Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

# A TeachingLearning-Based Marine Predators Algorithm for Feature Selection in Machine Learning-Driven Misbehavior Detection in Vehicular Ad Hoc Networks

**Abdulmumini Imam Ibrahim**
Department of Computer Science
Ramat Polytechnic Maiduguri
Maiduguri, Nigeria

**Fatima Muhammad Dawud**
Department of Computer Engineering
University of Maiduguri,
Maiduguri, Nigeria.
Department of Software Engineering, Al-Ansar University
Maiduguri, Maiduguri, Nigeria

***Abstract*** **- Vehicular Ad hoc Networks (VANETs) underpin the safety-messaging layer of Intelligent Transportation Systems, but their open wireless medium leaves them exposed to insider attacks that survive cryptographic authentication. Position falsification and Sybil attacks are particularly harmful because they corrupt the Basic Safety Messages (BSMs) on which collision-avoidance and platooning applications rely. This paper proposes a roadside-unit (RSU)-resident misbehavior detection framework that couples a teaching-learning-enhanced Marine Predators Algorithm (TL-MPA) for feature selection with k-Nearest Neighbors, Support Vector Machine, and fully connected deep neural network classifiers. BSMs from the VeReMi dataset are deduplicated, merged with ground truth, cleaned with a median absolute deviation estimator, and encoded, and two consecutive BSMs from each sender are fused into a single feature vector. TL-MPA then selects the most informative kinematic and positional features, namely velocity, position coordinates, and sender-receiver range. On VeReMi, the proposed pipeline achieves binary F1 scores above 99 percent for all five position-falsification attack types and raises multiclass precision and recall to 99.59 percent and 99.49 percent, outperforming unoptimized kNN and SVM baselines and a prior consecutive-BSM method on the same dataset. Ablation results show that two-BSM fusion is the main source of this gain, while feature selection reduces input dimensionality with comparable or better detection performance. The RSU-side design keeps feature selection and inference off resource-constrained on-board units. Limitations of single-dataset synthetic evaluation and directions for cross-dataset validation and edge deployment are discussed.**

***Keywords Vehicular Ad hoc Network (VANET); misbehavior detection; VeReMi dataset; position falsification; Sybil attack; feature selection; Marine Predators Algorithm; machine learning; deep learning; roadside unit***

## I. INTRODUCTION

### *A. Background and Motivation*

Road traffic injuries remain among the leading causes of death worldwide. The World Health Organization's Global Status Report on Road Safety 2023 estimates 1.19 million road traffic deaths annually and identifies road trauma as the leading cause of death for people aged 529 years, despite a 5% decline in fatalities since 2010 [1]. Intelligent Transportation Systems (ITS) are widely regarded as the most scalable route to reducing this toll, and Vehicular Ad hoc Networks (VANETs) are the communication substrate on which cooperative ITS safety applications forward-collision warning, intersection-movement assist, and platooning are built [2], [3]. VANETs enable vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication over IEEE 802.11p/WAVE Dedicated Short-Range Communication (DSRC), forming a hybrid topology in which fast-moving On-Board Units (OBUs) exchange Basic Safety Messages (BSMs) both with each other and with fixed Roadside Units (RSUs) connected to a central authority [3], [4].

Because most inter-vehicle safety messages must be broadcast with low latency, they are rarely encrypted beyond source authentication, and the open, decentralized nature of VANETs leaves them vulnerable to forgery, denial-of-service, Sybil, and position-falsification attacks [4], [5]. Cryptographic authentication can exclude outsiders who lack valid credentials, but it cannot, by itself, stop an authenticated insider from broadcasting false positional data. A vehicle with valid credentials can still lie about where it is. Detecting this class of behavior requires continuous, data-driven monitoring, which is where machine learning (ML)- and deep learning (DL)-based misbehavior detection systems (MDS) have shown consistent promise over the last decade [5][8].

### *B. Research Gap and Contributions*

Most existing VANET misbehavior detection studies train and evaluate classifiers on the single-BSM feature vector and place the detection burden on the OBU, which has limited memory, energy, and compute budget [9]. Comparatively little attention has been paid to (i) fusing information across consecutive BSMs to expose short-horizon kinematic inconsistencies, (ii) offloading detection to RSUs, which have materially higher processing capacity and a wired backbone to a central authority, and (iii) principled feature selection for the resulting feature space, as opposed to using all raw BSM/GPS fields. This paper addresses these gaps and makes the following contributions:

- A two-BSM data-fusion scheme that merges consecutive basic safety messages before feature extraction, exposing short-term positional and



(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

kinematic inconsistencies that single-message classifiers miss.

- An RSU-resident detection architecture that shifts feature selection and classification off the on-board unit, consistent with the asymmetric compute budgets of VANET nodes.
- A teaching-learning-enhanced Marine Predators Algorithm (TL-MPA) for feature selection that mitigates the premature convergence tendency of standard MPA and reduces the BSM feature set to the variables with the greatest discriminative value.
- An empirical comparison of kNN, SVM, and a fully connected deep neural classifier on the resulting feature set for both binary and five-class position-falsification classification on the VeReMi dataset, benchmarked against raw-feature and prior published baselines.

### *C. Paper Organization*

Section II reviews related work on VANET security requirements, cryptographic limitations, and ML/DL-based misbehavior detection, including recent (20232025) contributions. Section III presents the network and threat model. Section IV details the proposed methodology: dataset, preprocessing, TL-MPA feature selection, and classifiers. Section V describes the experimental setup. Section VI reports and discusses results. Section VII discusses limitations, and Section VIII concludes with directions for future work.

## II. RELATED WORK

### *A. VANET Security Requirements and Attack Taxonomy*

VANET security requirements are conventionally organized around availability, message integrity, confidentiality, source authentication, non-repudiation, and privacy/non-traceability [4], [10]. Attacks are correspondingly classified by the property they target: availability attacks include denial-of-service (DoS/DDoS), Sybil, and position falsification; integrity attacks include message tampering and replay; and confidentiality/privacy attacks include eavesdropping and traffic analysis [4], [11]. Because broadcast safety messages are time-critical and cannot tolerate heavyweight per-message cryptography, VANET security research has consistently identified insider misbehavior, legitimately authenticated nodes broadcasting false data, as the residual risk that authentication alone cannot close [4], [9], [12].

### *B. Cryptography-Based and Trust-Based Approaches*

Certificate-based authentication, conditional privacy-preserving authentication (CPPA) schemes, and trust/reputation frameworks reduce the attack surface from outsiders and provide accountability once misbehavior is flagged but rely on tamper-proof hardware or costly bilinear-pairing operations and cannot, on their own, distinguish an authenticated liar from a truthful node [13], [14]. This motivates data-driven, behavior-based detection layered on top of cryptographic authentication rather than as a replacement for it.

### *C. Machine Learning and Deep Learning-Based Misbehavior Detection*

A substantial body of work has applied supervised ML, chiefly SVM, kNN, Random Forest, and artificial neural networks, to VANET intrusion and misbehavior detection [5][9], [15]. So et al. showed that combining plausibility checks with kNN/SVM classifiers improves detection precision over plausibility checks alone [16]. Sharma and Jaekel proposed a consecutive-BSM feature representation for position-falsification classification on VeReMi, which is the closest prior baseline to the present work and is used for direct comparison in Section VI [17]. More recent studies have moved toward recurrent and attention-based architectures: Mahesh et al. proposed a HyperLSTM model evaluated on the larger VeReMi Extension dataset covering 20 attack categories [18], and Youness et al. introduced VeMisNet, which engineers DSRC-neighborhood-density and inter-message-timing features for LSTM/GRU/BiLSTM classifiers at scales up to two million samples while explicitly addressing the severe class imbalance (roughly 60% legitimate traffic) present in VeReMi-derived datasets [19]. Ensemble approaches combining bagging with kNN, Random Forest, and MLP classifiers have also been reported on VeReMi for RSU-level sequential-BSM detection [20]. Federated learning has emerged as a parallel research direction that keeps BSM data local to vehicles or RSUs: Mármol Campos et al. and Gurjar et al. both report competitive multiclass accuracy (9193%) on VeReMi/VeReMi Extension using federated aggregation strategies (FedAvg, FedProx, and FedYogi) [21], [22], and Qu and Cai proposed a federated intrusion detection system for vehicle-road cooperation scenarios [23]. Recent survey work situates these contributions and highlights that most reported gains have been demonstrated only in simulation, with limited attention to inference latency and on-RSU deployment cost, a gap this paper's RSU-resident design is intended to address [24], [25].

### *D. Feature Selection and Metaheuristic Optimization for Intrusion Detection*

Feature selection reduces training time and can improve generalization by removing non-contributing BSM/GPS fields (e.g., message ID and RSSI noise vectors) before classification. The Marine Predators Algorithm (MPA), a metaheuristic inspired by Lévy-flight and Brownian foraging strategies in marine predatorprey interactions, has been applied to feature selection and optimization problems across domains but is documented to be prone to premature convergence and entrapment in local optima [26]. Hybridization with teachinglearning-based strategies and crossover operators has been shown to mitigate this weakness and improve convergence in recent 20242025 studies applying enhanced MPA variants to intrusion detection in IoT and industrial settings [27][29]. The present work adopts this teaching-learning-enhanced MPA formulation for the VANET misbehavior detection setting, which, to the best of our knowledge, has not been directly benchmarked against un-optimized feature sets on VeReMi in prior published work.

### *E. Summary and Positioning*

Table I summarizes representative recent studies. Relative to this literature, the present work is distinguished by (a) RSU-side rather than OBU-side deployment, (b) two-BSM fusion prior to feature selection, and (c) an explicit ablation of metaheuristic feature selection versus raw features, evaluated consistently across kNN, SVM, and a deep neural classifier on the same VeReMi partitions.



(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

TABLE I. POSITIONING OF THE PROPOSED WORK RELATIVE TO REPRESENTATIVE RECENT VANET MISBEHAVIOR DETECTION STUDIES

| Ref. | Year | Technique | Dataset | Key Result / Gap Addressed |
|---|---|---|---|---|
| [16] | 2018 | Plausibility checks + kNN/SVM | VeReMi | +20% accuracy over plausibility checks alone; single-BSM features |
| [17] | 2022 | ML on consecutive-BSM pairs | VeReMi | Closest prior baseline; limited to 5 attack types, OBU-oriented |
| [18] | 2025 | HyperLSTM | VeReMi Extension | 20 attack categories; no RSU deployment analysis |
| [19] | 2025 | LSTM/GRU/BiLSTM + engineered spatiotemporal features | VeReMi Extension (up to 2M samples) | Addresses class imbalance; reports deployment constraints |
| [20] | 2025 | Bagging ensemble (kNN, RF, MLP) | VeReMi | RSU-level sequential-BSM detection |
| [21],[22] | 202425 | Federated learning (FedAvg/Prox/Yogi) | VeReMi Extension | Privacy-preserving, distributed training; 9193% accuracy |
| [27][29] | 202425 | Enhanced/hybrid MPA feature selection | IoT/IDS benchmarks (non-VANET) | Mitigates MPA premature convergence |
| This work | | TL-MPA feature selection + kNN/SVM/DNN, two-BSM fusion, RSU-resident | VeReMi | Combines feature-selection ablation with RSU-side two-BSM fusion |

## III. SYSTEM AND THREAT MODEL

### *A. Network Architecture*

The VANET architecture assumed here is the standard hybrid model: OBUs installed in vehicles communicate directly with nearby vehicles (V2V) and with fixed RSUs (V2I) over IEEE 802.11p/WAVE DSRC; RSUs are wired to a Central Authority (CA) responsible for periodic credential issuance and network-wide coordination [3], [4]. RSUs have materially higher computational capacity than OBUs and are not subject to the same energy constraints, which motivates placing feature selection and classifier inference at the RSU rather than the vehicle.

### *B. Threat Model*

We consider an insider adversary: a vehicle with valid cryptographic credentials that broadcasts falsified BSMs. Two attack families are considered. (i) Sybil attack: the adversary claims multiple simultaneous identities to simulate non-existent vehicles, which can trigger unwarranted collision-avoidance or congestion responses from honest vehicles [11]. (ii) Position falsification: the adversary reports an incorrect location, instantiated in the VeReMi dataset as five sub-types: Constant (ID 1), Constant Offset (ID 2), Random (ID 4), Random Offset (ID 8), and Eventual Stop (ID 16), described further in Section IV-A and Table II [30]. The detection objective is framed as both a binary problem (attacker vs. legitimate) and a five-class problem (identifying the specific attack subtype).

### *C. Design Rationale for RSU-Resident Detection*

Placing the trained classifier and feature-selection pipeline on the RSU rather than the OBU avoids adding inference latency and memory pressure to safety-critical on-vehicle computation. Once an RSU flags a vehicle's BSM stream as anomalous, it can (a) notify nearby vehicles directly, since it already possesses the wireless range to do so, and (b) forward the labeled instance to the shared database maintained by the central authority, incrementally improving the model. This design is consistent with, and extends, the node-centric/data-centric taxonomy of misbehavior detection schemes described in prior surveys [9], [24].

## IV. PROPOSED METHODOLOGY

### *A. Dataset: VeReMi*

We use the VeReMi dataset [30], generated with the LuST (Luxembourg SUMO Traffic) mobility scenario and the VEINS simulation framework and widely used as a comparable benchmark for VANET misbehavior detection [17], [18], [24]. VeReMi comprises 225 simulation runs spanning five position-falsification attack types, three traffic densities (low: 3539 vehicles, ~9081,144 BSMs; medium: 108 vehicles, ~4,489 BSMs; high: 519 vehicles, ~21,878 BSMs), and three attacker ratios (10%, 20%, 30%), each replicated five times with randomized seeds. Each scenario yields a ground-truth file and per-vehicle message logs containing both GPS and BSM records; BSMs carry the DSRC-received V2V signal fields, while GPS records provide each vehicle's own positional ground truth. Table II lists the attack identifiers used throughout this paper.

TABLE II. POSITION-FALSIFICATION ATTACK TYPES AND IDENTIFIERS IN THE VEREMI DATASET [30]

| Attack ID | Attack Name | Falsification Parameters |
|---|---|---|
| 1 | Constant | Fixed reported position (x=5560, y=5820) |
| 2 | Constant Offset | Fixed offset applied to true position (Δx=250, Δy=−150) |
| 4 | Random | New position drawn uniformly at random from the simulation area at every message |
| 8 | Random Offset | New Δx, Δy drawn uniformly from [−300, 300] around the true position at every message. |
| 16 | Eventual Stop | A genuine position was reported, then broadcast was frozen with stop probability increasing 0.025 per 10 Hz update. |
| 0 | Normal | No falsification (legitimate vehicle) |

### *B. Data Preprocessing*

- Deduplication: Each BSM is received by multiple vehicles, so duplicate log entries sharing a timestamp, sender ID, and message ID are removed prior to labeling.
- Ground-truth merge: deduplicated logs are joined with the scenario's ground-truth file to produce a supervised, labeled dataset.



(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

- Outlier filtering: inconsistent or missing field values are identified and filtered using a Median Absolute Deviation (MAD) estimator, which is more robust to the heavy-tailed noise present in raw BSM logs than a mean/standard-deviation approach.
- Categorical encoding: categorical fields (e.g., message/protocol type) are one-hot encoded into numerical form suitable for kNN, SVM, and neural network input.
- Two-BSM fusion: rather than classifying a single BSM in isolation, feature vectors from two temporally consecutive BSMs from the same sender are concatenated, exposing short-horizon inconsistencies (e.g., an implausible position jump) that a single message cannot reveal.

### *C. Feature Selection: Teaching-Learning-Based Marine Predators Algorithm (TL-MPA)*

The cleaned, labeled feature set is passed to a teaching-learning-enhanced Marine Predators Algorithm for feature subset selection. Standard MPA models three velocity-ratio regimes between predator (search agent) and prey (candidate feature subset): (i) an exploration phase, when the predator moves faster than the prey; (ii) a mixed exploitationexploration phase, when predator and prey move at comparable speed; and (iii) an exploitation phase, when the prey moves faster than the predator, at which point Lévy-flight steps dominate the search [26]. Positions are updated relative to an Elite matrix (the best-performing feature subset found so far) and a Prey matrix (the current population), with Fish Aggregating Device (FAD) effects periodically reinitializing a fraction of the population to reduce the risk of local-optimum entrapment. Because standard MPA is nonetheless prone to premature convergence, we hybridize it with a teachinglearning-based optimization step in which candidate solutions periodically move toward a population-mean-informed "teacher" solution before resuming MPA-style updates, consistent with recent enhanced-MPA formulations reported for intrusion detection tasks [27][29]. We use a population size of 10 and 100 iterations. TL-MPA retains the x, y, and z positional coordinates, velocity components, and inter-vehicle range (sender-to-receiver distance) as the most discriminative features, while eliminating position-noise and speed-noise vectors, RSSI, and message ID fields as low-contribution features.

### *D. Classification Models*

Three classifiers are trained and compared on the TL-MPA-selected feature set for both binary (attacker vs. legitimate) and five-class (attack-type) classification:

*k-Nearest Neighbors (kNN):* a non-parametric, instance-based classifier that assigns a query BSM pair to the majority class among its k nearest neighbors under Euclidean distance.

*Support Vector Machine (SVM):* trained with a radial basis function (RBF) kernel to accommodate the non-linear separability of the selected feature space.

*Proposed deep neural classifier:* a feed-forward network with an input layer that normalizes the TL-MPA-selected features, a stack of fully connected hidden layers with batch normalization and ReLU activation, and a softmax output layer trained with a cross-entropy loss. The network is trained for 30 epochs with a mini-batch size of 64 and a learning rate of 0.001, using an 80:20 train/test split.

### *E. Evaluation Metrics*

Performance is reported using precision, recall, F1-score, and overall/validation accuracy, computed from the confusion matrix (true positive TP, true negative TN, false positive FP, and false negative FN), where a positive label denotes an attacker and a negative label denotes a legitimate vehicle. The metrics are defined as

$$\text{Precision} = TP / (TP + FP) \quad (1)$$

$$\text{Recall} = TP / (TP + FN) \quad (2)$$

$$F1 = 2 \cdot \text{Precision} \cdot \text{Recall} / (\text{Precision} + \text{Recall}) \quad (3)$$

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (4)$$

The Matthews Correlation Coefficient (MCC) and Cohen's Kappa are also informative for the imbalanced multiclass setting; their use is discussed in Section VII.

## V. EXPERIMENTAL SETUP

Simulations were implemented in MATLAB 2021a on a Windows 10 workstation (Intel Core i5, 4th generation, 8 GB RAM). The preprocessed, TL-MPA-reduced dataset was split 80:20 into training and test partitions. Both high-density (108519 vehicles) and low-density (3539 vehicles) VeReMi scenarios were evaluated separately for the binary classification task, and a combined multiclass dataset spanning all five attack types, plus the normal class, was constructed for the multiclass task. The limitations of this single-split protocol are discussed in Section VII.

## VI. RESULTS AND DISCUSSION

### *A. Binary Classification: Sybil and Position-Falsification Detection with kNN*

Table III reports binary detection performance for the Sybil attack and for the constant and constant-offset position-falsification sub-types using the kNN classifier on the two-BSM feature representation.

TABLE III. KNN BINARY CLASSIFICATION RESULTS FOR SYBIL AND POSITION-FALSIFICATION ATTACK TYPES 1 AND 2

| Attack | Precision (%) | Recall (%) | F1-score (%) |
|---|---|---|---|
| Sybil | 95.27 | 94.52 | 94.89 |
| Constant (Type 1) | 96.27 | 97.52 | 96.89 |
| Constant Offset (Type 2) | 97.27 | 94.58 | 95.91 |

### *B. Binary Classification Across All Attack Types: kNN vs. SVM vs. Proposed Classifier*

Table IV compares kNN, SVM, and the proposed deep classifier across all five position-falsification subtypes under the combined (pooled-density) dataset. The proposed classifier matches or exceeds both baselines on every attack type, with the largest margin on Attack Type 1 (Constant), where it improves the F1-score by 2.3 points over kNN and 1.3 points over SVM. All three classifiers converge to near-ceiling performance (F1 $\geq$ 99.4%) on Types 8 and 16, where the falsification pattern (bounded random offset; frozen final position) produces feature distributions that are easily separable once two-BSM fusion is applied.



(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

TABLE IV. COMPARISON OF KNN, SVM, AND THE PROPOSED CLASSIFIER FOR BINARY CLASSIFICATION ACROSS ALL FIVE POSITION-FALSIFICATION ATTACK TYPES

| Attack Type | Classifier | Precision (%) | Recall (%) | F1-score (%) |
|---|---|---|---|---|
| Type 1 | k-NN | 96.27 | 97.52 | 96.89 |
| Type 1 | SVM | 97.24 | 98.52 | 97.87 |
| Type 1 | Proposed | 98.29 | 100.00 | 99.18 |
| Type 2 | kNN | 97.27 | 94.58 | 95.91 |
| Type 2 | SVM | 98.48 | 99.67 | 99.07 |
| Type 2 | Proposed | 99.24 | 100.00 | 99.62 |
| Type 4 | kNN | 99.50 | 98.67 | 99.08 |
| Type 4 | SVM | 100.00 | 98.72 | 99.36 |
| Type 4 | Proposed | 100.00 | 99.02 | 99.50 |
| Type 8 | kNN | 98.89 | 100.00 | 99.44 |
| Type 8 | SVM | 100.00 | 99.52 | 99.76 |
| Type 8 | Proposed | 100.00 | 100.00 | 100.00 |
| Type 16 | kNN | 100.00 | 100.00 | 100.00 |
| Type 16 | SVM | 99.85 | 100.00 | 99.92 |
| Type 16 | Proposed | 100.00 | 100.00 | 100.00 |

### *C. Multiclass Classification*

Table V reports five-class (attack-type) classification performance, including a comparison against the consecutive-BSM baseline of Sharma and Jaekel [17]. SVM performs comparatively poorly in the multiclass setting (61.15% recall), consistent with its known sensitivity to class overlap in one-vs-rest decomposition without careful kernel/parameter tuning; kNN performs modestly better. The proposed classifier attains 99.59% precision and 99.49% recall, outperforming both baselines and the prior published method.

TABLE V. MULTICLASS CLASSIFICATION COMPARISON ACROSS ALL FIVE ATTACK TYPES PLUS THE NORMAL CLASS

| Algorithm | Precision (%) | Recall (%) |
|---|---|---|
| kNN | 88.79 | 61.66 |
| SVM | 87.16 | 61.15 |
| Sharma & Jaekel [17] | 98.10 | 98.50 |
| Proposed | 99.59 | 99.49 |

### *D. Comparison with Prior Approaches and Ablation Against Raw Features*

To isolate the contribution of two-BSM fusion and feature selection, Table VI compares (i) a single-BSM, no-feature-selection baseline ("Original dataset"), (ii) a two-BSM fusion model without TL-MPA feature selection ("2BSM model"), and (iii) the full proposed pipeline (two-BSM fusion + TL-MPA feature selection + proposed classifier). The single-BSM baseline underperforms substantially on Types 2, 8, and 16 (precision as low as 2935%), confirming that a single message carries too little kinematic context to expose these falsification patterns. Two-BSM fusion alone closes most of this gap. TL-MPA feature selection then contributes the final few points, most visibly on Type 8 recall (97.2% to 100%) and on Type 16 precision and recall (96.8% and 95.2% to 100%), while maintaining comparable performance on Types 1, 2, and 4 with a reduced input feature set.

TABLE VI. ABLATION OF TWO-BSM FUSION AND FEATURE SELECTION AGAINST A SINGLE-BSM, NO-SELECTION BASELINE

| Attack | Metric | Original dataset (%) | 2BSM model (%) | Proposed (%) |
|---|---|---|---|---|
| Type 1 | Precision | 57.4 | 100.0 | 98.29 |
| Type 1 | Recall | 67.2 | 100.0 | 100.00 |
| Type 2 | Precision | 34.9 | 99.8 | 99.24 |
| Type 2 | Recall | 18.8 | 99.7 | 100.00 |
| Type 4 | Precision | 99.6 | 100.0 | 100.00 |
| Type 4 | Recall | 68.9 | 99.8 | 99.02 |
| Type 8 | Precision | 29.0 | 99.9 | 100.00 |
| Type 8 | Recall | 14.7 | 97.2 | 100.00 |
| Type 16 | Precision | 31.0 | 96.8 | 100.00 |
| Type 16 | Recall | 16.0 | 95.2 | 100.00 |

### *E. Overall Model Evaluation: Accuracy and Training Time*

Table VII compares overall model accuracy, validation accuracy, and training time. The proposed classifier achieves the highest accuracy (99.55%) and validation accuracy (98.64%) while requiring the least training time (24 s) of the three methods on this hardware, a result that should be interpreted with the caveats on hardware and single-run timing discussed in Section VII.

TABLE VII. OVERALL DETECTION ACCURACY, VALIDATION ACCURACY, AND TRAINING TIME FOR EACH CLASSIFIER

| Algorithm | Model Accuracy (%) | Validation Accuracy (%) | Training Time (s) |
|---|---|---|---|
| kNN | 95.07 | 93.58 | 114 |
| SVM | 92.20 | 92.20 | 35.8 |
| Proposed | 99.55 | 98.64 | 24 |

### *F. Discussion*

The results support three conclusions. First, temporal fusion of consecutive BSMs is the single largest contributor to detection performance, consistent with the intuition that position-falsification attacks are defined by implausible motion between messages rather than by any single message in isolation. Second, TL-MPA feature selection yields a further, smaller, but consistent improvement while reducing the dimensionality of the classifier input, which is directly relevant to RSU-side inference cost. Third, SVM's comparatively weak multiclass performance (61% recall) relative to its strong binary performance (>97% F1 on most attack types) suggests that the multiclass decision boundaries in this feature space are non-trivially overlapping for a one-vs-rest linear/RBF decomposition and a per-class confusion-matrix analysis, left for future work, would clarify which attack pairs are most frequently confused.

## VII. LIMITATIONS

The present study has several limitations that bound the generality of its conclusions.

- *Single, synthetic dataset:* all results are reported on VeReMi, a simulation-generated dataset with five attack types and three traffic densities. Generalization to the larger VeReMi Extension dataset [18], [19] or to an independently generated dataset has not yet been demonstrated.
- *Statistical validation:* the reported metrics are single-run point estimates obtained from one 80:20 split. k-fold cross-validation with mean and standard deviation, together with paired significance testing against each baseline, would strengthen the comparisons.
- *Model configuration:* the deep classifier was evaluated in a single configuration, and the sensitivity of the



(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

results to network depth, width, and other hyperparameters was not explored.

- *Class imbalance:* VeReMi-derived data are dominated by legitimate traffic [19]. The influence of this imbalance on the reported metrics, and of rebalancing strategies such as class weighting or SMOTE [31], was not analyzed.
- *Imbalance-robust metrics:* the Matthews Correlation Coefficient and Cohen's Kappa, which are more informative than accuracy under class imbalance, are not tabulated.
- *Deployment cost:* although RSU-side deployment is a central motivation of this work, per-message inference latency and memory footprint on RSU-class hardware were not measured.
- *Baselines:* the comparison is limited to unoptimized kNN and SVM baselines and the consecutive-BSM method of [17]. Direct comparison with recent methods such as [18] and [20][22] under identical data partitions remains for future work.
- *Timing measurements:* training times (Table VII) were measured in single runs on one commodity workstation and should be regarded as indicative rather than definitive.
- *Adversarial robustness:* adaptive attackers that mimic plausible kinematics to evade the selected features were not modeled.

## VIII. CONCLUSION AND FUTURE WORK

This paper presented an RSU-resident misbehavior detection framework for VANETs that combines two-BSM temporal fusion, teaching-learning-enhanced Marine Predators Algorithm feature selection, and supervised classification (kNN, SVM, and a deep feed-forward classifier) to detect Sybil and five position-falsification attack sub-types in the VeReMi dataset. The proposed pipeline improved binary F1 scores to above 99% on all five position-falsification attack types and multiclass precision/recall to 99.59%/99.49%, outperforming unoptimized baselines and a prior published consecutive-BSM method. Future work should (i) be validated on the VeReMi Extension dataset and, ideally, a second independently generated dataset; (ii) report cross-validated results with significance testing; (iii) benchmark on-RSU inference latency and memory footprint on representative edge hardware; (iv) explore federated variants of the proposed pipeline to preserve data locality across RSUs, consistent with recent federated VANET IDS work [21][23]; and (v) evaluate robustness against adversaries with partial knowledge of the detection feature set.

## DATA AVAILABILITY STATEMENT

The VeReMi dataset used in this study is publicly available [30]. The preprocessing and TL-MPA feature-selection code is available from the corresponding author upon reasonable request.

## CONFLICT OF INTEREST

The authors declare no conflict of interest.

## REFERENCES

[1] World Health Organization, *Global Status Report on Road Safety 2023*. Geneva, Switzerland: WHO, 2023.

[2] A. Dutta, L. M. Samaniego Campoverde, M. Tropea, and F. De Rango, "A comprehensive review of recent developments in VANET for traffic, safety & remote monitoring applications," *Journal of Network and Systems Management*, vol. 32, no. 4, Art. no. 73, 2024.

[3] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 11621182, Jul. 2011, doi: 10.1109/JPROC.2011.2132790.

[4] M. Arif, G. Wang, and M. Z. A. Bhuiyan, "A survey on security attacks in VANETs: Communication, applications and challenges," *Vehicular Communications*, vol. 19, Art. no. 100179, 2019.

[5] M. Almehdhar, A. Albaseer, M. A. Khan, M. Abdallah, H. Menouar, S. Al-Kuwari, and A. Al-Fuqaha, "Deep learning in the fast lane: A survey on advanced intrusion detection systems for intelligent vehicle networks," *IEEE Open Journal of Vehicular Technology*, vol. 5, pp. 869906, 2024, doi: 10.1109/OJVT.2024.3422253.

[6] X. Xu, Y. Wang, and P. Wang, "Comprehensive review on misbehavior detection for vehicular ad hoc networks," *Journal of Advanced Transportation*, vol. 2022, Art. no. 4725805, 2022, doi: 10.1155/2022/4725805.

[7] E. A. Shams, A. Rizaner, and A. H. Ulusoy, "Trust aware support vector machine intrusion detection and prevention system in vehicular ad hoc networks," *Computers & Security*, vol. 78, pp. 245254, 2018, doi: 10.1016/j.cose.2018.06.008.

[8] K. Rashid, Y. Saeed, A. Ali, F. Jamil, R. Alkanhel, and A. Muthanna, "An adaptive real-time malicious node detection framework using machine learning in vehicular ad-hoc networks (VANETs)," *Sensors*, vol. 23, no. 5, Art. no. 2594, 2023.

[9] N. A. Al-Khulaidi, A. T. Zahary, A. A. Al-Shargabi, and M. A. Hazaa, "Machine learning for intrusion detection in vehicular ad-hoc networks (VANETs): A survey," in *Proc. 4th Int. Conf. Emerging Smart Technologies and Applications (eSmarTA)*, 2024, pp. 110, doi: 10.1109/eSmarTA62850.2024.10639016.

[10] C. Tchepnda, H. Moustafa, H. Labiod, and G. Bourdon, "Vehicular networks security: Attacks, requirements, challenges and current contributions," *International Journal of Ambient Computing and Intelligence*, vol. 1, no. 1, pp. 3952, 2009, doi: 10.4018/jaci.2009010104.

[11] M. N. Mejri, J. Ben-Othman, and M. Hamdi, "Survey on VANET security challenges and possible cryptographic solutions," *Vehicular Communications*, vol. 1, no. 2, pp. 5366, 2014, doi: 10.1016/j.vehcom.2014.05.001.

[12] H. Setia, A. Chhabra, S. K. Singh, S. Kumar, S. Sharma, V. Arya, B. B. Gupta, and J. Wu, "Securing the road ahead: Machine learning-driven DDoS attack detection in VANET cloud environments," *Cyber Security and Applications*, vol. 2, Art. no. 100037, 2024.

[13] D. Manivannan, S. S. Moni, and S. Zeadally, "Secure authentication and privacy-preserving techniques in vehicular ad-hoc networks (VANETs)," *Vehicular Communications*, vol. 25, Art. no. 100247, 2020, doi: 10.1016/j.vehcom.2020.100247.

[14] W. El-Shafai, A. T. Azar, and S. Ahmed, "AI-driven ensemble classifier for jamming attack detection in VANETs to enhance security in smart cities," *IEEE Access*, vol. 13, 2025, doi: 10.1109/ACCESS.2025.3552544.

[15] F. A. Ghaleb, A. Zainal, M. A. Rassam, and F. Mohammed, "An effective misbehavior detection model using artificial neural network for vehicular ad hoc network applications," in *Proc. IEEE Conf. Application, Information and Network Security (AINS)*, Miri, Malaysia, 2017.

[16] S. So, P. Sharma, and J. Petit, "Integrating plausibility checks and machine learning for misbehavior detection in VANET," in *Proc. 17th IEEE Int. Conf. Machine Learning and Applications (ICMLA)*, 2018, pp. 564571.

[17] A. Sharma and A. Jaekel, "Machine learning based misbehaviour detection in VANET using consecutive BSM approach," *IEEE Open Journal of Vehicular Technology*, vol. 3, pp. 114, 2022, doi: 10.1109/OJVT.2021.3138354.

[18] T. R. Mahesh, G. Gangadevi, K. K. Mishra, A. Algarni, O. Saidani, and A. Kumar, "HyperLSTM for anomaly detection model for VANET



(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

security using the VeReMi extension dataset," *International Journal of Distributed Sensor Networks*, vol. 2025, Art. no. 6789771, 2025, doi: 10.1155/dsn/6789771.

[19] N. Youness, A. Mostafa, M. A. Sobh, A. M. Bahaa, and K. Nagaty, "VeMisNet: Enhanced feature engineering for deep learning-based misbehavior detection in vehicular ad hoc networks," *Journal of Sensor and Actuator Networks*, vol. 14, no. 5, Art. no. 100, 2025, doi: 10.3390/jsan14050100.

[20] B. K. Mekonen, L. Bane, and N. B. Fite, "Detection of false position attacks in VANETs through bagging ensemble learning," *PLOS ONE*, vol. 20, no. 8, Art. no. e0328829, 2025, doi: 10.1371/journal.pone.0328829.

[21] E. Mármol Campos, J. L. Hernández-Ramos, A. González-Vidal, G. Baldini, and A. Skarmeta, "Misbehavior detection in intelligent transportation systems based on federated learning," *Internet of Things*, vol. 25, Art. no. 101127, 2024, doi: 10.1016/j.iot.2024.101127.

[22] D. Gurjar, J. Grover, V. Kheterpal, and A. Vasilakos, "Federated learning-based misbehavior classification system for VANET intrusion detection," *Journal of Intelligent Information Systems*, vol. 63, no. 3, pp. 807830, 2025, doi: 10.1007/s10844-025-00920-0.

[23] Z. Qu and Z. Cai, "FEDSA-ResnetV2: An efficient intrusion detection system for vehicle road cooperation based on federated learning," *IEEE Internet of Things Journal*, vol. 11, no. 18, pp. 2985229863, 2024, doi: 10.1109/JIOT.2024.3410871.

[24] P. Kapoor and A. Prasad, "Machine and deep learning for VANET security: A survey of intrusion detection models and deployment realities," *International Journal of Computer Networks and Applications*, vol. 12, no. 5, pp. 720736, 2025, doi: 10.22247/ijcna/2025/43.

[25] A. Rodrigues, N. B. Rabah, A. Gorrab, and B. Le Grand, "A comprehensive review of explainable artificial intelligence to enhance intrusion detection in VANET networks," in *Proc. IEEE/ACS 21st Int. Conf. Computer Systems and Applications (AICCSA)*, 2024, pp. 16.

[26] M. A. Al-Betar, M. A. Awadallah, S. N. Makhadmeh, Z. A. A. Alyasseri, G. Al-Naymat, and S. Mirjalili, "Marine predators algorithm: A review," *Archives of Computational Methods in Engineering*, vol. 30, no. 5, pp. 34053435, 2023, doi: 10.1007/s11831-023-09912-1.

[27] Z. Wang, Y. Zhang, J. Yu, Y. Gao, G. Zhao, E. H. Houssein, and R. Zhong, "Multi-strategy enhanced marine predator algorithm: Performance investigation and application in intrusion detection," *Journal of Big Data*, vol. 12, no. 1, 2025, doi: 10.1186/s40537-025-01080-2.

[28] S. N. Makhadmeh, S. Fraihat, M. Awad, Y. K. Sanjalawe, M. A. Al-Betar, and M. A. Awadallah, "A crossover-integrated marine predator algorithm for feature selection in intrusion detection systems within IoT environments," *Internet of Things*, vol. 31, Art. no. 101536, 2025.

[29] R. Rai, K. G. Dhal, A. Das, and S. Ray, "An inclusive survey on marine predators algorithm: Variants and applications," *Archives of Computational Methods in Engineering*, vol. 30, no. 5, pp. 31333172, 2023, doi: 10.1007/s11831-023-09897-x.

[30] R. W. van der Heijden, T. Lukaseder, and F. Kargl, "VeReMi: A dataset for comparable evaluation of misbehavior detection in VANETs," in *Proc. Int. Conf. Security and Privacy in Communication Systems (SecureComm)*, Singapore, 2018, pp. 318337.

[31] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321357, 2002.



(This work is licensed under a Creative Commons Attribution 4.0 International License.)