MetroMind: A Machine Learning-Based Intelligent Metro Passenger Prediction System

Apsara PR[1] Fathima Sehna M[2] Nihala Shirin[3] Nafia Jebin[4] Mr. Juvin Wilson[5]
[5]Assistant Professor,
[1,2,3,4,5]Department of Computer Science and Engineering
[1,2,3,4,5]MGM Technological Campus Valanchery, Kerala, India

Abstract — Metropolitan railways have been facing many problems in terms of overcrowding, poor time management, and inefficiencies because of increasing urbanization. The METROMIND is a metro passenger prediction system based on machine learning to solve these challenges. It predicts passenger crowd density for better metro service operation. The system works through ML along with data analysis to predict passenger density. The proposed model has three layers including passenger application, operator control panel, and administration control panel. Machine learning algorithms are utilized to analyze data and make predictions on crowd. Passengers can get live crowd density data, buy tickets, and alert about emergencies, whereas the operator gets an effective way to manage operations of metro rails.

Keywords: Metro Systems, Machine Learning, Passen- Ger Flow Prediction, Crowd Density Prediction, Smart Trans- Portation, Predictive Analytics

I. INTRODUCTION

In recent years, rapid urbanization has made the requirement for effective public transportation systems more pressing than ever. Amongst other modes of transport, metro rail ways have received a lot of praise for being fast, reliable and ecofriendly means of transit. However, metro railways suffer from serious problems such as congestion, ineffective scheduling and lack of relevant passenger information. During peak hours, congestion results in passenger discomfort and potential dangers.

Traditional systems of managing metro railways are built upon the principles of fixed schedule planning and monitoring that is not able to efficiently cope with changing passenger dynamics. Consequently, traditional approaches lack analytical capabilities that enable operators to react to passenger dynamics and allocate resources efficiently.

This paper suggests METROMIND – an innovative system for intelligent prediction of metro passenger dynamics based on Machine Learning and data analysis technologies that allows to forecast crowd dynamics and predict the future situation on the railway.

The aim of this system is to increase the efficiency of metro operations and improve passenger experience.

II. LITERATURE REVIEW

Many researchers have focused on machine learning and deep learning techniques for predicting metro passenger flow. LSTM, GCN, and combined spatiotemporal models have been widely applied to capture temporal and spatial de- pendencies in metro data.

Researchers recently presented advanced methods based on TGALSTM and STFGCN models, where several techniques have been applied in one model to increase the accuracy. The advantage of these methods lies in the ability to understand the complex connections between metro stations and passengers during peak hours.

There are also other prediction techniques that use Origin Destination prediction methods with AFC datasets or optimization techniques for improving trains scheduling and resource allocation. There is a method based on reinforcement learning technique, Q-learning, which helps to adjust the operation of trains during delays.

These methods allow for obtaining high-quality results; however, they require big volumes of data and heavy computing processes, which makes it difficult to apply these techniques to real metro management tasks. Therefore, the introduced METROMIND system uses the Random Forest model, allowing combining accuracy with simplicity.

III. PROPOSED SYSTEM

The solution being considered, METROMIND, is an intelligence driven platform that uses machine learning algorithms and advanced data processing to help solve some key problems like congestion and suboptimal schedules. By incorporating Realtime analysis into traditional metro operations, the system will shift from the current reactive paradigm to become proactive and predictive in nature.

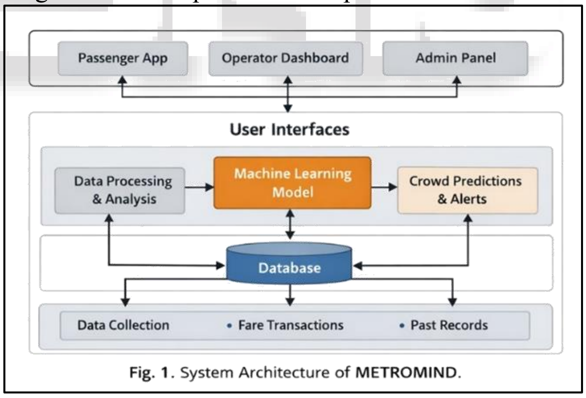

Fig. 1: System Architecture

METROMIND will consist of three modules, namely Passenger, Operator, and Administrator modules. The first one focuses on improving user experience through Realtime crowd analytics and the ability to access schedule data and book tickets in advance. The Operator module acts as an ex pert system that will allow operators to track passengers' behavior and make adjustments to the schedule based on these observations. The last module provides centralized management, which means more convenient operation and analysis of various data and resources.

As far as architecture goes, the system will include data collection, processing, prediction, and application layers. Passenger data collected through ticketing systems and past records will be processed through machine learning algorithms to provide accurate predictions. These predictions are then made available to passengers through intuitive ap plications.

Overall, by implementing intelligent algorithms and a user centric approach, METROMIND offers high potential for improving metro efficiency and passenger safety and comfort.

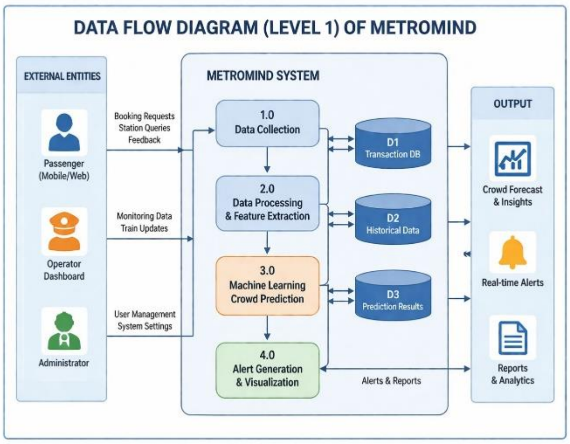

Fig. 2. Data Flow Diagram of METROMIND

Fig. 2: Data Flow Diagram

IV. METHODOLOGY

METROMIND makes use of datadriven approaches to predict the level of crowds within passenger populations us ing machine learning algorithms. The general workflow of the project includes data preprocessing, training models, and predicting crowds.

A. *Phase 1: Data Collection*

Data about passengers is gathered through the use of ticketing systems and historical data sets. Data may involve information on passenger travel behaviors and reservations

B. *Phase 2: Data Preprocessing*

Data cleaning and structuring processes take place using Python libraries like NumPy and Pandas. Missing data are accounted for and appropriate features are extracted for building the predictive models

C. *Phase 3: Model Selection and Training*

Random Forest model is chosen to make predictions. Uses an ensemble method by constructing several decision trees

D. *Phase 4: Model Serialization*

The created model is serialized using the Joblib library. Enables quick reuse of the model without training in the execution phase

E. *Phase 5: Prediction Process*

The trained model receives realtime data as well as historical data. Crowd level predictions are performed by the model

F. *Phase 6: Result Integration*

Predicted results are shown to passengers and operators using a website and mobile application.

V. IMPLEMENTATION

A. *Technology Stack*

MetroMind is developed using modern web, mobile, backend, and machine learning technologies as shown in Table I.

Component	Technology
Frontend Web Interface	HTML, CSS, JavaScript,Bootstrap
Mobile Application	Dart(Flutter)
Backend Framework	Django REST Framework
Programming Language	Python
Database	MySQL
Machine Learning Library	Scikitlearn,Numpy,Pandas
Mapping and Visualization	Google Maps API, Leaflet.js
Development Tools	PyCharm, Android Studio
Version Control	GitHub

Table I: Technology Stack Used in Metromind

B. *System Functional Modules*

The system has been split into three functional modules as follows:

1) Passenger Module: This module enables the user to check passenger loads, schedule, booking, emergency alerts, and other information.
2) Operator Module: The operator uses this unit to monitor passenger loads and manage the system during emergencies or congestion.
3) Administrator Module: The administrator can take control of the entire system through this module.

C. *Machine Learning Implementation*

The Random Forest technique is employed by the system to forecast the level of crowding of passengers. The trained model will use pre-processed training datasets from historical sources by means of libraries like NumPy and Pandas. The trained model is serialized by the use of the Joblib library for further reuse in the process of implementation. The model makes decisions based on data received.

D. *Interface and Backend Integration*

The frontend is connected to the backend through Django, a framework that allows communication between the user, database, and prediction models. There is a smooth flow of interaction between the web app and the mobile app, ensuring realtime data synchronization within all the mod ules.

E. *Performance Optimization*

For effective optimization, the machine learning system is saved for future use rather than undergoing training again. For efficient system operation, there must be an efficient database to facilitate fast data retrieval and storage. Due to its modularity nature, the system can easily be integrated into other systems.

VI. Results and Discussion

A. Prediction Performance

Historical data and simulation were used to test the pre diction ability of the METROMIND system. According to the results, the system is capable of analyzing the pat terns of passengers' movements and detecting periods when there will be congestion. The prediction algorithm has been proven to have an accuracy of about 80%.

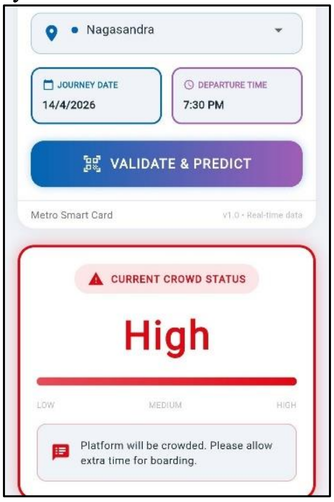

Fig. 3: Crowd prediction result showing realtime passenger density (High) with user guidance.

B. System Effectiveness

It offers realtime information that allows passengers to take an informed decision on their journey, thereby reducing congestion. The operator's module helps in monitoring passenger traffic efficiently and also helps in making changes to train schedules and resources.

C. System Integration and Reliability

The integration between different system mod ules—passenger, operator, and administrator—is seamless. The system ensures smooth data flow and consistent performance across web and mobile platforms, supporting realtime functionality.

D. Limitations and Improvements

While the system works well in standard circumstances, its accuracy could vary during unpredictable incidents, such as disruption of service or erratic behavior from passengers. Improvements in the model could be made through the use of more realtime data and better machine learning processes.

VII. Conclusion

METROMIND is an efficient approach for metro trans portation improvements through machine learning and data analytics. METROMIND makes predictions on passenger crowd density and allows for preemptive decision making, which helps avoid crowding.

The incorporation of passenger, operator, and administrative subsystems gives an integrated approach to the system operation and helps both passengers and operators benefit from it. METROMIND implementation shows its high prediction accuracy and effectiveness in monitoring and controlling metro transportations.

In conclusion, METROMIND plays a vital role in developing intelligent urban transportation systems. In the future, there can be implemented additional subsystems such as realtime sensors or IoT devices along with improved ML algorithms.

Acknowledgment

The authors sincerely thank Mr. Juvin wilson (Project Guide), Ms. Shabna M (Head of Department), and Prof. Dr. Binu B Pillai (Principal), MGM Technological Campus, for their guidance and support throughout this project. The authors also thank the Department of Computer Science and Engineering, APJ Abdul Kalam Technological University, as well as all faculty members, staff, and participants who contributed to the successful completion of MetroMind

References

[1] S. A. J. Cheng, G. Liu, and W. Juan, "Shortterm passenger flow prediction in urban rail transit based on points of interest," IEEE Access, vol. 12, pp. 95196–95208, 2024.
[2] C. Liu, "A novel method for passengers' origin destination distribution prediction over metro networks with expansions based on automatic fare collection data," IEEE Access, vol. 12, pp. 137418–137426, 2024.
[3] Y. Zhang, Z. Tian, and Y. Pan, "STFGCN: Spatio temporal fusion graph convolutional networks for sub way traffic prediction," IEEE Access, vol. 12, pp. 194449–194461, 2024.
[4] H. Zuo, S. Zhang, and J. Di, "Integrated optimization of train schedules and transportation plans for a pas sengerfreight metro line," vol. 17, no. 2, p. 730, 2025.
[5] W. Wang, "AttentionLSTM based modelling for Hangzhou city metro passenger flow forecasting study," vol. 113, pp. 66–76, 2024.
[6] Y. Hu and Q. Luo, "Realtime adjustment method for metro systems with train delays based on improved Q learning," vol. 14, no. 4, p. 1552, 2024.
[7] L. Wan, W. Cheng, and J. Yang, "Optimizing metro passenger flow prediction: Integrating machine learning and timeseries analysis with multimodal data fusion," 2024.
[8] N. Dong, T. Li, and T. Liu, "A method for shortterm passenger flow prediction in urban rail transit based on deep learning," vol. 83, pp. 61621–61643, 2024.
[9] C. Gao, H. Liu, J. Huang, Z. Wang, X. Li, and X. Li, "Regularized spatiotemporal graph convolutional networks for metro passenger flow prediction," IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 9, pp. 11241–11255, 2024.
[10] J. Wang, Y. Zhang, Y. Hu, and B. Yin, "Metro flow prediction with hierarchical hypergraph attention networks," IEEE Transactions on Artificial Intelligence, vol. 5, no. 6, pp. 3012–3021, 2024.
[11] R. R. Mendhe, S. Shingare, D. Shinde, A. Surve, and S. Kulkarni, "Forecasting passenger traffic in metro systems: ARIMA analysis of metro ticket reservation data," 2025, pp. 740–748.

