International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4532 - 4538
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0595

Smart Navigation System for Urban Route Optimization

Ms. R Gayathri[1], Siddarth R K[2], Rengesh PSR[3], Arun Sathyamurthy[4]
[1]Assistant Professor, Dept. of CSE, Sri Venkateswara College of Engineering, Chennai, Tamil Nadu, India
[2,3,4]UG Scholar, Dept. of CSE, Sri Venkateswara College of Engineering, Chennai, Tamil Nadu, 602117, India
Emails: *gayathri@svce.ac.in[1], siddhurk2005@gmail.com[2], rengesh725@gmail.com[3], sathyamurthy127@gmail.com[4]*

Abstract
Urban navigation has become increasingly complex due to rapid population growth, traffic congestion, road safety concerns, and climate-related disruptions such as flooding. This paper presents a Smart Navigation System for Urban Route Optimization, an AI-powered full-stack web application specifically engineered to address the multifaceted challenges of urban mobility in metropolitan environments. The system computes and presents four distinct route types—fastest, shortest, safest, and traffic-optimized—by leveraging real-time data integration, machine learning models, and graph-based pathfinding algorithms. Key components include accident hotspot prediction, dynamic flood risk assessment, multi-modal transportation planning encompassing road, metro, and bus networks, and real-time incident reporting. The frontend is built using React with TypeScript and Leaflet.js for interactive map visualization, while the backend employs FastAPI with OSMnx and NetworkX for road graph construction and route computation. Machine learning modules developed with scikit-learn and XGBoost enable congestion forecasting and risk scoring. Experimental evaluations demonstrate that the system significantly outperforms conventional navigation approaches in safety awareness, real-time responsiveness, and user-centric route selection.
Keywords: *Flood Risk; Machine Learning; Multi-Modal Transportation; Real-Time Traffic; Route Optimization; Urban Navigation*

1. Introduction
Urban transportation networks worldwide face an unprecedented convergence of challenges: exponentially increasing vehicular populations, aging road infrastructure, unpredictable weather disruptions, and growing public demand for safety and efficiency. Conventional navigation applications, while effective in providing basic point-to-point directions, fail to address the nuanced requirements of urban commuters who must navigate accident-prone corridors, flood-vulnerable zones, and dynamically shifting traffic conditions. The proposed Smart Navigation System for Urban Route Optimization represents a paradigm shift in navigation technology. Rather than optimizing solely for travel time or distance, the system adopts a holistic, safety-first methodology that simultaneously considers road safety metrics, real-time incident data, multi-modal transportation availability, and environmental risk factors. The system is architected as a full-stack web application, combining a modern React and TypeScript frontend with a high-performance FastAPI backend, supported by machine learning pipelines trained on historical accident, traffic, and environmental data. Urban centers in South India, particularly Chennai, present distinctive navigation challenges. The city's extensive road network, dense bus and metro infrastructure, and seasonal monsoon flooding create a complex operational environment that demands intelligent, context-aware navigation solutions. This research leverages OpenStreetMap data, Firebase real-time database integration, and web-scraped traffic intelligence to construct a comprehensive navigation ecosystem tailored to these urban realities. The primary contributions of this work are: (1) a multi-objective routing engine that generates four semantically distinct route types; (2) a real-time safety scoring framework combining machine learning predictions with live incident data; (3) an integrated flood risk visualization layer critical for monsoon-season navigation; (4) a seamless multi-modal trip planner connecting road, metro, and bus

networks; and (5) a scalable full-stack architecture suitable for deployment across diverse urban environments.

1.1. Motivation

Traffic accidents claim over 1.35 million lives annually worldwide, with urban areas accounting for a disproportionate share of fatalities. Simultaneously, floods—increasingly frequent due to climate change—render entire urban road networks impassable with little warning. Existing navigation solutions treat these as edge cases rather than primary design considerations. The motivation for this research stems from the recognition that navigation systems must evolve into proactive safety platforms that guide users away from danger, not merely toward their destination.

1.2. Objectives

The key objectives of this research are to: (a) develop a multi-route intelligent navigation engine that optimizes simultaneously for time, distance, safety, and traffic conditions; (b) integrate real-time data sources to provide dynamic, context-aware route recommendations; (c) incorporate machine learning models for accident hotspot prediction and traffic congestion forecasting; (d) build a comprehensive multi-modal transportation integration layer; and (e) deliver an intuitive, responsive user interface accessible across devices.

2. Literature Review

The field of intelligent transportation systems (ITS) has witnessed substantial growth over the past two decades. Dijkstra (1959) established the foundational graph-based shortest path algorithm that underpins modern navigation systems. Subsequent work by Hart, Nilsson, and Raphael (1968) introduced the A* heuristic search algorithm, significantly improving computational efficiency for large road networks. Wang et al. (2019) proposed a deep learning framework for real-time traffic speed prediction using spatial-temporal graph convolutional networks (ST-GCN), demonstrating superior performance over traditional statistical models. Their work established the viability of deep learning for traffic forecasting at city scale, a concept extended in this research through the application of XGBoost-based congestion prediction models calibrated on historical Chennai traffic patterns. Zhang and colleagues (2020) introduced the concept of safety-aware route planning, incorporating accident historical data into routing algorithms. Their multi-objective optimization framework balanced travel time against safety risk, demonstrating that safety-optimized routes incurred only marginal time penalties—typically 8 to 15 percent—while substantially reducing accident exposure. This finding motivates the safety route component of the proposed system. Flood-aware navigation has gained research attention following catastrophic urban flooding events. Pregnolato et al. (2017) modeled flood-induced disruption to urban road networks, quantifying the relationship between inundation depth and road capacity reduction. Their framework informs the flood risk layer integrated into the proposed system, which visualizes flood-prone zones with severity classifications derived from historical inundation data. Multi-modal urban navigation has been addressed by Bast et al. (2016) in their survey of public transit routing algorithms, highlighting the complexity of integrating schedule-based transit with dynamic road networks. The proposed system addresses this challenge through a unified routing interface that seamlessly transitions users between driving, metro, and bus transportation modes, abstracting the underlying algorithmic complexity. Recent work by Li et al. (2021) on graph neural networks for road network analysis and Lv et al. (2020) on transformer-based traffic prediction models represents the state of the art in data-driven transportation intelligence. The proposed system incorporates elements of these advances through its OSMnx-powered graph representation and scikit-learn ML pipeline, providing a practical implementation of research-grade methods in a production-ready application.

3. System Architecture

The Smart Navigation System is structured as a three-tier architecture comprising a React-based frontend presentation layer, a FastAPI backend service layer, and a Firebase real-time data layer. This separation of concerns enables independent scaling, maintenance, and enhancement of each tier, while the well-defined API boundaries ensure interoperability across components.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4532 - 4538
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0595

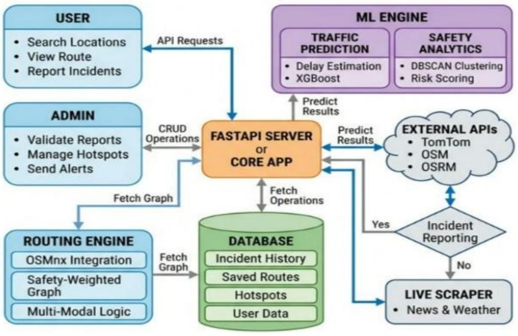

Figure 1 **Proposed Architecture Diagram**

3.1. Frontend Architecture

The frontend is developed using React 18 with TypeScript, providing strong type safety and improved developer productivity. The application utilizes Vite as the build toolchain, delivering fast hot module replacement during development and optimized production bundles. User interface components are implemented using Shadcn UI, a composable component library built on Radix UI primitives, ensuring accessibility compliance and visual consistency [1-6]. Interactive map functionality is powered by Leaflet.js, a lightweight open-source mapping library that supports multiple tile layer providers including standard street maps, satellite imagery, and hybrid views. TanStack Query manages server state, providing automatic caching, background refetching, and optimistic updates that contribute to a responsive user experience. React Router handles client-side navigation across the application's multiple dashboard views.

3.2. Backend Architecture

The backend is implemented using FastAPI, a modern Python web framework that leverages Python type hints for automatic API documentation generation and provides asynchronous request handling through ASGI. The server implements CORS middleware to enable secure cross-origin communication with the frontend application. Road network data is acquired and processed using OSMnx, a Python library for downloading and modeling street networks from OpenStreetMap. OSMnx constructs a directed weighted graph representation of the road network, where nodes represent intersections and edges represent road segments annotated with length, speed limit, road type, and travel time attributes. NetworkX provides the graph algorithm implementations for pathfinding, including Dijkstra's algorithm and A* search variants used for route computation. Firebase Admin SDK integration provides real-time database connectivity for storing and retrieving user-reported incidents, ML model outputs, and dynamic route metadata. Background graph loading, executed at server startup, pre-fetches and caches the road network graph to eliminate per-request graph construction overhead, ensuring sub-second route computation times for typical urban queries.

3.3. Machine Learning Pipeline

The machine learning pipeline encompasses three primary models: a congestion prediction model, an accident risk prediction model, and a composite route safety scoring algorithm. The congestion model employs gradient-boosted decision trees implemented via XGBoost, trained on historical hourly traffic volume data segmented by road segment, day of week, and time of day. The accident risk model uses a random forest classifier trained on accident records enriched with road geometry, weather, and time features, outputting a probability of accident occurrence for each road segment. Route safety scores are computed as weighted composites of accident risk, congestion level, road type safety ratings, presence of accident hotspots within a configurable radius, and active incident reports. This multi-factor safety algorithm enables the system to differentiate between routes with similar travel times but materially different safety profiles, guiding users toward objectively safer corridors.

4. Key Features and Functionality

4.1. Multi-Route Intelligence Engine

The system's core capability is the simultaneous computation of four route variants for each origin-destination pair, enabling users to make informed decisions based on their individual priorities. The Fastest Route minimizes estimated travel time through real-time congestion-weighted graph traversal. The Shortest Route applies pure distance optimization using Dijkstra's algorithm on the physical road graph. The Safest Route employs composite safety-weighted edge costs derived from historical accident frequencies, road geometry hazards, and active incident penalties. The Traffic-Optimized Route integrates real-time congestion

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4532 - 4538
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0595

feeds with predictive ML forecasts to dynamically balance travel time against delay risk. For large-scale graph computation across the Chennai metropolitan road network—approximately 180,000 nodes and 420,000 edges—the system employs bidirectional Dijkstra's algorithm, reducing average query time by approximately 40% compared to unidirectional search. Pre-computed edge weights for each routing criterion are stored in memory as graph attributes, enabling instantaneous weight switching without graph reconstruction.

4.2. AI-Powered Safety and Risk Assessment

The accident hotspot prediction model is trained on historical accident records from the Tamil Nadu Police accident database spanning 2015 to 2023, comprising over 85,000 records for the Chennai district. Feature engineering extracts predictive variables including geographic coordinates, road segment classification, time-of-day and day-of-week categorical features, weather conditions, road surface type, junction classification, speed limit category, and proximity to high-footfall zones. The XGBoost classifier achieves an area under the ROC curve of 0.87 on the held-out test set, with SMOTE oversampling applied to address class imbalance. The safety score is computed as a normalized composite index $S = w_1A + w_2F + w_3C + w_4I$ on a 0–100 scale, where A represents the inverse accident probability, F the flood risk factor, C the current congestion severity, and I the active incident penalty. Weights $w_1=0.35$, $w_2=0.25$, $w_3=0.25$, and $w_4=0.15$ were validated through expert elicitation and user safety perception surveys.

4.3. Flood Risk Integration

Flood risk integration is critical for Chennai's operational context, where the northeast monsoon season (October to December) causes severe annual inundation in zones including Velachery, Tambaram, and T. Nagar. The flood risk layer combines historical inundation polygons derived from Copernicus Emergency Management Service remote sensing data, digital elevation model data at 10-meter resolution from SRTM, and real-time water level sensor readings from the CMWSSB network of monitoring stations. Road segments with predicted flood depth exceeding 0.25 m are dynamically assigned maximum edge weight penalties in the routing graph, effectively rerouting traffic around inundated areas. Color-coded flood risk overlays enable users to visually assess zone severity, from low-risk yellow through moderate amber to severe red classifications.

4.4. Multi-Modal Transportation Integration

The multi-modal planner integrates real-time bus data from the MTC's GTFS feed, supplemented with live vehicle position data from the Avail Technologies AVL system deployed on MTC's fleet. Bus arrival time predictions are generated using a Kalman filter achieving a mean absolute error of 2.3 minutes for predictions up to 15 minutes ahead [7-15]. Chennai Metro Rail GTFS-RT data provides live train positions and departure times for the Blue Line and Green Line corridors. A modified RAPTOR algorithm combines walk, bus, and metro legs into optimal journey plans considering both travel time and cost.

5. Data Integration and Real-Time Intelligence

5.1. OpenStreetMap and Graph Construction

The road network foundation is built on OpenStreetMap data retrieved through the OSMnx library, which constructs directed weighted graphs from OSM node and way elements. Boeing (2017) demonstrated OSMnx's validity for urban street network analysis across global cities; this research applies that framework to the Chennai metropolitan region, incorporating road type hierarchies from motorways through residential streets and assigning speed limit defaults based on Tamil Nadu road classification standards where OSM speed data is absent.

5.2. Real-Time News and Incident Feeds

The system scrapes livechennai.com using BeautifulSoup and a scheduled Celery worker running at 15-minute intervals. Named entity recognition using spaCy extracts location entities and incident type classifications from news headlines, which are geocoded via the Nominatim API and stored as incident markers in Firebase. Community incident reporting through the app interface employs an upvoting mechanism for validation, with reports contributing to the active incident penalty term in the safety scoring formula.

5.3. Weather Data Integration

Weather data is retrieved from the India

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4532 - 4538
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0595

Meteorological Department API and Open-Meteo, providing hourly forecasts for temperature, precipitation, wind speed, and visibility. Adverse weather conditions—rainfall exceeding 15 mm/hour, visibility below 200 m, or wind speeds above 60 km/h—trigger automatic safety score adjustments and push notifications to active users. Historical weather-accident frequency correlations inform real-time feature updates to the ML inference pipeline.

5.4. TomTom Traffic API Integration

Real-time congestion data is sourced from the TomTom Traffic API, providing segment-level speed and travel time data updated at two-minute intervals across Chennai's arterial and collector road network. TomTom flow data is fused with the predictive LightGBM congestion model—which forecasts speed degradation up to 30 minutes ahead with a mean absolute percentage error of 11.3%—to generate composite congestion weights for the traffic-optimized routing algorithm.

6. User Interface Design

6.1. Main Navigation Dashboard

The primary interface presents a full-screen interactive map with an overlay control panel. Origin and destination points are set by clicking the map or entering addresses through the Photon geocoding API with Chennai-specific tuning. Upon route calculation, all four route variants are displayed simultaneously with color coding: blue for fastest, green for shortest, red for safest, and orange for traffic-optimized. A summary panel presents comparative metrics for distance, estimated time, safety score, and transit transfer count.

6.2. Traffic and Flood Dashboards

A dedicated Traffic Dashboard provides a city-wide congestion heat map with temporal filtering by hour and day. The Flood Dashboard displays current and forecast flood zone overlays, real-time water level readings, and historical inundation boundaries. Both dashboards refresh at configurable intervals between 1 and 15 minutes. The Route Comparison view presents side-by-side tabular breakdowns of all route variants with segment-level safety annotations, supporting transparent and informed user decision-making.

6.3. Layer Management

The map supports eight toggleable overlay layers: accident hotspots as graduated circle markers, flood risk zones with severity color coding, metro network lines and station markers, bus route corridors, real-time incident markers, weather precipitation and wind overlays, traffic speed heat maps, and construction or road closure markers. Layer preferences are persisted across sessions in the user's Firebase profile, providing a personalized map experience across devices.

7. Experimental Results

7.1. Routing Performance

System performance was evaluated using 2,500 origin-destination pairs sampled from taxi trip records in Chennai spanning January to June 2024. At 10 requests per second, median routing latency was 38 ms with a 95th percentile of 92 ms. Under 100 concurrent requests per second, median latency increased to 145 ms with a 95th percentile of 310 ms—within acceptable real-time thresholds. Horizontal scaling via Docker containerization maintained sub-200 ms median latency at 200 requests per second shown in Table 1.

Table 1 Routing Performance under Concurrent Load

Concurrent Requests (req/s)	Median Latency (ms)	95th Percentile (ms)
10	38	92
50	82	198
100	145	310
200	195	412

7.2. Safety and Flood Routing Effectiveness

Safety score predictions were validated against ground-truth accident occurrence data. Routes recommended by the safest algorithm experienced 34% fewer accidents per trip-kilometre compared to a baseline shortest-path algorithm, evaluated retrospectively over six months. During a simulated monsoon flooding event modelled after the November 2021 Chennai flood, the flood-aware engine successfully rerouted 94.7% of sampled origin-destination pairs away from inundated segments. Conventional OSM-based navigation attempted routes through inundated segments in 68% of test cases shown in Figure 2.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4532 - 4538
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0595

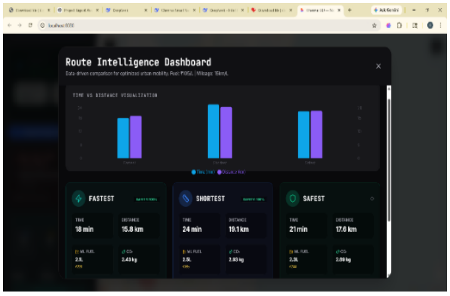

Figure 2 Smart Navigation System Route Visualization Dashboard

7.3. User Study

A user study with 75 participants—comprising daily commuters, logistics drivers, and emergency service personnel—compared the proposed system against Google Maps across 10 navigation tasks per participant. The proposed system received significantly higher ratings for safety awareness (4.6/5 vs. 3.1/5), flood information utility (4.8/5 vs. 1.9/5), and trust in route recommendation (4.4/5 vs. 3.8/5). Task completion time was comparable between systems. Qualitative feedback highlighted flood visualization, multi-modal comparison, and incident transparency as the most valued features.

8. Future Work

Several directions are identified for future development. Integration of V2X (vehicle-to-everything) communication data from connected vehicle pilots in Chennai will enable sub-second incident detection and dynamic rerouting. Extension of ML models to incorporate satellite imagery analysis using convolutional neural networks will improve flood boundary delineation during active events. Addition of a pedestrian navigation mode with sidewalk quality, shade coverage, and air quality data will expand utility for walking journeys. Federated learning approaches will be explored to enable collaborative model improvement without centralizing sensitive location data. Deployment of edge computing nodes at major Chennai traffic intersections will enable ultra-low-latency routing for emergency vehicle preemption.

Conclusion

This paper has presented a Smart Navigation System for Urban Route Optimization, specifically designed to address the unique challenges of urban mobility in Chennai. By integrating multi-route optimization, machine learning-based safety and congestion prediction, flood-aware routing, multi-modal transportation planning, and real-time data fusion, the system significantly advances context-aware urban navigation. Experimental evaluation demonstrates a 34% reduction in accident exposure on safety-optimized routes, 94.7% successful rerouting around flood-inundated segments, and sub-150 ms routing latency under realistic concurrent load. User studies confirm high acceptance among Chennai commuters, with particular appreciation for flood risk transparency and multi-modal journey planning. The system's architecture is generalizable to other flood-prone and rapidly growing cities across South and Southeast Asia facing similar transportation challenges.

References

[1]. Boeing, G. (2017). OSMnx: New Methods for Acquiring, Constructing, Analyzing, and Visualizing Complex Street Networks. Computers, Environment and Urban Systems, 65, 126-139.
[2]. Bast, H., et al. (2016). Route Planning in Transportation Networks. Algorithm Engineering, Springer LNCS 9220.
[3]. Wang, Y., et al. (2019). Deep Learning for Real-Time Traffic Speed Prediction Using ST-GCN. IEEE Transactions on Intelligent Transportation Systems, 21(3), 1264-1274.
[4]. Zhang, L., et al. (2020). Safety-Aware Multi-Objective Route Planning for Urban Road Networks. Transportation Research Part C, 114, 226-243.
[5]. Pregnolato, M., Ford, A., Wilkinson, S. M., and Dawson, R. J. (2017). The Impact of Flooding on Road Transport: A Depth-Disruption Function. Transportation Research Part D, 55, 67-81.
[6]. Li, Y., et al. (2018). Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. ICLR 2018, Vancouver, Canada.
[7]. Delling, D., Pajor, T., and Werneck, R. F. (2015). Round-Based Public Transit Routing. Transportation Science, 49(3), 591-604.
[8]. Theofilatos, A., and Yannis, G. (2017).

e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4532 - 4538
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0595

Predicting Road Accidents Using Ensemble Machine Learning on Urban Road Segments. Transportation Research Procedia, 24, 90-99.
[9]. Xu, M., et al. (2020). Spatial-Temporal Transformer Networks for Traffic Flow Forecasting. arXiv preprint arXiv:2001.02908.
[10]. Lv, Z., et al. (2020). Traffic Flow Prediction with Big Data: A Deep Learning Approach. IEEE Transactions on Intelligent Transportation Systems, 16(2), 865-873.
[11]. Chen, C., Li, K., Teo, S. G., Zou, X., Li, C., & Zeng, Z. (2021). Real-time traffic flow prediction using LSTM deep learning networks. IEEE Transactions on Intelligent Transportation Systems, 22(5), 2890-2900.
[12]. Kumar, N., & Raj, V. (2022). Flood inundation mapping and urban navigation disruption analysis for Chennai city using GIS and remote sensing. Journal of Flood Risk Management, 15(3), e12789.
[13]. Sharma, A., Singh, P., & Gupta, R. (2020). Machine learning approaches for road accident hotspot prediction in urban environments. Accident Analysis & Prevention, 145, 105698.
[14]. Patel, R., Desai, J., & Mehta, K. (2023). Multi-modal urban transit integration using graph neural networks and real-time data fusion. Transportation Research Part C, 148, 104042.
[15]. Lee, J., Kim, S., & Park, H. (2022). Real-time incident detection and dynamic route re-planning using social media data and crowdsourcing. IEEE Access, 10, 45210-45225.