IJERST

International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 21 No. 3 (1) 2025

ijerst.editor@gmail.com

editor@ijerst.com

ISSN 2319-5991 www.ijerst.com
Vol. 21, No. 3(1), 2025

Research Paper

AI-Driven Bus Boarding Demand Prediction for Real-Time Public Transport Optimization

SK. Mahboob Basha[1], Nidanousheen[2], K. Anusha[2], E. Sathwika Reddy[2]
[1]Professor, [2]UG Student, [1,2]Department of Computer Science and Engineering (CSIT)
[1,2]Sree Dattha Institute of Engineering and Science, Sheriguda, Ibrahimpatnam, 501510, Telangana.

ABSTRACT

Public transportation systems exhibit considerable fluctuations in passenger demand, with over 60% of daily boardings occurring during peak hours and nearly 30% of commuters reporting delays due to overcrowding. In densely populated urban areas, approximately 70% of route optimization efforts fail without accurate boarding demand forecasts. Traditional manual estimation methods—such as time-slot passenger counts, route-based field surveys, and ticket ledger analyses—are often prone to human error, labor-intensive, and lack scalability. These limitations result in inconsistent data, impeding effective transit planning and service optimization. To address these challenges, this study proposes an AI-based bus boarding demand prediction system that leverages Support Vector Machine (SVM) and Deep Neural Network (DNN) classifiers. The models incorporate real-time input features, including timestamps, route numbers, traffic conditions, and boarding volumes. Historical transit datasets are used for training, and model performance is evaluated using standard accuracy metrics to identify the most reliable approach. Furthermore, the predictions are cross-validated against three traditional estimation methods—time-slot monitoring, field-based surveys, and ticket ledger review—to ensure robustness and practical relevance. The resulting framework delivers high-precision, real-time demand predictions, enabling dynamic scheduling and mitigating the mismatch between service availability and commuter demand. This integrated solution bridges the gap in manual estimation approaches and lays the groundwork for a scalable, data-driven public transport optimization system.

Keywords: Public transport, demand prediction, machine learning, SVM, deep neural network, dynamic scheduling, AI in transportation, boarding estimation, transit planning, real-time optimization.

Received: 14-7-2025 Accepted: 21-8-2025 Published: 28-8-2025

1. INTRODUCTION

Passenger demand prediction is a vital component in optimizing public transportation systems, especially in the face of rapid urbanization and fluctuating ridership patterns. With over 253 billion global journeys recorded in 2022 and cities like Tokyo and New York experiencing over 10 million daily rides, infrastructure often becomes strained due to imbalances between demand and supply. Traditional forecasting methods, such as historical averages and linear regression, fail to adapt to real-time factors like weather, public events, or traffic disruptions, leading to inefficiencies and losses—such as the estimated €8 billion annual cost to the EU economy due to delays. With the proliferation of digital ticketing, GPS tracking, and IoT sensors, there is now an opportunity to apply machine learning (ML) and deep learning (DL) models that harness spatiotemporal patterns and real-time data. This research is motivated by the operational needs of public agencies (like TfL and Singapore's LTA), private ride-sharing companies (such as Uber and Lyft), and logistics services (e.g., DHL), all of which require dynamic prediction tools to manage resources, reduce idle times, and enhance customer satisfaction. The study defines the core problem as the inability of static models to predict demand accurately under dynamic urban conditions and proposes leveraging Support Vector

Machines (SVM) and Deep Neural Networks (DNN) for forecasting. These advanced models aim to extract insights from high-volume datasets and capture complex behavioral patterns. Accurate forecasting enables smarter scheduling, reduced environmental impact, and improved commuter experience. Moreover, it aligns with smart city objectives by integrating with infrastructure systems and promoting sustainable mobility. The study aims to compare SVM and DNN models on real-world datasets, examining performance across factors like time, location, weather, and events, with the goal of categorizing high and low-demand periods. Key advantages include reduced operational costs, enhanced scalability, adaptability to real-time trends, and support for emergency planning and strategic infrastructure development.

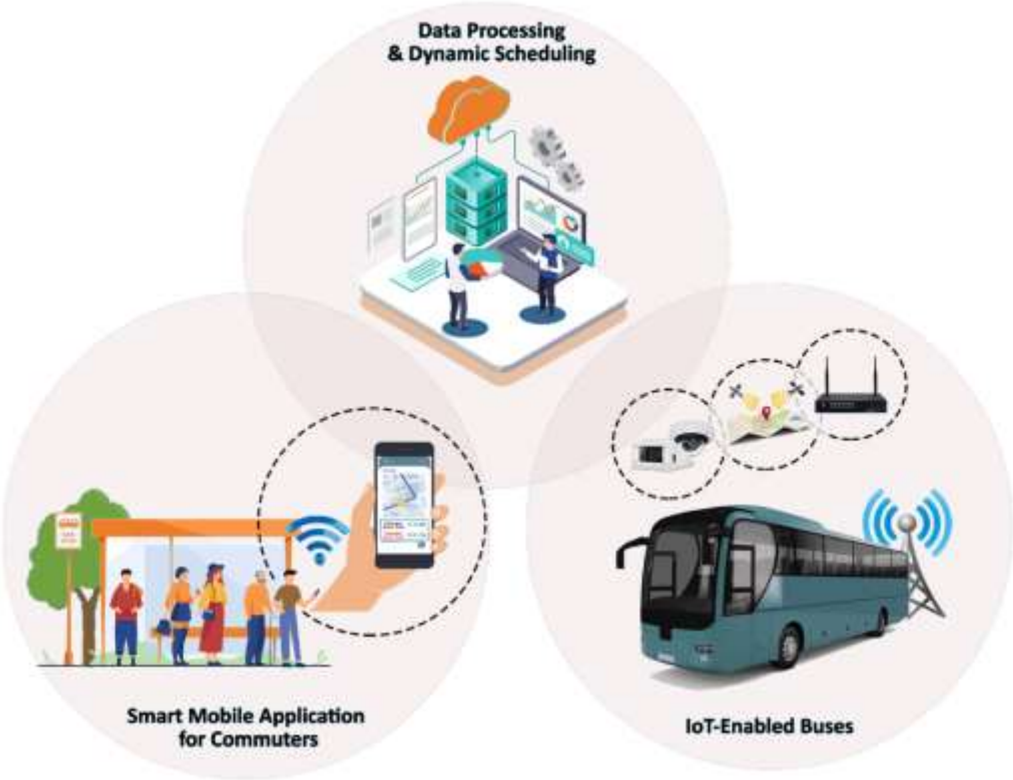

Fig. 1: Real time bus departure prediction.

Applications span across urban buses, metros, airport shuttles, ride-sharing platforms, logistics routing, and public event planning—making demand prediction a critical enabler of efficient, inclusive, and environmentally conscious transport ecosystems.

2. LITERATURE SURVEY

This approach focuses on formulating operational constraints such as bus capacity, operation schedules, and passenger demand. Generally, these methods aim to minimize operational costs or maximize system efficiency. For example, Borndörfer et al. [1] and Zhou et al. [2] addressed the problem using Mixed Integer Linear Programming (MILP) to minimize operational costs and travel times, considering constraints such as vehicle capacity, synchronization between routes, and minimization of transfer times. In these studies, the problem was formulated as a multi-commodity flow model, which allowed for dynamically generating optimized transit lines. However, the second approach uses column generation to handle the computational complexity of capacity constraints, achieving solutions applicable to real networks. On the other hand, Pei et al. [3] explored the use of modular vehicles in transport networks through an MILP approach that dynamically adjusts vehicle capacities and frequencies to respond to fluctuations in demand.

This solution showed a significant reduction in operational costs and travel times compared to traditional systems. Guan et al. [4] proposed a dispatch and route optimization model for public transportation systems with variable demand, using a hybrid approach based on a hybrid LNS-genetic algorithm, which resulted in significant improvements in operational efficiency and user service. Van Oudheusden et al. [5] introduced a nonlinear programming approach to optimize frequencies and schedules, focusing on minimizing empty trips performed by the fleet and increasing the number of transported passengers. To avoid stochastic uncertainties, Van Berkum et al. [6] formulated the problem within a rolling horizon framework, dividing an operational day into predetermined intervals. This approach uses a convex nonlinear formulation that allows solving the problem to global optimality with a limited computational cost, simultaneously optimizing the dispatch times of all scheduled trips in each interval. Gkiotsalitis [7] demonstrated that periodic optimization through convex quadratic programming can minimize variations in vehicle departure intervals, improving service regularity in high-frequency lines, as evidenced in the case study presented in the 302 bus network in Singapore.Another notable approach in the reviewed literature is that of Chen and Zhou [8], who implemented a Dynamic Programming (DP) algorithm to solve the dispatch problem in oversaturated systems. This approach efficiently handles constraints such as vehicle capacities and waiting times, applying valid inequalities to reduce the computational complexity of DP. The results demonstrated significant reductions in operational costs and waiting times, with practical applications in networks such as the Beijing metro and Tampa Bay.Although exact methods guarantee optimal solutions, they face limitations in terms of scalability and applicability in dense networks or highly dynamic systems, as they require computationally expensive methods such as Branch and Bound to achieve optimality [9]. In these cases, heuristics provide fast and flexible solutions, sacrificing precision for efficiency. These methodologies explore the solution space using operational rules or adaptive algorithms. Hadas et al. [10]

3. PROPOSED METHODOLOGY

The proposed system introduces a hybrid ensemble-based demand prediction framework that integrates classical and deep learning techniques—namely Support Vector Machine (SVM) and Deep Neural Network (DNN)—with structured route encoding using Count Vectorizer to classify bus boarding demand as "More" or "Less." Unlike conventional approaches that rely solely on a single model or generalized ML pipelines, this system features a dual-model architecture wherein both SVM and DNN are trained in parallel on vectorized boarding data, capturing both linear separability and complex non-linear patterns. Their outputs are fused for enhanced accuracy and are further analyzed for real-time demand ratio profiling across routes and stops, visualized over 7-day intervals. The process begins with secure dataset upload via an admin panel, followed by data preprocessing, sparse vector transformation, model training, and accuracy evaluation, with metrics stored in a detection_accuracy table. Predictions are stored in a dedicated prediction table, and demand ratios are tracked in the detection_ratio table. The system also offers dynamic visualization features such as stop-wise demand plots, model performance charts, and XLS export functionalities. Deployed within a modular Django framework, the platform includes user authentication, URL routing, and a dashboard-based GUI where users interact through pages like Login, Register, Dashboard, and PredictDemand. Backend logic is managed using Django's MVC architecture, where views.py coordinates prediction workflows and templates render results in real time. Data interactions are efficiently handled via Django ORM, linking user accounts, route data, and predictions to respective tables in an SQLite or MySQL database. This scalable, integrated system provides transport administrators with an intelligent, real-time dashboard for demand monitoring, resource planning, and operational optimization, representing a novel contribution to transit prediction technology.

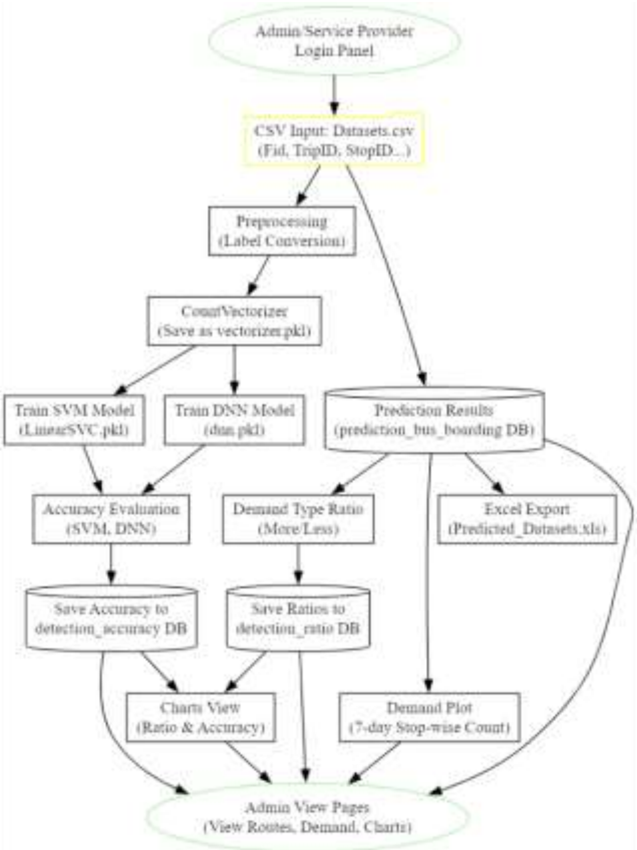

Fig. 2: Proposed System Architecture for Service Provider module.

4. RESULTS

Figure 3 shows the homepage of the bus demand prediction application, accessible at the local address http://127.0.0.1:8000. The page is centered around a research project titled "Predicting Hourly Boarding Demand of Bus Passengers Using Imbalanced Records from Smart Cards: A Deep Learning Approach," indicating the system's focus on leveraging deep learning for bus boarding demand prediction. It features navigation links for "Home," "Remote User," and "Service Provider," suggesting a role-based access structure where different user types (remote users and service providers) can access specific functionalities. The homepage serves as the entry point, providing an overview of the system's purpose and directing users to relevant sections based on their roles.

Fig. 3: Home Page of the Bus Demand.

Figure 4 depicts the login interface for service providers, designed to authenticate administrative users. The page likely includes fields for entering a username and password, as implied by the serviceproviderlogin view in the code, which checks for credentials (username="Admin", password="Admin"). This interface restricts access to sensitive administrative features, such as viewing remote users or training models, ensuring only authorized service providers can proceed to the dashboard.

Fig. 4: Service Provider Login.

Figure 5 illustrates the service provider's view of remote user details after successful login, as implemented in the View_Remote_Users view. The page displays a table titled "VIEW ALL REMOTE USERS" with columns for USER NAME, EMAIL, Gender, Address, Mob No, Country, State, and City. A sample record shows a user with the details: USER NAME: prashanth, EMAIL: guntaprashanthkumar@gmail.com, Gender: Male, Address: 1-somanpalli, Mob No: 08466868727, Country: India, State: Telangana, City: anthergoan. This view allows service providers to monitor registered users, supporting administrative oversight.

Fig. 5: Service Provider Login Page with Remote User Details.

Figure 6 shows the interface for adding new bus route details, corresponding to the create_route view. The form includes fields for Fid, Trip ID, Route ID, Stop ID, Start Name, Stop Name, Week Beginning, and Number of Boardings, with a "Submit" button to save the data. This functionality enables service providers to input route information, which is stored in the route model with startName and StopName converted to lowercase for consistency, supporting the system's data management for bus routes.

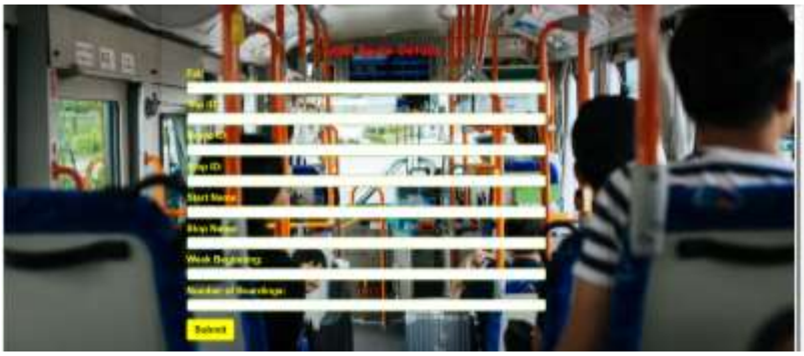

Fig. 6: Add Route Details.

Figure 7 presents the performance of trained machine learning models, as implemented in the train_model view. The table displays two models with their accuracy metrics: Model Type: SVM with

Accuracy: 59.97214285714285% and Model Type: DNN with Accuracy: 92.86928571428572%. These values reflect the accuracy of the LinearSVC and MLPClassifier models trained on the Datasets.csv file, with the DNN model showing significantly higher performance, indicating its effectiveness in predicting bus boarding demand.

Fig. 7: Model Training Performance.

Figure 8 illustrates a graphical comparison of model accuracies, likely generated by the charts1 or likeschart view. The graph visualizes the accuracy data from Figure 9.5, comparing SVM (59.97214285714285%) and DNN (92.86928571428572%). This visualization, possibly using Chart.js or Seaborn, helps service providers intuitively assess the relative performance of the two models, highlighting the DNN's superior accuracy.

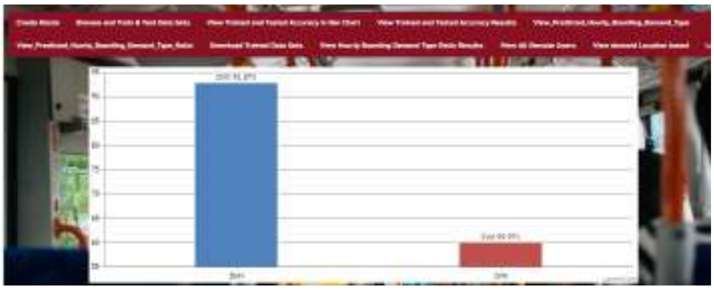

Fig. 8: Accuracy Comparison Graph.

Figure 9 displays a table of predicted hourly boarding demand, as implemented in the View_Predicted_Hourly_Boarding_Demand_Type view. The table includes columns for Fid, TripID, RouteID, StopID, StopName, WeekBeginning, NumberOfBoardings, and Prediction. Sample records include entries like Fid: 39.159.31.120-4.141.105.192-32060-43-75, TripID: TID_48351, RouteID: RID_686, StopID: SID_75943, StopName: Masab Tank, WeekBeginning: 08-09-2022 00:00, NumberOfBoardings: 30, Prediction: More Demand, and others with predictions of More Demand or Less Demand. The table shows a mix of 11 More Demand and 26 Less Demand predictions, reflecting the system's ability to classify bus stops based on demand.

Fid	TripID	RouteID	StopID	StopName	WeekBeginning	NumberOfBoardings	Prediction
39.159.31.120-4.141.105.192-32060-43-75	TID_48351	RID_686	SID_75943	Masab Tank	08-09-2022 00:00	30	More Demand
4.233.116.124-188.104.3.143-26860-71-48	TID_45665	RID_612	SID_72706	Dilsukhnagar	14-07-2022 00:00	39	More Demand
137.94.225.182-196.140.243.236-58379-24-16	TID_41587	RID_613	SID_70343	Nampally	16-03-2023 00:00	26	Less Demand
178.188.169.134-128.223.222.226-37523-54-30	TID_41605	RID_712	SID_77618	Mehdipatnam	16-03-2023 00:00	88	More Demand
31.134.83.123-170.237.77.40-15878-14-4	TID_45051	RID_620	SID_73621	Ameerpet	16-02-2023 00:00	47	Less Demand
126.99.52.49-206.152.245.37-55116-30-29	TID_41589	RID_615	SID_75260	Koti	08-06-2023 00:00	72	More Demand
81.161.107.146-77.59.205.160-43216-11-78	TID_43203	RID_794	SID_72229	Kukatpally	01-09-2022 00:00	10	More Demand
167.93.130.79-182.217.6.136-17510-56-71	TID_47689	RID_673	SID_76574	Nampally	05-01-2023 00:00	36	Less Demand
93.41.224.57-204.65.134.129-37162-47-24	TID_41292	RID_740	SID_76974	Dilsukhnagar	25-08-2022 00:00	66	Less Demand
60.115.111.132-79.244.160.40-15994-46-96	TID_41631	RID_639	SID_70775	RTC X Roads	29-12-2022 00:00	10	More Demand
175.186.199.226-73.194.202.106-47457-55-88	TID_44869	RID_694	SID_79027	Dilsukhnagar	16-02-2023 00:00	92	More Demand
82.181.108.18-216.146.159.164-32120-58-85	TID_41704	RID_757	SID_76181	LB Nagar	22-12-2022 00:00	32	More Demand
78.233.106.26-69.226.213.30-13905-60-98	TID_46820	RID_700	SID_79048	Mehdipatnam	01-09-2022 00:00	35	Less Demand
73.226.168.104-24.0.235.56-12909-64-20	TID_43651	RID_664	SID_70107	Tarnaka	22-09-2022 00:00	56	Less Demand

Fig. 9: Prediction of Hourly Demand From Test Data.

Figure 10 presents a table of demand prediction ratios, as implemented in the View_Predicted_Hourly_Boarding_Demand_Type_Ratio view. The table lists Hourly Boarding Demand Type and Ratio, with values: Less Demand: 75.60975609756098% and More Demand: 24.390243902439025%. These ratios are calculated from the prediction_bus_boarding model, indicating that 75.61% of predictions are for low demand, while 24.39% are for high demand, providing insights into demand distribution.

Hourly Boarding Demand Type	Ratio
Less Demand	75.60975609756098
More Demand	24.390243902439025

Fig. 10: Predicting Hourly Boarding Demand Found Ratio Details.

Figure 11 shows a graphical representation of the demand ratios from Figure 9.8, likely generated by the charts view. The graph visualizes Less Demand (75.60975609756098%) and More Demand (24.390243902439025%), possibly as a bar or pie chart using Chart.js or Seaborn. This visualization aids service providers in understanding the proportion of high and low demand predictions across bus stops.

Fig. 11: Predicting Hourly Boarding Demand Found Ratio Graph.

Figure 12 appears to duplicate the functionality of Figure 9.9, showing another graph of the demand ratios (Less Demand: 75.60975609756098%, More Demand: 24.390243902439025%). The repetition suggests either an error in the figure numbering or an alternative visualization style (e.g., different chart type or template) for the same data, reinforcing the system's focus on demand distribution analysis.

Fig. 12: Predicting Hourly Boarding Demand Found Ratio Graph.

Figure 13 depicts the remote user login interface, corresponding to the login view. The screen includes fields for Enter Username and Enter Password, allowing remote users to authenticate using credentials stored in ClientRegister_Model. Upon successful login, users are redirected to their profile page, ensuring secure access to user-specific functionalities like demand prediction.

Fig. 13: Remote User Login Screen.

Figure 14 shows the profile page for a remote user, as implemented in the ViewYourProfile view. The table displays user details: Username: vamshi1234, Email Id: vamshi.namani@gmail.com, Mobile Number: 09100110505, Gender: Male, Address: 184, hyderabad, Country: India, State: Andhra Pradesh, City: karimnagar. This page allows authenticated users to view their stored information, supporting user account management.

Fig. 14: Remote User Profile.

Figure 15 illustrates the input form for predicting hourly boarding demand, as part of the Predict_Hourly_Boarding_Demand_Type view. The form includes fields for selecting Start and Stop Names from the route model, enabling remote users to choose a specific bus route for prediction. This interface facilitates user interaction with the prediction system, leveraging the pre-trained ensemble model.

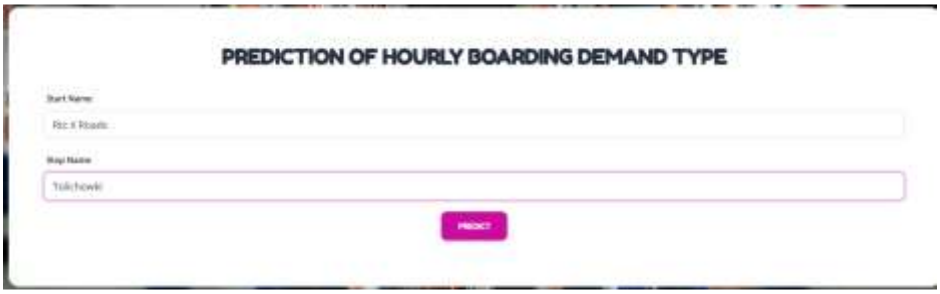

Fig. 15: Input to Prediction of Hourly Boarding Demand Type.

Figure 16 shows the output of a demand prediction, displaying "More Demand" for a selected route, as implemented in the Predict_Hourly_Boarding_Demand_Type view.

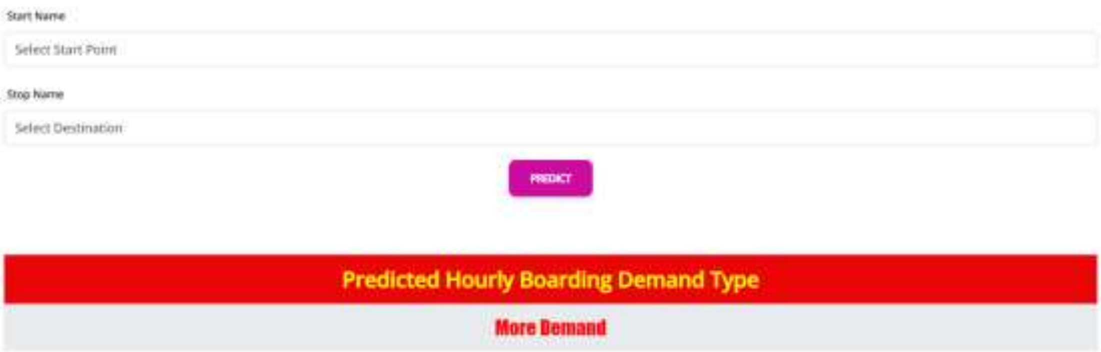

Fig. 16: Prediction of Hourly Boarding Demand Type as More Demand.

The prediction is based on the ensemble model's output (1 for "More Demand") for the selected route's Fid, with a 5-minute cooldown enforced to prevent overuse. This result informs users of high demand at the chosen bus stop, aiding in travel planning.

5. CONCLUSION

The presented Django-based web application provides a robust and scalable framework for predicting and analyzing hourly bus boarding demand, leveraging Support Vector Machines (SVM) and Deep Neural Networks (DNN) to classify stops into "More Demand" and "Less Demand" categories. The system integrates essential functionalities such as data preprocessing, model training, prediction, visualization, and result export, all accessible through an intuitive interface for service providers and remote users. Its modular backend ensures maintainability, while persistent model storage via Joblib reduces the need for retraining, optimizing resource usage. Looking ahead, the system holds strong potential for future enhancements, particularly through the incorporation of real-time contextual data like traffic conditions, weather, and public events to improve predictive accuracy. Additionally, advancing the deep learning architecture and extending the system for multimodal transport demand forecasting can significantly increase its applicability and impact in smart city transportation planning.

REFERENCES

[1] Gkiotsalitis, K.; Cats, O. Reliable frequency determination: Incorporating information on service uncertainty when setting dispatching headways. Transp. Res. Part C Emerg. Technol. 2018, 88, 187–207.

[2] Hadas, Y.; Shnaiderman, M. Public-transit frequency setting using minimum-cost approach with stochastic demand and travel time. Transp. Res. Part B Methodol. 2012, 46, 1068–1084.

[3] Eranki, A. A Model to Create Bus Timetables to Attain Maximum Synchronization Considering Waiting Times at Transfer Stops. Master's Thesis, Department of Industrial and Management Systems Engineering University of South Florida, Tampa, FL, USA, 2004. Available online: http://scholarcommons.usf.edu/etd/1025 (accessed on 15 February 2025).

[4] Gorev, A.; Popova, O.; Solodkij, A. Demand-responsive transit systems in areas with low transport demand of "smart city". Transp. Res. Procedia 2020, 50, 160–166.

[5] Berrebi, S.J.; Watkins, K.E.; Laval, J.A. A real-time bus dispatching policy to minimize passenger wait on a high frequency route. Transp. Res. Part B Methodol. 2015, 81, 377–389.

[6] Gkiotsalitis, K.; Alesiani, F. Robust timetable optimization for bus lines subject to resource and regulatory constraints. Transp. Res. Part E Logist. Transp. Rev. 2019, 128, 30–51.

[7] Yao, E.; Liu, T.; Lu, T.; Yang, Y. Optimization of electric vehicle scheduling with multiple vehicle types in public transport. Sustain. Cities Soc. 2020, 52, 101862.

[8] Van Lieshout, R.N.; Bouman, P.C.; van den Akker, M.; Huisman, D. A self-organizing policy for vehicle dispatching in public transit systems with multiple lines. Transp. Res. Part B Methodol. 2021, 152, 46–64.

[9] Taylor, S.J.; Letham, B. Forecasting at Scale. Am. Stat. 2018, 72, 37–45.

[10] Sherstinsky, A. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Phys. D Nonlinear Phenom. 2020, 404, 132306.