Delhi Metro Operations Optimization

[2] Ms. Pooja K N [1] Suchithra C

[1] *Student, 4th Semester MCA, Department of MCA, BIET, Davanagere*

[2] *Assistant Professor, Department of MCA, BIET, Davanagere*

Abstract

The Delhi Metro, one of the most extensive and sophisticated urban transit systems globally, faces continuous challenges in optimizing operations due to fluctuating passenger demand and complex scheduling dynamics. This paper presents a machine learning-based approach to analyze and predict operational patterns within the Delhi Metro network. Utilizing datasets comprising metro routes, stops, schedules, and trip frequencies, the study explores spatial and temporal variations in metro operations across different times of the day and days of the week. Key operational metrics, such as stop-wise route frequency and trip intervals, are examined through data visualization and exploratory analysis. A Random Forest Regressor model is developed to predict the number of routes at specific metro stops based on time, day, and location. The system supports two user types: Admin, who manages users and FAQ content, and Users, who can register, log in, and access route prediction insights and FAQs. The results demonstrate the potential of machine learning in enhancing metro scheduling, optimizing resource allocation, and supporting data-driven decision-making for more efficient and sustainable urban transportation.

Keywords:

Delhi Metro, Urban Transportation, Machine Learning, Random Forest Regressor, Travel Pattern Prediction

I. INTRODUCTION

The Delhi Metro is one of the largest and most advanced urban transit systems in the world, serving millions of passengers daily. With a vast network of routes, stations, and complex scheduling, ensuring efficient operations and optimizing services is a constant challenge. As passenger demand fluctuates based on time of day, day of the week, and other factors, traditional methods of analysis often fall short in capturing the dynamic nature of metro operations. This is where machine learning becomes essential. By leveraging historical metro data, machine learning techniques can help predict travel patterns, optimize schedules, and improve route planning. These models can provide valuable insights into factors such as passenger demand, travel intervals, and route frequency, enabling more accurate and data-driven decision-making to enhance the overall efficiency, reliability, and sustainability of the metro system.

The aim of this proposed system is to analyze and predict the operational patterns of the Delhi Metro by exploring various datasets related to metro routes, stops, and schedules. By examining the geographical distribution of stops, the frequency of trips, and the number of routes passing through each stop, the proposed system seeks to identify patterns in metro operations across different times of the day and days of the week. The predictive model developed in the proposed system uses machine learning techniques, specifically a Random Forest Regressor, to estimate the number of routes at specific metro stops based on factors like time of day, day of the week, and location. This can be used to optimize metro operations, improve scheduling, and support decision-making for better service delivery and resource allocation.

The efficient operation of metro systems plays a critical role in urban transportation, providing millions of passengers with reliable and timely services. In this proposed system, we aim to analyze the operational patterns of the Delhi Metro, one of the busiest metro systems in the world. By leveraging various datasets containing information about metro routes, stops, schedules, and trip frequencies, the proposed system seeks to uncover insights into how the system functions across different times of the day and days of the week.

Through data exploration and visualization, we examine key aspects of the metro's operations, such as the geographical distribution of stops, the number of routes serving each stop, and the intervals between subsequent trips. Additionally, a predictive model is developed using machine learning techniques, specifically a Random Forest Regressor, to forecast the number of routes at a given stop based on factors like time of day, day of the week, and location.

The results of this proposed system are aimed at optimizing metro operations, improving scheduling efficiency, and providing valuable insights for decision- makers to enhance service delivery. By understanding the factors that influence metro traffic and predicting demand, the proposed system contributes to the development of smarter, more responsive transportation systems.

The system is designed with two types of users: Admin and User. The Admin has the authority to log in, manage registered users, and maintain the FAQ section by adding or deleting entries as needed. On the other hand, the User can register with valid details and log in using their credentials. Once logged in, Users can access detailed metro operation analysis, including insights on route stop predictions, allowing them to make informed decisions based on the available data. Additionally, Users can view the FAQ section, where common questions are answered, enhancing their overall experience and helping them navigate the system with ease.

II. RELATED WORK

1. Operations Research in Metro Rails, Authors:Deshna Mehta, Dhairya Jain, Dhruvi Nathwani, Ekansh Arora, Gunjan Kapoor,

The paper "Operations Research in Metro Rails" examines how Operations Research (OR) techniques, particularly the simplex method, can optimize key aspects of metro rail systems such as timetabling and platform assignments. As metro networks grow more complex, the study highlights the need for efficient scheduling and resource management. By applying quantitative models, the authors demonstrate that metro operations can achieve better cost efficiency and infrastructure utilization, ultimately improving overall system performance.

2. Metro Train Operation Plan Analysis Based on Station Travel Time Reliability, Authors: Ruihua Xu, Fangsheng Wang, Feng Zhou.

This study presents a novel method for analyzing metro train operation plans using Station Travel Time Reliability (STTR) derived from Automated Fare Collection (AFC) data. By integrating a data-driven framework—including preprocessing, STTR computation, SOM neural network clustering, and optimization—the authors demonstrate how STTR serves as a valuable metric for identifying inefficiencies in metro services. Applied to the Beijing metro network, the method reveals how mismatches between transport capacity and passenger demand cause localized congestion. The findings show that STTR can effectively support the optimization of metro operation plans, enhancing efficiency and passenger experience.

3. Optimizing Delhi Metro Operations with XGBoost Ridership Levels in Urban Transit Systems. Authors: S Anu Priya, Anoop Bahuguna, P Jaisankar, R Ramesh, Abebe Kindie Awuraris, R Reka.

This paper proposes a predictive model using the XGBoost algorithm to classify metro station ridership levels in Delhi as low, medium, or high. By incorporating spatial and operational features such as station age, location, and distance from the line's origin, and applying thorough feature engineering and hyperparameter tuning (via GridSearchCV), the model achieved over 99% accuracy, precision, recall, and F1 score. This research highlights how machine learning can significantly enhance urban transit planning and offers scalable solutions adaptable to other metro systems facing similar mobility challenges.

4. On-demand ridesharing with optimized pick-up and drop-off walking locations. Authors: Andres Fielbaum, Xiaoshan Bai, Javier Alonso-Mora.

The paper ―On-demand ridesharing with optimized pick-up and drop-off walking locations‖ by Andres Fielbaum, Xiaoshan Bai, and Javier Alonso-Mora presents a system that enhances on-demand ridesharing by allowing users to walk short distances to optimized pick-up and drop-off (PUDO) points. Instead of strict door- to-door service, the model proposes walking to nearby PUDO points to improve overall system efficiency. Using simulations on real-world Manhattan taxi data, the study shows that short walks (around one minute) can significantly reduce ride request rejections and vehicle travel time. The authors develop heuristics to solve the complex assignment and optimization problem and demonstrate that efficiency gains are greatest in high- demand scenarios.

5. Incorporating Travel Behavior Regularity into Passenger Flow Forecasting. Authors: Zhanhong Cheng, Martin Trépanier, Lijun This paper introduces a new framework for metro ridership forecasting that goes beyond traditional time series models by incorporating travel behavior regularity. The authors propose using―returning flow‖ — derived from previous alighting trips — as a covariate to capture the causal relationship between morning and evening travel patterns. They develop the Return Probability Parallelogram (RPP) to estimate this return flow, which enables the model to account for long-range dependencies.

Evaluation with real-world metro data demonstrates significant accuracy improvements in both short- and long- term forecasting, especially in business-centric stations where passengers tend to follow daily round trips.

6. Passenger Flow Prediction Based on a Hybrid Method in the Nanjing Metro System. Authors: Jiaxiao Feng, Xiangyu Chang, Qiang Tu, Zimu Li, Leyu Zhou, Xiaoyu Cai.

This study introduces a hybrid prediction model called DWT– SARIMA for short-term passenger flow forecasting in urban metro systems. The method uses Discrete Wavelet Transform (DWT) to decompose time-series data into high- and low-frequency components, followed by applying Seasonal ARIMA (SARIMA) to predict each component separately. The outputs are then recombined for final predictions. Tested on Nanjing Metro Line 2 data, this model showed superior accuracy compared to SARIMA alone and backpropagation neural networks, with a MAPE of 9.372%, VAPE of 0.670%, and RMSE of 36.364.

III. METHODOLOGY

This study adopts a data-driven approach to analyze and predict passenger traffic at metro stops using GTFS (General Transit Feed Specification) datasets, with a focus on Delhi Metro. The methodology comprises five major phases: data acquisition, preprocessing, feature engineering, predictive modeling, and visualization—integrated into a web-based application using the Flask framework.

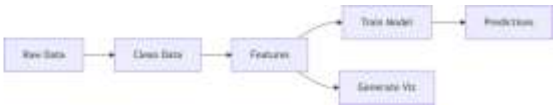


Fig 1. Proposed Methodology

3.1 Data Acquisition

The GTFS dataset used in this study includes seven primary text files: `agency.txt`, `calendar.txt`, `routes.txt`, `shapes.txt`, `stop_times.txt`, `stops.txt`, and `trips.txt`. These files provide comprehensive information regarding metro agency operations, route structures, service availability, stop locations, and arrival times.

3.2 Data Preprocessing

The `stop_times.txt` file contains `arrival_time` values in the HH\:MM\:SS format, which are converted into Python `datetime.time` objects. Special handling is implemented for entries exceeding 24 hours (e.g., 25:00:00), which are normalized to the 24-hour format. The data files are merged to establish a link between stops, trips, and routes. The `calendar.txt` file is joined with `trips.txt` to include day- of-week service information.

3.3 Feature Engineering

Each stop is characterized by temporal and spatial features. The `arrival_time` values are transformed to extract `day_of_week` and `hour_of_day` features. Spatial coordinates (`stop_lat`, `stop_lon`) are taken from `stops.txt`. A new target variable, `number_of_routes`, is created by calculating the number of unique routes servicing each stop. Additionally, trips are categorized into time intervals such as Early Morning, Morning Peak, Midday, Evening Peak, and Late Evening to support temporal analysis.

3.4. Predictive Modeling

A supervised machine learning model is developed to estimate traffic density (number of routes) at metro stops based on spatial and temporal features. A Random Forest Regressor is employed for its robustness against overfitting and capability to model nonlinear relationships. The dataset is split into training and testing sets using an 80-20 ratio. Performance is evaluated using Root Mean Squared Error (RMSE) and 5-fold cross-validation to ensure generalization. The model is then used to generate traffic predictions based on user inputs.

3.5 Visualization and Application Interface

Exploratory Data Analysis (EDA) is conducted to extract patterns and trends. Visualizations include:

- Geospatial plots of metro routes and stops.
- Bar plots showing trip frequencies per weekday and time interval.
- Interval analysis between consecutive trips.
- Traffic adjustment simulation under hypothetical peak-hour scaling.

IV. TECHNOLOGIES USED

4.1 Python Programming Language:

Python serves as the backbone for the entire system due to its versatility, readability, and extensive support for data analysis, machine learning, and web development. Its rich ecosystem of libraries makes it ideal for handling complex data-driven applications such as metro operations analysis.

4.2 Data Analysis and Visualization Libraries (Pandas, NumPy, Matplotlib, Seaborn): Pandas and NumPy are essential for data manipulation, cleaning, and exploration of large datasets related to metro routes, stops, and schedules. Matplotlib and Seaborn are powerful visualization tools that help in generating insightful charts and graphs, enabling users to better understand geographical distributions, trip frequencies, and operational patterns.

4.3 Machine Learning Frameworks (Scikit-learn):

Scikit-learn is utilized for building and training the predictive models, such as the Random Forest Regressor. This library provides robust machine learning algorithms and utilities for model selection, training, evaluation, and deployment, making it suitable for forecasting the number of routes at specific metro stops.

4.4 Database Management Systems (MySQL, PostgreSQL, or MongoDB):

A reliable database system is crucial for storing and managing user data, metro operational data, and frequently asked questions (FAQs). Relational databases like MySQL or PostgreSQL are suitable for structured data, while MongoDB can be used for more flexible, document-based storage.

4.5 Web Development Frameworks (Django or Flask):

Django or Flask, both Python-based web frameworks, are used to create the web application that serves as the user interface for both Admin and User roles. These frameworks facilitate user authentication, session management, and integration with backend services, ensuring a secure and responsive experience.

4.6 User Authentication and Authorization:

Technologies such as Django's built-in authentication system or Flask extensions (like Flask-Login) are implemented to manage user registration, login, and role- based access control, ensuring that Admins and Users have appropriate permissions within the system.

4.7 RESTful API Development:

RESTful APIs are developed to enable seamless communication between the frontend and backend, allowing for efficient data retrieval, prediction requests, and FAQ management. This modular approach supports scalability and integration with other systems if needed.

4.8 Frontend Technologies (HTML, CSS, JavaScript, Bootstrap):

A responsive and user-friendly interface is built using HTML, CSS, and JavaScript, with frameworks like Bootstrap to ensure the application is accessible and visually appealing across different devices.

4.9 Cloud Deployment and Hosting (AWS, Azure, or Heroku):

To make the system accessible to users, cloud platforms such as AWS, Microsoft Azure, or Heroku are used for hosting the web application, ensuring scalability, reliability, and secure access.

4.10 Version Control (Git and GitHub):

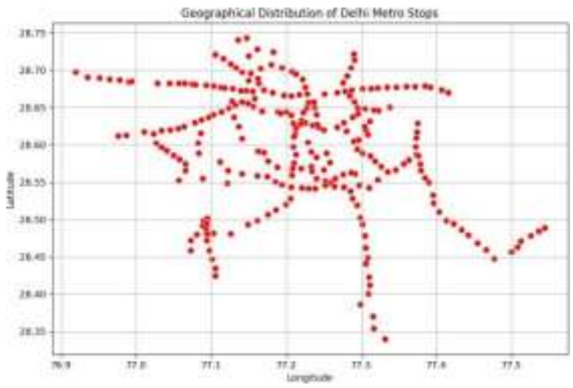
Git is used for source code management, enabling collaborative development, version tracking, and seamless deployment via platforms like GitHub. These technologies collectively enable the development, deployment, and maintenance of a robust, scalable, and user-friendly metro operations analysis and prediction system, supporting both administrative and end-user functionalities.

IV. Result

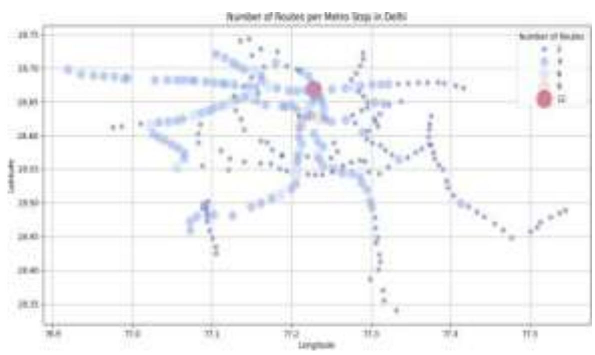
Geographical Paths of Delhi Metro Routes

This showing Delhi metro paths.

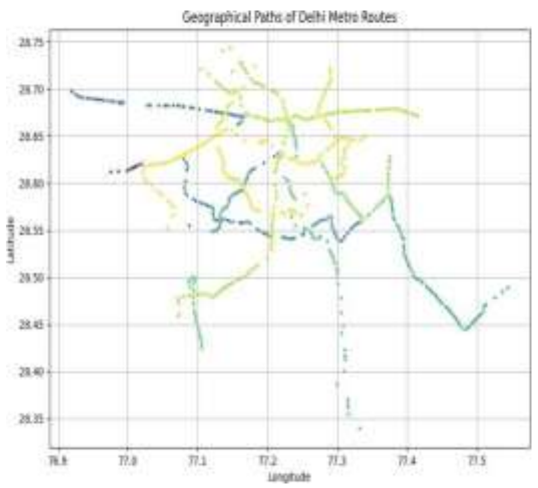
Geographical Distribution of Delhi metro Stops

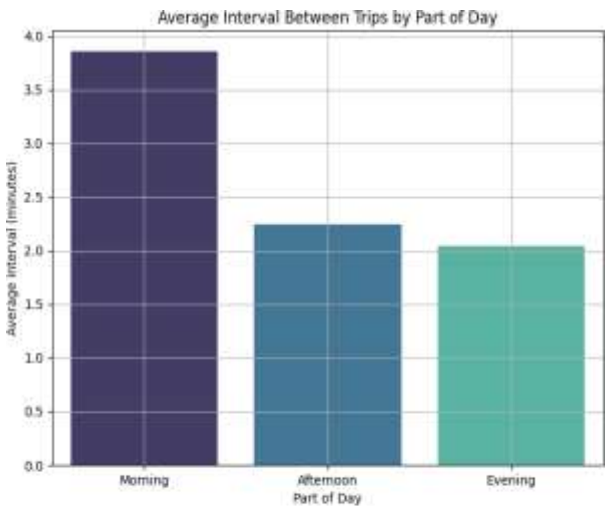


Number of Routes per Metro Stop in Delhi

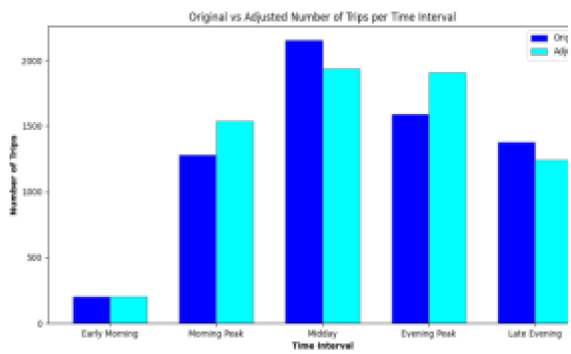


Average Interval between Trips by Part of Day





Original vs adjusted number of trips per Time Interval



V. CONCLUSION

The study effectively demonstrates how machine learning, particularly the Random Forest Regressor algorithm, can be utilized to analyze and predict operational patterns in a large urban transit network such as the Delhi Metro. By systematically examining datasets related to metro stops, routes, and schedules, this research identifies significant trends in metro traffic based on time of day, day of the week, and geographical distribution. The predictive model developed in this work enables accurate forecasting of the number of routes serving specific stops, offering a valuable tool for enhancing schedule planning and resource allocation.

The results highlight the potential of data- driven approaches in supporting the optimization of metro operations. The integration of the model into a user-centric web platform further enhances its practical utility, allowing users to interact with the predictions and insights in real time. Administrators benefit from improved system management capabilities, while end users gain access to route-based insights for better commuting decisions.

Overall, the paper contributes to the field of intelligent transportation systems by offering a scalable and interpretable solution for operational forecasting. Future enhancements may include incorporating real-time passenger data, extending the model to cover passenger load predictions, and integrating adaptive learning techniques to further refine prediction accuracy and system responsiveness.

VI. REFERENCES

1. Operations Research in Metro Rails, Authors:Deshna Mehta, Dhairya Jain, Dhruvi Nathwani, Ekansh Arora, Gunjan Kapoor, Journal Name / Publisher:Journal of Emerging Technologies and Innovative Research (JETIR), ISSN:2349-5162, Volume
and Issue: Volume 6, Issue 6, June 2019.

2. Metro Train Operation Plan Analysis Based on Station Travel Time Reliability, Authors: Ruihua Xu, Fangsheng Wang, Feng Zhou. Journal: Journal of Advanced Transportation. Publisher: Hindawi. Volume: 2021. Article ID: 8813461.DOI:[https://doi.org/10.1155/

2021/8813461]

3. Optimizing Delhi Metro Operations with XGBoost Ridership Levels in Urban Transit Systems. Authors: S Anu Priya, Anoop Bahuguna, P Jaisankar, R Ramesh, Abebe Kindie Awuraris, R Reka. Published In: Proceedings of the 3rd International Conference on Optimization Techniques in the Field of Engineering (ICOFE-2024)

4. On-demand ridesharing with optimized pick-up and drop-off walking locations. Authors: Andres Fielbaum, Xiaoshan Bai, Javier Alonso-Mora. Journal: Transportation Research Part C: Emerging Technologies. Volume:128,2021., DOI: [10.1016/j.trc.2021.103061].
Publisher: Elsevier

5. Incorporating Travel Behavior Regularity into Passenger Flow Forecasting. Authors: Zhanhong Cheng, Martin Trépanier, Lijun Sun. DOI:[https://doi.org/10.1016/j.trc.2021.103200]. Publisher: Transportation Research Part C:Emerging Technologies, Elsevier.

6. Passenger Flow Prediction Based on a Hybrid Method in the Nanjing Metro System. Authors: Jiaxiao Feng, Xiangyu Chang, Qiang Tu, Zimu Li, Leyu Zhou, Xiaoyu Cai. DOI: [https://doi.org/10.1061/JTEPBS.TEE NG-8830. Publisher: Journal of Transportation Engineering, Part A: Systems, ASCE. Volume & Issue: Volume 151, Issue 5, 2025.