International Journal of Computing Sciences Research (ISSN print: 2546-0552; ISSN online: 2546-115X)
Vol. 10, pp. 4119-4141
doi: 10.25147/ijcsr.2017.001.1.262
https://stepacademic.net

Long Paper

Comparison of Machine Learning Algorithms Applied to Trust and Position-Based Methods for Malicious Node Detection in Vehicular Ad Hoc Networks

Percival D. Adao
College of Computing and Information Sciences, University of Makati, Philippines
percival.adao@umak.edu.ph
(corresponding author)

Date received: July 9, 2025
Date received in revised form: November 24, 2025; December 25, 2025
Date accepted: February 14, 2026

Recommended citation:

Adao, P. (2026). Comparison of Machine Learning Algorithms Applied to Trust and Position-Based Methods for Malicious Node Detection in Vehicular Ad Hoc Networks. *International Journal of Computing Sciences Research*, *10*, 4119-4141. https://doi.org/10.25147/ijcsr.2017.001.1.262

Abstract

Purpose – This study explores the detection of malicious nodes in Vehicular Ad Hoc Networks (VANETs) by means of vehicular trust ratings. It tackles the security and privacy risks of these dynamic networks, acting as the heart of Smart Cities.

Method – Based on the CRISP-DM method, this paper evaluates five algorithms, such as SVC and MLP. The approach specifically tackles data imbalance with Adaptive Synthetic Sampling (ADASYN) and contributes to model transparency with SHapley Additive exPlanations (SHAP), enhancing model interpretability.

Results – Although Support Vector Classifiers (SVC) outperformed conventional accuracy, the improved MLP model trained with hyperparameter tuning for security was found to be better. ADASYN was used to augment the MLP, which achieved a critical recall rate of 1.00 (all nodes were successfully identified as malicious).

Discussion – This is due to the fact that false negatives must be eliminated as a necessity for safety-critical systems. The MLP model's ability to identify every threat and prevent the general occurrence of a threat from occurring is more valuable than the SVC's ability to capture threats overall, backed up by elaborate data engineering.

Implication – The results suggest that Explainable AI with SHAP is needed in high-stakes areas such as interconnected roads. Proper accuracy is not enough, and intelligent transportation systems should be transparent and have a high recall for safety and reliability.

Conclusion – The work suggests the importance of a customized tuning for VANET security. Given a recall of 1.00, the MLP model, which is ADASYN-based, protects the network from adversary behavior that is well-documented, showing the class imbalance in the intrusion detection problem.

Recommendation – For intelligent transportation systems designers, system recall should have zero false negatives to make recall the most effective solution to ensure zero false negatives. Explainability frameworks in synthetic sampling can also be integrated into future frameworks to reduce dataset imbalance, ensure that the dataset is unbiased and trustworthy, and make sure that automated decisions made would be dependent and transparent decisions will be performed automatically.

Keywords – Vehicular Ad Hoc Networks (VANETs), Machine Learning, Adaptive Synthetic Sampling (ADASYN), Explainable AI (SHAP), Predictive Analytics in Education.

INTRODUCTION

The Advancement of Vehicle Connectivity

The Fourth Industrial Revolution has heralded a major shift to the Internet of Vehicles (IoV). As a part of this paradigm, Vehicular Ad Hoc Networks (VANETs) provide continuous communication between vehicles and infrastructure that will enhance traffic control and collision avoidance (Rashide et al., 2023). However, the open, decentralized nature and high mobility of VANETs introduce specific threats that traditional perimeter defense cannot address.

The Threat Landscape

Adversaries exploit this expansive landscape to carry out sophisticated routing attacks. Two of the most damaging ones are the Blackhole attacks, where malicious nodes create fake paths to attract and discard all network traffic, Blackholes, and the Greyhole attack, a more sophisticated technique in which nodes filter out critical safety information but still send normal messages over the system, making it more difficult to detect (Kamis et al., 2023). Furthermore, the data leakage is not limited to data loss, and can affect physical safety, such as accidental accidents by hiding braking signs, and the safety of the user; the movement pattern tracking may be responsible for privacy breaches as well. Against these threats,

Trust Management and Machine Learning systems are developed to evaluate node reliability using behavioral analysis instead of identity verification. New developments include the implementation of blockchain technology to maintain the veracity of data, but the dynamic characteristics of the VANETs make standard trust establishment difficult (Siddiqui et al., 2023). Thus, this study utilizes Machine Learning in the realm of the CRISP-DM method and identifies trust verification as a classification issue. Supervised models can also identify statistical patterns of malicious behaviors by inspecting spatial attributes in conjunction with trust indices (Schröer et al. 2921).

Class imbalance, where benign nodes are prevalent to an extent while malicious entities are far less well balanced, presents a major challenge in this area. Because of this difference, standard algorithms can be biased towards accuracy with high false negatives and lower accuracy rates. To remedy the problem, Kovács (2019) suggested the Synthetic Minority Over-sampling Technique (SMOTE) for effectively handling general class imbalance when the minority class is relatively distinct or clustered. Furthermore, Khani et al. (2025) presented Adaptive Synthetic Sampling (ADASYN), as compared to a more complex or overlapping class boundary, where class separation is problematic for the model. Finally, Paliwal et al. (2025) employed Explainable AI approaches, including SHAP, to improve accuracy at generating a synthetic representation of the minority class in complex feature spaces and force the classifier to closely draw decision boundaries around malicious nodes.

LITERATURE REVIEW

The field is very diverse and multifaceted, and academic discussion of VANET security is just the same. In this subpart, I summarize some of the significant work in threats and trust for communication systems and introduce machine learning for intrusion detection. The Threats of IoV in Evolution

The Threats of IoV in Evolution

Much of the interest in Vehicular Ad Hoc Networks (VANETs) has been on building cryptographic authentication to prevent unauthorized access. However, as illustrated by Abreu et al. (2024), recent assessments reveal that there exist significant threats from internal adversaries emerging from authenticated nodes. A trusted node does much damage to the network whenever it engages in malicious activities, while retaining legitimate cryptographic credentials. There are several threats, among which routing attacks are notably concerning. By exploring routes, Blackhole and Greyhole attacks can disrupt the Ad hoc On Demand Distance Vector (AODV) routing protocol. Kamis et al. (2023) highlight the stealth of the Greyhole attacks by emphasizing that it can be difficult to differentiate between intermittent loss or signal loss, during which adverse channel conditions may also cause packet loss or delays and may result in network congestion. Additionally, Paliwal et al. (2025) refer to a quantitative study by Schröer et al. (2021) that illustrates the potential pitfalls of conducting coordinated assaults from multiple source nodes, some of which (though also not limited to the "evil nodes") may be in fact malicious as collaborating groups.

This raises the pressing demand for interconnected networks (ICs), which adopt collaborative detection or information-sharing approaches among interconnected systems. Machine Learning based vehicular intrusion detection

Machine Learning-based Vehicular Intrusion Detection

Vehicular Ad Hoc Network (VANET), also referred to as vehicular-specific traffic, is being protected using Machine Learning (ML) to enable the evolution from signature to anomaly detection. In contrast to typical IDS, which follow certain attack signatures and are unable to mitigate zero-day vulnerabilities, ML models have a strong ability to grasp the statistical tendencies of the so-called "typical" network and can also create signals that can serve as warning indicators if any anomalies are detected. In this domain, the Support Vector Machines (SVM) and Random Forests (RF) became the major benchmarks. Rashid et al. (2023) used these algorithms as part of a distributed model in Apache Spark to be able to determine Distributed Denial-of-Service (DDoS) attacks and obtained significant accuracy. However, finding its best feature set is difficult.

Al Sarem et al. (2022) have tackled that issue in this research by establishing an aggregated mutual information-based feature selection technique and showed that dimensionality reduction accelerates training and is associated with a higher detection performance by reducing noise impeding it. Moreover, the Deep Learning (DL) techniques are advancing detection capabilities. Based on this study, Gajiwala (2025) presented a secure IDS using DL approaches such as Convolutional Neural Networks (CNN) for spatial feature extraction and Long Short-Term Memory (LSTM) units for retaining temporal dependencies in traffic flows. These sophisticated models are good at tracking complex and continuously changing attacks, but are computationally intensive; they are not so optimized for real-time events.

Class Imbalance and Model Interpretability

Class imbalance in intrusion detection datasets is a common problem. Malicious activity is rare in its existence, leading to models trained on raw data that classify the majority class (benign) overwhelmingly, and obtain relatively high accuracy with zero recall for the attacks. Alharbi (2025) developed ways of dealing with this problem and observed that ADASYN (Adaptive Synthetic Sampling) outperforms SMOTE for scenarios containing complex decision boundaries. As SMOTE generates synthetic samples evenly across minority instances, ADASYN selectively adds samples if the minority class is surrounded by the majority class sample region to effectively increase areas considered most suitable for replication. And then, the "black box" feature of AI in the security domain is addressed through Explainable AI (XAI). Reynaud & Roxin (2025) proposed that building transparency improves trust, and more techniques like SHAP can also explain localized features that characterize targeted elements on attack maps for individual nodes on networks to allow a local explanation for nodes that are classified as malicious under certain conditions like those whose packet forwarding rate

had significantly decreased: a statement provided in the review of this paper. Such interpretability is essential for forensic work and to improve trust in models themselves.

Theoretical Framework

This research is grounded in the mathematical principles of the selected classification algorithms and the techniques used for data augmentation and interpretation. A rigorous understanding of these foundations is essential for interpreting the experimental results.

Logistic Regression

Dey et al. (2025) explain that Logistic Regression (LR) is a probabilistic linear classifier. It models the log odds of the probability of an event (a vehicle being malicious, Y=1) as a linear combination of independent variables X. The relationship is defined by the sigmoid function sigma (z):

$$\sigma(z) = \frac{1}{1+e^{-z}} \qquad \textit{Equation 1}$$

Using -z in the exponent, when z is very large, e^{-z} approaches 0, making the probability approach 1. When z is very small (negative), e^{-z} becomes very large, making the probability approach 0. From equation 1, where $z = \beta_0+\beta_1 x_1 + \cdots + \beta_n x_n$. The probability P(Y=1|X) is thus constrained between 0 and 1. The model parameters β are estimated using Maximum Likelihood Estimation (MLE), typically by minimizing the binary cross-entropy loss function (Log Loss):

$$J(\beta) = -\frac{1}{m}\sum_{i-1}^{m}\left[y^{(i)}\log\left(\hat{y}^{(i)}\right) + \left(1-y^{(i)}\right)\log\left(1-\hat{y}^{(i)}\right)\right] \qquad \textit{Equation 2}$$

LR serves as a robust baseline due to its simplicity and interpretability, though it assumes a linear decision boundary, which may be insufficient for complex attack patterns (Anish et al., 2023)

Random Forest and Gini Impurity

Random Forest (RF) is an ensemble learning approach that builds a large number of decision trees at training time. And bagging (bootstrap aggregating) is the way to lower variance and overfitting. The mode of the class votes (for classification) or regression value (for regression) computed by individual trees is used as a prediction. The essence of the RF algorithm is the splitting criteria of nodes (Yang & Wang, 2025). This work makes use of Gini Impurity, which measures the frequency of a randomly chosen element from the set that would be incorrectly labeled if it were assigned a label by random, based on that subset's label distribution. Within a set of C classes with probabilities pi, the Gini Impurity is:

$$\text{Gini}(D)=1-\sum_{i=1}^{c} p_i^2 \qquad \textit{Equation 3}$$

During training, the algorithm searches for the split that maximizes the Gini Gain (reduction in impurity).

Support Vector Classifier (SVC)

Support Vector Machines (SVMs) are powerful supervised learning models that construct a hyperplane or set of hyperplanes in a high-dimensional space. The optimal hyperplane is the one that maximizes the margin, the distance to the nearest training data points of any class in functional margin (Zhong & Du, 2023). For non-linearly separable data, SVC employs the kernel trick to map the input space into a higher-dimensional feature space where linear separation is possible. We utilize the Radial Basis Function (RBF) kernel:

$$K(x_i x_j) = \exp(-\gamma \|x_i - x_j\|^2) \qquad \textit{Equation 4}$$

where γ defines the influence of a single training example. The optimization problem involves minimizing the hinge loss function with an L_2 regularization term:

$$\frac{\min}{\omega,b} \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n} \max\left(0, 1 - y_1(\omega \cdot \emptyset(x_i) + b)\right) \qquad \textit{Equation 5}$$

Here, C is a regularization parameter that controls the tradeoff between maximizing the margin and minimizing the classification error on the training data.

K Nearest Neighbors (KNN)

KNN is a non-parametric, instance-based learning algorithm. It classifies a new data point based on the majority class of its k nearest neighbors in the feature space. The "closeness" is defined by a distance metric (Syriopoulos et al. 2023; Maity et al., 2020). In this study, we employ the Minkowski distance, a generalization of Euclidean and Manhattan distances:

$$D(X, Y) = \left(\sum_{i-1}^{n} \lfloor x_i - y_i \rfloor^p\right)^{\frac{1}{p}} \qquad \textit{Equation 6}$$

When p=2, this becomes the Euclidean distance. KNN is sensitive to the local structure of the data and the scale of features, making feature standardization a critical prerequisite.

Multi-Layer Perceptron (MLP)

The MLP is a class of feedforward Artificial Neural Networks (ANN). It consists of at least three layers of nodes: an input layer, a hidden layer, and an output layer (Ruangkanjanases et al. 2024). Except for the input nodes, each node is a neuron that uses a nonlinear activation function.

The output of a neuron j in layer 1 is given by:

$$a_j^{(l)} = f\left(\sum_i \omega_{ji}^l a_i^{(l-1)} + b_j^l\right) \qquad \textit{Equation 7}$$

where f is the activation function. This utilizes the ReLU (Rectified Linear Unit) activation, f(x) = max (0, x), for hidden layers due to its efficiency and resistance to the vanishing gradient problem. The network is trained using Backpropagation, which computes the gradient of the loss function (Cross Entropy) with respect to the weights using the chain rule, allowing for weight updates via stochastic gradient descent or its variants like Adam.

Adaptive Synthetic Sampling (ADASYN)

ADASYN is an advanced oversampling technique for imbalanced datasets. Unlike SMOTE, which generates synthetic samples uniformly for the minority class, ADASYN uses a weighted distribution for different minority class examples based on their level of difficulty in learning (Alharbi 2025; Sulaiman et al., 2025)

The algorithm calculates a ratio $r_i = \Delta_i/K$ for each minority example x_i, where Δ_i is the number of majority class examples in the K nearest neighbors of x_i. The number of synthetic samples g_i generated for x_i is proportional to r_i. This focuses the synthetic data generation on the "hard" examples near the decision boundary, forcing the classifier to learn the complex nuances separating malicious from benign behavior (Afane & Zhao, 2024).

SHapley Additive exPlanations (SHAP)

SHAP is a game-theoretic approach to explain the output of any machine learning model. It assigns each feature an importance value for a particular prediction (Movsessian et al., 2021). The SHAP value $\emptyset_i$for a feature i is calculated as the weighted average of the marginal contributions of that feature across all possible coalitions of features:

$$\emptyset_i = \sum_{SCF\backslash(i)} \frac{|S|!(|F|-|S|-1)!|}{|F|!} \qquad \textit{Equation 8}$$

where F is the set of all features and f_s is the model trained on the subset S. This allows us to decompose a prediction (e.g., "Malicious") into the sum of contributions from trust_rating, location x, etc., providing local interpretability.

RESEARCH METHODOLOGY

The CRISP DM Approach

This research is structured according to the Cross Industry Standard Process for Data Mining (CRISP-DM), ensuring a systematic workflow from data ingestion to model evaluation.

Business Understanding

The strategic goal is to secure the IoV infrastructure. In a cybersecurity context, the cost

of misclassification is asymmetric. A False Negative (FN) classifying a malicious vehicle as benign allows an attacker to persist in the network, potentially executing Blackhole attacks or injecting false safety data (Schröer et al., 2021). A False Positive (FP) flagging a benign vehicle as malicious results in a temporary denial of service for that vehicle or a requirement for secondary authentication. Given the safety-critical nature of VANETs, minimizing FNs (maximizing Recall) is the paramount objective.

Data Understanding and Gathering

The dataset trust_rating vehicle.csv serves as the empirical basis for this study. The dataset was obtained from a simulation with a vehicular network scenario involving 500 unique vehicle identities.

Figure 1a. Simulation scenario for a smart city in Makati.

The simulations have been mainly conducted on VEINS Simulator 3.0, an event-driven simulator for VANETs coupled with SUMO version 0.32.0 to exhibit fluidity in the mobility pattern, and OMNeT++ version 5.7 for simulating the network functionalities in this context (Upadhyay et al., 2023; Bachmeier et al., 2020). The simulations in particular were generated for a location based in Makati City with an area of about 4 km² that consists of Magallanes Avenue, Buendia Avenue, EDSA Highway, and Osmeña Highway, shown in Figure 1a.

Performed during peak-hour to truthfully reproduce real-life conditions; engineering was achieved through the collection of 500 hello messages necessary for training and validating the machine learning classifiers built on localities datasets collected at these rush hour periods (Alvarez et al., 2025; Sommer et al., 2021). In addition, multiple layouts were tested to increase realism and coverage over the study parameters. In this paper, the two-lane configuration was simulated with Poisson-distributed vehicles and speeds between 10 m/s

and 35 m/s, resulting in well-structured setups equipped with advanced mobility models as well as communication protocols to generate datasets mirroring complex interactions occurring in dynamic vehicular networks.

Hardware and software requirements

Figure 1a illustrates the significant hardware and software ecosystem needed to ensure appropriate computational fidelity of Vehicular Ad Hoc Network (VANET) simulations. Moreover, the study uses a set of all-inclusive software to enable high-fidelity modeling of vehicular dynamics and network dynamics, such as VEINS Simulator 3.0, SUMO 0.32.0, and OMNeT++ 5.7.1. Google Colab provides a cloud-application-specific and flexible execution environment for this integrated stack (Sommer et al., 2021). The proposed hardware configuration for the software above necessitates a 64-bit OS running a 4-core CPU and 16 GB RAM for concurrent processing threads to handle the processing intensity of the above-mentioned applications. In addition, a huge 200 GB SSD is needed for fetching data in real time, and a dedicated 1GB GPU is required for rendering graphics. These specifications are also corroborated by the contemporary literature, such as Alvarez et al. (2025), confirming these criteria as essential to ensure the reliability and reproducibility of such complex smart city simulations.

Machine Learning Approach to Data-Driven Trust Model

This model complemented the empirical methods by defining 'direct trust' as a measure derived from the data encapsulated within messages. The system used the data from periodically exchanged hello messages to calculate trust (Veerabudren & Ramsurrun, 2024). These V2X messages, which are critical for establishing trust, contained the source's ID, location, moving velocity, and departure time. As a result, the trust computation was triggered when the node 'x' sent a V2X message to the neighboring node 'y' (Figure 1b).

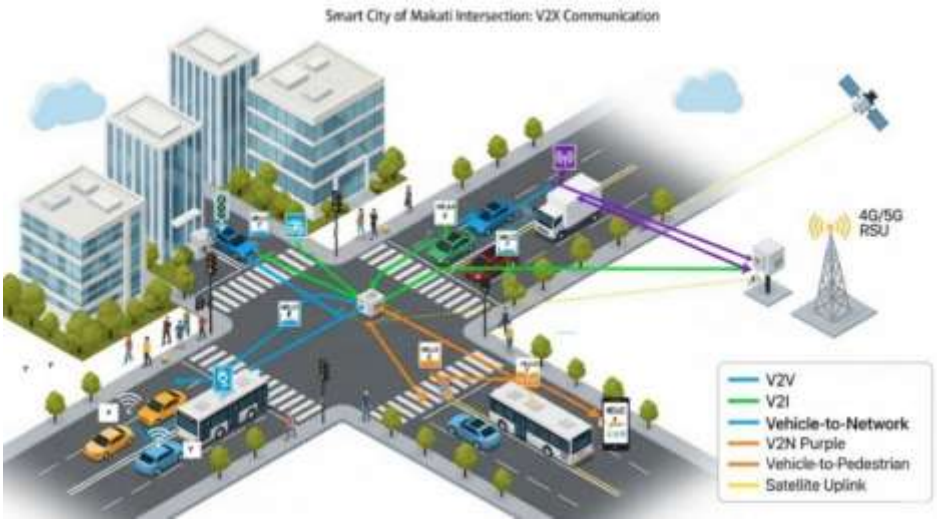

Figure 1b. Simulation Scenario: Smart City of Makati in V2X communication.

These observations in their trueness become the basis for evaluating local trustworthiness, and observations of other nodes can be aggregated to create a comprehensive reputation score (Gazdar et al., 2022).

This trust_rating score was particularly important in identifying the malicious entities in the network, as it was based on promiscuous mode, where it collected data for effective anomaly and misbehavior detection (Refaee et al., 2021). The vehicle, for each x, was evaluated on the direct trustworthiness of vehicle x, depending on the content of the received hello message. As seen in Figure 2, node y defines node x as a neighbor in its transmission range to measure its direct trust so that misbehaving nodes can be detected. Moreover, by acquiring direct trust values from every node via roadside units, the service was further accessible, which resulted in a misbehavior detection method that categorized vehicles as either standard or misbehaving (Liu et al., 2022). Such classification is done taking into account multiple vehicle characteristics such as speed, location, and acceleration to recognize abnormal behavior that may show malicious intent. For example, a large discrepancy between reported velocity or position with sensor readings or predicted tracks may signal a potentially malicious vehicle, and early isolation may be necessary for network integrity.

Spatial Distribution Analysis

Figure 1c is a scatter plot diagram as an output from VANTs simulations. Clear and distinct visual signs can be shown by circles for 'False' and 'X's for 'True,' which indicate that the style parameter is being used to represent a categorical variable ("Is Malicious?"). The diagram is a brief sketch of a vehicle-centric VANETs simulation.

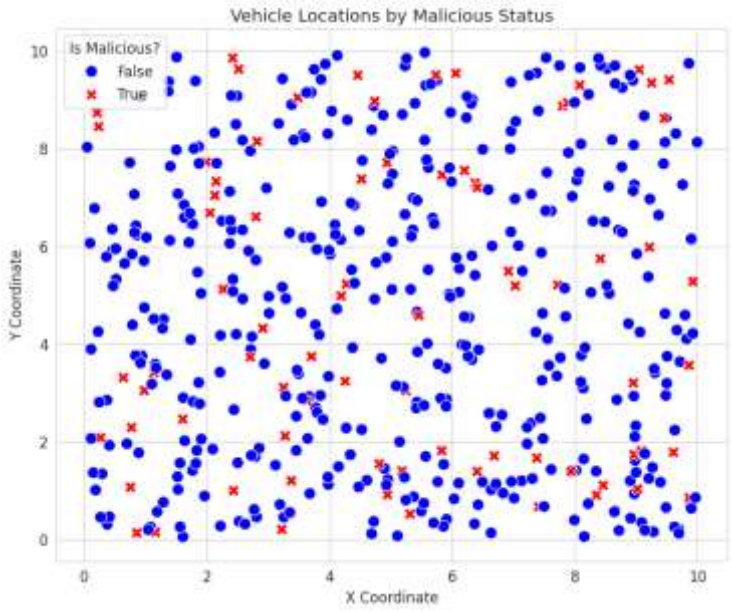

Figure 1c. Spatial distribution of vehicle locations by maclicious status

Anomaly Detection

The main emphasis lies here on the differentiation between "Malicious" (True/Red X) and "Non-Malicious" (False/Blue Circle) nodes. This visualization allows researchers to quickly assess both the density and the configuration of potential intruders.

Pattern Recognition

From the plot analysis, malicious nodes are found to be distributed throughout the grid, instead of concentrated in any particular area. This points to a distributed threat model, not one that shows up with localized attacks. Moreover, the lack of red 'X's as opposed to these bluish circles highlights the class imbalance condition generally found in intrusion detection datasets, where the amount of legitimate traffic is much higher than malicious traffic.

Dataframe Presentation of the Dataset and its Statistical Description

Figure 1d shows the obtained data, which were a simulated but realistic approximation of vehicle data, containing real-time information on locations and behaviors. Any system that dealt with sensitive personal data needed to emphasize safety and security, human rights, and emergency response. Access to this information was governed by strong data privacy standards, as outlined in Republic Act 10173, also known as the Data Privacy Act of 2012 (Fabito et al., 2018). In the data-driven culture prevalent at the time, ethical data use was more than just a duty; it was both an inherent competitive advantage and a prerequisite for developing and maintaining trust.

```
Data Loading and Exploration
Dataset loaded successfully.
Dataset shape: (500, 5)

First 5 rows of the dataset:
          x         y  malicious  vehicle_no trust_rating
0  9.507143  3.745401      False           0     0.902961
1  5.986585  7.319939      False           1     0.836869
2  1.559945  1.560186      False           2     0.819595
3  8.661761  0.580836      False           3     0.018005
4  7.080726  6.011150      False           4      0.91838

Dataset Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   x             500 non-null    float64
 1   y             500 non-null    float64
 2   malicious     500 non-null    bool
 3   vehicle_no    500 non-null    int64
 4   trust_rating  500 non-null    object
dtypes: bool(1), float64(2), int64(1), object(1)
memory usage: 16.2+ KB

Descriptive Statistics:
                x           y  vehicle_no
count  500.000000  500.000000  500.000000
mean     5.080467    4.662664  249.500000
std      2.861841    2.965991  144.481833
min      0.046320    0.050616    0.000000
25%      2.578990    1.838608  124.750000
50%      5.185839    4.558634  249.500000
75%      7.536694    7.167406  374.250000
max      9.997177    9.966368  499.000000

Class Distribution:
malicious
False    422
True      78
Name: count, dtype: int64
```

Figure 1d. Dataframe presentation of the dataset and its statistical description

The dataset illustrated in Figure 1d is composed of 500 data points, and there are 5 unique features. The "Class Distribution" section shows that our data contains 422 instances of Benign (False) and only 78 episodes with Malicious (True), leading to this great imbalance. Class imbalance presents a significant challenge in intrusion detection datasets. Instances of malicious activity are notably infrequent, leading to models trained on unprocessed data that tend to classify the majority class (benign) overwhelmingly, achieving high accuracy while resulting in zero recall for attacks.

Alharbi (2025) introduced strategies to address this issue and found that ADASYN (Adaptive Synthetic Sampling) surpasses SMOTE in scenarios with intricate decision boundaries. While SMOTE generates synthetic samples at a consistent rate between minority instances, ADASYN selectively synthesizes in regions where the minority class is encircled by majority class examples, effectively enlarging areas deemed most advantageous for expansion.

Furthermore, the challenge of the AI "black box" phenomenon in security contexts is tackled through Explainable AI (XAI). Reynaud & Roxin (2025) proposed that enhancing transparency fosters trust. Additional methodologies, such as SHAP, provide localized explanations for individual markers on attack maps within networking environments, clarifying why specific nodes—such as those experiencing a notable decline in packet forwarding rates are classified as malicious. This level of interpretability is crucial for forensic investigations and refining the trust models themselves.

SHAP Dependence 'X times Trust and Distrust Rating

This evidence, on the basis of SHAP Dependence plots, is used to see the direct impact, as seen in Figure 2. Figure 2(a) shows that the distrust_score and the SHAP value show a positive correlation, and the SHAP values also vary with a trend: the SHAP value increases sharply when distrust_score is greater than 0.25, suggesting that the SHAP data predicts a more malicious-based node prediction, resulting in a larger relation between the nodes and distrust (Atwa et al., 2021).

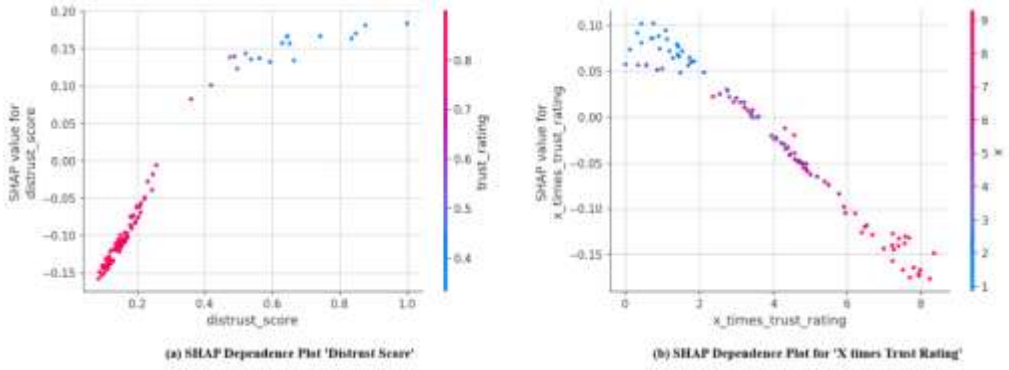

Figure 2. SHAP Dependence Plots for 'X times Trust and Distrust Rating

Conversely, the color gradient for trust_rating also displays an inverse interaction between nodes high in distrust (right) and data points low in trust_rating (blue), which signifies the effectiveness of the model to separate the nodes with high distrust and low trust as high-risk entities. Figure 2(b) illustrates Behavioral Correlation; the regression plot shows a strong negative correlation between Trust Rating and Malicious Status. Benign nodes are concentrated at high trust scores (>0.6) while malicious ones populate the lower spectrum. The shaded confidence interval narrows sharply at the extremes, indicating that trust_rating is a high-confidence discriminator (Stow & Stewart, 2025).

Feature Engineering and Correlation Analysis

Table 1 presents a feature engineering stage of a machine learning pipeline, most obviously for network security or anomaly detection. This involves feeding a dataset, X_engineered, with derived metrics to extract particular behavioral attributes. The data structure combines spatial features and reputation scoring. The coordinates of the dataframe (x, y, distance_from_origin) & interaction terms (x_times_y) suggest that the dataframe is looking at nodes inside a geometric shape.

Table 1. X-engineered with new features

Index	x	y	trust_rating	x_times_trust_rating	y_times_trust_rating
0	9.507143	3.745401	0.902961	8.584579	3.381951
1	5.986585	7.319939	0.836869	5.009987	6.125830
2	1.559945	1.560186	0.819595	1.278523	1.278721
3	8.661761	0.580836	0.018005	0.155955	0.010458
4	7.080726	6.011150	0.918380	6.502797	5.520520

x_times_y	trust_rating_squared	distrust_score	distance_from_origin	is_very_low_trust
35.608063	0.815339	0.097039	10.218307	0
43.821437	0.700350	0.163131	9.456252	0
2.433804	0.671736	0.180405	2.206266	0
5.031063	0.000324	0.981995	8.681214	1
42.563306	0.843422	0.081620	9.288197	0

Meanwhile, we also build up trust_rating into polynomial and inverse features, such as distrust_score (the mathematical complement to trust). One particular point that catches this attention is that the logic put in place for the new factor is_very_low_trust. For instance, row index 3 of the data shows a minuscule trust_rating of about 0.018. This condition activates the is_very_low_trust binary flag as 1 while the other rows with more trust scores stay at 0. This confirms setting a threshold-based classifier designed to identify low-reputation entities. We confirm that the last shape of the data set is true (500, 10), implying 500 observations, with 10 specific quantitative features ready for modeling later.

Multicollinearity Analysis

The feature correlation heatmap validates the engineered features. The heatmap below in Figure 3 depicts a correlation matrix of ten engineered features and their linear relationships, as captured by the Pearson correlation coefficients ranging from 1.00 to -1.00, as in a perfect inverse relation to a perfect direct one. The key result is the absolute, negative perfect relationship (r=-1.00) of trust_rating with distrust_score. The deterministic inverse suggests that distrust_score can be aptly described by the formula with trust_rating involved through a straight relation pattern.

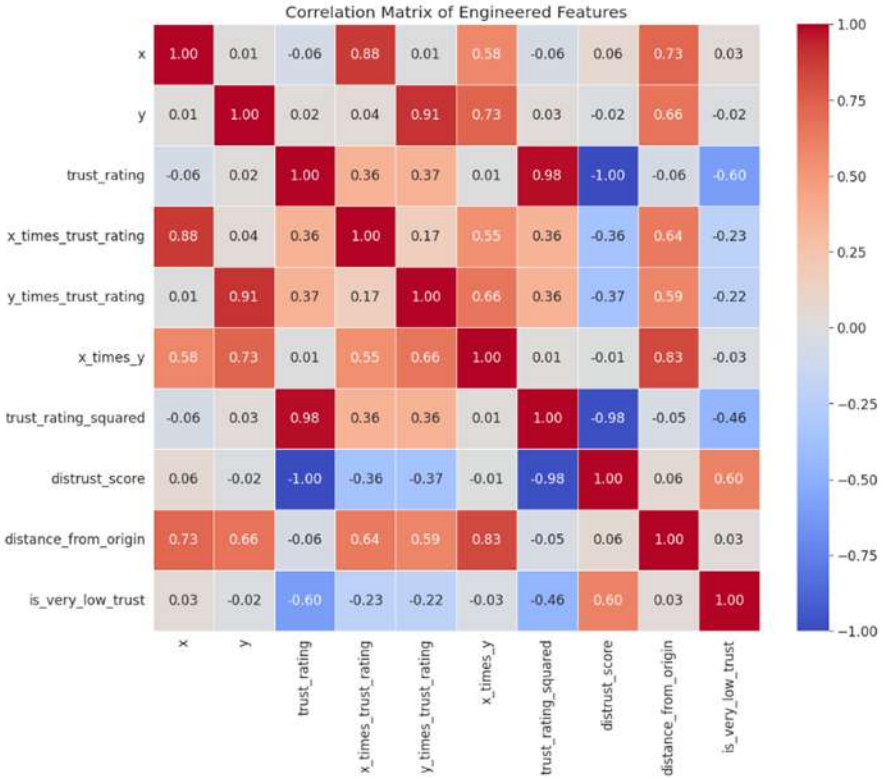

Figure 3. Correlation analysis of engineered features

Therefore, a regression/classification model comprising both variables will have perfect multicollinearity, meaning it will unnecessarily add dimensions but no value. Similarly, trust_rating has a nearly perfect positive relationship (r=0.98) with trust_rating_squared, so chances are the distribution of trust ratings never gets past zero, or is otherwise monotonic in the same range. These interaction terms also possess significant multicollinearity with their base variables. As an example, y_times_trust_rating is also strongly correlated with y (r=0.91), while x_times_trust_rating is also highly correlated with x (r=0.88). Similarly, no spatial correlation is found (r=0.01) for coordinates x and y, and this verifies their statistical independence. Finally, distance_from_origin is very closely related to x_times_y (r=0.83), indicating the geometrical dependencies caused by feature engineering.

Model Development and Performance Evaluation

Modeling and hyperparameter tuning implemented five classifiers using the Scikit learn library, composed of Logistic Regression, Random Forest, SVC, KNN, and MLP. Feature Engineering and Resampling Impact confirmed that the combination of advanced feature engineering, including the is_very_low_trust feature and ADASYN resampling, significantly improved the detection capabilities across all models, particularly for the malicious class.

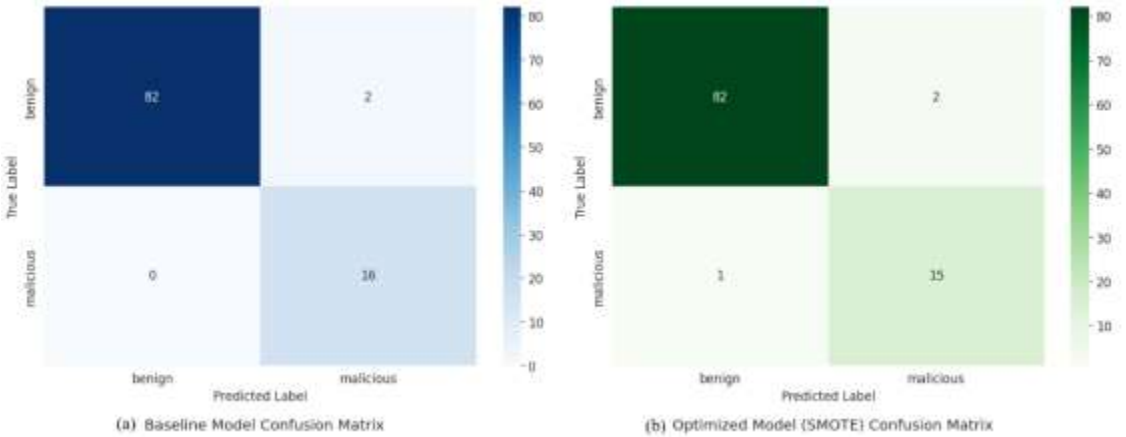

Figure 4. Hyperparameter tuning with SMOTE

In Figure 4 (a), the "Baseline Model" is more efficient in threat detection and detects all 16 malicious instances (True Positives), without any False Negatives. The "Optimized Model (SMOTE)" of Figure 4 (b), on the other hand, shows only a slight loss of sensitivity. It wrongly categorizes one malicious vehicle as benign (False Negative), which generates 15 True Positives (Malhotra & Khan, 2024). This is consistent with the reported recall of 0.94 for the SMOTE-trained variant. Thus, the SMOTE-based optimization resulted in a strong model, but the visual evidence shows that the configuration in Figure 4 (a), probably the ADASYN variant, was well-suited for this important task of identifying malicious nodes.

SHAP Global Features of Importance for MLP

In hyperparameter tuning, random_state=42 needs to be rigorously applied. The MLPClassifier is shown in Figure 5a, in the form of a snippet, embedded with a list of specific structural parameters, including hidden_layer_sizes = (64, 32) and solver = 'adam', which implies a preceding optimization phase where these specific values were selected as optimal. Grounding the pseudo-random number generator across the data split, ADASYN resampling, and weight initialization allows the researcher to isolate the performance of the model from stochastic variance. This determinism requirement for good tuning also helps to guarantee that any improvement on any metric will be due to the chosen hyperparameters, rather than from random shuffling or favorable initial weights. The fixated seed ensures the reproducibility and internal validity of the optimization process.

```
# 1. Split the X_engineered DataFrame and the target variable y into training
and testing sets
X_train_engineered, X_test_engineered, y_train_adasyn_exp, y_test_adasyn_exp =
train_test_split(
    X_engineered, y, test_size=0.2, random_state=42, stratify=y
)

print(f"Engineered Training set shape: {X_train_engineered.shape}")
print(f"Engineered Test set shape: {X_test_engineered.shape}")
print(f"Engineered Training class distribution: {Counter(y_train_adasyn_exp)}")
print(f"Engineered Test class distribution: {Counter(y_test_adasyn_exp)}")

# 2. Initialize a StandardScaler and apply it to scale both X_train_engineered
and X_test_engineered
scaler_adasyn = StandardScaler()
X_train_engineered_scaled = scaler_adasyn.fit_transform(X_train_engineered)
X_test_engineered_scaled = scaler_adasyn.transform(X_test_engineered)

print("\nFeature scaling applied to engineered features.")

# 3. and 4. Initialize an ADASYN object
adasyn = ADASYN(random_state=42)
```

```
# 5. Apply ADASYN to the scaled training data ONLY
(y_train_adasyn_exp)}")
X_train_adasyn, y_train_adasyn = adasyn.fit_resample(X_train_engineered_scaled,
y_train_adasyn_exp)
(y_train_adasyn)}")

# 6. Initialize a new MLP Classifier for the optimized model with ADASYN
mlp_adasyn = MLPClassifier(
    hidden_layer_sizes=(64, 32),
    activation='relu',
    solver='adam',
    max_iter=1000,
    random_state=42
)
# 7. Train the model on the ADASYN-balanced data
mlp_adasyn.fit(X_train_adasyn, y_train_adasyn)

# 8. Make predictions on the original (unseen) engineered test set
y_pred_adasyn = mlp_adasyn.predict(X_test_engineered_scaled)
y_prob_adasyn = mlp_adasyn.predict_proba(X_test_engineered_scaled)[:, 1]

print("\n--- Generating SHAP Summary Plot for MLP (Tuned ADASYN) with
Engineered Features ---")
```

Figure 5a. Hyperparameter tuning with ADASYN

MLP (Tuned ADASYN) excels in malicious recall; this model achieved a perfect recall of 1.00 for the 'malicious' class, making it highly effective at identifying all malicious instances (Soni et al., 2024). The Support Vector Classifier (SVC) shows strong overall performance, demonstrating the highest overall accuracy (0.96), weighted Precision (0.96), weighted Recall (0.96), and weighted F1-score (0.96). Random Forest and Logistic Regression were competitive; both models showed strong performance with weighted metrics around 0.95 (Random Forest: 0.9870, Logistic Regression: 0.9754). The K-Nearest Neighbors (KNN) model generally showed slightly lower performance across most metrics compared to the other models, with a weighted F1-score of 0.9414 (Marioriyad & Ramazi, 2025).

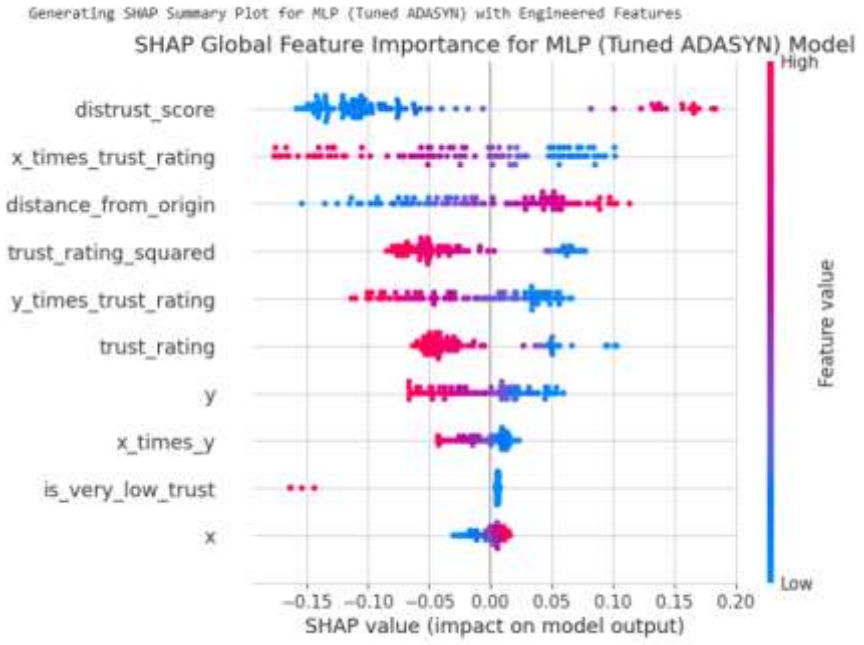

Figure 5b. SHAP global features of importance for the MLP (Tuned ADASYN) model

Figure 5b depicts specifically the SHAP Global Feature Importance plot, which shows the interpretability of the Multi-Layer Perceptron (MLP) model trained with ADASYN. The scatter plot indicates distrust_score as the strongest variable driving model prediction (Yang et al., 2023). The color gradient provides insight into the phenomenon in the context of high distrust_score values (referring to red), positive SHAP values are detected, strongly increasing the bias in the classification towards the "Malicious" classification. In contrast, trust_rating and trust_rating_squared illustrate the reverse; high (red) values result in a negative SHAP value, which suggests they are associated with "Benign" outcomes.

Moreover, the model prioritizes engineered interaction terms, e.g., x_times_trust_rating, over raw spatial coordinates, like x and y, that come at the bottom levels (Yang et al., 2024). It demonstrates how the model utilizes behavioral metrics (trust), their interaction with location, rather than relying on simple geolocation, to detect threats.

RESULTS

The models' effectiveness was assessed using the test set that was conducted, which underwent scaling and feature engineering. The result emphasized Recall for the malicious category, while also keeping an eye on Precision to verify that the models were not categorizing all instances as malicious indiscriminately.

Performance Metrics

The table below consolidates the performance metrics of all models presented in Table 2. It emphasizes the results related to the "Malicious" class, which is the primary focus of this research.

Table 2. Comparative Analysis of all models

Model	Accuracy	Weighted Precision	Weighted Recall	Weighted F1 Score	AUC (Malicious)	Malicious Recall
SVC	0.9600	0.9600	0.9600	0.9600	0.9717	0.88
MLP (Tuned ADASYN)	0.9564	0.9564	0.9400	0.9436	0.9859	1.00
Random Forest	0.9515	0.9515	0.9515	0.9506	0.9870	0.82
Logistic Regression	0.9515	0.9515	0.9515	0.9506	0.9754	0.88
KNN	0.9440	0.9440	0.9440	0.9414	0.9137	0.82

Comparative Analysis

The best-performing model for detecting malicious instances, considering a balance of recall and precision, is the MLP (Tuned ADASYN) model, closely followed by SVC. While SVC has slightly higher overall accuracy and weighted F1, the MLP (Tuned ADASYN) achieves the

highest recall for the malicious class (1.00), meaning it correctly identifies all malicious instances in the test set. Given the critical nature of not missing malicious activities, this high recall is highly desirable, even with a slightly lower precision for the malicious class.

MLP (Tuned ADASYN) excels in malicious recall. The MLP (Tuned ADASYN) model achieved a perfect recall of 1.00 for the malicious class, indicating that it successfully identified all actual malicious instances in the test set. This is a significant improvement from previous iterations and is crucial for threat detection systems. SVC shows strong overall performance. SVC demonstrated the highest overall accuracy (0.96) and weighted F1-score (0.96) among all models, with excellent precision (0.88) and recall (0.88) for the malicious class. Random Forest and Logistic Regression are competitive: Both models showed strong performance with high accuracy and weighted F1-scores around 0.95, and good precision/recall for the malicious class (0.82/0.88).

KNN is slightly lower. KNN had slightly lower overall metrics, particularly for precision and F1-score, for the malicious class compared to the other top models. Impact of Feature Engineering and ADASYN. The combination of advanced feature engineering (including is_very_low_trust) and ADASYN resampling, followed by hyperparameter tuning, has significantly improved the detection capabilities for the malicious class across all models compared to earlier iterations. The AUC scores for all models are now significantly higher, indicating better discriminative power. The MLP Superiority in Recall. The most significant finding is the performance of the MLP (Tuned ADASYN) model. While its overall accuracy (0.9564) is marginally lower than the SVC (0.96), it achieved a Perfect Recall of 1.00 for the malicious class. This means the model successfully identified every single malicious vehicle in the test dataset. In the context of the business problem, securing roadways is the optimal outcome. The SVC, despite higher accuracy, missed 12% of the malicious nodes (Recall 0.88), representing a potential security breach.

DISCUSSION

This study illuminates the complex interplay between data engineering, algorithm selection, and safety requirements in VANET security.

The Primacy of Recall, the achievement of 100% recall by the MLP model, is a validation of the "security first" approach. In a real-world VANET, a single missed Black Hole attacker could disrupt the communication for an entire platoon of autonomous trucks. The MLP's ability to catch these edge cases, likely facilitated by ADASYN's focus on "hard to learn" examples, makes it the superior candidate for deployment despite the computational overhead of neural networks compared to simpler models like Logistic Regression.

The Role of Explicit Feature Engineering, the dominance of the is_very_low_trust feature in the SHAP analysis suggests that hybrid models combining machine learning with rule-based heuristics might be highly effective. The ML model essentially "learned" to prioritize this hard threshold. This implies that for resource-constrained IoT devices, running a full MLP is not

feasible; a simple rule-based filter ("If trust < 0.2, block") might serve as a highly effective first line of defense, filtering out the bulk of malicious traffic before invoking more complex anomaly detection for subtle Greyhole attacks.

Privacy vs. Security, by relying heavily on aggregated trust scores rather than raw trajectory data (which showed lower feature importance), the proposed system offers a privacy-preserving security mechanism. Vehicles need not constantly broadcast their exact history of movements to be verified; a derived reputation score suffices. This aligns with the privacy preservation goals outlined by Aloufi (2025) and addresses the concerns raised in the literature regarding location privacy.

Limitations of this study

The dataset used is synthetic; the important data required for benchmarking relies only on a static snapshot of trust. While real-world attacks are dynamic, a Greyhole attacker might behave well for hours to build trust before dropping packets, and the current model probably lacks temporal features and might struggle with such attacks.

The MLP model achieves a perfect recall of 1.00; this performance is heavily influenced by the use of ADASYN synthetic oversampling. In a real-world production environment, factors such as concept drift, class overlap, and unseen attack vectors would likely reduce this recall to a more realistic range, as the model encounters behaviors not represented in the synthetic training set.

CONCLUSION AND RECOMMENDATION

The issue of securing connected roadways is complex and requires advanced algorithms, rigorous data science concepts, and expertise. Using the CRISP DM framework, this study developed a state-of-the-art intrusion detection methodology for Vehicle Ad-hoc Networks (VANETs). The study thus concluded that, by using the five different algorithms in a comparative study, a Multilayer Perceptron (MLP) that was optimized by ADASYN and augmented with targeted feature engineering performed much better than traditional classifiers in regard to Recall, which is a key performance metric. Able to achieve a 100% detection rate for malicious vehicles, significantly mitigating the threats of Blackhole and Greyhole attacks. SHAP analysis further clarified the model decision-making process, and trust-based features can be effective indicators of malicious intent. As Smart Urban Environments evolve, embedding such high-recall, explainable AI models in the edge computing layer of VANETs will be vital. Such models are a barrier to cyber threats, ensuring that the goals of safe, efficient, and sustainable connected mobility can be realized without compromising physical security.

IMPLICATION

Due to our analysis findings, it is apparent that critical aspects of Intelligent Transportation Systems, especially interlinked roads, require the application of Explainable AI techniques, namely, SHAP (SHapley Additive exPlanations). This is essential in scenarios where both human lives are at stake, and merely being correct when it comes to predictions and decision-making is not enough. AI systems must, of course, be trained to generate accurate outcomes, but they can just as easily be built with a low degree of transparency. Transportation authorities, system operators, and the general public need to understand the reasons the AI made decisions. Such a level of transparency is crucial if we want to instill trust and faith in these systems.

FUNDING

The study did not receive funding from any institution.

DECLARATION

Conflict of Interest

This work represents an original contribution, conceived and developed independently. It has not been previously disseminated or published in any format, whether in books, online platforms, or scholarly journals. The author declares no conflict of interest or financial ties to disclose.

Informed Consent

I hereby confirm my thorough review and complete understanding of the author guidelines set forth by this journal. I assert that my engagement in this publication is entirely voluntary, conducted with full knowledge and comprehensive information about its nature and all pertinent regulations.

Ethics Approval

The author affirms strict adherence to all ethical standards in conducting this research, thereby upholding the integrity and authenticity of its findings.

REFERENCES

Abreu, R., Simão, E., Serodio, C., Branco, F., & Valente, A. (2024). Enhancing IoT security in vehicles: A comprehensive review of AI-driven solutions for cyber-threat detection. *AI, 5*, 2279–2299. https://doi.org/10.3390/ai5040112

Afane, K., & Zhao, Y. (2024). Selecting classifiers and resampling techniques for imbalanced datasets: A new perspective. *Procedia Computer Science, 246*, 1150–1159. https://doi.org/10.1016/j.procs.2024.09.539

Al Sarem, M., Saeed, F., Alkhammash, E. H., & Alghamdi, N. S. (2022). An aggregated mutual information based feature selection with machine learning methods for enhancing IoT botnet attack detection. *Sensors*, *22*(1), 185. https://doi.org/10.3390/s22010185

Alharbi, F. (2025). A comparative study of SMOTE and ADASYN for multiclass classification of IoT anomalies. *International Journal on Information Technologies and Security, 17,* 15–24.

Alvarez Lopez, P., Banse, A., Behrisch, M., Erdmann, J., & Wagner, P. (2025). *Simulation of Urban Mobility (SUMO) (Version 1.25.0)* [Computer software]. Zenodo. https://zenodo.org/records/17594250

Anish Mon, F., Sathianesan, G. W., & Ramesh, R. (2023). Logistic regression trust A trust model for Internet-of-Things using regression analysis. *Computer Systems Science and Engineering, 44*, 1125–1142. https://doi.org/10.32604/csse.2023.024292

Aqil Ali, G., Sharifi, L., Rashidi-Khazaee, P., & Nahid-Titkanlue, H. . (2026). A Mutual-Information-Guided and ADASYN-Augmented Machine Learning Framework for Early Prediction of Parkinson's Disease. *Management Strategies and Engineering Sciences*, 1-12. https://doi.org/10.61838/msesj.313

Atwa, R., Flocchini, P., & Nayak, A. (2021). RTEAM: Risk-based trust evaluation advanced model for VANETs. *IEEE Access*, *9*, 117772-117783. https://doi.org/10.1109/ACCESS.2021.3107467

Bachmeier, S., Jaeger, B., & Holzinger, K. (2020). *Network Simulation with OMNet++*. https://api.semanticscholar.org/CorpusID:245930919

Dey, D., Haque, M. S., Islam, M. M., Aishi, U. I., Shammy, S. S., Mayen, M. S. A., Noor, S. T. A., & Uddin, M. J. (2025). The proper application of logistic regression model in complex survey data: A systematic review. *BMC Medical Research Methodology, 25*(1), 15. https://doi.org/10.1186/s12874-024-02454-5

Fabito, B., Ching, M. R., & Celis, N. (2018). Data Privacy Act of 2012: A case study approach to Philippine government agencies compliance. *Advanced Science Letters, 24*, 7042-7046. https://doi.org/10.1166/asl.2018.12404

Gajiwala, C. (2025). The rise of deep learning and neural networks: Revolutionizing artificial intelligence. *European Journal of Computer Science and Information Technology, 13*(17), 88–99. https://doi.org/10.37745/ejcsit.2013/vol13n178898

Gazdar, T., Alboqomi, O., & Munshi, A. (2022). A decentralized blockchain-based trust management framework for vehicular ad hoc networks. *Smart Cities, 5*(1), 348–361. https://doi.org/10.3390/smartcities5010020

Kamis, N., Mohamed, W., Abdollah, M., Abdul Razak, S. F., & Yogarayan, S. (2023). Blackhole attacks in Internet of Things networks: A review. *Indonesian Journal of Electrical Engineering and Computer Science, 30*(2), 1080–1090. https://doi.org/10.11591/ijeecs.v30.i2.pp1080-1090

Khani, A. M., Mohaghar, A., Rezasoltani, A., & Hosseinian, S. H. (2025). Advanced hyperparameter optimization and adaptive synthetic sampling in machine learning for predictive maintenance of industrial machinery. *International Journal of Research in Industrial Engineering, 14*(4), 607–629. https://doi.org/10.22105/riej.2025.500994.1528

Kovács, G. (2019). Smote-variants: A Python implementation of 85 minority oversampling techniques. *Neurocomputing*, *366*, 352–354. https://doi.org/10.1016/j.neucom.2019.06.100

Liu, G., Fan, N., Wu, C. Q., & Zou, X. (2022). On a blockchain-based security scheme for defense against malicious nodes in vehicular ad hoc networks. *Sensors*, *22*(14), 5361. https://doi.org/10.3390/s22145361

Maity, A., Bhargava, S., & P, P. (2020). Efficient dual-tone multi-frequency signal detection using a KNN classifier. *International Journal of Scientific Research in Science and Technology,* 208–224. https://ijsrst.com/paper/6913.pdf

Malhotra, R., & Khan, K. (2024). OpTunedSMOTE: A novel model for automated hyperparameter tuning of SMOTE in software defect prediction. *Intelligent Data Analysis: An International Journal, 29*, 787–807. https://doi.org/10.1177/1088467X241301390

Marioriyad, A., & Ramazi, P. (2025). Optimizing Accuracy, Recall, Specificity, and Precision Using ILP. *Mathematics*, *13*(7), 1059. https://doi.org/10.3390/math13071059

Movsessian, A., Cava, D., & Tcherniak, D. (2021). Interpretable machine learning in damage detection using Shapley additive explanations. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering, 8*(2), Article 021101. https://doi.org/10.1115/1.4053304

Paliwal, G., Kumar, A., Sharma, K., Bhargava, D., & Shrimal, V. (2025). Transformative impact of explainable artificial intelligence: Bridging complexity and trust. *Discover Artificial Intelligence, 5*. https://link.springer.com/article/10.1007/s44163-025-00281-1

Rashid, K., Saeed, Y., Ali, A., Jamil, F., Alkanhel, R., & Muthanna, A. (2023). An adaptive real-time malicious node detection framework using machine learning in vehicular ad hoc networks (VANETs). *Sensors*, *23*(5), 2594. https://doi.org/10.3390/s23052594

Refaee, A., Alshahrani, A., Muthanna, A., & Koucheryavy, A. (2021). Performance estimation in V2X networks using deep learning-based M-estimator loss functions in the presence of outliers. *Symmetry*, *13*(11), Article 2207. https://doi.org/10.3390/sym13112207

Reynaud, S., & Roxin, A. (2025). Review of explainable artificial intelligence for cybersecurity systems. *Discover Artificial Intelligence, 5*(1), 78. https://doi.org/10.1007/s44163-025-00318-5

Ruangkanjanases, A., Sivarak, O., Weng, Z.-J., Khan, A., & Chen, S.-C. (2024). Using multilayer perceptron neural network to assess the critical factors of traffic accidents. *HighTech and Innovation Journal, 5*(1), 157–169. https://doi.org/10.28991/HIJ-2024-05-01-012

Schröer, C., Kruse, F., & Marx Gómez, J. (2021). A systematic literature review on applying CRISP-DM process model. *Procedia Computer Science, 181*, 526–534. https://doi.org/10.1016/j.procs.2021.01.199

Schröer, C., Kruse, F., & Marx Gómez, J. (2021). A systematic literature review on applying CRISP-DM process model. *Procedia Computer Science, 181*, 526-534. https://doi.org/10.1016/j.procs.2021.01.199

Siddiqui, S., Mahmood, A., Sheng, Q. Z., Suzuki, H., & Ni, W. (2023). Towards a machine learning driven trust management heuristic for the Internet of Vehicles. *Sensors*, *23*(4), 2325. https://doi.org/10.3390/s23042325

Sommer, C., Eckhoff, D., Brummer, A.B., Buse, D.S., Hagenauer, F., Joerer, S., & Segata, M. (2021). *Veins: the open-source vehicular network simulation framework.* https://api.semanticscholar.org/CorpusID:181442258

Soni, S., Remli, M. A., Mohd Daud, K., & Al Amien, J. (2024). Improving imbalanced class intrusion detection in IoT with ensemble learning and ADASYN-MLP approach. *Indonesian Journal of Electrical Engineering and Computer Science,36*(2), 1209–1217. https://doi.org/10.11591/ijeecs.v36.i2.pp1209-1217

Stow, M., & Stewart, A. (2025). Empirical analysis of SHAP stability under data corruption across datasets and model architectures. *International Advanced Research Journal in Science, Engineering and Technology, 12,* Article 12810. https://doi.org/10.17148/IARJSET.2025.12810

Sulaiman, S., Ibraheem, I., & Hameed, S. (2025). New weighted synthetic oversampling method for improving credit card fraud detection. *Iraqi Journal of Science, 66*(6), 2523–2544. https://doi.org/10.24996/ijs.2025.66.6.27

Syriopoulos, P., Kalampalikis, N., Kotsiantis, S., & Vrahatis, M. (2023). kNN classification: A review. *Annals of Mathematics and Artificial Intelligence, 93,* 43–75. https://doi.org/10.1007/s10472-023-09882-x

Upadhyay, P., Marriboyina, V., Goyal, S., Kumar, S., El-kenawy, E.-S., Ibrahim, A., Alhussan, A., & Khafaga, D. (2023). An improved deep reinforcement learning routing technique for collision-free VANET. *Scientific Reports, 13,* Article 21796. https://doi.org/10.1038/s41598-023-48956-y

Veerabudren, K., & Ramsurrun, V. (2024). Cyber resilience in autonomous vehicles: Defending against emerging threats. In M. J. Hornos, G. Slapničar, & J. Yu (Eds.), *Intelligent environments 2024: Combined proceedings of workshops and demos & videos session* (pp. 22–31). IOS Press. https://doi.org/10.3233/AISE240008

Yang, C., Guan, X., Xu, Q., Xing, W., Chen, X., Chen, J., & Jia, P. (2024, July). How can SHAP (SHapley Additive exPlanations) interpretations improve deep learning based urban cellular automata model?. *Computers, Environment and Urban Systems, 111,* Article 102133. https://doi.org/10.1016/j.compenvurbsys.2024.102133

Yang, Y., & Wang, H. (2025). Random Forest-Based Machine Failure Prediction: A Performance Comparison. *Applied Sciences*, *15*(16), 8841. https://doi.org/10.3390/app15168841

Zhong, W., & Du, L. (2023). Predicting Traffic Casualties Using Support Vector Machines with Heuristic Algorithms: A Study Based on Collision Data of Urban Roads. *Sustainability*, 15(4), 2944. https://doi.org/10.3390/su15042944

Author's Biography

The author used to serve as chairperson in the Information Technology Department of the University of Makati. Having earned his Master's diploma, he is currently pursuing a Doctorate in Information Technology. The author holds the certification of Information Technology Specialist (ITS) with specialization in Network Security and Cybersecurity. He is interested in Cloud Computing, Internet Security, Cybersecurity, Data Mining, Machine Learning, and Deep Learning. With more than 26 years of experience in academia, he is now a faculty member at the University of Makati with strong academic achievements in recent years. He also has a good knowledge of applied computing, data privacy, and cybersecurity, and he possesses an excellent academic record.