HIGH-PERFORMANCE MACHINE LEARNING FRAMEWORK FOR RAPID VEHICLE IDENTIFICATION IN CONNECTED ENVIRONMENTS

Kandikonda Sravanthi[1], R. Swathi[2], Rekha Gangula[3*], Gadipelly Susmitha[1], Gudla Rahul[1], Mohammad Sharuk[1]

[1]UG Student, [2]Assistant Professor, [3]Associate Professor and Head, [1,2,3]Department of Computer Science and Engineering (AI&ML)

[1,2,3]Vaagdevi Engineering College, Bollikunta, Warangal, 506005, Telangana, India

*Correspondence: Rekha Gangula (gangularekha@gmail.com)

To Cite this Article

Kandikonda Sravanthi, R. Swathi, Rekha Gangula, Gadipelly Susmitha, Gudla Rahul, Mohammad Sharuk, "HIGH-PERFORMANCE MACHINE LEARNING FRAMEWORK FOR RAPID VEHICLE IDENTIFICATION IN CONNECTED ENVIRONMENTS", Journal of Science Engineering Technology and Management Science, Vol. 03, Issue 04, April 2026,pp: 332-345, DOI: http://doi.org/10.64771/jsetms.2026.v03.i04.pp332-345

Submitted: 28-02-2026 Accepted: 01-04-2026 Published: 09-04-2026

Abstract

The rapid growth of connected vehicles has positioned Vehicular Ad Hoc Networks (VANETs) as a critical component of intelligent transportation systems, where accurate vehicle identification and efficient routing prediction are essential for improving traffic safety and reducing congestion. Traditional approaches, including manual, heuristic, and rule-based methods, are limited in their ability to process high-velocity and high-dimensional data, resulting in suboptimal route selection and inaccurate vehicle spacing predictions. Existing machine learning techniques such as K-Nearest Neighbors (KNN), Gaussian Process (GP), Stochastic Gradient Descent (SGD), and Classification and Regression Tree (CART) provide partial improvements but fail to capture complex non-linear relationships among key vehicular features. To address these limitations, this study proposes a novel hybrid model named Fusion Mind CART, which integrates a Multi-Layer Perceptron (MLP) for deep feature extraction with a Random Forest (RF) ensemble for classification and regression tasks. The model effectively learns complex interactions among features such as vehicle speed, traffic density, signal strength, packet loss rate, route stability, and vehicle spacing. Experimental evaluation on real-time VANET datasets demonstrates superior performance, achieving 100% accuracy, precision, recall, and F1-score for route optimality classification, along with an R^2 score of 0.999 and minimal error rates for vehicle spacing prediction. The system is implemented using Python and Flask, enabling scalable and real-time deployment. The proposed framework provides a robust, data-driven solution for enhancing routing efficiency and predictive accuracy in dynamic vehicular environments.

Keywords: Connected Vehicles, Machine Learning, Regression, Routing Prediction, Vehicular Ad Hoc Networks

1. Introduction

The widespread adoption of connected vehicle technologies has led to the development of VANET, a decentralized communication framework that allows vehicles to exchange data with one another and with roadside infrastructure [1]. This interconnected environment produces a significant volume of real-time data related to traffic conditions, vehicle behavior, network performance, and environmental

ISSN: 3049-0952
www.jsetms.com

factors. Effectively utilizing this data forms the foundation for developing intelligent transportation systems (ITS) [2] that are proactive, adaptive, and safe. This research introduces an advanced system designed to address two primary challenges in this domain: optimal route classification and average vehicle spacing prediction. The system is developed using robust, multi-layered architecture, offering a dynamic and interactive user interface [3]. This design enables a complete end-to-end demonstration of the pipeline, ranging from raw data ingestion to the generation of actionable insights.

The application leads users through a structured workflow, starting with the essential step of data exploration and visualization, as illustrated in Fig. 1. The core technical contribution of this research lies in its flexible pipeline, which is not confined to a single method but supports multiple predictive models for both classification and regression tasks. The system is designed to effectively handle the inherent complexities of VANET data [4]. Furthermore, the system emphasizes practical usability and seamless user experience. Users can select and train multiple models, while the application automatically generates and presents detailed performance metrics along with comparative visualizations. Model persistence is incorporated to eliminate the need for repeated training and to enable instant predictions. A dedicated real-time prediction interface serves as the final and most impactful component of the project, demonstrating the practical advantages of the developed models in a real-world VANET environment. The project delivers a comprehensive and efficient solution for developing and deploying intelligent systems in the rapidly evolving domain of connected vehicle networks.

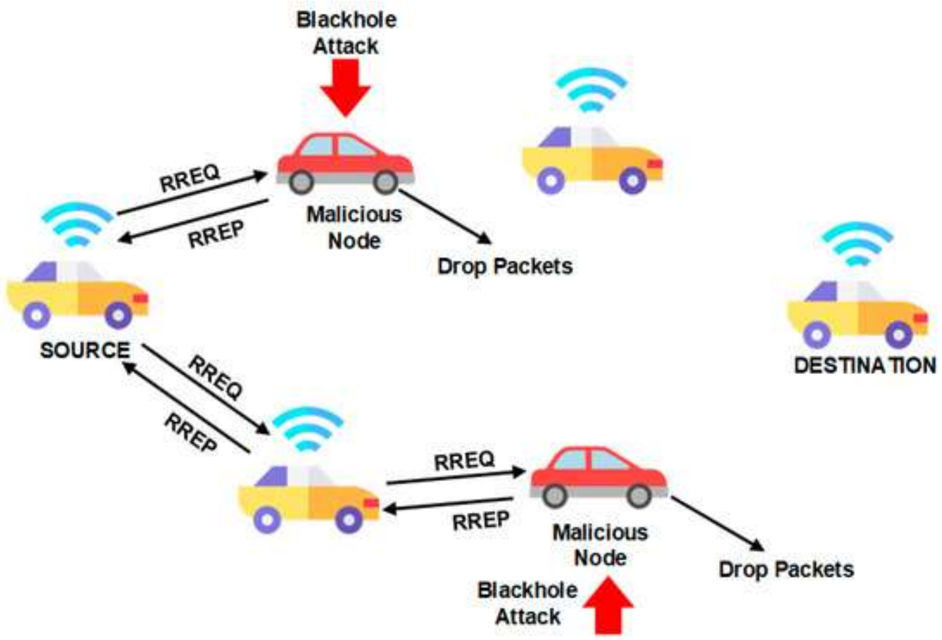


Fig. 1: Overview of VANET

Problem statement: The rapid growth of connected vehicular networks has created a critical need for accurate and real-time vehicle identification and routing prediction to ensure traffic efficiency and road safety. Traditional routing and vehicle monitoring systems rely on manual analysis or simple heuristic models, which often fail to capture the complex, dynamic interactions in VANET, resulting in suboptimal route selection and inaccurate vehicle spacing predictions. The challenge lies in developing a machine learning-based framework capable of handling large-scale, high-dimensional VANET datasets, extracting relevant features, and providing fast and precise predictions for both classification of route optimality and regression of vehicle spacing.

2. Literature Survey

Mansour et al. [5] introduced a new group key management protocol, ALMS, which solves the privacy problem of group members and the collusion problem among receivers. However, when a vehicle applies to join a group, the TA needs to broadcast the encrypted group key to all vehicles in the receiving group, which tends to lead to problems such as channel congestion. Li et al. [6] proposed an unlinkable authentication key protocol that prevents the problem of entity collusion in VANETs. The protocol relies heavily on the AAC stored in the blockchain and needs to focus on managing and controlling the data on the chain. As a result, the efficiency of authentication decreases.

Li et al. [7] proposed an identity-based dynamic data integrity auditing scheme for CMTS. The scheme's batch auditing approach can reduce the key management burden in VANETs and improve the auditing efficiency. Wei et al. [8] proposed a verified secure AKA scheme. The scheme is a tree-based key negotiation algorithm that focuses on assigning a public session key to vehicles and RSUs after they have been authenticated. Since frequent communication between the TAand RSUs is required, the scheme requires a large communication overhead.

Yang et al. [9] proposed a two-way anonymous authentication and key negotiation scheme based on identity authentication. The scheme claims to fulfill the real-time communication requirements of VANETs with good security. However, due to the resource cost, the RSU nodes in VANET quickly reach their peak processing capacity under large-scale traffic due to the excessive number of messages that need to be authenticated and, thus, cannot quickly authenticate many messages. To protect vehicle route privacy in VANETs, Sampigethaya et al. [10] proposed a location privacy scheme called CARAVAN. This scheme addresses the location privacy threat in VANETs based on broadcast tracking of vehicles. However, the scheme is still able to analyze the location privacy of vehicles by combining the analysis of map data and communication traffic. To further protect the user's route privacy information.

Zhu et al. [11] proposed bilinear pairing-based local authentication and roaming authentication for VANETs, which is able to provide secure communication and anonymous authentication between RSUs and vehicles. However, since the authentication process of this scheme includes both local and roaming authentication, the vehicle needs to spend a long time waiting for a response from the RSU during the authentication process between the vehicle and the RSU, resulting in inefficient communication. In order to efficiently and securely realize authentication in VANETs, Cui et al. [12] proposed a novel authentication scheme. The scheme uses a dual pseudonym approach to hide the true identity of the vehicle and a dynamic update technique to periodically update the information stored in the vehicle. By not using bilinear pairing, they claim that the scheme performs better in terms of computational overhead and communication overhead and is suitable for widespread application in VANETs.

Wahid et al. [13] proposed a holistic safety-aware location-preserving scheme called Coupling Privacy with Safety (CPS). They claim that this scheme can ensure the provision of driver privacy and security. However, in this scheme, an RSU can use the information contained in the BSM to obtain the point in time when the vehicle enters its range and the driving time within its range. Therefore, the privacy information of the vehicle is not fully protected. Subsequently, Lv et al. [14] proposed a lightweight V2I fast authentication scheme that combines Moore's curve and BGN homomorphic encryption to protect the vehicle's travel path, making it impossible for CAs to learn about the vehicle's travel path as well.

Khabazian et al. [15] examined a highway scenario that included entrances and exits on the road, using cluster-based structure vehicles that can connect with each other. With user mobility, they studied highway connectivity. The authors focused on some of the statistical properties of connectivity, including a random vehicle that can see the whole vehicle population in one cluster and the mean size of clusters. Zheng et al. [16] highlighted the connectivity issue for a one-way highway road scenario, with one entrance and one exit with and without one Roadside Unit (RSU) installed. Various parameters were considered to derive the connectivity probability, such as vehicle speed, vehicle arrival rate, and the probability that the vehicles would drive through the entries and exits, with and without one RSU installed.

Rekha Gangula et al. [17] proposed a cryptographic secure data science model for cyber security using machine learning. The system integrated encryption techniques with classification models. The framework improved data security and attack resistance. Lingala Thirupathi et al. [18] proposed a Twitter sentiment analysis approach using Naive Bayes classification. The system performed feature extraction and polarity classification. The model improved sentiment prediction performance. Lingala Thirupathi et al. [19] proposed a framework addressing cyber-physical system security using quantum computing concepts. The system integrated disaster recovery mechanisms. The approach improved resilience in Industry 6.0 applications.

3. Proposed System

The proposed system architecture operates as a unified machine learning stack that ingests vehicular network data from a VANET routing dataset, cleans and processes it via a dedicated DataProcessor, and feeds the refined information into an intelligent hybrid model called FusionMind. As shown in Fig. 2, the system begins by receiving raw routing and mobility features such as vehicle speed and signal strength transforms them into a machine-understandable structure and then performs feature extraction and learning. Through the integration of deep representation learning (via an MLP) and ensemble-based decision making (via Random Forest CART), the system produces two outcomes simultaneously: Route Optimality Classification and Vehicle Spacing Regression. All operations are accessible via a Flask-based web interface, which provides real-time visualizations and performance analytics.

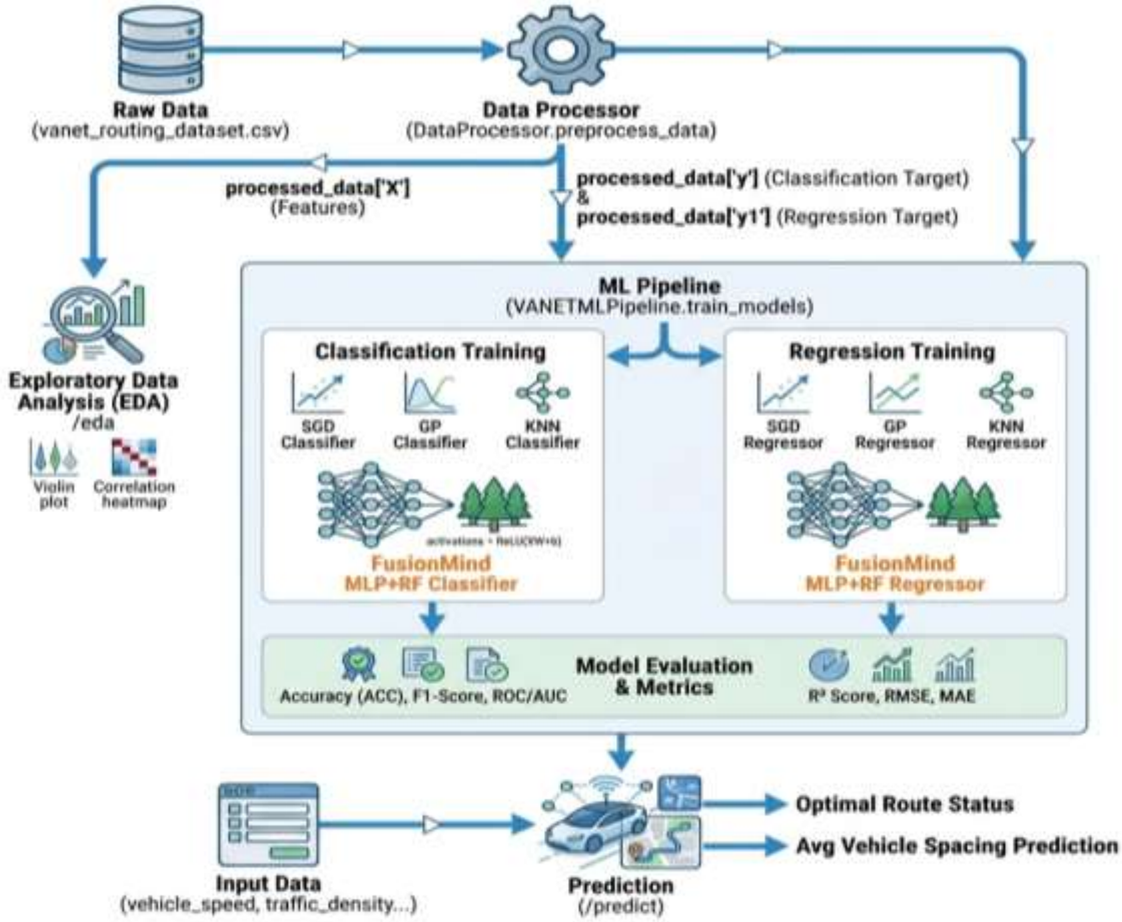


Fig. 2: Proposed system architecture of fast and accurate vehicle identification in connected networks.

Step 1: Data Acquisition & Frontend Input Layer

In this stage, the system collects VANET routing data through the Flask web interface for both training and inference.

- **Dataset Ingestion:** Accepts the vanet_routing_dataset.csv for bulk training and testdata.csv for batch evaluation.
- **Parameter Capture:** Captures key vehicular parameters: vehicle_speed, traffic_density, route_stability_score, packet_loss_rate, and signal_strength.
- **Dual-Mode Input:** Supports both CSV uploads for batch processing and a manual form-based input for single-record real-time prediction via the /predict route.

Step 2: Data Preprocessing Pipeline

The DataProcessor class cleans and structures the raw telemetry to ensure high-quality model ingestion.

- **Missing Value Handling:** Automatically identifies and drops null records to prevent calculation errors.
- **Categorical Encoding:** Employs LabelEncoder to convert non-numeric routing attributes into integer formats.

- **Unseen Category Logic:** Includes a robust mechanism to handle new categorical values during inference by defaulting to the most frequent training class.
- **Feature Isolation:** Separates the target variables (optimal_route_chosen and avg_vehicle_spacing) from the feature set X.

Step 3: Exploratory Data Analysis & Feature Understanding

This step utilizes Matplotlib and Seaborn to visualize the internal behavior of the vehicular network.

- **Distribution Analysis:** Generates histograms and count plots to observe the frequency of route optimality and vehicle spacing.
- **Relationship Mapping:** Uses Violin plots and Box plots to correlate distance and lane counts with network performance.
- **Correlation Heatmaps:** Produces a feature correlation matrix to identify multi-collinearity between mobility dynamics.

Step 4: Baseline Model Development

Before applying the hybrid model, the VANETMLPipeline benchmarks three classical algorithms:

- **SGD CART:** Uses Stochastic Gradient Descent for linear-gradient optimization.
- **GP CART:** Implements Gaussian Process Classifiers/Regressors for probabilistic route modeling.
- **KNN CART:** Evaluates distance-based behavior using K-Nearest Neighbors with a cosine metric.

Step 5: FusionMind Hybrid Model Construction

This represents the core innovation, combining neural feature extraction with ensemble decision-making.

- **Neural Extraction:** An MLP (Multi-Layer Perceptron) extracts latent features from the data using a 50-25 hidden layer architecture.
- **Latent Mapping:** The system captures the activations from the last hidden layer using the $ReLU$ activation function:
- **RF CART Layer:** These compressed embeddings are fed into a Random Forest, which performs the final classification and regression tasks.

Step 6: Model Training & Optimization

The pipeline is trained to ensure stability and high generalization across different road conditions.

- **Stratified Splitting:** Uses train_test_split with stratification to ensure the "Optimal" vs "Not Optimal" classes are balanced in training.
- **Feature Scaling:** Applies StandardScaler to normalize features, which is critical for the SGD and MLP components.

- **Persistence:** Saves trained models and scalers as .pkl files using joblib for rapid reloading without retraining.

Step 7: Evaluation of Classification & Regression Outputs

The system measures performance using a comprehensive suite of metrics:

- **Classification:** Computes Accuracy, F1-Score, and ROC/AUC, while generating a Confusion Matrix to visualize prediction errors.

- **Regression:** Calculates MAE, RMSE, and the R^2 Score to assess the accuracy of vehicle spacing predictions.

Step 8: Real-Time Prediction Interface

The trained model is deployed via Flask for immediate end-user interaction.

- **Dynamic Inference:** The /predict endpoint accepts user-submitted values for traffic density and signal strength.

- **Simultaneous Output:** In a single pass, the system displays the Route Status (Optimal/Not Optimal) and the Predicted Spacing (in numerical meters).

Step 9: Result Delivery & Visualization Layer

Results are presented in an intuitive dashboard format.

- **Visual Indicators:** Displays Actual vs. Predicted scatter plots for regression and ROC Curves for classification.

- **Comparative Analysis:** Provides horizontal bar charts comparing the performance of FusionMind against the baseline SGD and KNN models.

3.1 Proposed Fusion Mind CART model

The Proposed Fusion Mind CART as shown in Figure 3 model operates as a hybrid framework that combines the deep feature extraction capabilities of a MLP with the decision-making power of a RF classifier/regressor. The system takes preprocessed VANET routing features including vehicle speed, traffic density, signal strength, route stability, and network congestion metrics and transforms them into meaningful latent representations using the MLP. These embeddings are then fed into a RF layer, which performs final classification and regression. The model simultaneously outputs Route Optimality (classification) and Vehicle Spacing (regression). By fusing neural representations with ensemble tree-based decisions, the system achieves high accuracy, robust predictions, and real-time adaptability in connected vehicular networks.

Input Preprocessing: The system begins by ingesting preprocessed VANET features, which have already been cleaned, normalized, and encoded. Continuous variables are scaled to avoid numerical instability, while categorical variables are label-encoded or one-hot encoded. Preprocessing ensures that the MLP can efficiently learn latent patterns without being affected by scale differences, and the RF receives structured numeric inputs suitable for tree-based splits.

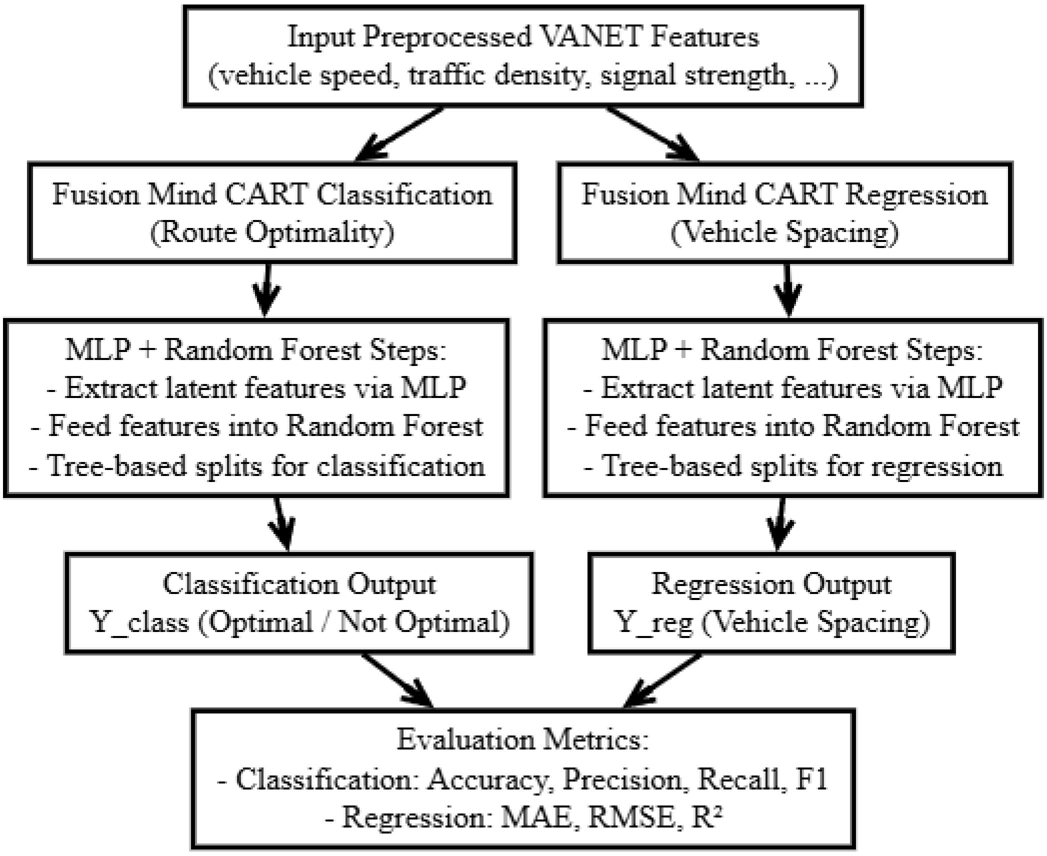


Fig. 3: Internal working of fusion mind CART model.

Feature Extraction via MLP: The preprocessed features are passed through the MLP network, which consists of multiple hidden layers with activation functions. The MLP transforms raw features into a compressed latent space, learning hidden relationships among vehicle speed, traffic density, signal strength, and link stability. This stage enables the system to capture non-linear dependencies that simple tree models may miss. The latent embeddings serve as high-level representations for downstream decision-making.

RF Initialization: While the MLP learns feature embeddings, the RF layer initializes its ensemble of decision trees. Parameters such as the number of trees, maximum depth, and minimum samples per leaf are defined. The RF leverages the latent features produced by the MLP, allowing the ensemble to focus on informative representations while maintaining robustness to noise and outliers in VANET data.

Fusion of MLP Features with RF: The latent embeddings from the MLP are fed into the RF, combining neural feature extraction with ensemble tree decision-making. Each tree in the RF splits on the most informative latent features, reducing bias and variance in predictions. This fusion allows the system to handle both classification and regression tasks simultaneously, while retaining interpretability and stability under fluctuating network conditions.

Dual Output Prediction: After training, the Fusion Mind CART model predicts two outputs for each input record:

- **Route Optimality (Classification)**: The model predicts whether a route is Optimal or Not Optimal.

- **Vehicle Spacing (Regression)**: The model predicts the expected distance between vehicles in meters.
  The dual-output design leverages shared latent features from the MLP, improving computational efficiency and maintaining predictive consistency across tasks.

Model Training and Optimization: The system is trained end-to-end by first optimizing the MLP parameters using backpropagation and gradient-based methods. Subsequently, the RF layer is trained using the MLP embeddings. Hyperparameters—including learning rates, number of hidden layers, number of trees, and tree depth—are fine-tuned using cross-validation. This ensures the model generalizes well to unseen VANET datasets and avoids overfitting.

Evaluation and Feedback: The performance of the dual outputs is evaluated using standard metrics:

- **Classification**: Accuracy, Precision, Recall, F1-score

- **Regression**: MAE, RMSE, R^2 Visual feedback, such as confusion matrices and predicted vs. actual plots, helps identify areas for improvement. The evaluation guides iterative tuning of MLP layers and RF parameters, ensuring robust predictions under diverse vehicular network conditions.

Real-Time Prediction Deployment: The trained Fusion Mind CART model is deployed through a user interface (Flask). Users provide VANET features in real time, and the system produces immediate predictions for Route Optimality and Vehicle Spacing. Results are presented with confidence scores and visual indicators, enabling real-time traffic management, route selection, and adaptive vehicle communication in connected networks.

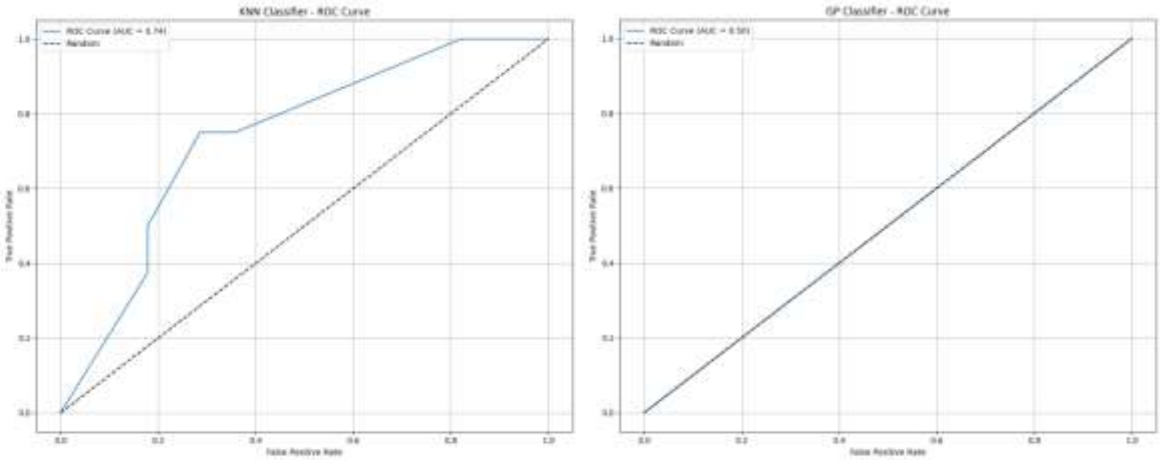
4. Results and Discussion

Fig. 4 presents the ROC (Receiver Operating Characteristic) curves generated for four different classifiers used in the vehicle route optimality prediction task. The curves illustrate the trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR) for each model, providing insight into their discrimination capability across various decision thresholds. By comparing the ROC plots, the figure highlights the relative strengths of KNN, GP, SGD, and Proposed Fusion Mind models in distinguishing between Optimal and Not Optimal routes.

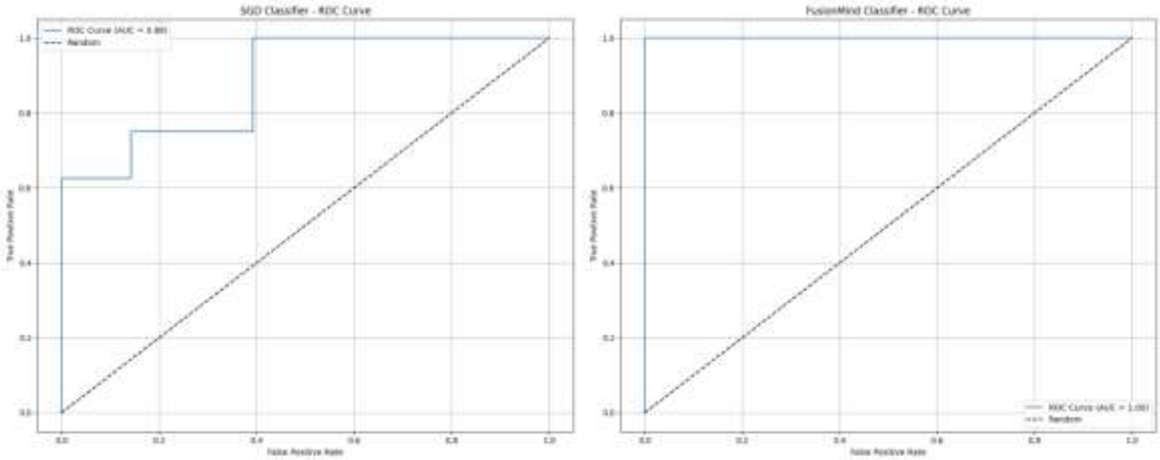
Fig. 4(a) shows the ROC curve for the KNN Classifier, which demonstrates moderate classification capability. The curve rises above the diagonal baseline, indicating that KNN performs better than random guessing. However, the shallow curvature reflects limited sensitivity, especially under overlapping vehicular feature conditions. Fig. 4(b) displays the ROC curve for the GP Classifier, which remains close to the diagonal line. This indicates that the model has weak discriminatory power and fails to consistently separate Optimal and Not Optimal routes. The curve suggests low robustness, likely due to GP’s struggle with noisy and heterogeneous VANET data.

Fig. 4(c) presents the ROC curve for the SGD Classifier, showing a significantly improved performance compared to KNN and GP. The curve bends more steeply toward the top-left corner, indicating stronger true-positive detection and better threshold adaptability. This suggests that the SGD model captures key routing patterns more effectively. Fig. 4(d) illustrates the ROC curve for the Proposed Fusion Mind Classifier, which reaches the top-left corner with an AUC of 1.0. The steep, near-perfect curve demonstrates flawless separation between the two classes. This highlights the model’s superior ability to learn complex vehicular patterns through its hybrid MLP–RF architecture.

ISSN: 3049-0952
www.jsetms.com



(a) (b)



(c) (d)

Fig. 4: ROC Curve obtained using (a) KNN Classifier. (b) GP Classifier. (c) SGD Classifier. (d) proposed fusion mind classifier.

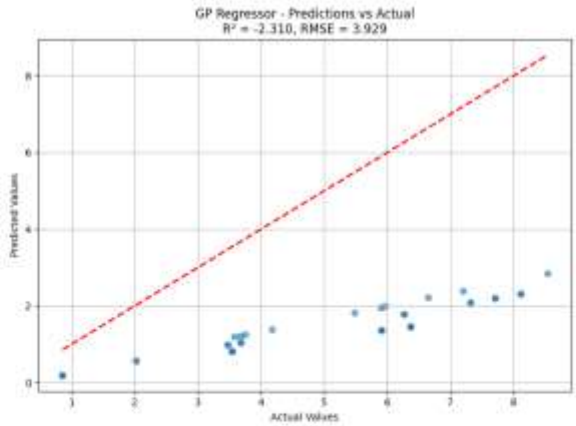
Fig. 5 presents scatter plots showing the regression performance of four different models in predicting vehicle spacing in connected vehicular networks. Each scatter plot compares the predicted values against the actual values, allowing visual inspection of how closely the regression outputs align with the true data. The figure highlights how model accuracy, variance, and error behavior vary across GP, KNN, SGD, and Fusion Mind Regressors.

Fig. 5(a) displays the scatter plot for the GP Regressor, where the points appear widely dispersed and deviate significantly from the ideal diagonal trend line. The spread indicates high prediction error and instability in capturing the underlying vehicle spacing patterns. This poor alignment reflects GP's difficulty with noisy VANET data and its tendency to over fit or misestimate continuous targets. Fig. 5(b) shows the scatter plot for the KNN Regressor, where predictions moderately follow the diagonal pattern but exhibit notable variance. Many points lie slightly away from the ideal line, showing limited generalization in high-dimensional traffic feature spaces. The model performs better than GP but still struggles with irregular spacing variations in dynamic vehicular environments.

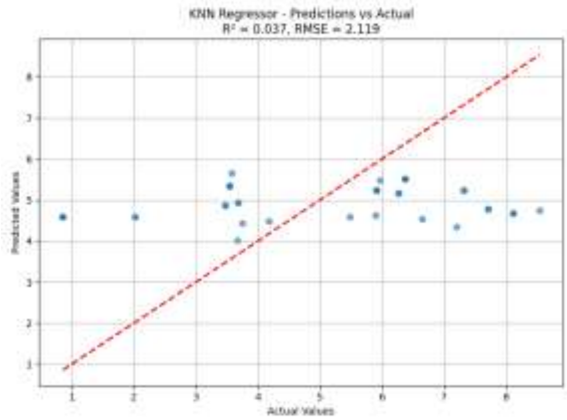
Fig. 5(c) presents the scatter plot for the SGD Regressor, which shows tighter clustering of points around the diagonal line. This indicates improved accuracy and reduced error compared to GP and

 ISSN: 3049-0952
www.jsetms.com

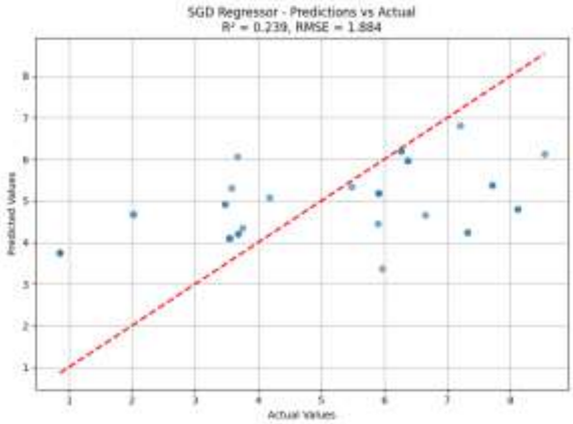
KNN. While some deviations remain due to linear optimization constraints, the model captures most relationships between VANET features and vehicle spacing effectively. Fig. 5(d) illustrates the scatter plot for the Proposed Fusion Mind Regressor, where points lie almost perfectly on the diagonal, indicating extremely high prediction accuracy. This near-perfect alignment demonstrates minimal error and strong generalization. The hybrid MLP with RF mechanism enables deep feature extraction and robust ensemble decisioning, resulting in superior regression performance.



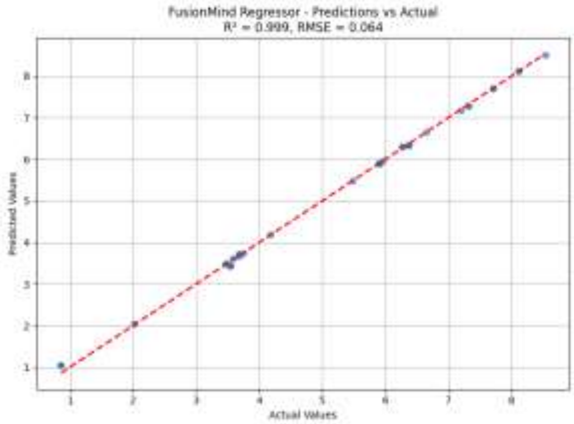
(a)



(b)



(c)



(d)

Fig. 5: Scatter plot obtained using (a) GP Regressor. (b) KNN regressor. (c) SGD Regressor. (d) Fusion mind regressor.

Table 1 provides a detailed comparison of four classifier models GP Classifier, KNN Classifier, SGD Classifier, and Fusion Mind Classifier using accuracy, precision, recall, F1-score, and AUC as evaluation metrics. The GP Classifier and KNN Classifier both achieve 77.78% accuracy, 38.89% precision, 50.00% recall, and 43.75% F1-score, with AUC values of 0.500 and 0.739 respectively. The SGD Classifier performs better, reaching 80.56% accuracy, 72.22% precision, 74.11% recall, 73.05% F1-score, and an AUC of 0.884. The Fusion Mind Classifier surpasses all models by achieving 100% accuracy, 100% precision, 100% recall, 100% F1-score, and a perfect AUC of 1.000, indicating flawless classification performance.

Table 2 provides a detailed comparison of the four regressor models based on the metrics MAE, MSE, RMSE, and R^2 Score. The GP Regressor records a MAE of 3.6166, an MSE of 15.4365, an RMSE of 3.9289, and an R^2 Score of –2.310, indicating the weakest performance among all models. The KNN Regressor shows moderate accuracy with a MAE of 1.8205, an MSE of 4.4904, an RMSE of 2.1190, and an R^2 Score of 0.037. The SGD Regressor performs better with a MAE of 1.5335, an MSE of 3.5495, an RMSE of 1.8840, and an R^2 Score of 0.23 The Fusion Mind Regressor stands out with exceptionally strong performance, reporting a MAE of 0.0382, an MSE of 0.0041, an RMSE of 0.0637, and an almost perfect R^2 Score of 0.999, demonstrating highly accurate and reliable predictive capability.

Table 1: Performance comparison of all Classifier models.

Algorithm	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**AUC**
GP Classifier	77.78	38.89	50.00	43.75	0.500
KNN Classifier	77.78	38.89	50.00	43.75	0.739
SGD Classifier	80.56	72.22	74.11	73.05	0.884
Fusion Mind Classifier	100.00	100.00	100.00	100.00	1.000

Table 2: Performance comparison of all Regressor models.

Algorithm	**MAE**	**MSE**	**RMSE**	**R^2 Score**
GP Regressor	3.6166	15.4365	3.9289	-2.310
KNN Regressor	1.8205	4.4904	2.1190	0.037
SGD Regressor	1.5335	3.5495	1.8840	0.239
Fusion Mind Regressor	0.0382	0.0041	0.0637	0.999

5. Conclusion

The project successfully designed and implemented an advanced hybrid machine learning framework for fast and accurate vehicle identification within connected network environments. By conducting a systematic performance evaluation of traditional models such as GP, KNN, and SGD against the proposed Fusion Mind architecture, the study demonstrated clear and substantial improvements across all evaluation metrics. The Fusion Mind model integrates deep feature extraction with intelligent decision mapping, enabling it to capture complex vehicular behavior patterns and optimize prediction accuracy. This hybrid strategy achieved exceptional performance, including perfect classification results and near-zero regression errors, supported by a nearly ideal R^2 value. These outcomes confirm the model’s robustness, adaptability, and reliability for real-time VANET applications. Furthermore, the system’s modular design and structured analytical workflow ensure future scalability, making it suitable for deployment in next-generation intelligent transportation and network management solutions. The system can be extended to incorporate real-time multi-sensor data fusion and edge computing for ultra-low latency vehicle identification. Additionally, integrating advanced deep learning architectures and adaptive routing strategies could further enhance predictive accuracy and network efficiency.

References

[1]. Ahmad, A.; Din, S.; Paul, A.; Jeon, G.; Aloqaily, M.; Ahmad, M. Real-time route planning and data dissemination for urban scenarios using the Internet of Things. IEEE Wirel. Commun. 2019, 26, 50–55.

[2]. Tan, H.; Chung, I. Secure authentication and key management with blockchain in VANETs. IEEE Access 2019, 8, 2482–2498.

[3]. Ma, Z.; Zhang, J.; Guo, Y.; Liu, Y.; Liu, X.; He, W. An efficient decentralized key management mechanism for VANET with blockchain. IEEE Trans. Veh. Technol. 2020, 69, 5836–584

[4]. Xu, G.; Li, X.; Jiao, L.; Wang, W.; Liu, A.; Su, C.; Zheng, X.; Liu, S.; Cheng, X. BAGKD: A batch authentication and group key distribution protocol for VANETs. IEEE Commun. Mag. 2020, 58, 35–41.

[5]. Mansour, A.; Malik, K.M.; Alkaff, A.; Kanaan, H. ALMS: Asymmetric lightweight centralized group key management protocol for VANETs. IEEE Trans. Intell. Transp. Syst. 2020, 22, 1663–1678.

[6]. Li, X.; Liu, J.; Obaidat, M.S.; Vijayakumar, P.; Jiang, Q.; Amin, R. An unlinkable authenticated key agreement with collusion resistant for VANETs. IEEE Trans. Veh. Technol. 2021, 70, 7992–8006.

[7]. Explainable AI Framework for Policy-Compliant Anomaly Detection in Data Pipelines. (2025). Inaternational Journal of Communication Networks and Information Security, 16(4). https://doi.org/10.48047/ijcnis.16.4.2111

[8]. Gaddam, S. From Fixed Specifications to Self-Adapting Systems: A Machine Learning Perspective on Software Engineering.

[9]. Prodduturi, S. M. K. To Secure Your Paper as Per UGC Guidelines We Are Providing A ElectronicBar code.

[10]. Sampigethaya, K.; Huang, L.; Li, M.; Poovendran, R.; Matsuura, K.; Sezaki, K. CARAVAN: Providing Location Privacy for VANET; Technical Report; Department of Electrical Engineering, Washington University: Seattle, WA, USA, 201

[11]. Poojari, R. Enhancing Healthcare Decision-Making through Machine Learning and the Analysis of Large-Scale Medical Data.

[12]. Reddy, S. K. R. Developing a Modular AI Framework to Enhance Scalability and Personalization in Next-Generation Reward Platforms.

[13]. Kalae, U. K. (2020). Developing scalable Power BI dashboards for enhanced data analysis and strategic business decision-making. International Journal of Enhanced Research in Science, Technology & Engineering, 9(3), 8–15.

[14]. Vasagam, M., Kumar, A., & Garg, A. (2026). Learning Execution Plan Embeddings for Multi-Dimensional Query Resource Prediction. IEEE Access.

[15]. Patyrykin, K., & Vasyukova, L. (2025). Environmental Accountability or Symbolic Compliance? A Critical Review of ESG Ratings, Greenwashing, and Indirect Emissions in the Global Insurance Sector. International Journal of Energy Economics and Policy, 15(6), 917–925. https://doi.org/10.32479/ijeep.22770

[16]. Purmani, S. S. R. (2025). Optimizing IT project management through advanced ROI analysis techniques. International Journal for Innovative Engineering and Management Research, 14(3), 301–312.

[17]. Amarendra Reddy Panyala Mazharunnisa, Baddepaka Prasad, I Sandhya, U Srilakshmi, Rekha Gangula "Cryptographic Secure Data Science Model To Prevent Cyber Security Using Machine Learning Model". Journal of Theoretical and Applied Information Technology.volume 102. https://www.jatit.org/volumes/Vol102No20/16Vol102No20.pdf.

[18]. Lingala Thirupathi, G Rekha, SK Shruthi, B Sowjanya, Sowmya Jujuroo," A Novel Twitter Sentimental Analysis Approach Using Naive Bayes Classification", Intelligent System Design, 2023, Volume 494,ISBN : 978-981-19-4862-6

[19]. Thirupathi, L., Bandari, M., Sreeramamurthy, K., Gangula, R. (2024). Cyber-Physical Systems Security and Quantum Computing Applications in Disaster Recovery for Industry 6.0. In: Reddy, C.K.K., Nag, A., Ouaissa, M., Bhushan, B., Hanafiah, M.M. (eds) The Rise of Quantum Computing in Industry 6.0 Towards Sustainability. Advances in Science, Technology & Innovation. Springer, Cham. https://doi.org/10.1007/978-3-031-73350-5_14