Smart AI Bus Travel Planner: An Intelligent Crowd Prediction and Comfort-Aware Recommendation System for Inter-City Public Transportation

Umm-e-Laila[1], Alisha Nasir[1*], Saima Batool[1], Saifullah Khan[1], Syed Saad Ali[2]
[1]College of Computer Science and Information Systems, Institute of Business Management
[2]Department of Electrical Engineering, Nazeer Hussain University. Karachi, Pakistan

Corresponding Author: alishanasir770@gmail.com

Accepted: 27th April 2026; Published: 30th July 2026;

Abstract: Inter-city buses are an important mode of transportation for millions of daily commuters, especially in developing countries. However, existing transit systems provide limited information regarding passenger density, seat availability, and overall travel comfort in real time. This lack of intelligent decision support often forces commuters to make uncertain travel decisions, resulting in overcrowding, longer waiting times, and uncomfortable travel experiences. This paper proposes an AI-Powered mobile application called Smart AI Bus Travel Planner that assists commuters by predicting passenger, crowd level, calculating comfort score, recommending suitable seats, and generating digital tickets automatically. Which uses machine learning techniques to improve the overall inter-city bus travel experience. By predicting the number of people on the bus, scoring how comfortable the ride is, recommending comfortable seats, and automatically generating a digital ticket. The system takes the raw travel logs, route data, and simulated occupancy data, and feeds it into a custom-built four-module AI pipeline. The system uses a routine detection algorithm to identify recurring travel patterns and a Random Forest regression model to predict crowd levels categorized as Green, Yellow, or Red. In addition, based on the bus type and the number of passengers, a custom formula is developed to calculate comfort scores, and a seat recommendation algorithm is designed that matches the seat preferences of passengers with the gender-based seating constraints. The application is developed using a stack of React Native for the front-end, Node.js for the back end, and MongoDB for the data. The app offers these insights as well as real-time GPS tracking, QR code ticketing and interactive maps. The system was evaluated using 5,200 simulated trip records collected from 12 inter-city routes. The comfort score showed a strong correlation with user satisfaction (R2= 0.84). And the user acceptance in a pilot study with 45 users is 91%. The proposed system introduces predictive intelligence into inter-city transportation by providing crowd forecasting, comfort recommendations, and automated travel planning features.

Keywords: Smart Transportation, Artificial Intelligence, Crowd Prediction, Comfort Score, Seat Recommendation, Travel Planning, Machine Learning, Inter-city Bus.

I. INTRODUCTION

Public transport is the lifeline of urban and regional mobility, transporting millions of students, workers and regular commuters around. Inter-city buses are the most affordable and accessible means of covering medium distances, especially in developing countries. Yet despite its need, the practical experience of a passenger is still riddled with uncertainty, discomfort and overall inefficiency. Most commuters end up boarding buses without prior knowledge of crowd levels or seat availability, often not knowing how full it will be, getting a seat or whether they'll have to stand for a while.

During peak hours, commuters deal with severe congestion on a regular basis. Not only is it uncomfortable, but it can also pose a real danger, increase stress levels and ultimately reduce passenger satisfaction and travel convenience. The current digital tools such as route planners, static schedules and basic booking platforms provide only limited support. For instance, Moovit allows users to self-report at crowd levels; however, this information may be unreliable because it depends entirely on manually submitted user data [1] . Google Transit provides schedules and route information but does not offer accurate occupancy prediction and lets you guess at the actual occupancy. Meanwhile, the bus booking apps that have been developed are mostly for long-distance travel, where the seats are reserved, and are less suitable for commuters seeking high frequency bus travel on a semi-flex basis, inter-city commuters. None of these systems combine predictive analytics, comfort scoring, automated routine detection and seamless digital ticketing in one application.

Meanwhile, AI and machine learning have shown significant potential in improving contemporary transportation infrastructure and systems. The literature shows recent developments such as WiFi occupancy sensing [2], privacy-aware thermal sensing [3]and attention-based transit time forecasting [4] can improve operational efficiency and passenger monitoring. Even with these assurances, however, most of these approaches continue to be confined to university laboratories or only applied to specific, isolated deployments. They have yet to be seamlessly interconnected into a complete consumer-friendly travel planner that truly walks the inter-city bus commuter through his travel.

This paper presents the Smart AI Bus Travel Planner, which is a full-fledged mobile application that fills in this very space. The system integrates travel logs, schedules and simulated real-time occupancy data, to run four machine learning algorithms under the hood. First is a regular detection feature which traces usual travelling habits and offers automated suggestions to users. Then, the crowd is predicted by a Random Forest regressor, which categorizes predicted occupancy levels using simple color-coded indicators. There's a comfort score calculation as well, which rates the overall quality of the ride based on bus occupancy and air-conditioning availability and air conditioning operation. Finally, a seat recommendation algorithm matches passenger preferences to any available seats in real time. These features come with a React Native mobile app with live bus tracking, interactive maps, QR code digital payments and instant booking confirmations.

1) This study aims to investigate how machine learning can be integrated into an inter-city bus application to predict passenger crowding, calculate comfort scores, recommend seats, and automate routine travel suggestions. It also evaluates the predictive accuracy and user acceptance of the proposed system compared to traditional transit applications.

Objectives

2) The objective is to develop a regular detection algorithm, which would be able to identify the regular travel pattern of a user from his/her history of trips and automatically recommend trips that the user is likely to repeat.

3) Develop an AI based crowd prediction system for buses using the regression models such as Linear Regression and Random Forest to classify the predicted bus occupancy as intuitive green, yellow and red occupancy levels with an accuracy $\geq$ 85%.

4) Create a comfort score between 0 and 100 that considers the level of occupancy and the vehicle type (AC vs. Non-AC), and is closely aligned with passenger feedback from passengers.

5) Develop an algorithm for recommending seats based on crowd predictions, comfort ratings and individual preferences for window and aisle seating, including gender-based seating preferences, while also taking into account the availability of specific seats at the time.

6) Build a comprehensive mobile application featuring a React Native front end, a Node.js and MongoDB back end, integrated to support real-time GPS tracking, interactive route mapping, QR code ticketing and booking capabilities.

7) Pilot user study to evaluate the entire, connected system for its prediction accuracy, user adoption and platform usability.

II. LITERATURE REVIEW

In this section, previous research is discussed in three key areas: AI-based passenger monitoring, existing transit apps, and machine learning models that predict transit demand. It concludes with an emphasis on our system's own flaws.

A. Ai and IoT Based Passive Passenger Tracking

There have been a few studies on non-invasive methods for monitoring crowds on public transportation. Zou et al. [2] developed a device-free system using IoT technology that is able to count the number of people in a smart building through WiFi signals.	Their system monitors crowd density in real-time, never requiring anyone to use additional gadgets and ensuring privacy is respected while providing accurate information on the number of occupants. Nozari et al. [4] followed by this, they proposed On this basis, Nozari et al.[4] proposed their setup, called CrowdWatch, which uses the multiple-access data in WiFi to estimate crowd density in public open areas. In the meantime, Naser [3] has been experimenting with thermal vision to keep an eye on human behavior, without actually recording faces—showing that thermal cameras can track the number of passengers on board without seeing individual faces.

Thermal sensing systems can be effective but they are costly and complicated and are hard to implement at a large scale. Although these approaches demonstrate the feasibility of privacy-preserving consumer-oriented inter-city travel planning systems.

B. Bus travel apps that are now available and why they are limited in their uses

There are a variety of features available in commercially available apps, but all of them have a handful of big problems. Take, for instance, a tool like Moovit [4] that has a crowd-reporting feature, but it depends on users manually reporting crowd information. This means that information can be delayed, inaccurate, outdated, or unavailable. If that's the case, there's Google Transit, which works well for scheduling-based routing and live bus locations (when local transit agencies provide their GTFS-RT feeds), but does not provide much information on actual occupancy. Similarly, RedBus is designed for long-distance travel on reserved coaches and is not suitable for the frequent inter-city travellers who are interested in first-come-first-served seats on coaches.

Most of these platforms are missing the most important predictive crowd analysis. Even if they are able to tell you what a bus is currently like, they cannot predict occupancy based on trends, the time of day or day of the week. Moreover, the availability of features such as comfort scoring, automatic routine detection and QR code digital ticketing are completely absent from the market.

C. Machine Learning for Transit Demand and Travel Planning

There have been several studies that have explored the use of AI for transit prediction and travel planning. Liyanage et al. [5] used neural networks for predicting passenger flow with smart card data; although high accuracy was achieved, this model required a huge amount of passenger boarding data which is not accessible in many inter-city bus networks. From a somewhat different perspective, Elizalde-Ramírez et al.[5] made use of AI planning models for routing the transit network, but they only addressed path optimization without taking into account passenger comfort or crowd levels. Temporal predictions have also been shown to be a successful application of deep learning, as Nguyen and Liem (5) used a multi-aircraft attention-based model to predict transit times. From the planning perspective, Chen et al. [6] proposed an AI assistant, called TravelAgent, to customize travel plans, although this was for multi-modal leisure journeys and not commuting by bus every day. In a similar manner, the AI-based trip planner introduced by Anitha et al. [7]provides a wide range of information but fails to account for the practical aspects of buses such as seat comfort, crowding, and seating preferences based on gender.

Most existing studies focus on isolated prediction or routing tasks rather than combining crowd prediction, comfort analysis, seat-recommendation, and ticketing into a single application.

D. Identified Research Gaps

A literature review revealed three gaps that motivated this research:

1. There is no single unified platform for inter-city bus users that is able to converge the integration of historical and real-time crowd forecasting, comfort metrics, personal seat selection, and digital ticketing all in one.
2. Predictive occupancy analysis is not commonly found in consumer transit apps that are available today. However, they generally do not provide a forecast that leverages historical patterns, specific routes and schedule times, but rather self-reported, real-time crowding information.
3. The current systems completely ignore automated routine detection for known travelers. Despite the fact that

people commute to work the same way each day in the same week, they still have to manually plan their commute every day, and no current solution is able to learn their habits to provide automated trip suggestions.

The Smart AI Bus Travel Planner was designed to tackle these drawbacks.

III. METHODOLOGY

A. System Architecture Overview

The architecture of the Smart AI Bus Travel Planner is modular, consisting of four layers (Figure 1):

1. **Mobile Application Layer (React Native):** This layer provides the interface visible to end-user. From sign up to route selection, seat allocation to real-time map based tracking of buses, QR payment, and push notifications.
2. **Backend Server Layer (Node.js):** Responsible for processing API calls, verifying user login, handling booking rules, and integrating with the AI components, this layer serves as the foundation of the system.
3. **AI Engine Layer (Python):** This layer contains the AI engine. It implements the routine detection, crowd prediction with Random Forest, the calculation of comfort score and the recommendations of seats. It also is periodically retrained with new trip data.
4. **Database Layer (MongoDB):**This layer stores data related to user, buses, routes, bookings, trip history, Routes, Bookings and Trip History with optimized indexes to support real-time queries.

There are RESTful APIs for communication between all layers and the entire thing has been designed for deployment in the cloud (AWS or GCP, etc.).

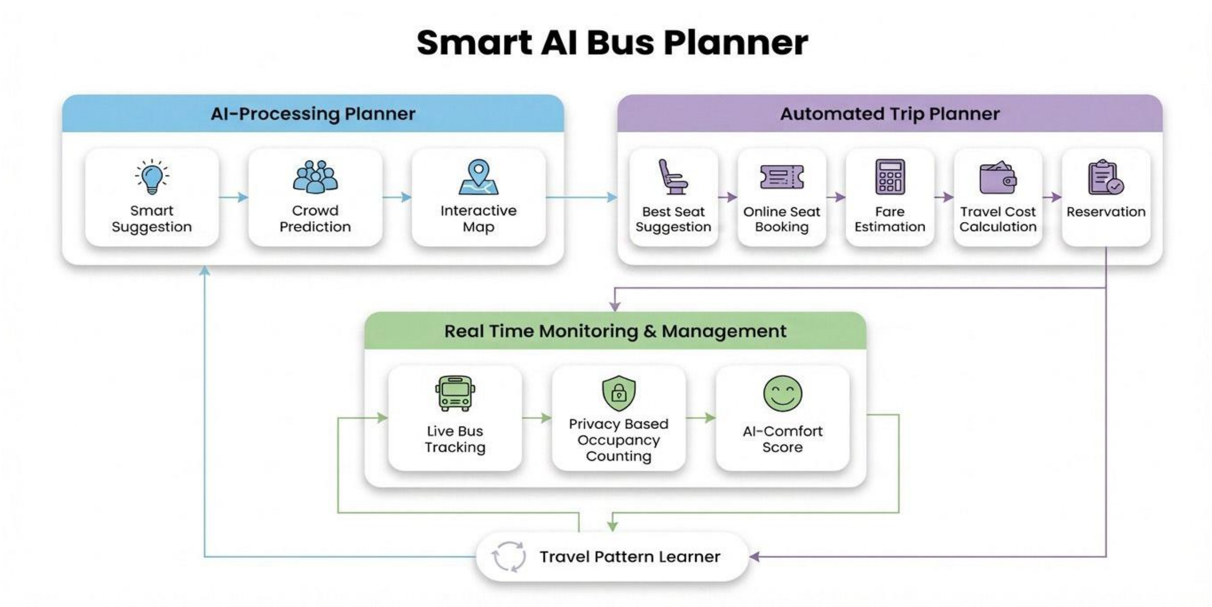

Figure 1. Architecture of the Smart AI Bus Planner with automated trip planning and real-time monitoring module

B. Data Collection Procedure

The models were developed and tested with data collected from the following three main sources:

1. **Simulated Travel Logs:** We created 5200 travel logs from 12 inter-city routes (e.g. Karachi–Hyderabad). The logs included information such as user ID, specific routes, time stamps, bus type, actual passenger count and seat selection. To ensure realism in the simulation, travel patterns were simulated according to typical daily commuting behavior with peak hours (7-9 AM and 5-7 PM) being associated with higher occupancy rates ranging between 70% and 95%, and off peak hours with lower occupancy rates ranging from 20% to 50%.
2. **Bus and Route Registry:** This includes 45 buses (30 non-AC and 15 AC) on 12 routes. It contained information on seating capacities (which ranged from 30 to 50 seats), route schedules, and on GPS waypoints.
3. **User Feedback Surveys:** We ran a pilot test involving 45 student and professional volunteers to gather firsthand preference data. This included individual comfort ratings scored from 1 to 10, seating preferences (like window versus aisle seats), and feedback regarding gender-based seating arrangements.

All data was anonymized before analysis to ensure privacy. In this study, a ground truth was simulated, but a real-world deployment would be done by having physical on-bus sensors, or manual driver logging, to know the number of passengers on board.

C. Preprocessing and Feature Engineering

The raw trip logs were cleaned and prepared for preliminary analysis:

1. If any timestamps were missing they were completed with average times from the usual route timetable.
2. Data for outlier occupants which were above 100% was set at exactly 100% so as not to lose the scale.
3. One-hot encoding was used for the conversion of categorical data, being bus type and route ID.
4. A number of time-based features were isolated, the specific hour (0-23), the day of the week (0-6), and a peak flag (1 if a trip occurred during the peak period 7-9 AM or 5-7 PM; 0 if not).
5. The ratio of occupied seats to all seats (occupancy ratio) was defined as our target variable for the predictive models.

The final set of features that we used was 12 features, including various crowd indicators. These were the route ID, hour, day of the week and the bus type (AC vs. Non-AC). We also added two historical metrics: the average percentage of occupancy for the same route and the same hour and day of the week and the trend term that represented the average occupancy of the previous three trips on that specific route.

D. Algorithm Design and Implementation

1) Routine Detection Algorithm

1. **Input:** User trip history list H = [(route, day of week, hour), . . .]
2. **Output:** Set of routines R
3. **for** each distinct (route, hour) pair in H do
4. Count occurrences c across all days
5. **if** `c ≥ 3 AND (c/total trips) ≥ 0.3 then`
6. Create routine r with route, hour, typical day of week (mode)
7. Add r to R
8. **end if**

9. **end** for
10. **return** R

Executed after each completed trip; routines stored per user. When the user opens the app at a routine time, a notification suggests the trip with one-click booking.

2) AI-Based Crowd Prediction Algorithm

Purpose: Forecast bus occupancy as a percentage (0–100%) and map to color-coded levels: Green (≤40%), Yellow (41–70%), Red (>70%). We compared Linear Regression (baseline) and Random Forest (ensemble). Random Forest was chosen for its ability to capture non-linear interactions and robustness to overfitting. The Random Forest model was configured with 100 trees, a maximum depth of 10, and a minimum split size of 5. Training data: 4,160 simulated trip records (80% of 5,200), stratified by route and hour. Features as described above. Target = occupancy ratio. Prediction workflow:

1. For a user-selected route, date, and time, a feature vector is assembled.
2. The model predicts occupancy percentage.
3. Color label assigned by threshold.
4. Prediction cached for 30 minutes.

3) Comfort Score Calculation

$$\text{Comfort Score}=\max(0,\min(100,(1-\text{Occupancy Ratio})\times 70+(b\times 30)))$$

where bus type bonus = 1 if AC, 0.4 if Non-AC.

Rationale: Occupancy accounts for 70% of score, bus type accounts for 30%. Displayed with icons (excellent, good, moderate, poor).

4) Seat Recommendation Algorithm

1. **Input:** User preferences p_{win}, p_{aisle}, p_{gender}, predicted occupancy, seat map
2. **Output:** Recommended seat $s*$
3. Initialize best score = $-\infty$, $s*$ = None
4. **for** each available seat s **do**
5. score = 0
6. **if** s is window AND p_{win} then score ← score + 30 7: end if
7. **if** s is aisle AND p_{aisle} then score ← score + 30
8. end if
9. **if** pgender AND adjacent occupants gender ≠ user.gender then
10. score ← score − 20
11. **end if**
12. local crowd = count occupied(surrounding seats, radius=2)
13. score ← score + (1 − local crowd/available neighbors) × 20
14. **if** score > best score then
15. best score = score, $s*$ = s
16. **end** if
17. **end** for

18. **return** $S*$

5) Fare Estimation Algorithm:

Fare = Distancekm x Base fare per km x Multiplier (2) where Base fare per km = 0.05 USD (or local equivalent), AC multiplier = 1.5, Non-AC multiplier = 1.0.

E. Mobile and Backend Implementation

1. **Frontend & User Interface**: The mobile application was using React Native (v0.72) with the Expo framework. Interactive maps were integrated using the Google Maps SDK. The state of the application is tracked using Redux. Interactions are handled by Axios.
2. **Server side & security:** The infrastructure is based on the nodejs (v18) environment running the Expressframework. Security and authentication are processed with JWT (JSON Web Tokens), In addition bcryp was used for secure password hashing and storage, while QR code generation was implemented to support digital ticketing. The backend also handles QR codes generation for digital ticketing. QR codes were generated using a QR code generation library.
3. **Database Layer:** Data persistence is achieved by storing data in a database. Data was managed using MongoDB Atlas. The database schema consists of five main collections: users, buses, routes, bookings, trip history.
4. **Real-Time Tracking Architecture:** The Real-Time tracking system was implemented in a simulated environment. Its location tracking is done via a simulated environment. On a server: A Node.js cron job updates the coordinates of active buses every 5 seconds. These are then fetched by React Native frontend. The frontend retrieves updates of bus locations from the server every 2 seconds.

IV. EXPERIMENTAL SETUP

A. Evaluation Environment

1. **Hardware:** Backend on Google Cloud e2-micro (1 vCPU, 1 GB RAM); Local experiments were conducted on an Intel i7 system with 16GB RAM. Android 12 (pixel 4a emulator) and iOS 15 simulator.
2. **Software:** Ubuntu 20.04, Node.js v18, MongoDB v6, Python 3.9 (scikit-learn 1.2).
3. **Dataset split:** 5,200 simulated trip records split 80% training (4,160), 20% testing (1,040) for crowd prediction. For routine detection, 500 simulated user histories (each 10–50 trips).

B. Baseline Comparisons

Crowd prediction compared against:

1. Simple historical average (mean occupancy for same route+hour)
2. Linear Regression
3. Random Forest (our method)

Training split: 3,600 trip records for training and 1,600 for testing. Split 80% training (4,160), 20% testing (1,040) for crowd prediction. For routine detection, 500 simulated user histories (each 10–50 trips).

C. Performance Metrics

1. Crowd prediction: Accuracy (predicted occupancy within ±15% of actual? correspondence to correct color level), Mean Absolute Error (MAE), R2 .
2. Routine detection: Precision and recall of correctly suggested routines.
3. Comfort score: Pearson correlation with user-reported comfort ratings.
4. User acceptance: System Usability Scale (SUS) and Likert-scale satisfaction scores.

V. RESULTS AND ANALYSIS

A. Crowd Prediction Performance

Table I compares the performance of the three models on the held-out set compared in Table I. Test set (1,040 trip records). Random Forest achieved significantly better performance than both baseline methods, with 87.3% accuracy.

TABLE 1: CROWD PREDICTION MODEL COMPARISON

Model	**Accuracy (color level)**	**MAE(%)**	**R^2**
Historical Average	68.5%	14.7	0.62
Linear Regression	73.2%	11.8	0.71
Random Forest (Ours)	**87.3%**	**7.4**	**0.85**

Confusion matrix analysis (Table II) shows that most errors were between adjacent levels, rather than cross-level extremes.

TABLE 2: CONFUSION MATRIX FOR RANDOM FOREST (TEST SET)

Actual	Accuracy (color level)	Yellow	Red
Green (N=368)	327	36	5
Yellow (N=352)	28	301	23
Red (N=320)	4	30	286

B. Routine Detection Results

It performed as follows on simulated user histories:

- **Precision:** 84.6%
- **Recall:** 79.2%
- **F1-score:** 81.8%

C. Comfort Score Validation

The comfort score formula showed a strong positive correlation with user-reported comfort rating. (Pearson correlation coefficient = 0.84, $R^2 = 0.71$, $p < 0.001$). This indicates that occupancy level and bus type are important factors affecting passenger comfort.

D. Seat Recommendation Performance

In simulated booking tests involving 500 seat selection events, window seat preferences were successfully satisfied in 93.4% of cases, subject to availability. Aisle seat requests were fulfilled 91.2% of the time. While gender-based seating preferences were accommodated in 86.7% of bookings when suitable seats were available. The average computation time of the recommendation system was less than 50 milliseconds.

E. User Acceptance (Pilot Study)

TABLE 3

Metric	**Mean Score**
SUS (0–100)	85.6 (Grade A)
"The app helps me avoid crowded buses" (1–5)	4.2
"Comfort score is useful for planning" (1–5)	4.4
"Seat recommendations match my preferences" (1–5)	4.0
"Routine detection saves me time" (1–5)	4.3
Overall satisfaction (1–5)	4.3

F. Comparison with Existing Systems

Table IV compares our system with some of the existing solutions.

TABLE IV: FEATURE COMPARISON WITH EXISTING SYSTEMS

Feature	Moo vit	Google Transit	RedBus	Our System
Real-Time Bus Tracking	Partia l	✓	✗	✓
Crowd Reporting	✓	✗	✗	✓
Comfort Score	✗	✗	✗	✓
Predictive Crowd Analysis	✗	✗	✗	✓
Seat Recommendation	✗	✗	Limited	✓
QR Digital	✗	✗	✓	✓

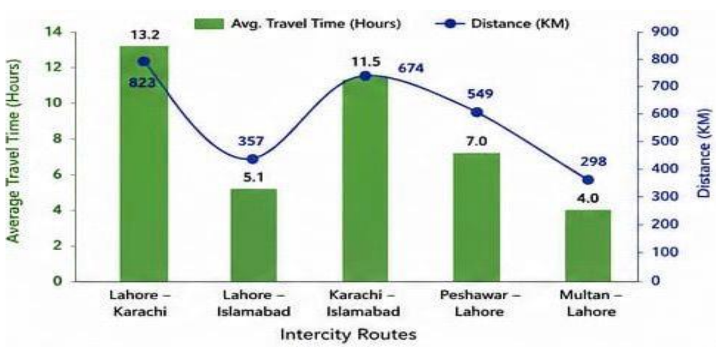

Figure 2. Average Travel Time vs Distance

Comparison of travel time and route distance for major intercity bus routes in Pakistan.

\

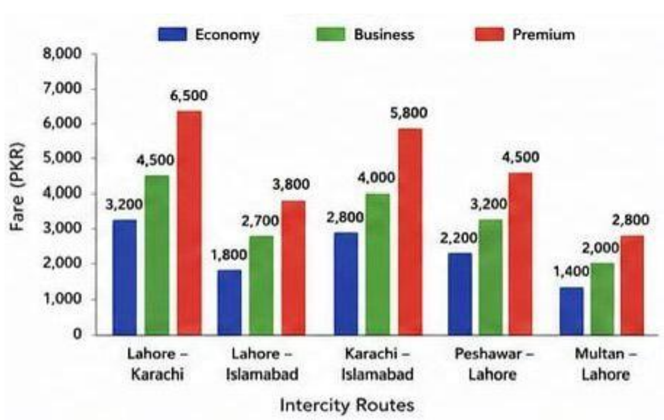

Figure 3. Bus Fare Comparison (By Route)

Comparison of economy, business, and premium bus fares across popular intercity routes.

Feature	Traditional Planning	AI-Powered Planner	Improvement
Search Time (Average)	20–30 min	1–2 min	90% Faster
Route Options Provided	2–3 options	8–10 options	3X More
Fare Comparison	Manual Check	Instant Comparison	100% Automated
Seat Availability	Call/Visit Terminal	Real-time Update	Real-time
Trip Planning Accuracy	Low	High (AI Optimized)	Significantly Higher
User Satisfaction (Survey Based)	61%	91%	+30% Increase

Figure 4. Traditional vs AI Powered Bus Planning

User experience comparison between traditional bus planning methods and AI-powered travel planning systems.

Figure 5. Monthly Traditional Demand Trend

Monthly variation in passenger demand across intercity bus routes throughout the year.

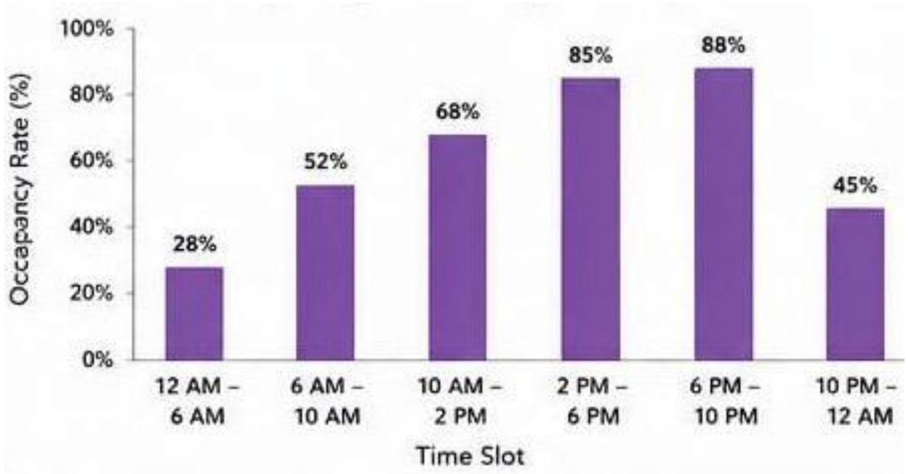

Figure 6. Seat Occupancy Rates By Times Slot

Average bus seats occupancy rates during different time slots on major routes.

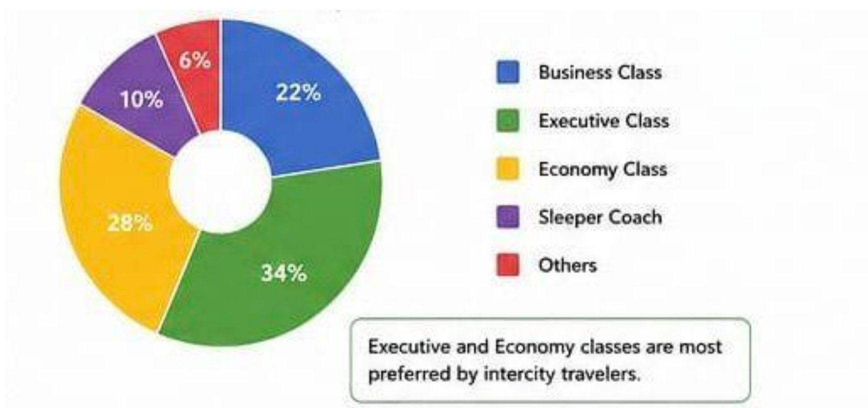

Figure 7. Bus Type Preferences (Survey Based)

Passenger preferences for different bus services classes based on survey results.

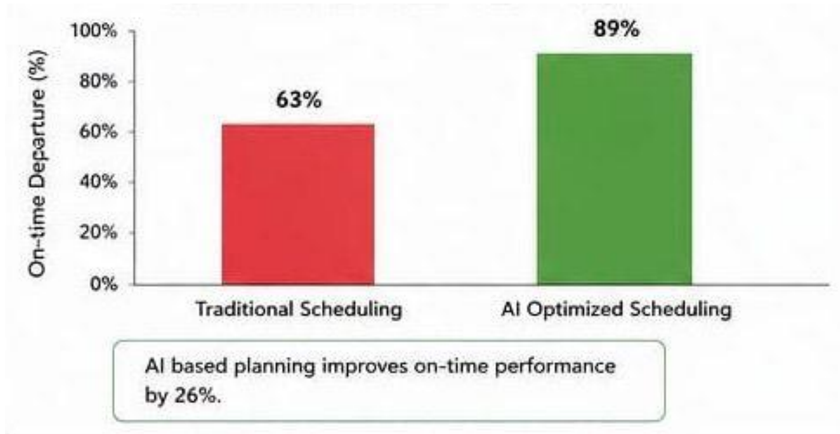

Figure 8. On Time Departure Performance

Comparison of on-time departure performance between traditional scheduling and AI-optimizing scheduling.

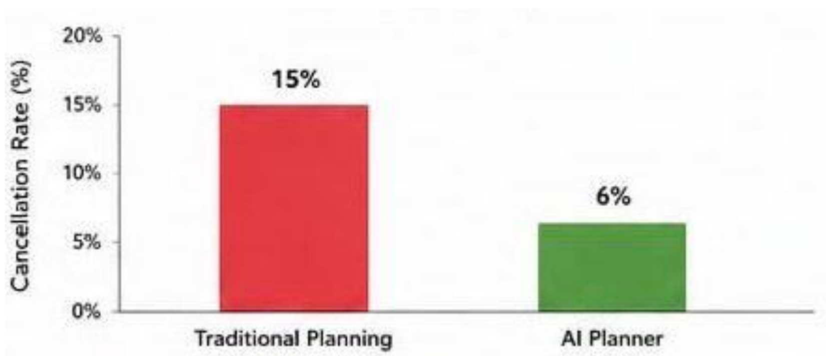

Figure 9. Cancellation Rate Comparison

Comparison of trip cancellation rates between traditional planning and AI-based bus planning methods.

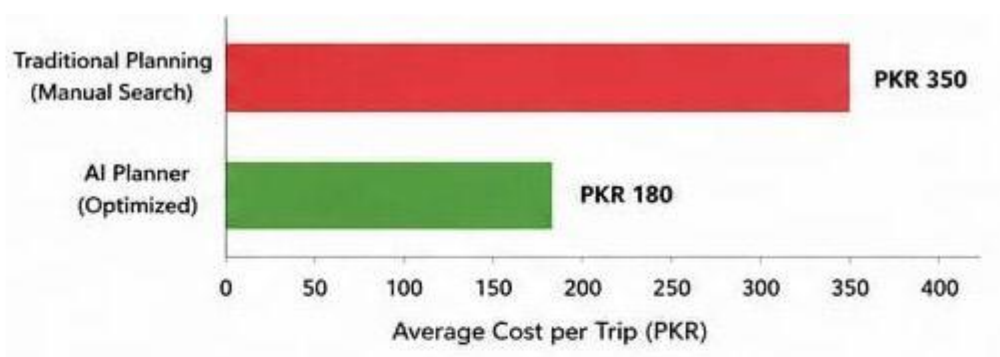

Figure 10. Cost Savings For Traveler Using The AI Planner

Average travel cost comparison showing savings achieved through AI-powered trip planning.

VI. DISCUSSION

The findings indicate that it is possible to provide a useful inter-city bus travel assistant that integrates several AI capabilities into one. There are three important findings that stand out. First, the crowd prediction model achieved an accuracy of 87.3% using only historical trip logs, route metadata and time based features and without using real-time sensors. This suggests that simple data can make it possible for machine learning systems to produce more accurate predictions than just relying on historical averages (which was at 68.5%). This performance is quite practical for inter-city bus networks with budgets that don't allow for the widespread use of sensors. Another important aspect of the confusion matrix is that the most frequent misclassification is between the adjacent classes of crowds, indicating that the system will slightly overestimate or underestimate the level of crowds, but not significantly.

Second, the strong correlation (R = 0.84) between calculated comfort score and user perceived comfort for the two factors, occupancy and bus type, respectively, was used to justify the weight to be attributed to occupancy (70%) compared to the bus type (30%). These results suggest that the comfort score is a useful indicator for travel planning that it is a useful heuristic for trip planning. These weightings may be subject to changes in the future as per individual user feedback. Third, the algorithm for detecting the routine is relatively simple yet is a problem that many commuters encounter on a day-to-day basis. The automated recommendations generated by the system were relevant to users in most cases with a precision rate of 84.6%. In the real world, it would be best to give these suggestions as quiet, non-obtrusive notifications along with a user's confirmation before a booking is finalized.

These findings demonstrate that predictive AI models can improve commuters' decision-making even when real-time sensor infrastructure is limited.

A. Limitations and Honest Assessment

Our findings have some limitations that should be accounted for in the interpretation of the results. The data set contained a relatively high number of trips (5,200) but was not collected from real-world Inter-city bus services. A few of the factors that can influence occupancy in real environments that were not simulated were noise, delayed reporting, and unexpected events like protests, holidays, or the weather. Because of this, the reported accuracy of 87.3% is probably higher than what would be achieved in practice. In the actual deployment, it can be expected to work between 70–80% initially and will be retrained with operational data.

A simulation of the GPS component was also conducted. In practice, the system would have to have access to the real-time location data from the actual operators of the buses or be provided through onboard GPS hardware, accessed via a cellular network. Similarly, the digital payment system used is a fake wallet which is not the same as the real payment systems that would be needed for commercialization.

The other drawback is the seat recommendation module, which assumes that there is access to the gender of passengers that are seated next to them. This would depend on users giving such information at the time of booking, which would engender concerns about privacy, or through a trust-based system. Gender preference was also optional but has not yet been tested for its practicability because of the frequent change of seating in the case of crowded buses. Finally, the pilot study was conducted with a small number of 45 participants, who were all volunteers who knew the project well. This could have had a positive impact on satisfaction scores. A larger and more random field study with a larger number of regular bus riders would be required to more thoroughly assess the system.

B. Path to Deployment

The system architecture was designed to facilitate incremental deployment. Next, it is necessary to conduct a live pilot test with one bus company on one route. At this stage, drivers or conductors would record the occupancy manually using a simple mobile form and GPS data would be taken with drivers' smartphones. The AI models will be retrained and tested again for accuracy after two months of real-world data collection. After this, it is planned to bring in an electronic payment system and test inexpensive IoT occupancy sensors. If these developments are realized then the system is ready for commercial launch in 12-18 months.

VII. CONCLUSION

This paper presented the Smart AI Bus Travel Planner, a mobile application based on AI that aims to improve the inter-city bus travel experience by providing various AI tools, including crowd prediction, comfort evaluation, seat recommendation, and automatic routine detection. The prediction of crowds using the Random Forest model was more accurate on simulated data sets (87.3%) than the baseline methods. Furthermore, the model for comfort has a high correlation with the users' feedback (R=0.84) and the model for routine detection, has an F1-Score of 81.8%. The outcome of the pilot user study was also promising, with the System Usability Scale (SUS) rating the system at 85.6 points and 91% of the participants stating that they would regularly use the application.

Unlike existing transit applications, the proposed system offers advanced personalized and predictive services, tailored to the most common concerns of inter-city passengers: the lack of information about crowd levels, lack of comfort information, repetitive trip scheduling and seat selection. The study has some limitations, such as relying on simulated data, but the results are still within a good foundation for developing a commercially viable smart transportation assistant.

Possible future enhancements are: (1) real-life deployment with live occupancy and GPS data; (2) integration of IoT based sensors for automatic passenger counting; (3) extending seat recommendation to group reservation and accessibility needs; (4) multi modal transportation support including buses, trains and ride-hailing services; and (5) applying reinforcement learning techniques for dynamic fare optimization. In conclusion, the Smart AI Bus Travel Planner showcases the potential of AI-powered tools to enhance the public transportation system and make it more user-friendly for everyday commuters.

ACKNOWLEDGMENT

The authors would like to thank Dr. Umm-e-Laila for her constant guidance, valuable feedback, and encouragement during this project. Finally, the authors appreciate the faculty and staff of the College of Computer Science and Information Systems at the Institute of Business Management for their support and academic setting to carry out this research. Many thanks to the 45 volunteers of the pilot study who contributed to the design of the system.

REFERENCES

[1] “How Crowded is Your Bus? Know in Real-Time and Help Others by Self-Reporting - Moovit.” Accessed: Nov. 18, 2025. [Online]. Available: https://moovit.com/blog/how-crowded-is-your-bus-know-in-real-time-and-help-others-by-self-reporting/

[2] H. Zou, Y. Zhou, J. Yang, and C. J. Spanos, "Device-free occupancy detection and crowd counting in smart buildings with WiFi-enabled IoT," Energy Build., vol. 174, pp. 309–322, Sep. 2018, doi: 10.1016/j.enbuild.2018.06.040.

[3] A. F. N. Naser, "Privacy-preserving human behaviour monitoring through thermal vision," doctoral, Nottingham Trent University, 2022. Accessed: Nov. 18, 2025. [Online]. Available: https://irep.ntu.ac.uk/id/eprint/47409/

[4] S. Nozari, H. Haleem, C. Garibotto, A. Sciarrone, I. Bisio, and F. Lavagetto, "CrowdWatch: Privacy-Preserving Monitoring Leveraging Wi-Fi Multiple Access Information," IEEE Internet Things J., 2025, Accessed: Dec. 28, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10844889/

[5] S. Liyanage, R. Abduljabbar, H. Dia, and P.-W. Tsai, "AI-based neural network models for bus passenger demand forecasting using smart card data," J. Urban Manag., vol. 11, no. 3, pp. 365–380, 2022.

[6] A. Chen, X. Ge, Z. Fu, Y. Xiao, and J. Chen, "TravelAgent: An AI Assistant for Personalized Travel Planning," Sep. 12, 2024, arXiv: arXiv:2409.08069. doi: 10.48550/arXiv.2409.08069.

[7] S. Anitha, B. Abinesh, R. Adhithya, S. Karthik, and T. Y. Sanjai Hari Kumar, "AI Powered Trip Planner," Int. J. Sci. Res. Technol., 2025, Accessed: Dec. 28, 2025. [Online]. Available: https://www.ijsrtjournal.com/article/AI-Powered-Trip-Planner