



International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 2(1) (2026)



ijerst.editor@gmail.com
editor@ijerst.com

Research Paper

AI DELIVERY TIME PREDICTION SYSTEM

¹K Ravi Naik, ²M ViswaSimha, ³B Naveen, ⁴A Ashwith kumar, ⁵P Pranay

¹AssistantProfessor, ²³⁴⁵Students

Department of CSE(AI & ML)

Siddhartha Institute of Technology & Sciences, Narapally

ravinaik_cse@siddhartha.co.in, 23TQ1A6602@siddhartha.co.in, 23TQ1A6614@siddhartha.co.in,
23TQ1A6661@siddhartha.co.in, 23TQ1A6662@siddhartha.co.in.

Abstract

The AI Delivery Time Prediction System is designed to estimate the delivery time of food orders using machine learning techniques, aiming to improve efficiency and customer satisfaction in logistics and food delivery services. The system utilizes a dataset of 45,593 food delivery records with 20 features, including delivery personnel details, vehicle type and condition, weather conditions, traffic density, distance between locations, festival occurrences, city type, and order type. Additionally, a new feature called Haversine distance is engineered using GPS coordinates to calculate the real-world distance between the restaurant and the delivery location, enhancing prediction accuracy.

Multiple machine learning regression models, including Linear Regression, Decision Tree Regressor, Random Forest Regressor, and Gradient Boosting Regressor, are implemented and evaluated. The models are assessed using standard performance metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R^2 Score, and Mean Absolute Percentage Error (MAPE). Among these, the Gradient Boosting Regressor achieves the best performance with a high R^2 score of 0.89, along with low error values, indicating strong prediction accuracy. The reliability of the model is further validated using 5-fold cross-validation.

The system is deployed as an interactive desktop application developed using Python's Tkinter library, featuring a user-friendly graphical interface. Users can input order details and receive instant delivery time predictions without system delays. Additionally, the application includes a PDF report generation feature for documenting prediction results.

I. Introduction

In today's fast-paced world, the food delivery industry has experienced tremendous growth due to the increasing demand for convenience and the rapid expansion of online food ordering platforms. Companies such as Swiggy, Zomato, and Uber Eats handle millions of orders daily, where accurate delivery time estimation plays a crucial role in customer satisfaction. When a customer places an order, one of the most important expectations is knowing how long it will take for the food to arrive. Inaccurate delivery time predictions can lead to frustration, negative reviews, and a loss of customer trust.

Delivery time is influenced by multiple dynamic factors, making it a complex problem to solve. These factors include traffic conditions, weather, distance between the restaurant and customer, delivery person experience, vehicle type and condition, festival or peak seasons, and order load. Traditional systems often rely on simple rule-based methods such as distance and average speed, which are not sufficient to handle the complexity of real-world scenarios. As a result, these systems often produce inaccurate and unreliable predictions.

Artificial Intelligence and Machine Learning provide an effective solution to this challenge by enabling data-driven prediction models. By analyzing large volumes of historical delivery data, machine learning algorithms can identify hidden patterns and relationships between different influencing factors and the actual delivery time. These models can then be used to predict delivery times more accurately for new orders in real time.

II. Literature Survey

The literature survey highlights the growing importance of machine learning in delivery time prediction and logistics optimization. Several researchers have demonstrated that data-driven approaches significantly outperform traditional rule-based systems in handling complex and dynamic factors such as traffic conditions, weather, and delivery routes. Studies in logistics and transportation show that ensemble models like Random Forest and Gradient Boosting provide higher accuracy compared to simple regression models. In the context of food delivery, research indicates that additional factors such as order density, preparation time, and customer location further increase prediction complexity. Deep learning models like LSTM have been used to capture temporal patterns, but they require large datasets and high computational resources, making them less suitable for smaller applications. Feature engineering has been identified as a critical component, with factors such as distance, traffic density, delivery partner rating, and festival conditions significantly influencing delivery time. The use of Haversine distance instead of raw GPS coordinates has also been proven to improve prediction accuracy. Furthermore, various regression algorithms have been compared in prior studies, with Gradient Boosting consistently achieving superior performance due to its ability to handle non-linear relationships. Research also emphasizes the importance of deploying machine learning models through user-friendly interfaces to improve usability and trust. However, many existing studies focus only on model development and lack practical implementation or integration of all relevant real-world factors. Therefore, the proposed system addresses these gaps by combining comprehensive feature engineering, multiple model evaluation, and a fully functional application to provide accurate and efficient delivery time prediction.

III. System Analysis

System analysis is a critical phase in developing the AI Delivery Time Prediction System, as it helps identify existing challenges and define system requirements. In current food delivery platforms, accurate prediction of delivery time is difficult due to multiple dynamic factors such as traffic conditions, weather, distance, and delivery workload. Traditional systems rely on simple calculations based on distance and average speed, which often produce inaccurate results. This leads to customer

dissatisfaction and inefficiency in operations. The analysis focuses on developing a system that can handle complex data and provide real-time predictions. It considers requirements such as accuracy, scalability, performance, and usability. Machine learning is identified as a key approach to solve this problem by learning patterns from historical data. The system must also support feature engineering and efficient data processing. Overall, system analysis helps design a robust and intelligent prediction system.

Existing System

Existing delivery time prediction systems mainly depend on rule-based approaches that use distance and average speed to estimate delivery duration. These systems do not consider important real-world factors such as traffic density, weather conditions, or delivery person performance. As a result, their predictions are often inaccurate and unreliable. Some systems use basic statistical methods, but they lack the ability to handle complex relationships between variables. Additionally, these systems are not adaptive and cannot learn from new data. They also do not provide real-time updates or personalized predictions. Most existing systems lack proper feature engineering techniques, which limits their accuracy. Furthermore, they do not offer user-friendly interfaces for easy interaction. Overall, existing systems are limited in intelligence, scalability, and efficiency.

Disadvantages of Existing System

- Inaccurate predictions due to limited factors
- Dependence on simple rule-based methods
- Inability to handle dynamic conditions like traffic and weather
- No learning from historical data
- Lack of real-time prediction capability
- Poor scalability for large datasets
- Limited user interaction and usability

Proposed System

The proposed system is an AI-based Delivery Time Prediction System that uses machine learning techniques to generate accurate delivery time estimates. It considers multiple factors such as distance, traffic density, weather conditions, delivery person details, vehicle condition, and order type. The system performs feature engineering, including the calculation of Haversine distance, to improve accuracy. Multiple regression models such as Linear Regression, Decision Tree, Random Forest, and Gradient Boosting are trained and evaluated. The system selects the best-performing model based on evaluation metrics. It is designed to handle large datasets and provide real-time predictions. A user-friendly graphical interface is developed using Python Tkinter, allowing users to input order details and receive instant predictions. The system also includes a PDF report generation feature. Overall, the proposed system provides a more accurate, scalable, and efficient solution.

Advantages of Proposed System

- High prediction accuracy using machine learning

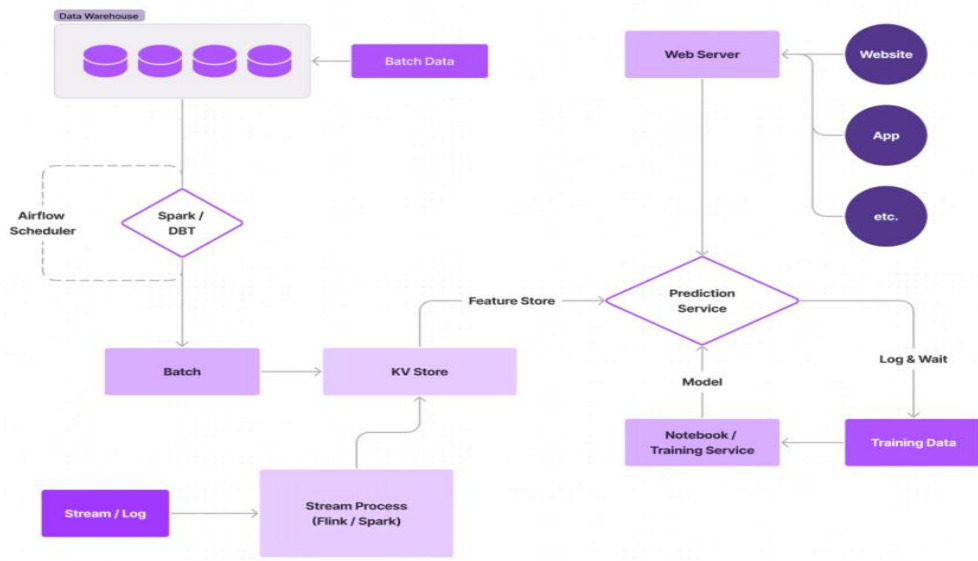
- Considers multiple real-world factors
- Real-time delivery time prediction
- Scalable and efficient system
- Improved customer satisfaction
- User-friendly interface

IV. Methodology

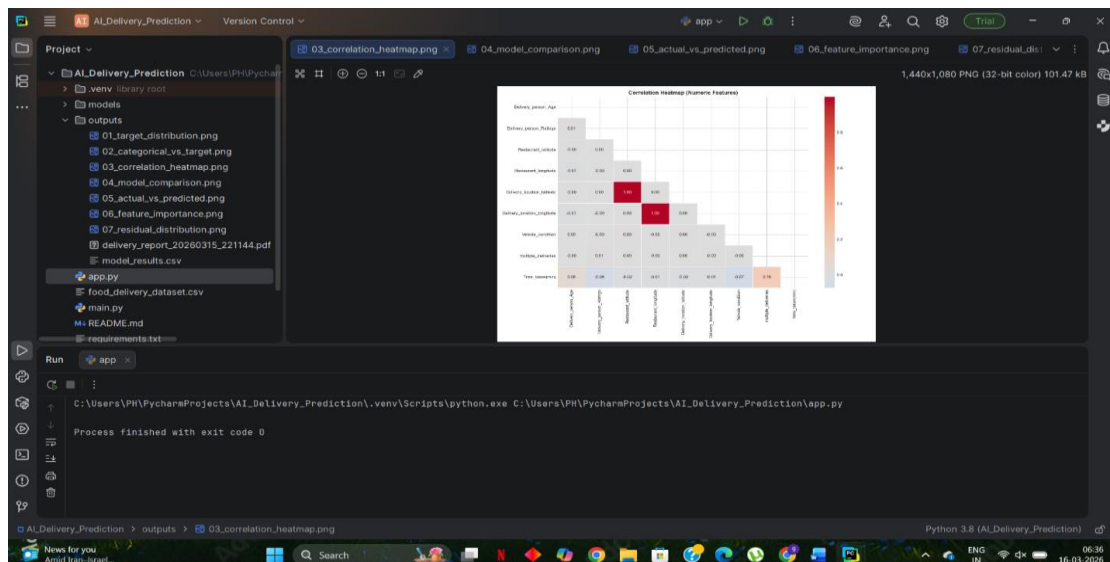
The methodology of the AI Delivery Time Prediction System involves several stages, starting with data collection and preprocessing. A dataset containing delivery records is gathered and cleaned by removing missing or irrelevant values. Feature engineering is performed to create meaningful inputs, including the calculation of Haversine distance from GPS coordinates. The dataset is then split into training and testing sets. Multiple machine learning regression models such as Linear Regression, Decision Tree, Random Forest, and Gradient Boosting are trained on the data. The models are evaluated using performance metrics such as MAE, RMSE, R^2 score, and MAPE. The best-performing model is selected for prediction. The system is implemented using Python and integrated into a graphical user interface using Tkinter. Users input order details, and the system processes the data to generate real-time delivery time predictions. The results are displayed and can be exported as a PDF report, ensuring usability and efficiency.

System Architecture

The system architecture of the AI Delivery Time Prediction System consists of multiple modules that work together to provide accurate delivery time predictions. The process begins with the user interface, developed using Python Tkinter, where users input delivery-related details such as location, traffic conditions, weather, and order type. This input data is passed to the preprocessing module, where missing values are handled and data is transformed into a suitable format. The feature engineering module then calculates additional features such as the Haversine distance using GPS coordinates to determine the actual distance between the restaurant and the delivery location. The processed data is then fed into the machine learning model, which has been trained using various regression algorithms. The model analyzes the input features and predicts the delivery time. The results are displayed instantly on the interface, and the system also provides an option to generate a PDF report for documentation. The database module stores historical data for future analysis and model improvement. This architecture ensures efficient data flow, real-time prediction, scalability, and a user-friendly experience.



V. Result and Output



AI Delivery Time Prediction System

Order Details

Fill in all fields, then click Run Prediction

Delivery Person Age: 28, Person Rating (2.5-5.0): 4.5, Distance (km): 5.0, Multiple Deliveries (0-3): 0, Vehicle Condition (0-3): 2, Weather Conditions: Sunny, Traffic Density: Medium, Type of Order: Meal, Type of Vehicle: motorcycle, Festival: No, City: Urban

Age: 18-55, Rating: 2.5-5.0, Distance: 0.1-50 km, Multiple Deliveries: 0-3, Vehicle Condition: 0-3

Run Prediction, Retrain Models, Clear Form

Prediction Result

Waiting — fill the form above and click Run Prediction

The screenshots show the 'AI Delivery Time Prediction System' interface. The first screenshot shows the 'Order Details' form with fields for Delivery Person Age (28), Person Rating (4.5), Distance (5.0), Multiple Deliveries (0-3), Vehicle Condition (2), Weather Conditions (Sunny), Traffic Density (Medium), Type of Order (Meal), Type of Vehicle (motorcycle), Festival (No), and City (Urban). Below the form are buttons for 'Run Prediction', 'Retrain Models', and 'Clear Form'. The second screenshot shows the 'Prediction Result' section with a message: 'Waiting — fill the form above and click Run Prediction'. The third screenshot shows the 'Background Process Log' with the message: 'PROCESS MONITOR: [04/04/24] System ready. Waiting for prediction request...'. Below the log is a 'Prediction Result' section with a message: 'Prediction completed successfully! Estimated Delivery Time: 57.9 minutes'. The result also includes 'Slow Delivery', 'Distance: 10.0 km', 'Traffic: 3', 'Weather: 0', 'City: 2', and 'Vehicle: 2'. There is an 'Export PDF Report' button and a link to 'Save a professional PDF report of this prediction'.

AI Delivery Time Prediction System

Age: 18-55 Rating: 2.5-5.0 Distance: 0.1-50 km Multiple Deliveries: 0-3 Vehicle Conditions: 0-3

Run Prediction **Retrain Models** **Clear Form**

Prediction Result

Prediction completed successfully!

Estimated Delivery Time: 57.9 minutes

Slow Delivery

Distance: 10.0 km Traffic: 3 Weather: 0 City: 2 Vehicle: 2

Export PDF Report Save a professional PDF report of this prediction

Background Process Log Real-time output from the AI engine

```

PROCESS MONITOR
[06:37:39] Festival : No
[06:37:39] City : Urban
[06:37:39] distance_km : 10.0
[06:37:39] Loading model from disk...
[06:37:39] Prediction done!
[06:37:39] Estimated Time : 57.9 minutes
[06:37:39] Category : Slow Delivery
    
```

AI Delivery Time Prediction System

Age: 18-55 Rating: 2.5-5.0 Distance: 0.1-50 km Multiple Deliveries: 0-3 Vehicle Conditions: 0-3

Run Prediction **Retrain Models** **Clear Form**

Prediction Result

Prediction completed successfully!

Estimated Delivery Time: 57.9 minutes

Slow Delivery

Distance: 10.0 km Traffic: 3 Weather: 0 City: 2 Vehicle: 2

Export PDF Report Save a professional PDF report of this prediction

PDF Exported

Report saved successfully!

Location: C:\Users\PH\PycharmProjects\AI_Delivery_Prediction\outputs\delivery_report_20260316_063758.pdf

Background Process Log Real-time output from the AI engine

```

PROCESS MONITOR
[06:37:39] Festival : No
[06:37:39] City : Urban
[06:37:39] distance_km : 10.0
[06:37:39] Loading model from disk...
[06:37:39] Prediction done!
[06:37:39] Estimated Time : 57.9 minutes
[06:37:39] Category : Slow Delivery
    
```

AI Delivery Time Prediction Report

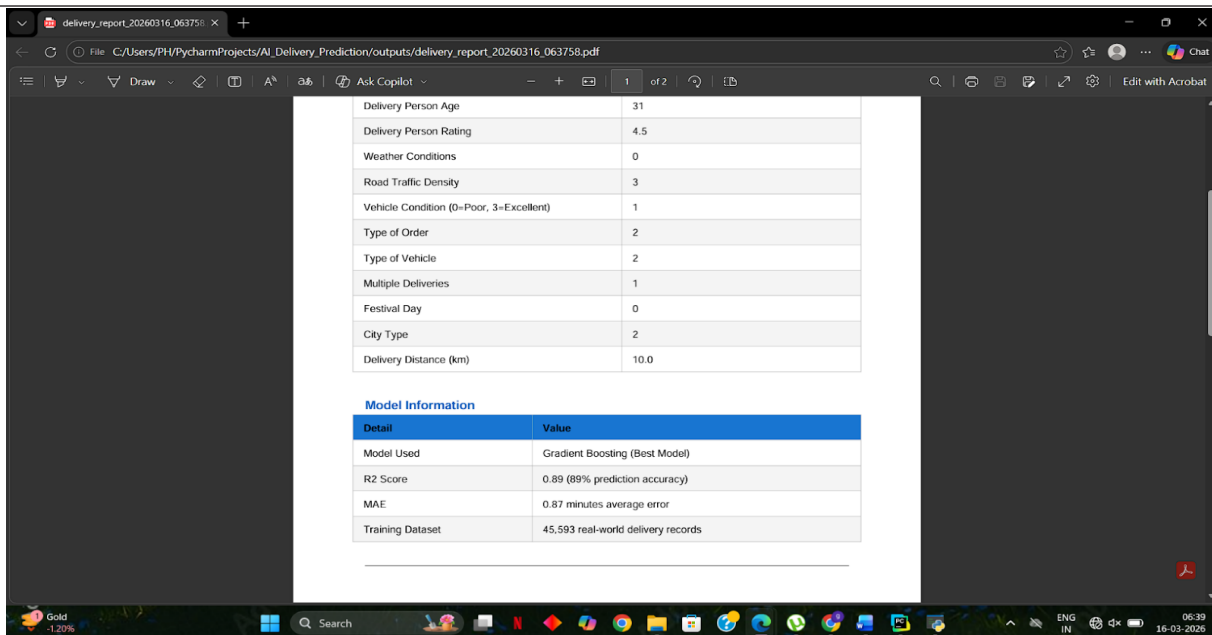
Generated: 16 March 2026 at 06:37:58

Prediction Result

Predicted Delivery Time	57.9 minutes
Category	Slow Delivery

Order Details

Field	Value
Delivery Person Age	31
Delivery Person Rating	4.5
Weather Conditions	0
Road Traffic Density	3
Vehicle Condition (0=Poor, 3=Excellent)	1
Type of Order	2
Type of Vehicle	2
Multiple Deliveries	1
Festival Day	0



Delivery Person Age	31
Delivery Person Rating	4.5
Weather Conditions	0
Road Traffic Density	3
Vehicle Condition (0=Poor, 3=Excellent)	1
Type of Order	2
Type of Vehicle	2
Multiple Deliveries	1
Festival Day	0
City Type	2
Delivery Distance (km)	10.0

Model Information	
Detail	Value
Model Used	Gradient Boosting (Best Model)
R2 Score	0.89 (89% prediction accuracy)
MAE	0.87 minutes average error
Training Dataset	45,593 real-world delivery records

VI. Conclusion

The AI Delivery Time Prediction System demonstrates the effective use of machine learning techniques in solving real-world logistics problems. By analyzing multiple influencing factors such as traffic, weather, distance, and delivery conditions, the system provides accurate and reliable delivery time predictions. The implementation of various regression models and the selection of the best-performing model ensures high prediction accuracy. The system also offers a user-friendly interface and real-time results, improving usability and efficiency. Compared to traditional methods, the proposed system significantly enhances prediction accuracy and customer satisfaction. Overall, the project highlights the importance of data-driven approaches in optimizing delivery operations and improving service quality.

References

- [1] Kumar, R. D., Prudhviraaj, G., Vijay, K., Kumar, P. S., & Plugmann, P. (2024). Exploring COVID-19 through intensive investigation with supervised machine learning algorithm. In Handbook of Artificial Intelligence and Wearables (pp. 145-158). CRC Press.
- [2] Swathi, B., Vijay, K., Sushanth Babu, M., & Dinesh Kumar, R. (2024, November). Machine Learning Techniques in Cloud Based Intrusion Detection. In The International Conference on Artificial Intelligence and Smart Environment (pp. 557-564). Cham: Springer Nature Switzerland.
- [3] Sv satyakrishna, shirisha rang, bhargavi nalacheruve.(2024) Prospective investigation on colorectal cancer with SMOTE on machine learning Algorithm
- [4] Dr.G.Vishnu Murthy, BhargaviNalacheruve 1Professor, Department of computer Science & engineering, Anurag University, TS, India. 2Student, Department of computer Science & engineering, Anurag University, TS, India.

- [5] V. N. S. Manaswini, K. K, C. Nigam, S. S. Ali, R. Niranjana, and Suman, “Real-Time Object Detection in Drone Surveillance Using YOLOv5,” in Proc. 2025 3rd Int. Conf. IoT, Communication and Automation Technology (ICICAT), Gorakhpur, India, 2025, pp. 1–6, doi: 10.1109/ICICAT68430.2025.11414670.
- [6] B. Soundarya, V. N. S. Manaswini, M. Ayyakrishnan, R. D. Kumar, “Contextual Analysis of Big Data Analytics in Intelligent Transportation Frameworks,” in Intersection of Artificial Intelligence, Data Science, and Cutting-Edge Technologies: From Concepts to Applications in Smart Environment, Lecture Notes in Networks and Systems, vol. 1353, Cham: Springer, 2025, doi: 10.1007/978-3-031-88304-0_79.
- [7] R. D. Kumar, V. N. S. Manaswini, “Applications of blockchain in smart cities: detecting fake documents from land records using blockchain technology,” in Blockchain for Smart Cities, Elsevier, 2021, pp. 105–117, doi: 10.1016/B978-0-12-824446-3.00017-X.
- [8] Tejavath Veeramma, Badarla Anil, Guguloth Ravinder, “An advanced movie recommender using collaborative filtering and sentiment analysis,” International Research Journal of Modernization in Engineering Technology and Science, vol. 7, no. 7, July 2025, doi: 10.56726/IRJMETS81618.
- [9] Ravi Kumar Banoth, Ramana Murthy B V, “Automatic crop recommendation system using LightGBM and decision tree machine learning models,” Journal of Machine and Computing, vol. 5, no. 1, pp. 343, Jan. 2025, doi: 10.53759/7669/jmc202505026.
- [10] Ravi Kumar Banoth, Dr. B.V. Ramana Murthy, “Smart agriculture through IoT and machine learning for analyzing carbon footprints,” in Proc. Int. Conf. Computer Science and Communication Engineering (ICCSCE), Apr. 2025.
- [11] Ravi Kumar Banoth, B. V. Ramana Murthy, “Soil image classification using transfer learning approach: MobileNetV2 with CNN,” SN Computer Science, vol. 5, art. no. 199, 2024, doi: 10.1007/s42979-023-02500-x.