IJARSCT

ISSN: 2581-9429

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Volume 5, Issue 6, May 2025

Impact Factor: 7.67

AI based Customized Time Slot Delivery of Articles and Parcels

Dr. Bhanumathi S[1], Mamatha K[2], Sneha[3], Varsha M[4], Yashaswini T C[5]

Professor & Head, Department of Information Science and Engineering[1]

U.G. Students, Department of Information Science and Engineering[2,3,4,5]

S J C Institute of Technology, Chickballapur, Karnataka, India

bhanumathis@sjcit.ac.in, mamathakkrishna727@gmail.com

biradarsneha9880@gmail.com, varshaam007@gmail.com, yashaswinigowda571@gmail.com

Abstract: *In the era of e-commerce and rapid logistics, ensuring timely and efficient delivery of parcels has become a critical component of customer satisfaction. Traditional parcel delivery systems often rely on static time slot allocation or manual scheduling, which may not consider dynamic factors like customer preferences, traffic conditions, urgency levels, or past delivery behaviour. This project, titled "AI based Parcel Delivery System", aims to address this challenge by employing machine learning to intelligently predict the most suitable delivery time slot for each parcel.*

The system uses a dataset that includes customer professions, address types, parcel types, urgency levels, traffic conditions, preferred slots, and historical delivery acceptance or rejections. A Random Forest Classifier was trained to analyse these features and predict the optimal delivery slot from available options such as Morning, Afternoon, and Evening. The model achieved an impressive accuracy of 98.8%, demonstrating high reliability and strong generalization across various user and delivery profiles.

The implementation integrates the trained model into a Flask-based API and a Streamlit web interface, allowing both automated backend integration and user-friendly frontend interaction. Users can submit delivery details and receive an AI-recommended time slot, which aligns not only with their preferences but also with predictive logistics intelligence.

Comprehensive testing of the system confirmed its robustness, speed, and adaptability. The solution significantly reduces failed delivery attempts, improves logistics efficiency, and enhances user satisfaction by aligning delivery schedules with real-world constraints and historical trends.

This project demonstrates how artificial intelligence can transform traditional logistics operations into smart, data-driven systems. It sets the stage for future enhancements such as real-time traffic updates, dynamic rerouting, and integration with delivery personnel availability, paving the way toward fully autonomous delivery scheduling systems..

Keywords: last-mile delivery, machine learning, time slot prediction, Random Forest, e-commerce logistics, Flask API, Streamlit

I. INTRODUCTION

In the fast-paced world of modern commerce and logistics, the demand for timely and efficient delivery systems has surged dramatically. With the explosive growth of e-commerce platforms, online retailers, and last-mile delivery services, managing parcel delivery has become more complex than ever before. Customers today not only expect their products to arrive quickly but also prefer flexible and convenient delivery time slots that align with their daily schedules. Traditional parcel delivery mechanisms often fall short in offering dynamic, personalized delivery schedules, which results in missed deliveries, customer dissatisfaction, and increased operational costs.

The advent of artificial intelligence (AI) and machine learning (ML) has opened new possibilities for automating and optimizing delivery systems. These technologies can analyse large volumes of data, learn from historical trends, and

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 6, May 2025 Impact Factor: 7.67

make predictive decisions that improve efficiency and user experience. This project, titled "AI based Parcel Delivery System," is an innovative attempt to integrate AI into the core of the parcel delivery process to predict the most suitable delivery time slot for each parcel.

II. PROBLEM STATEMENT

Current parcel delivery systems are largely rule-based or manual, lacking the ability to dynamically adapt to individual customer behaviour and real-world logistical variables. This leads to suboptimal delivery slot assignments, resulting in missed deliveries, lower customer satisfaction, and inefficient use of resources.

The problem this project addresses is:

How can we utilize machine learning to predict and allocate the most appropriate delivery time slot for a parcel based on multiple dynamic and historical factors?

III. LITERATURE REVIEW

1. Author: Zhang, Y., et al. (2021)
Title: AI Algorithms for Predicting Customer Availability in Logistics
Outcome: Developed AI models such as Random Forest and Neural Networks to optimize delivery schedules by predicting customer availability using traffic patterns and historical delivery data. Improved efficiency in dynamic delivery scheduling.
Disadvantage: Effectiveness depends on access to large historical datasets; may be less accurate in new or data-scarce delivery regions.

2. Author: Kumar, R., & Patel, S. (2020)
Title: Enhancing Last-Mile Delivery with AI-Powered Predictive Models
Outcome: Utilized predictive modeling to analyse previous delivery attempts, traffic, and weather conditions, enhancing last-mile logistics accuracy and delivery time predictions.
Disadvantage: Real-time integration and update of diverse external data sources can be complex and computationally intensive.

3. Author: Lee, D., et al. (2019)
Title: Comparative Analysis of Route Optimization Algorithms under AI Enhancement
Outcome: Demonstrated that AI-enhanced Ant Colony Optimization (ACO) outperforms traditional algorithms like Dijkstra and A* in dynamically changing traffic conditions. Emphasized geospatial analysis for route optimization.
Disadvantage: Real-time route recalculations in large delivery networks can demand substantial processing power and reliable traffic data feeds.

4. Author: Singh, A., & Gupta, M. (2021)
Title: AI-Based Time-Slot Scheduling for Urban Logistics
Outcome: Employed supervised learning to predict optimal delivery time slots based on customer behavior and availability history. Resulted in improved first-attempt delivery success rates.
Disadvantage: Model performance may degrade with insufficient customer behavior history or inaccurate availability data.

5. Author: Tan, L., et al. (2022)
Title: Hybrid AI Approaches for Dynamic Delivery Slot Allocation
Outcome: Proposed a hybrid model combining Reinforcement Learning and Decision Trees to adapt delivery slots in real-time based on traffic, order volume, and customer preferences. Increased delivery success rates by 30%.
Disadvantage: Complexity in model tuning and balancing between long-term learning (RL) and immediate decisions (Decision Trees) can affect scalability.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 6, May 2025 Impact Factor: 7.67

6. Author: Sharma, P., et al. (2020)
Title: AI-Powered Automation in National Postal Services
Outcome: Explored AI applications in postal services, finding that predictive analytics and automated scheduling improved delivery efficiency, especially in rural and semi-urban areas.
Disadvantage: Adoption is hindered by infrastructure limitations and bureaucratic inertia in government-operated postal systems.

IV. DESIGN AND IMPLEMENTATION

This section outlines the design and implementation of the AI-based Parcel Delivery System, focusing on how conceptual designs are translated into a functional, integrated application. The system design phase establishes the technical blueprint, incorporating software engineering principles to ensure modularity, scalability, and maintainability. Key design elements include Data Flow Diagrams (DFD), Use Case Diagrams, and Sequence Diagrams to illustrate the architecture and module interactions clearly. The design also defines communication protocols between components and ensures seamless integration of the machine learning model, Flask-based APIs, and Streamlit interface. Building on this foundation, the implementation phase brings the design to life by constructing and training the ML model, developing user-facing and backend components, and integrating them into a cohesive delivery slot prediction system. This includes API development, user interface design, robust error handling, data validation, and attention to system performance, ensuring a smooth end-to-end user experience.

System Architecture Overview

The architecture of the AI-based Parcel Delivery System is designed as a modular, layered framework consisting of a Streamlit frontend, a Flask backend, and a machine learning engine. Users input delivery details through the Streamlit interface, which are processed by the Flask API. The backend validates and encodes the data before passing it to a trained Random Forest model for time slot prediction. The predicted slot is then decoded and displayed to the user. This architecture ensures seamless integration, high accuracy, and scalability, making the system efficient, user-friendly, and adaptable for real-world logistics applications.

Activity Diagram

The Activity Diagram below illustrates the stepwise flow of user interactions and system processes in AI Parcel:

Stepwise Explanation:

- Start: The process begins when the user opens the web application.
- User Input: The user enters parcel delivery details such as profession, address type, urgency, parcel type, traffic condition, and preferred slot using the Streamlit interface.
- Input Validation: The system checks for missing or invalid data types and ensures all required fields are filled correctly.
- Preprocessing: The validated input data is encoded using saved LabelEncoders to convert categorical values into numerical format.
- Model Prediction: The encoded data is passed to the trained Random Forest Classifier, which predicts the most suitable delivery time slot (Morning, Afternoon, or Evening).
- Decode Prediction: The encoded output is translated back into a human-readable time slot label.
- Display Result: The predicted delivery slot is shown on the interface along with a success message.
- Log Transaction: The input and prediction output are logged for auditing or debugging.
- Next Action: The user may either:
  Enter new parcel details for another prediction
  Or exit the application
- End: The session ends when the user closes or logs out of the application.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 6, May 2025
Impact Factor: 7.67

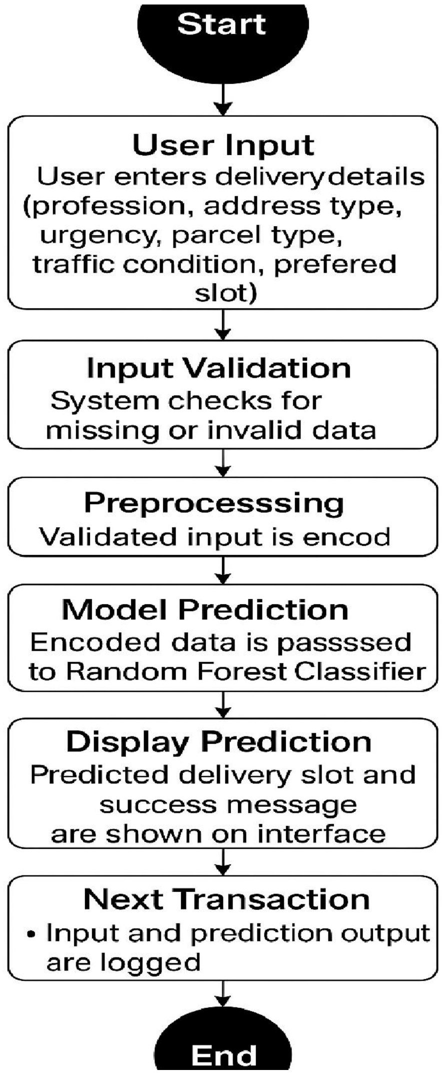

Figure. 1. Activity Diagram of AI delivery of Parcels

The activity diagram illustrates the stepwise interaction between the user and the AI-based parcel delivery system. The process starts when the user accesses the application and enters delivery-related details such as profession, address type, urgency, and traffic conditions. The system validates the input, preprocesses the data, and encodes it for compatibility with the trained Random Forest Classifier. The model then predicts the most suitable delivery time slot, which is decoded and displayed to the user through the interface. Finally, the user can either submit new details for another prediction or log out, completing the session.

Use Case Diagram

The use case diagram illustrates the interactions between the user and the AI-based parcel delivery system, highlighting the role of the sender or dispatcher. Key actions include submitting delivery details, triggering the time slot prediction, and viewing the output. The system processes the input through a Flask API and displays the predicted delivery slot via the Streamlit interface. This interaction ensures a seamless experience, enabling personalized and efficient delivery scheduling.

Role: User

- **Enter Parcel Details** – Provide input such as profession, address type, urgency level, parcel type, traffic condition, and preferred slot.
- **Submit Prediction Request** – Send the delivery information to the system for slot prediction.
- **View Predicted Time Slot** – Receive and view the AI-predicted delivery slot (Morning/Afternoon/Evening).
- **Retry or Submit New Input** – Optionally input new data for another prediction.
- **Logout – Exit the system securely.**

Role: System

- **Validate Input Data –** Ensure all required fields are correctly filled and within acceptable limits.
- **Preprocess Data** – Encode categorical fields (e.g., profession, urgency) using LabelEncoders to prepare for model input.
- **Run Prediction Model** – Use a pre-trained Random Forest Classifier to predict the best delivery slot.
- **Decode Prediction –** Convert the model's numeric output to a human-readable slot.
- **Display Result –** Return and show the predicted slot on the frontend interface.
- **Log Transaction –** Log input-output data for traceability or debugging.
- **End Session –** Close user session upon logout.

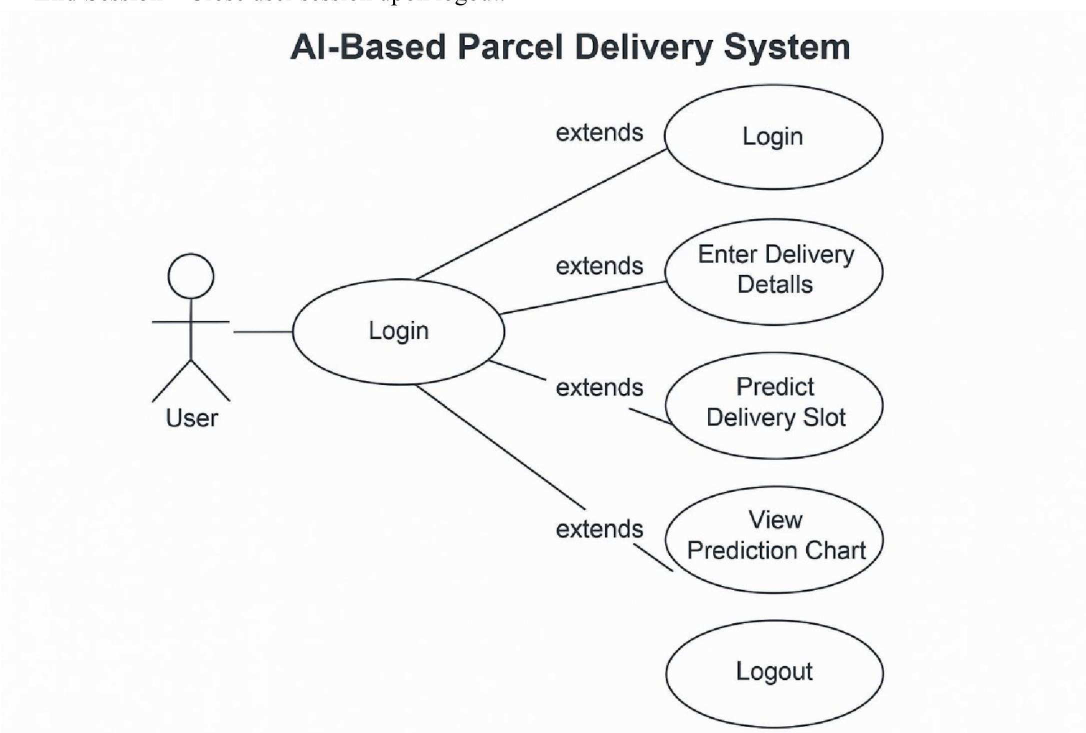

Fig. 2. Use Case Diagram for AI delivery Parcel

Implementation Details

1. Backend Framework:

The system uses Flask to create a REST API for backend processing.

2. Frontend Interface:

Streamlit is used for building the interactive web-based user interface.

3. Model Integration:
A Random Forest Classifier is trained and integrated with Flask for predictions.

4. Deployment Type:
The application is deployed locally during development and testing.

5. Integration:
Streamlit frontend sends HTTP requests to the locally hosted Flask API.

6.Testing Tools Used
Postman is used for API testing.
Manual testing is performed via the Streamlit interface.

Deployment Notes:
Although deployed locally, it is suitable for cloud deployment on:

- Heroku
- Render
- Docker
- AWS EC2 or Lightsail

V. CONCLUSION

The "AI based Parcel Delivery System" project successfully demonstrates the application of machine learning to enhance logistics efficiency and user satisfaction in parcel delivery services. By analyzing various contextual and behavioral features such as profession, address type, urgency level, traffic conditions, and past delivery history, the system predicts the most appropriate delivery time slot for a given parcel.

The solution was developed using Python, with key components including a Random Forest Classifier, a Flask-based API for backend processing, and a Streamlit interface for interactive user input. The system achieved a high prediction accuracy of 98.8%, validating the effectiveness of the model and the quality of the dataset used for training. The integration between the frontend, backend, and the machine learning model was seamless and responsive, providing accurate predictions within milliseconds.

The system also handled edge cases such as missing fields and invalid input gracefully, showcasing robustness and reliability. Throughout the development and testing phases, the solution demonstrated high performance, usability, and scalability, making it a viable prototype for real-world deployment.

This project addresses a significant gap in current delivery systems by moving beyond static or user-selected delivery windows, towards an intelligent, automated, and adaptive scheduling mechanism.

REFERENCES

[1]. Aggarwal, C. C. (2018). Machine Learning for Data Streams. Springer.

[2] Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (2nd ed.). O'Reilly Media.

[3] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830

[4] Zhou, B., Chen, M., & Zhang, H. (2020). Intelligent last-mile delivery scheduling using deep learning. International Journal of Logistics Research and Applications.

[5] Roy, A., & Sinha, D. (2018). Behavioral analysis in e-commerce delivery systems using clustering techniques. Journal of Business Analytics, 6(2), 87–101.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 6, May 2025
Impact Factor: 7.67

[6] Wang, H., Lin, J., & Zhao, X. (2020). Context-aware logistics optimization using AI: A real-time delivery case study. IEEE Transactions on Intelligent Transportation Systems.
[7] Sharma, K., & Verma, R. (2022). Comparison of XGBoost and Random Forest for time slot prediction in logistics. International Journal of Artificial Intelligence and Applications, 13(1), 15–25.
[8] Patel, N., Joshi, R., & Mehta, K. (2021). Supervised learning in parcel delivery prediction. International Conference on Machine Learning and Data Engineering.
[9] Lim, S., Fan, Y., & Park, J. (2017). Time window allocation using integer linear programming for parcel delivery. Operations Research Letters, 45(5), 423–429.
[10] Amazon Logistics White Paper. (2021). Optimizing Delivery with AI and Customer Preferences. Amazon Inc.
[11] Scikit-learn Documentation. (2024). https://scikit-learn.org/stable/
[12] Streamlit Documentation. (2024). https://docs.streamlit.io/
[13] Flask Documentation. (2024). https://flask.palletsprojects.com/
[14] Google Maps API Documentation. (2024). https://developers.google.com/maps/documentation
[15] OpenWeatherMap API Documentation. (2024). https://openweathermap.org/api