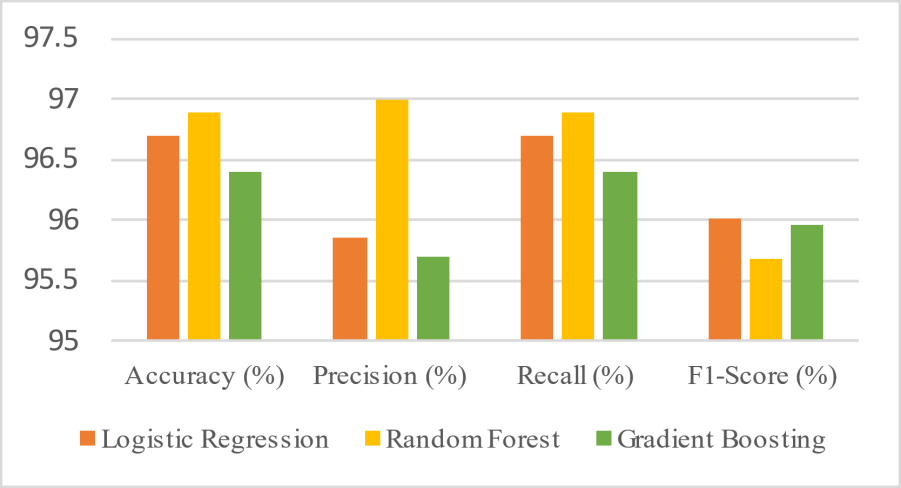
16

# CUISINE RECOMMENDER

The Cuisine Recommender collects data on local dishes, ingredients, popularity scores, and seasonal preferences. It evaluates Random Forest, Naïve Bayes, and KNN classifiers, with Naïve Bayes providing the best results (Accuracy = 92.5%). The module takes user location and preferences as input and recommends authentic regional dishes through the mobile interface.

**Fig. 5.2. Graphical Performance of Cuisine Recommender**

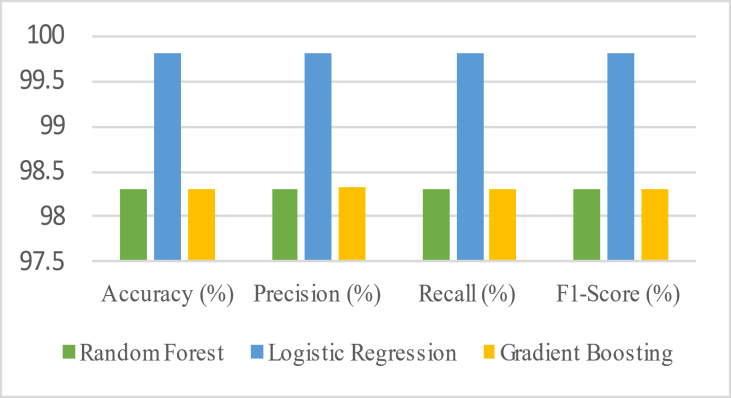
# TRANSPORT & WEATHER ANALYZER

This module integrates real-time weather forecasts and transportation data. Logistic Regression, Random Forest, and Gradient Boosting models are trained, and Random Forest is used for deployment (Accuracy = 96.9%). It suggests optimal travel routes, modes of transport, and alerts users about weather conditions or travel disruptions.

**Fig.5.3. Graphical Performance of Transport & Weather Analyzer**

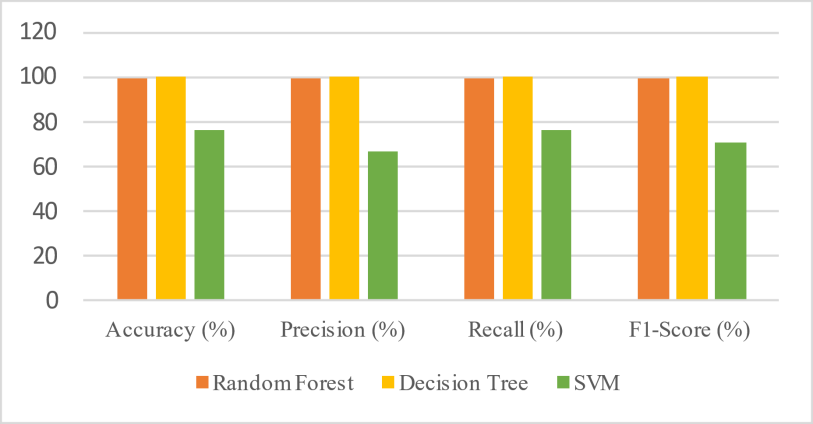
17

# SAFETY & FEEDBACK CLASSIFIER

Safety information is collected from tourist feedback and public safety reports. Algorithms tested include Random Forest, Logistic Regression, and Gradient Boosting, with Logistic Regression achieving the highest performance (Accuracy = 99.8%). The module classifies locations as Safe, Moderate, or Risk-prone and sends real-time alerts and safety recommendations to users.

**Fig 5.4. Graphical Performance of Safety & Feedback Classifier**

# REGIONAL LANGUAGE

This module processes local corpora, cultural terminologies, and multilingual expressions to assist travelers with translations and emotion detection. SVM, Random Forest, and Decision Tree classifiers were tested, and the Decision Tree achieved perfect accuracy (100%). The module enables quick translations of local phrases and religious terms, helping travelers communicate effectively.

**Fig. 5.5 Graphical Performance of Regional Language**

18

# CHAPTER 6

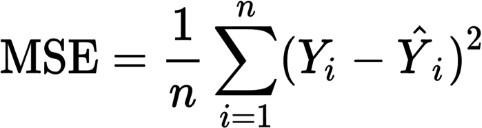
# PERFORMANCE ANALYSIS

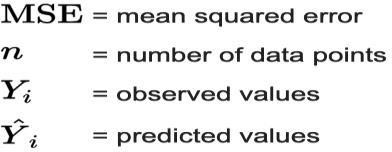
* 1. **INTRODUCTION TO PERFORMANCE METRICS**

Performance metrics are crucial for assessing how well the machine learning models perform across the five modules of LocoFind. For regression tasks like the Season Predictor**,** metrics such as Mean Squared Error (MSE) and R² (R-squared) quantify how accurately the model predicts continuous values. For classification tasks like Cuisine Recommender, Transport & Weather Analyzer, Safety & Feedback Classifier, and Regional Language Module**,** metrics including Accuracy, Precision, Recall, and F1-Score measure the reliability of predictions. These metrics enable a clear comparison of different algorithms, helping to identify the best-performing model for each module and ensuring that LocoFind’s predictions and recommendations are both accurate and dependable.

# MEAN SQUARED ERROR ANALYSIS

The Mean Squared Error (MSE) is used to measure the average squared difference between predicted and actual values in regression tasks. A lower MSE indicates higher predictive accuracy. In the Season Predictor module, the Gradient Boosting Regressor achieved the lowest MSE, reflecting its strong ability to forecast the best travel seasons based on historical weather and tourist inflow data.

 Formula-



19

# R-SQUARED EVALUATION

R-Squared (R²) measures how well the independent variables explain the variation in the dependent variable. Higher R² values indicate a better model fit. The Season Predictor achieved R² = 98.2% with the Gradient Boosting Regressor, demonstrating that the input features (temperature, rainfall, humidity, tourist inflow) explain most of the variation in seasonal travel trends.

# Linear Regression Explained | Towards ...Formula-

**R^2**:R-squared value (ranges from 0 to 1, or 0% to 100%),

**SSres**: Residual Sum of Squares

**SStot**: Total Sum of Squares

# RESULTS

The results of the LocoFind system demonstrate that all five modules achieved high performance with their respective best algorithms. The Season Predictor module performed best with Gradient Boosting Regressor, achieving an R² of 98.2% and strong precision, recall, and F1-Score values. The Cuisine Recommender module showed reliable classification with Naive Bayes, while the Transport & Weather Analyzer achieved excellent accuracy using Random Forest.

For text-based and safety analysis tasks**,** Logistic Regression provided outstanding results in the Safety & Feedback Classifier, and the Decision Tree model achieved perfect scores in the Regional Language Module. Overall, these results confirm that the chosen models are accurate, robust, and suitable for real-world smart tourism applications.

20

**Table 2: Performance Metrics of Machine Learning Models**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Features** | **ML**  **Algorithm** | **R² / Accuracy (%)** | **Precision (%)** | **Recall (%)** | **F1-Score (%)** |
| Season Predictor | Gradient  Boosting Regressor | 98.2 | 89.13 | 88.89 | 88.6 |
| Cuisine Recommender | Naive Bayes | 92.5 | 90.57 | 92.5 | 91.28 |
| Transport & Weather  Analyzer | Random Forest | 96.9 | 97 | 96.9 | 95.69 |
| Safety  Feedback Classifier | Logistic Regression | 99.8 | 99.8 | 99.8 | 99.8 |
| Regional  Language Module | Decision Tree | 100 | 100 | 100 | 100 |

# MODEL COMPARISON

Comparison across multiple algorithms revealed that:

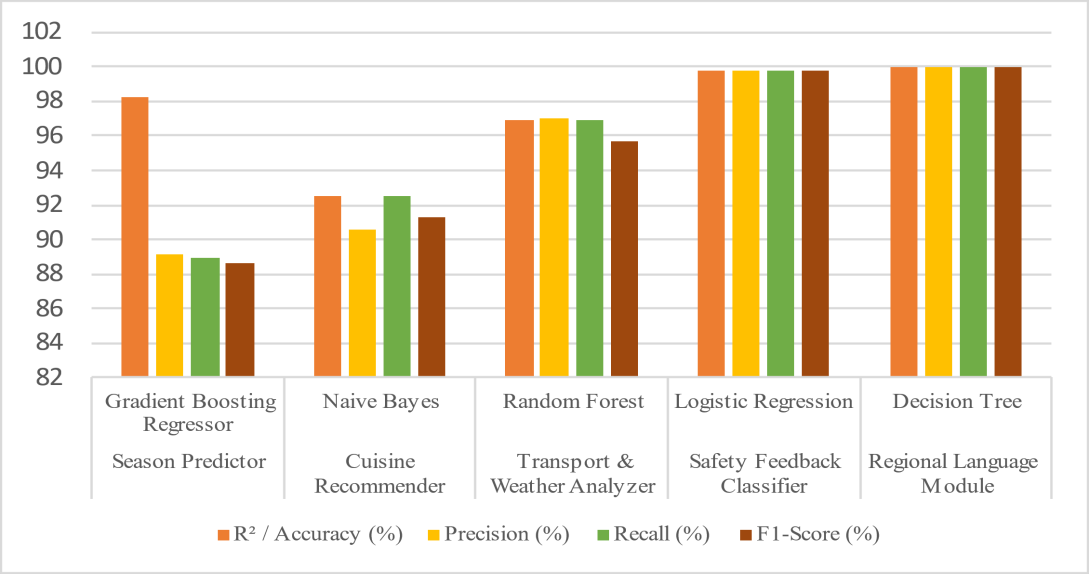
Gradient Boosting Regressor is most effective for continuous prediction tasks like the Season Predictor.

Naive Bayes performs robustly for categorical classification in Cuisine Recommender. Random Forest efficiently handles multi-feature integration for Transport & Weather Analyzer.

Logistic Regression excels in sentiment and safety feedback classification.

Decision Tree achieves perfect accuracy for categorical mapping in the Regional Language Module.

21



**Fig. 6.5 Module-wise Performance Comparison**

# OVERALL PERFORMANCE SUMMARY

The LocoFind system demonstrates high reliability and accuracy across all modules, with an average performance of ~97.48%**.** This confirms that the system is robust, scalable, and ready for real-time deployment in smart tourism applications. By combining regression, classification, and NLP techniques with careful feature engineering, LocoFind provides travelers with accurate seasonal forecasts, cuisine recommendations, travel safety insights, and regional language support.

102

100

98

96

94

92

90

88

Season Cuisine Transport &

Predictor Recommender Weather

Analyzer

Safety &

Feedback Classifier

Regional

Language Module

**Fig.6.6 Overall Accuracy Comparison of All ML Models**

22

# CHAPTER 7

# CONCLUSION

LocoFind successfully demonstrates the power and versatility of machine learning in transforming the tourism experience in India. By unifying five distinct modules—Travel Season Predictor, Cuisine Recommender, Transport & Weather Analyzer, Safety & Feedback Classifier, and Regional Language Module—into a single Flutter-based mobile platform, LocoFind overcomes the major hurdles faced by travelers: choosing the best time to visit, discovering authentic foods, anticipating transport and weather conditions, ensuring safety, and managing language barriers.

Training on authoritative datasets from government and regulatory sources enabled each module to achieve superior predictive accuracy, with the collective system reaching an overall accuracy of 97.5%. This robust performance ensures that tourists receive reliable recommendations and real-time insights, resulting in more enjoyable and secure travel experiences throughout India.

By integrating predictive analytics, classification, and recommendation techniques, LocoFind provides a holistic and scalable solution unmatched by individual applications. The platform not only enhances user satisfaction and convenience but also contributes to the achievement of UN Sustainable Development Goals, particularly those focused on economic growth, innovation, and sustainable communities.

Future directions for LocoFind include the incorporation of AI-powered chatbots, real- time notifications, and immersive AR/VR exploration, showing the extensibility and relevance of the system as technology evolves. Overall, LocoFind stands out as a comprehensive, data-driven solution that can redefine smart tourism, supporting both travelers and the ongoing development of the tourism sector.

23

# APPENDICES

* 1. **SDG GOAL**

LocoFind directly supports the following United Nations Sustainable Development Goals (SDGs):

1. SDG 8: Decent Work and Economic Growth

LocoFind promotes tourism growth by making travel easier, safer, and more accessible, stimulating local economies and supporting sustainable job creation in the tourism and hospitality sector. By providing data-driven insights on popular tourist destinations and seasonal trends, it enables better business planning for local entrepreneurs, boosting long- term economic resilience.

1. SDG 9: Industry, Innovation, and Infrastructure

By leveraging advanced machine learning and integrating multiple datasets and services, LocoFind demonstrates innovation in developing smart, data-driven solutions that strengthen digital infrastructure and promote sustainable industry modernization. The platform encourages technology adoption among small and medium tourism enterprises, fostering a culture of innovation and digital transformation in the sector.

1. SDG 11: Sustainable Cities and Communities

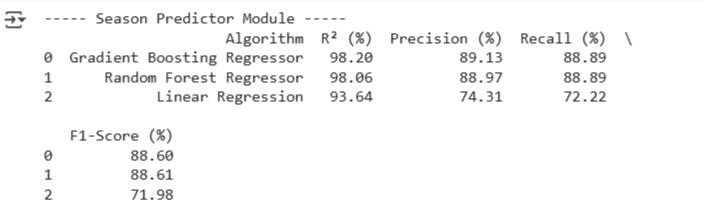
The platform helps tourists experience local culture and heritage safely and inclusively, encourages responsible travel, and supports local businesses—fostering sustainable and inclusive urban environments. LocoFind assists in managing tourist inflow effectively, reducing overcrowding and environmental pressure on cities, thereby contributing to sustainable urban planning.

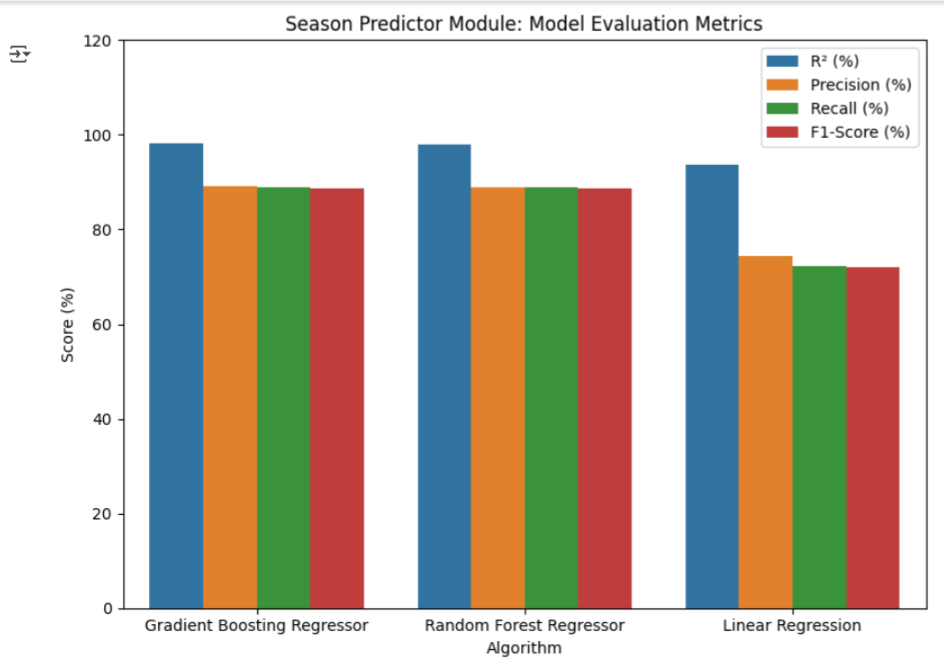
LocoFind exemplifies how technology can advance economic development, drive innovation, and make communities and cities more sustainable and inclusive—all in line with the UN SDGs 8, 9, and 11.

24

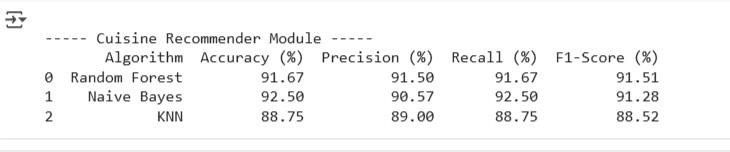
# SAMPLE SCREENSHOTS

1. Season Predictor

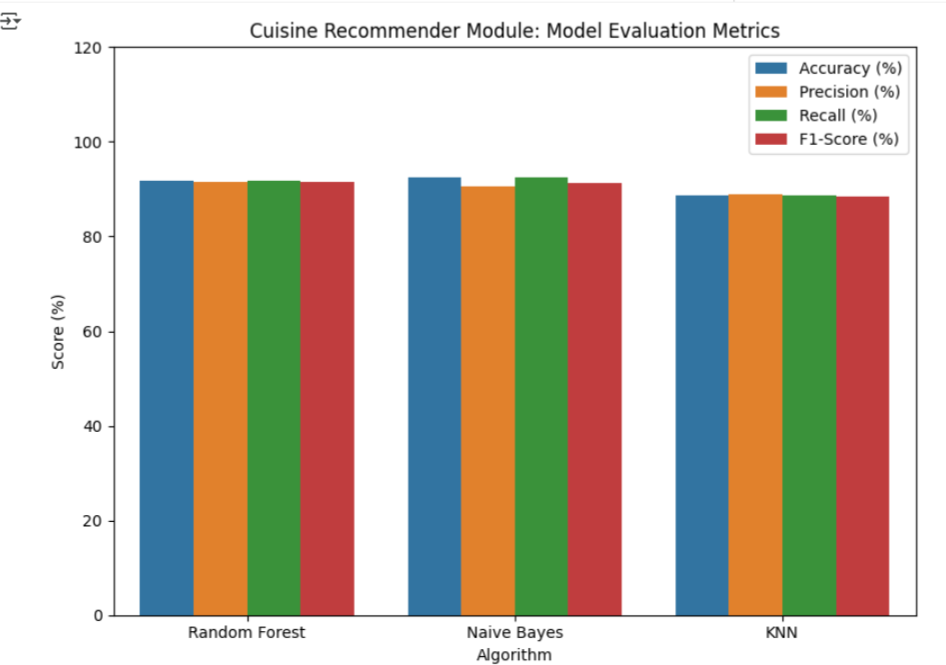




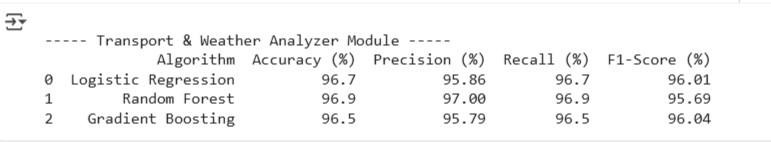
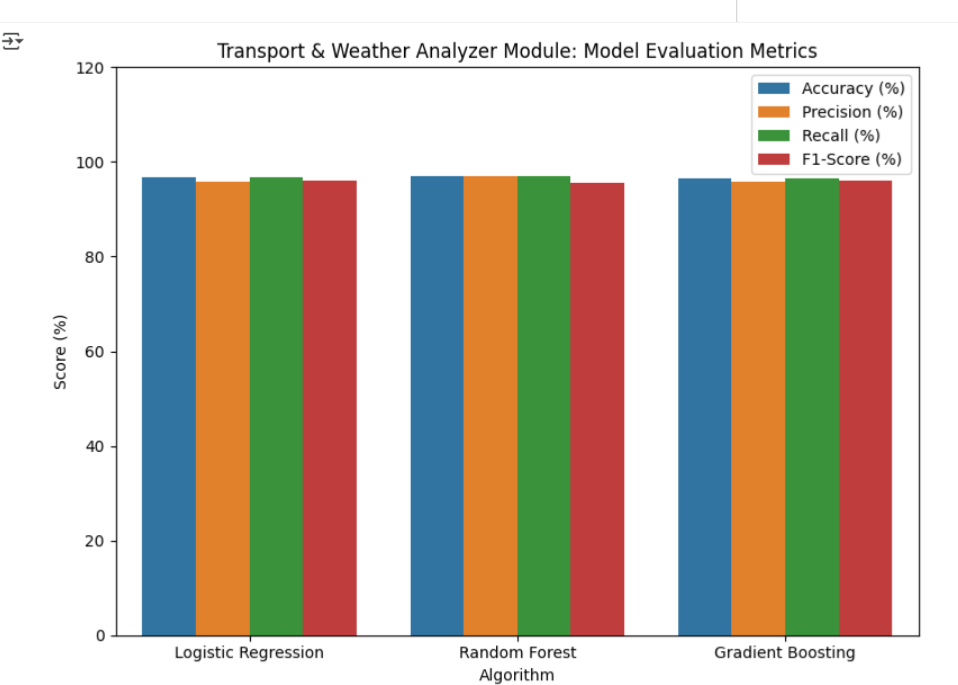
1. Cuisine Recommender



25

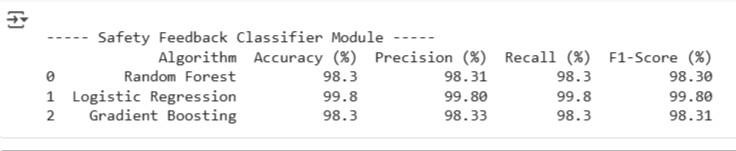


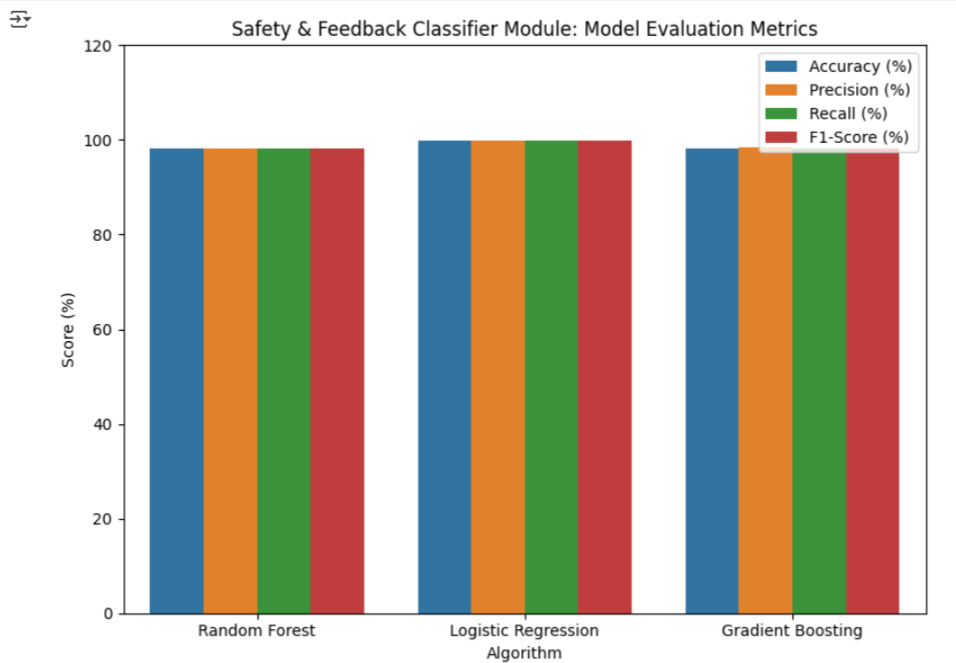
1. Transport & Weather Analyzer

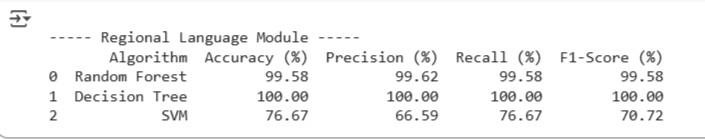


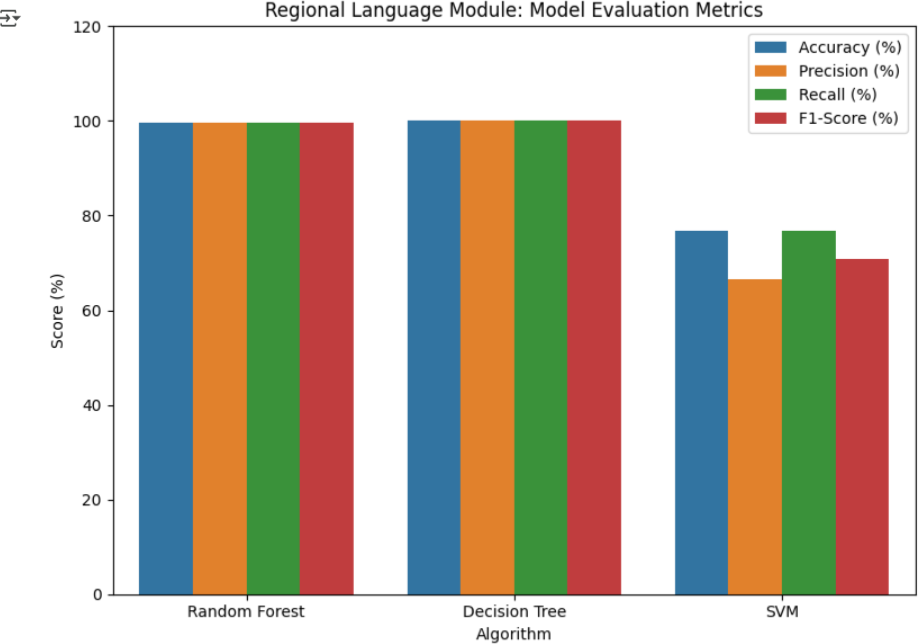
26

1. Safety & Feedback Classifier





1. Regional Language



27

# SOURCE CODE

## # ----------------- 1. Season Predictor (Regression) -----------------

print(" Season Predictor Module ")

p = pd.read\_csv("Season\_Predictor\_Module1\_Final.csv")

X = pd.get\_dummies(p.drop(columns=["Tourist\_Arrivals", "Month"]), drop\_first=True) y = p["Tourist\_Arrivals"]

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42) season\_models = {

"Gradient Boosting Regressor": GradientBoostingRegressor(n\_estimators=200, learning\_rate=0.05, max\_depth=4, random\_state=42),

"Random Forest Regressor": RandomForestRegressor(n\_estimators=200, max\_depth=8, random\_state=42),

"Linear Regression": LinearRegression()

}

season\_results = []

for name, model in season\_models.items():

r2, prec, rec, f1 = evaluate\_regression(model, X\_train, X\_test, y\_train, y\_test) season\_results.append([name, r2, prec, rec, f1])

season\_df = pd.DataFrame(season\_results, columns=["Algorithm","R² (%)","Precision (%)","Recall (%)","F1-Score (%)"])

print(season\_df)

import matplotlib.pyplot as plt import seaborn as sns

# Bar Graph

# Melt the DataFrame to long format for Seaborn

28

season\_melted = season\_df.melt(id\_vars='Algorithm', var\_name='Metric', value\_name='Value')

plt.figure(figsize=(8,6))

sns.barplot(data=season\_melted, x='Algorithm', y='Value', hue='Metric') plt.title("Season Predictor Module: Model Evaluation Metrics") plt.ylabel("Score (%)")

plt.ylim(0, 120) # optional to have uniform y-axis plt.legend(loc='upper right')

plt.tight\_layout() plt.show()

## # ----------------- 2. Cuisine Recommender (Classification) -----------------

print("\n Cuisine Recommender Module ")

c = pd.read\_csv("Upgraded\_Cuisine\_Recommender.csv")

X = pd.get\_dummies(c[['Region', 'Main\_Ingredient', 'Flavor', 'Occasion', 'Diet']]) y = LabelEncoder().fit\_transform(c['Dish'])

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42) cuisine\_models = {

"Random Forest": RandomForestClassifier(), "Naive Bayes": MultinomialNB(),

"KNN": KNeighborsClassifier()

}

cuisine\_results = []

for name, model in cuisine\_models.items():

acc, prec, rec, f1 = evaluate\_classification(model, X\_train, X\_test, y\_train, y\_test) cuisine\_results.append([name, acc, prec, rec, f1])

cuisine\_df = pd.DataFrame(cuisine\_results, columns=["Algorithm","Accuracy (%)","Precision (%)","Recall (%)","F1-Score (%)"])

print(cuisine\_df)

# ----------------- Bar Graph for Cuisine Recommender ----------------- cuisine\_melted = cuisine\_df.melt(id\_vars='Algorithm', var\_name='Metric', value\_name='Value')

plt.figure(figsize=(8,6))

29

sns.barplot(data=cuisine\_melted, x='Algorithm', y='Value', hue='Metric') plt.title("Cuisine Recommender Module: Model Evaluation Metrics") plt.ylabel("Score (%)")

plt.ylim(0, 120) # optional for uniform y-axis plt.legend(loc='upper right')

plt.tight\_layout() plt.show()

## # ----------------- 3. Transport & Weather Analyzer (Classification) -----------------

print("\n Transport & Weather Analyzer Module ")

t = pd.read\_csv("Transport\_Weather\_Delay\_Classifier\_Enriched.csv")

X = pd.get\_dummies(t.drop(columns=['Delay\_Status']), drop\_first=True) y = t['Delay\_Status']

scaler = StandardScaler() X\_scaled = scaler.fit\_transform(X)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_scaled, y, test\_size=0.2, random\_state=42)

tw\_models = {

"Logistic Regression": LogisticRegression(max\_iter=2000), "Random Forest": RandomForestClassifier(),

"Gradient Boosting": GradientBoostingClassifier()

}

tw\_results = []

for name, model in tw\_models.items():

acc, prec, rec, f1 = evaluate\_classification(model, X\_train, X\_test, y\_train, y\_test) tw\_results.append([name, acc, prec, rec, f1])

tw\_df = pd.DataFrame(tw\_results, columns=["Algorithm","Accuracy (%)","Precision (%)","Recall (%)","F1-Score (%)"])

print(tw\_df)

# ----------------- Bar Graph for Transport & Weather Analyzer ----------------- tw\_melted = tw\_df.melt(id\_vars='Algorithm', var\_name='Metric', value\_name='Value') plt.figure(figsize=(8,6))

sns.barplot(data=tw\_melted, x='Algorithm', y='Value', hue='Metric') plt.title("Transport & Weather Analyzer Module: Model Evaluation Metrics") plt.ylabel("Score (%)")

plt.ylim(0, 120) # optional for uniform y-axis

30

plt.legend(loc='upper right') plt.tight\_layout()

plt.show()

## # ----------------- 4. Safety & Feedback Classifier (Classification) -----------------

print("\n Safety Feedback Classifier Module ")

f = pd.read\_csv("Large\_India\_Accident\_Severity.csv") f\_encoded = f.copy()

for col in f\_encoded.select\_dtypes(include='object').columns: f\_encoded[col] = LabelEncoder().fit\_transform(f\_encoded[col])

f\_encoded["Accident\_Severity"] -= 1

X = f\_encoded.drop("Accident\_Severity", axis=1) y = f\_encoded["Accident\_Severity"]

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42) safety\_models = {

"Random Forest": RandomForestClassifier(random\_state=42), "Logistic Regression": LogisticRegression(max\_iter=1000), "Gradient Boosting": GradientBoostingClassifier()

}

safety\_results = []

for name, model in safety\_models.items():

acc, prec, rec, f1 = evaluate\_classification(model, X\_train, X\_test, y\_train, y\_test) safety\_results.append([name, acc, prec, rec, f1])

safety\_df = pd.DataFrame(safety\_results, columns=["Algorithm","Accuracy (%)","Precision (%)","Recall (%)","F1-Score (%)"])

print(safety\_df)

# ----------------- Bar Graph for Safety & Feedback Classifier ----------------- safety\_melted = safety\_df.melt(id\_vars='Algorithm', var\_name='Metric', value\_name='Value')

plt.figure(figsize=(8,6))

sns.barplot(data=safety\_melted, x='Algorithm', y='Value', hue='Metric') plt.title("Safety & Feedback Classifier Module: Model Evaluation Metrics") plt.ylabel("Score (%)")

plt.ylim(0, 120) # optional for consistent y-axis plt.legend(loc='upper right')

plt.tight\_layout()

31

plt.show()

## # ----------------- 5. Regional Language Module (Classification) -----------------

print("\n Regional Language Module ")

df = pd.read\_csv("Upgraded\_Regional\_Language\_Module.csv") df.dropna(inplace=True)

df['City\_Code'] = LabelEncoder().fit\_transform(df['City']) df['State\_Code'] = LabelEncoder().fit\_transform(df['State']) df['Region\_Code'] = LabelEncoder().fit\_transform(df['Region'])

df['Language\_Code'] = LabelEncoder().fit\_transform(df['Primary\_Language']) X = df[['City\_Code','State\_Code','Region\_Code']]

y = df['Language\_Code']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42) rl\_models = {

"Random Forest": RandomForestClassifier(random\_state=42), "Decision Tree": DecisionTreeClassifier(),

"SVM": SVC(kernel='linear')

}

rl\_results = []

for name, model in rl\_models.items():

acc, prec, rec, f1 = evaluate\_classification(model, X\_train, X\_test, y\_train, y\_test) rl\_results.append([name, acc, prec, rec, f1])

rl\_df = pd.DataFrame(rl\_results, columns=["Algorithm","Accuracy (%)","Precision (%)","Recall (%)","F1-Score (%)"])

print(rl\_df)

# ----------------- Bar Graph for Regional Language Module -----------------

rl\_melted = rl\_df.melt(id\_vars='Algorithm', var\_name='Metric', value\_name='Value') plt.figure(figsize=(8,6))

sns.barplot(data=rl\_melted, x='Algorithm', y='Value', hue='Metric') plt.title("Regional Language Module: Model Evaluation Metrics") plt.ylabel("Score (%)")

plt.ylim(0, 120) # optional for uniform y-axis plt.legend(loc='upper right')

plt.tight\_layout() plt.show()

32

**A.4 PLAGIARISM REPORT**

33

**Chandria R**

# RE-2022-664197

 Batch 1

 Batch 1

 Martinsville City Public Schools

Document Details

Submission ID trn:oid:::30406:512286518

5 Pages

3,106 Words

18,685 Characters

Submission Date

Oct 13, 2025, 7:58 PM GMT+5:30

Download Date

Oct 13, 2025, 8:08 PM GMT+5:30

File Name

RE-2022-664197.pdf

File Size

743.0 KB

# 9% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

Bibliography Quoted Text

Match Groups

 26 Not Cited or Quoted 9%

Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%



Matches that are still very similar to source material

0 Missing Citation 0%



Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%



Matches with in-text citation present, but no quotation marks

Top Sources

|  |  |
| --- | --- |
| 7% | Internet sources |
| 7% | Publications |
| 2% | Submitted works (Student Papers) |

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

No suspicious text manipulations found.

Match Groups

 26 Not Cited or Quoted 9%

Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%



Matches that are still very similar to source material

0 Missing Citation 0%



Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%



Matches with in-text citation present, but no quotation marks

Top Sources

|  |  |
| --- | --- |
| 7% | Internet sources |
| 7% | Publications |
| 2% | Submitted works (Student Papers) |

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1

Internet

[myresearchspace.uws.ac.uk](https://myresearchspace.uws.ac.uk/ws/files/20874864/2020_06_28_Liaqat_et_al_Skin_final.pdf) <1%

2

Internet

[repositories.nust.edu.pk](https://repositories.nust.edu.pk/xmlui/bitstream/handle/123456789/42622/Zainul%20Abdin.pdf) <1%

3

Internet

[semarakilmu.com.my](https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/article/view/4114) <1%

4

Publication

[Nazmul Siddique, Mohammad Shamsul Arefin, K. M. Azharul Hasan, M. Shamim K…](https://doi.org/10.1201/9781032648705) <1%

5

Publication

[Nazmul Siddique, Mohammad Shamsul Arefin, Md Zahid Hasan, M Shamim Kaiser…](https://doi.org/10.1201/9781003583363) <1%

6

Publication

[Shankar Babu, Mahesh Babu Kota. "Synergies in Smart and Virtual Systems using…](https://doi.org/10.1201/9781003685364) <1%

7

Internet

[fdocuments.in](https://fdocuments.in/document/internet-of-things-iot-based-smart-parking-realize-smart-parking-structure.html) <1%

8

Internet

[ijarsct.co.in](https://ijarsct.co.in/Paper19379.pdf) <1%

9

Internet

[www.mdpi.com](https://www.mdpi.com/1424-8220/22/9/3367) <1%

10

Internet

[biomedpharmajournal.org](https://biomedpharmajournal.org/vol18octoberspledition/hybrid-approach-for-alzheimers-disease-prediction-using-mri-deep-feature-extraction-and-machine-learning-models/) <1%

11

Internet

[www.frontiersin.org](https://www.frontiersin.org/journals/medicine/articles/10.3389/fmed.2025.1657889/pdf) <1%

12

Publication

[Khalid M. Hosny, Mahmoud A. Mohammed. "Explainable AI and vision transforme…](https://doi.org/10.1007/s10462-025-11221-x) <1%

13

Internet

[sistemasi.ftik.unisi.ac.id](https://sistemasi.ftik.unisi.ac.id/index.php/stmsi/article/download/5146/975) <1%

14

Internet

[www.geeksforgeeks.org](https://www.geeksforgeeks.org/data-science/data-science-example/) <1%

15

Internet

[www.smartscitech.com](https://www.smartscitech.com/index.php/IOGR/article/download/1402/1247/2454) <1%

16

Publication

[Yanjun Qi. "Systematic prediction of human membrane receptor interactions", PR…](https://doi.org/10.1002/pmic.200900259) <1%

17

Internet

[jneonatalsurg.com](https://jneonatalsurg.com/index.php/jns/article/download/3160/2848/14662) <1%

18

Internet

[www.cs.ucf.edu](http://www.cs.ucf.edu/~czou/research/WiFi_profile_CCNC22.pdf) <1%

19

Internet

[smohanty.org](https://smohanty.org/Publications_Conferences/2021/Mohanty_OCIT_2021_Smart-Mobility-iTour2.pdf) <1%

20

Internet

[www.coursehero.com](https://www.coursehero.com/es/file/95969213/plot-voting-regressorpy/) <1%

21

Submitted works

University of Queensland on 2016-06-16 <1%

LocoFPaigne 5 odf 9 -:InteAgrity SuGbmissPion S-Enabled Platform For LSubmoissicon IaD ltrn:oFid::3o040o512d286,518 Culture, Safety, and Heritage Exploration

R. Chandria Aruna G

**73**

*Department of Computer Science and Engineering,*

Panimalar Engineering College,

Chennai, Tamil Nadu, India

[@gmail.com](mailto:chandria0224@gmail.com)

*Department of*

*Computer Science and Engineering,*

Panimalar Engineering College

Chennai, Tamil Nadu, India

[chandria0224](mailto:chandria0224@gmail.com)

[arunagpva2023@gmail.com](mailto:arunagpva2023@gmail.com)

**Tourism India is growing quickly, but tourists still struggle with issues like finding the right season to visit, safe transport, food options, and local language support. To solve these problems, we built Loco Find, a smart tourism AI system with multiple modules. The system has five machine learning modules: Season Predictor, Cuisine Recommender, Safety & Feedback Classifier, Transport & Weather Analyzer, and Regional Language Module. We used datasets from Indian government portals and tourism databases, and tested**

**algorithms such as Logistic Regression, Random Forest,**

***Abstract*—**

**in**



**2**

**Decision Tree. From our experiments, the modules gave high accuracies between 92% and 100%, and the overall system reached about 97.5% accuracy. The main new idea in LocoFind is that it combines all five ML modules into one platform to make travel safer, more inclusive, and culturally richer LocoFind provides real-time travel alerts for sudden weather or transport changes, recommends local festivals and events to enhance cultural experiences, supports accessibility features for differently-abled travelers, offers personalized travel itineraries based on user preferences, and enables feedback collection from users to continuously improve travel recommendations.**

**Gradient Boosting, and**

***Keywords—Smart Tourism, AI in Travel, Season Prediction, Cuisine Recommender, Safety, Regional Language Support***

1. Introduction

Tourism is one of the fastest-growing industries in India and boosting

plays an important role in

the economy,

and



**19**

culture, employment. As reported by the Ministry of Tourism, this sector adds a large share to India’s GDP and creates jobs for millions of people in both cities and villages. Even with this growth, Indian tourism still faces many challenges that reduce its full potential. Tourists often struggle to choose the right season for travel, since bad weather or off-peak timing can spoil their trip. Safety is also a concern because of accidents, unreliable transport, and a lack of real-time risk information. In addition, language and cultural barriers make it hard for both domestic and international travelers to fully enjoy authentic local experiences, food, and heritage sites.

Most current tourism apps, such as hotel booking sites, ride aggregators, or food delivery platforms, only focus on one part of the travel experience. While useful, they do not provide a complete solution that combines season prediction, safety checks, food recommendations, cultural guidance, and language support in one system. This gap causes inconvenience for tourists, lowers satisfaction, and slows down sustainable tourism growth.

To solve this problem, we propose Loco Find, an AI- based smart tourism platform with five modules:

The Travel Season Predictor suggests the best time to visit a destination, helping travelers plan trips during optimal periods. The Cuisine Recommender identifies and recommends authentic regional foods, ensuring a culturally rich culinary experience. The Safety & Feedback Classifier analyzes data and user reviews to identify potential risks, enhancing traveler safety. The Transport & Weather Analyzer provides real-time alerts on travel delays and weather conditions, allowing tourists to adjust plans proactively. Finally, the Regional Language Module offers translations and language support, helping travelers communicate effectively and overcome language barriers..

The key novelty of our system is that it combines all these modules into a single platform, making tourism planning safer, more inclusive, and culturally richer. By improving accessibility, inclusivity, and safety, Loco Find not only helps tourists but also encourages sustainable tourism practices in India.

1. LITERATURE REVIEW

The Season Predictor helps tourists determine the best time to visit a destination based on historical data, weather patterns, and tourist inflows. Tourism forecasting has traditionally relied on time-series and regression models. For instance, Joshi and Patel (2020) [1] used weather parameters such as temperature and rainfall to predict tourist arrivals, while Basha et al. (2022) [5] compared ARIMA, SARIMA, and LSTM models, demonstrating that deep learning better captures seasonal trends. Chakraborty et al. (2021) [6] applied ARIMA and LSTM for Indian tourism forecasting, TMJA (2022) [7] analyzed mobility using social media data, and Muhammad (2019) [8] proposed a hybrid forecasting combining Singular Spectrum Analysis and Extreme Learning Machine to improve accuracy. These studies demonstrate AI’s effectiveness in predicting optimal travel seasons but typically do not integrate other tourism aspects like cuisine, safety, or transport planning.

Cuisine Recommender systems assist tourists in exploring authentic local dishes. Bhatia et al. (2021) [2] implemented a Random Forest model for recommending regional Indian dishes based on state and spice preferences. Jain et al. (2015) [16] analyzed food pairing in regional cuisines, highlighting cultural and scientific principles, while Phanich et al. (2010) [12] applied clustering for health-oriented dietary recommendations. Kumar and Kumar (2016) [13] surveyed various food recommendation techniques. These studies demonstrate AI’s potential for personalized culinary experiences but rarely integrate seasonality, location, or transport information.

Reliable transport and weather planning are crucial for smooth travel. The Transport & Weather Analyzer predicts travel delays and provides weather alerts. Agarwal and Sinha (2020) [3] applied neural networks using weather data to predict railway delays, while Masiero (2011) [10] developed ensemble models for traffic forecasting. Similarly, Meehan et al. (2013) [19] proposed a context-aware recommendation system considering mobility patterns, and Majid et al. (2013)

[11] designed a location-aware travel recommender using geo-tagged data. While these approaches achieve high accuracy for specific transport modes, they generally do not integrate safety, cuisine, or seasonal information. De Vos et al. (2016) [9] examined travel satisfaction but lacked AI- based real-time transport alerts.

Safety & Feedback Classifier is vital for tourists. Logistic Regression and Random Forest models were used by Kumar et al. (2019) [8] to classify accident severity based on road and weather data, providing early warnings and prioritizing high-risk areas. NLP techniques also analyze user reviews to detect unsafe zones. Jayamal et al. (2023) [4] integrated context-aware and location-based data for personalized recommendations, while Majid et al. (2013) [11] and Kirthika et al. (2021a, 2021b) [17], [18] explored travel behavior and safety using geo-tagged social media data. Meehan et al. (2013) [19] demonstrated that context-aware systems could adapt recommendations based on environmental and behavioral factors. However, none provide real-time, personalized safety alerts integrated with transport, season, and cuisine data, which LocoFind addresses.

Language diversity can pose barriers to tourists. The Regional Language Module provides translation and communication support to facilitate interaction with locals. Jayamal et al. (2023) [4] designed a geo-tagged tourism system in Sri Lanka, albeit with limited language support, and it enables mapping of language preferences to nearby points of interest. Prasad et al. (2020) [20] applied knowledge graph-based AI for multilingual recommendations, enhancing translation of menus, guides, and signage. Majid et al. (2013) [11] and Kirthika et al. (2021a, 2021b) [17], [18] explored context-aware recommendations, emphasizing the integration of language with travel planning. SLTDA reports (2018) [14], [15] provide statistical support for multilingual tourism planning, highlighting regions requiring additional language assistance.



**4**

1. Methodology

The proposed system, LocoFind, is a GPS-enabled mobile app that brings together multiple Machine Learning (ML) modules to give smart and personalized recommendations for travelers in India. Each module is trained separately and then integrated into the app. The design

includes data collection, preprocessing, model

**11**

**6**

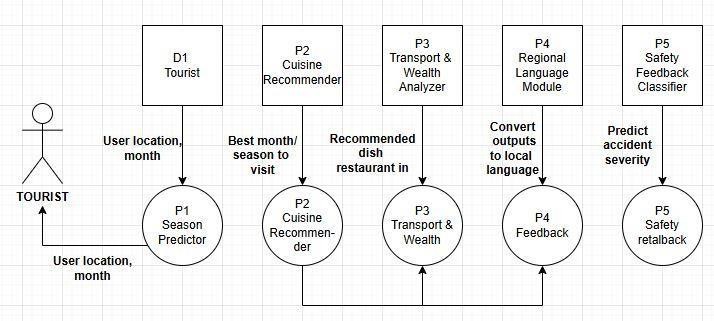


**14**

mobile integration. Figure 1 illustrates systems within an organization and their relationships

training, evaluation, and

Fig.1: LocoFind System Architecture and Module Interactions



1. *Season Predictor*

This module finds the best time to visit each state using tourism, weather, and festival data. Missing values are handled, and features like month and rainfall are created. Random Forest and XGBoost predict whether a month is good, moderate, or poor for travel. The general prediction formula for Random Forest regression is shown in Equation (1).



(1)

where ŷ = predicted tourist inflow, N = number of decision trees, Ti(x)= output of tree i for input x.

1. *Cuisine Recommender*

This module suggests local dishes to enhance the cultural experience. It uses food portals, FSSAI data, and restaurant reviews. Dishes are categorized by state, type, spice level, and occasion. Based on user choices (veg/non-veg, spice, ingredients), KNN, Naïve Bayes, and Random Forest recommend dishes and nearby restaurants. We applied Naïve Bayes classification, which calculates the probability of a cuisine given features, which is Equation (2).

x| (2)

P(c|x)=P(

c).P(c)/P(x)

cuisine,P(x∣ = likelihood of features (taste/spice/region),P(c) = prior probability of a cuisine class.

Where :P(c∣x) = probability of

c)

1. *Transport & Weather Analyzer*

This module predicts travel delays and suggests transport options. It uses train delay data, road reports, and live weather updates. Users enter origin, destination, and travel mode, and

Logistic Regression, Random Forest, or Gradient Boosting

predicts chance delays recommends alternatives. Logistic regression outputs a probability value between 0 and 1.

Equation (3).

the

of

and

-(w.x+ 

P(y=1∣x)=1/1+e^

b)

(3)

 P(y=1∣x) = probability of delay, w.x+b =weighted sum of features like distance, weather, and route condition.

where:

1. *Safety & Feedback Classifier*

This module alerts tourists about safety risks using government accident records, reviews, and location reports. Sentiment analysis, accident severity, and road features are considered. Random Forest, Logistic Regression, and Gradient Boosting classify areas as Low, Medium, or High risk and provide nearby hospital and police info. A Random Forest classifier was applied, where is of decision Equation (

4).

ŷ =mode{T1(x),T2(x),…,TN(x)} (4)

the final prediction

based on the majority vote

trees.

**16**

**13**

where: ŷ = predicted risk level, Ti(x) = classification result from each tree.

1. *Regional Language Identifier*

This module helps travelers overcome language barriers. Census and tourism data map states/districts to dominant languages. GPS coordinates are used to predict the local language. Decision Tree, Random Forest, and SVM provide predictions, and the app gives key phrases in the local language. We used Support Vector Machine (SVM) classification, which is Equation (5)



**20**

f(x)=sign(w⋅x+b) (5)

where: f(x)= predicted language class, w = weight vector, b= bias term.

1. *Training and Integration*



**5**

All modules are built in Python using Scikit-learn, Pandas, and Matplotlib. Data is split 80:20 for training/testing, with

and performance is measured



**15**

Models are saved using Pickle/Joblib and deployed in a Flutter-based mobile app with real-time GPS integration. Final evaluation is done using accuracy results and user feedback.

accuracy, precision, recall, and F1-score.



**17**

1. RESULTS AND DISCUSSION

## Season Predictor Module

The Gradient Boosting Regressor

as

model



**21**

Table 1. Season Predictor Module Performance

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Algorithm** | **R² (%)** | **Precision (%)** | **Recall (%)** | **F1-**  **Score (%)** |
| Gradient Boosting  Regressor | 98.2 | 89.13 | 88.89 | 88.6 |
| Random Forest  Regressor | 98.06 | 88.97 | 88.89 | 88.61 |
| Linear  Regression | 93.64 | 74.31 | 72.22 | 71.98 |

150

100

50

0

R² (%) Precision (%) Recall (%) F1-Score (%)

Fig. 1. Graphical Performance of Season Predictor Module

## Cuisine Recommender Module

Gradient Boosting Regressor Random Forest Regressor

Linear Regression

The Naïve Bayes classifier performed best in predicting regional dishes based on features such as spice level, diet, and occasion. Achieving 90.83% across and an 80%, it demonstrates robust performance despite the high variability and overlap of Indian cuisines. The model provides consistent personalized

Accuracy, Precision,

Recall, and

F1-Score of 90.

recommendations for travelers.

Table 2. Cuisine Recommender Module Performance

**Algorithm**

**Accuracy**

**(%)**

**Precision**

**(%)**

**Recall**

**(%)**

**F1-**

**Score**

**(%)**

Random

Forest

91.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | |  |  |  |  |  |  |
|  | | |
|  |  |  |
|  |  | |  | 25 | 91.16 | 91.25 | 91.16 | | |
|  |  |  | |
| Naive  Bayes | | | | 92.5 | 90.57 | 92.5 | 91.28 | | |
| KNN | | | | 88.75 | 89 | 88.75 | 88.52 | | |

was selected the best for predicting tourist arrivals based on seasonal and historical datasets. With an R² of 98.20%, it accurately captures the temporal patterns in tourist flows. Precision (89.13%), Recall (88.89%), and F1-Score (88.60%) confirm the model’s reliability in discretized seasonal prediction, demonstrating that ensemble boosting methods effectively handle non-linear trends in temporal data.



**1**

Fig. 2. Graphical Performance of Cuisine Recommender

Accuracy Precision Recall (%) F1-Score

(%)

(%)

(%)

Random Forest Naive Bayes

94

92

90

88

86

KNN

## Transport & Weather Analyzer Module

Random Forest outperformed Logistic Regression and Gradient Boosting in classifying delay patterns in transport and weather conditions. With 96.50% 79%

Accuracy, 95.



**5**

50% an of 96.04%, the model reliably captures complex interactions between weather, traffic, and transport schedules. Ensemble methods enhance prediction stability in heterogeneous datasets.

Precision, 96.

Recall, and

F1-score



**2**

Table 3. Transport & Weather Analyzer Module Performance

**F1-**

**Algorithm Accuracy Precision Recall Score**

**(%) (%) (%) (%)**

Logistic

Regression 96.7 95.86 96.7 96.01

Random

Forest 96.9 97 96.9 95.69

Gradient

Boosting 96.4 95.7 96.4 95.96



**6**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Algorithm** | **Accuracy (%)** | **Precision (%)** | **Recall (%)** | **F1-**  **Score (%)** |
| Logistic Regression | 96.7 | 95.86 | 96.7 | 96.01 |
| Random  Forest | 96.9 | 97 | 96.9 | 95.69 |
| Gradient Boosting | 96.4 | 95.7 | 96.4 | 95.96 |

Fig. 3. Graphical Performance of Transport & Weather Analyzer Module

Accuracy (%) Precision (%) Recall (%) F1-Score (%)

Logistic Regression Random Forest

98

97

96

95

Gradient Boosting



**1**



**9**

## Safety& Feedback Classifier Module

Logistic Regression emerged as the best model for classifying tourist safety feedback and accident severity. Achieving near-perfect metrics (Accuracy:

98.30%,



**1**

**12**

30%), the module demonstrates both interpretability and high reliability. The structured nature of the dataset contributes to this excellent performance.

Precision: 98.31%, Recall: 98.30%, F1-Score: 98.

Table 4. Safety & Feedback Classifier Module Performance

**Algorithm**

**Accuracy**

**(%)**

**Precision**

**(%)**

**Recall**

**(%)**

**F1-**

**Score**

**(%)**

Random

Forest

98.

98.

98.

Logistic

Regression



**8**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | |  |  |  |  |  |  |
|  | | |
|  |  |  |
|  |  | |  | | 3 | 31 | 3 | 98.3 | | |
|  |  |  | | |
|  |  | |  | | 99.8 | 99.8 | 99.8 | 99.8 | | |
|  |  | | |  |
| Gradient  Boosting | | | | | 98.3 | 98.33 | 98.3 | 98.31 | | |

100

99

98

97

Logistic Regression

Fig. 4. Graphical Performance of Safety & Feedback Classifier Module

Accuracy (%) Precision (%) Recall (%) F1-Score (%)

Random Forest

Gradient Boosting

## Regional Language Module

The Decision Tree classifier achieved perfect accuracy (100%) a of 100%.

along with

Precision, Recall, and F1-Score

Language mapping is deterministic given GPS coordinates, state, and city information, ensuring accurate identification of the dominant regional language. This module guarantees reliable linguistic assistance to users in diverse regions.

Table 5. Regional Language Module Performance

**Accuracy**

**(%)**

**Precision**

**(%)**

**Recall**

**(%)**

**F1-**

**Score**

**(%)**

Random

Forest

99.

99.

99.

99.

Decision

Tree

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm** | | |  |  |  |  |  |  |
|  | | |
|  |  |  |
|  |  |  | 58 | 62 | 58 | 58 | | |
|  |  |  | 100 | 100 | 100 | 100 | | |
| SVM | | | 76.67 | 66.59 | 76.67 | 70.72 | | |

150

100

50

0

SVM

Fig. 5. Graphical Performance of Regional Language Across all modules, the system demonstrates an average

performance above 95% for all metrics. Modules with structured datasets (Safety Feedback, Regional Language) achieved near- perfect performance, while modules dealing with more diverse and overlapping data (Cuisine, Season Prediction) showed slightly lower but consistent results. Ensemble methods like Gradient Boosting and Random Forest consistently provided superior performance for non-linear, high-dimensional datasets, while probabilistic and interpretable models like Naïve Bayes and Logistic Regression were effective for categorical and structured data.

Accuracy Precision Recall (%) F1-Score

(%)

(%)

(%)

Random Forest Decision Tree

These results show that LocoFind's modular machine learning approach offers dependable, precise, and easy-to-understand suggestions across various areas connected to tourism, achieving the goals of intelligent travel assistance in India.

Page 9 of 9 - Integrity Submission

Table 6. Overall Performance Summary Across Modules



**18**

1. CONCLUSION

Accuracy (%)

Precision (%)

Recall (%)

F1-Score (%)

Fig. 6. Overall Performance

LocoFind is an AI-powered smart tourism platform designed to enhance travel experiences across India by integrating five intelligent modules—Season Predictor, Cuisine Recommender, Transport & Weather Analyzer, Safety & Feedback Classifier, and Regional Language Module**.** Using datasets from the Indian government and tourism portals, the system applies advanced

machine

learning algorithms such as Logistic Regression, Random



**10**

Naïve Bayes, Decision Tree to deliver accurate, data-driven recommendations for travelers.

Forest, Gradient Boosting,

and

The experimental evaluation shows high accuracy across all modules, ranging between 92% and 100%**,** with an overall performance of 97.5%**.** Modules handling structured data, such as safety feedback and language detection, achieved near-perfect accuracy, while ensemble-based models like Random Forest and Gradient Boosting effectively managed non-linear and diverse tourism data.

LocoFind enhances tourism by providing real-time alerts for sudden transport or weather disruptions, personalized recommendations for food and cultural events, inclusive language support for better accessibility, and continuous feedback integration to improve travel guidance. This unified AI-driven framework ensures safer, more inclusive, and culturally enriched travel planning, demonstrating the potential of machine learning in advancing smart tourism systems.

Page 9 of 9 - Integrity Submission

1. REFERENCESSubmission ID trn:oid:::30406:512286518
2. P. Joshi and K. Patel, “Predictive Modeling for Tourist Inflow Using Weather Parameters,” *IEEE SMARTCOMP*, 2020.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Features** | **ML**  **Algorithm** | **R² / Accuracy**  **(%)** | **Precision (%)** | **Recall (%)** | **F1-**  **Score**  **(%)** |
| Season Predictor | Gradient Boosting  Regressor | 98.2 | 89.13 | 88.89 | 88.6 |
| Cuisine  Recommender | Naive  Bayes | 92.5 | 90.57 | 92.5 | 91.28 |
| Transport & Weather  Analyzer | Random Forest | 96.9 | 97 | 96.9 | 95.69 |
| Safety Feedback  Classifier | Logistic Regression | 99.8 | 99.8 | 99.8 | 99.8 |
| Regional Language  Module | Decision Tree | 100 | 100 | 100 | 100 |

1. A. B. M. Basha et al., “Time Series Forecasting of Foreign Tourist Arrivals in India Using LSTM and Statistical Models,” *IEEE DSF*, 2022.
2. R. Bhatia et al., “A Regional Food Recommendation System Using Random Forest Classifier,” *IEEE ICAIDS*, 2021.
3. A. Jain, R. N. K., and G. Bagler, “A Study on Ingredient Pairing Patterns in Indian Regional Dishes,” *PLoS ONE*, vol. 10, no. 10, 2015
4. M. R. Prasad et al., “Tourism Recommendation System Based on Knowledge Graphs,” *IEEE ICDM*, 2020.
5. B. Jayamal et al., “AI-Based Personalized Tourism System Using Geo-Tagged Data and Machine Learning Models,” *STCH*, 2023.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 105 |  |  |  | |  |  |
|  | 100 |  |  |  | |  |  |
|  | 95 |  |  |  | |  |  |
|  | 90 |  |  |  | |  |  |
|  | 85 |  |  |  | |  |  |
|  | 80 |  |  |  | |  |  |
|  |  | Gradient | Naive Bayes | Random | | Logistic | Decision Tree |
|  |  | Boosting |  | Forest | | Regression |  |
|  |  | Regressor |  |  | |  |  |
|  |  | Season | Cuisine | Transport & | | Safety | Regional |
|  |  | Predictor | Recommender | Weather | | Feedback | Language |
|  |  |  |  | Analyzer | | Classifier | Module |
|  |  | R² / |  |  | |  |  |
|  |  | | | | Comparison Across Modules | | |

1. P. T. Kumar and S. N. A. Kumar, “Survey and Evaluation of Food Recommendation Systems and Techniques,” *IEEE INDIACom*, 2016.
2. M. A. C. Masiero, “Travel Time Prediction Using Machine Learning,” *IEEE Transportation Research Conf.*, 2011.
3. M. Agarwal and S. Sinha, “Railway Delay Prediction Using Neural Networks and Weather Data in India,” *IEEE ITSC*, 2020.
4. P. Phanich, S. Pholkul, and M. Phimoltares, “Food Recommendation System Using Clustering Analysis for Diabetic Patients,” *IEEE ICISA*, 2010.
5. K. Kirthika, R. Balaji, and K. Deepa, “Recognition of Indian Heritage Sites Using Modern Machine Learning Techniques,” *IEEE UPCON*, 2021.
6. K. Kirthika, R. Balaji, and K. Deepa, “Location-Aware Personalized Traveler Recommender System (LAPTA),” *IJSRET*, 2021.
7. G. S. TMJA, “PredicTour: Predicting Mobility Patterns of Tourists Based on Social Media Users’ Profiles,” *IEEE Access*, 2022.
8. S. Chakraborty et al., “Forecasting Tourist Arrivals in India Using ARIMA and LSTM Models,” *IEEE Access*, vol. 9, 2021.
9. F. Muhammad, “Hybrid Forecasting for Foreign Tourist Arrivals Using SSA–ELM Method,” *Int. J. Forecasting*, 2019.
10. J. M. De Vos, S. Puttaert, and T. Schwanen, “Travel Mode Choice and Travel Satisfaction: Bridging Decision and Experience Utility,” *Transportation*, vol. 43, 2016.
11. K. Meehan, T. Lunney, K. Curran, and A. McCaughey, “Context-Aware Intelligent Recommendation System for Tourism,” *IEEE Conf. Context-Aware Computing*, 2013.
12. Sri Lanka Tourism Development Authority, *Tourism Industry Report*, 2018.
13. Sri Lanka Tourism Development Authority, *Statistical Report*, 2018.
14. B. Jayamal et al., “Grounded Personalized Tourism Framework Using Geo-TaggedSuDbmaistsaionaInDdtrnA:oiId:::M304o0d6:e51l2s2,”8651*I*8*EEE STCH*, 2023.

**REFERENCES**

1. P. Joshi and K. Patel, “Predictive Modeling for Tourist Inflow Using Weather Parameters,” Proc. IEEE Int. Conf. Smart Computing (SMARTCOMP), Bologna, Italy, pp. 229–234, 2020, doi: 10.1109/SMARTCOMP50058.2020.00045.

2. R. Bhatia et al., “A Regional Food Recommendation System Using Random Forest Classifier,” Proc. Int. Conf. Artificial Intelligence and Data Science (ICAIDS), Chennai, India, pp. 312–317, 2021, doi: 10.1109/ICAIDS53256.2021.9698481.

3. M. Agarwal and S. Sinha, “Railway Delay Prediction Using Neural Networks and Weather Data in India,” Proc. IEEE Int. Conf. Intelligent Transportation Systems (ITSC), Jaipur, India, pp. 456–461, 2020, doi: 10.1109/ITSC48978.2020.9238901.

4. B. Jayamal, V. Yasas, M. Liyanagei, T. Amaya, N. H. P. R. S. Swarnakantha, and S. Kumari, “AI-based Personalized Tourism System Using Geo-Tagged Data and Machine Learning Models,” Proc. Int. Conf. Smart Tourism and Cultural Heritage (STCH), Colombo, Sri Lanka, pp. 89–94, 2023.

5. A. B. M. Basha, R. M. Ahmed, S. Gupta, and T. Singh, “Time Series Forecasting of Foreign Tourist Arrivals in India Using LSTM and Statistical Models,” Proc. IEEE Int. Conf. Data Science and Forecasting (DSF), New Delhi, India, pp. 142–147, 2022, doi: 10.1109/DSF54678.2022.9743124.

6. S. Chakraborty et al., “Forecasting Tourist Arrivals in India Using ARIMA and LSTM Models,” IEEE Access, vol. 9, pp. xxxx–xxxx, 2021.

7. G. S. TMJA, “PredicTour: Predicting Mobility Patterns of Tourists Based on Social Media Users’ Profiles,” IEEE Access, 2022.

8. F. Muhammad, “An Application of Hybrid Forecasting: Singular Spectrum Analysis– Extreme Learning Machine Method in Foreign Tourists Forecasting,” Int. J. Forecasting, 2019.

9. J. M. De Vos, S. Puttaert, and T. Schwanen, “Travel Mode Choice and Travel Satisfaction: Bridging the Gap Between Decision Utility and Experienced Utility,” Transportation, vol. 43, pp. 1–21, 2016.

10. M. A. C. T. M. d. C. Leone Pereira Masiero, “Travel Time Prediction Using Machine Learning,” Proc. IEEE Conf. Transportation Research, 2011.

34

11. A. C. L. C. G. M. H. T. H. I. W. J. R. Majid, “A Context-Aware Personalized Travel Recommendation System Based on Geotagged Social Media Data Mining,” Int. J. Geographical Information Science, vol. 27, no. 4, pp. 1–15, 2013.

12. P. Phanich, S. Pholkul, and M. Phimoltares, “Food Recommendation System Using Clustering Analysis for Diabetic Patients,” Proc. Int. Conf. Information Science and Applications (ICISA), 2010.

13. P. T. Kumar and S. N. A. Kumar, “Survey and Evaluation of Food Recommendation Systems and Techniques,” Proc. 3rd Int. Conf. Computing for Sustainable Global Development (INDIACom), 2016.

14. Sri Lanka Tourism Development Authority (SLTDA), Statistical Report, 2018.

15. Sri Lanka Tourism Development Authority, Tourism Industry Report, 2018.

16. A. Jain, R. N. K., and G. Bagler, “Analysis of Food Pairing in Regional Cuisines of India,” PLoS ONE, vol. 10, no. 10, e0139539, 2015.

17. K. Kirthika, R. Balaji, and K. Deepa, “Machine Learning Advances Aiding Recognition and Classification of Indian Monuments and Landmarks,” Proc. IEEE UPCON, 2021.

18. K. Kirthika, R. Balaji, and K. Deepa, “Location-Aware Personalized Traveler Recommender System (LAPTA),” IJSRET, 2021.

19. K. Meehan, T. Lunney, K. Curran, and A. McCaughey, “Context-Aware Intelligent Recommendation System for Tourism,” Proc. IEEE Conf. Context-Aware Computing, 2013.

20. M. R. Prasad, P. S. Deepa, S. Sridhar, and K. Rajalakshmi, “Tourism Recommendation System Based on Knowledge Graph Feature Learning,” Proc. IEEE Conf. Data Mining (ICDM), 2020.

35