# Intrusion Detection System in CAN Messages

Project Report Submitted
in partial fulfilment of the requirements for the ward of the

## Bachelor of Technology

by

**Abhimanyu Tripathi** (B222003)

**Anshuman Mahabhoi** (B422010)

**Saumyajeet Varma** (B522053)

Under the supervision of

**Dr. Puspanjali Mohapatra**


Department of Computer Science and Engineering
International Institute of Information Technology, Bhubaneswar

# APPROVAL OF THE VIVA-VOCE BOARD

Thu, 20 November 2025

Certified that the report entitled **"Intrusion Detection System in CAN Messages"** submitted by **Abhimanyu Tripathi (B22003)**, **Anshuman Mahabhoi (B422010)** and **Saumyajeet Varma (B522053)** to **International Institute of Information Technology, Bhubaneswar** in partial fulfillment of the requirements for **Technical Writing in Computer Science Engineering (7th Semester)** under the BTech Programme has been accepted by the examiners during the viva-voce examination held today.

(Supervisor) (Panel Head)

(Internal Examiner 1) (Internal Examiner 2)

# CERTIFICATE

This is to certify that the report entitled **"Intrusion Detection System in CAN Messages"** submitted by **Abhimanyu Tripathi (B222003)**, **Anshuman Mahabhoi (B422010)** and **Saumyajeet Varma (B522053)** to **International Institute of Information Technology, Bhubaneswar** is a record of bonafide project work under my supervision, and the report is submitted for end-semester evaluation of **Technical Writing, B.Tech, 7th Semester.**

Dr. Puspanjali Mohapatra
(Supervisor)

# DECLARATION

We certify that

1. The work contained in the report has been done by me under the general supervision of my supervisor.

2. The work has not been submitted to any other Institute for any degree or diploma.

3. I have followed the guidelines provided by the Institute in writing the thesis.

4. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

5. Whenever I have used materials (data, theoretical analysis, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.

6. Whenever I have quoted written materials from other sources, I have put them under quotation marks and given due credit to the sources by citing them and giving required details in the references.

| Abhimanyu Tripathi | Anshuman Mahabhoi | Saumyajeet Varma |
|---|---|---|
| (B222003) | (B422010) | (B522053) |

# ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to all those who have guided us throughout the development of this report. This work would not have been possible without the continuous guidance, invaluable insights, and dedicated support of our esteemed mentor, **Dr. Puspanjali Mohapatra**, who supervised us closely during the project.
We would also like to extend our sincere appreciation to our institution for providing the encouragement and resources necessary for this undertaking. This experience has allowed us to gain significant knowledge and exposure in our field, enriching our understanding and enhancing our skills.

Abhimanyu Tripathi
(B22003)

Anshuman Mahabhoi
(B422010)

Saumyajeet Varma
(B522053)

# Contents

# 1 Introduction

The rapid integration of electronic control units (ECUs) in modern vehicles has fundamentally transformed the automotive landscape, making contemporary cars complex cyber-physical systems. These ECUs are interconnected through the Controller Area Network (CAN), a lightweight and efficient communication protocol originally designed to ensure fast, reliable, and real-time data exchange between vehicle subsystems. However, despite its technical strengths, CAN was developed at a time when vehicular connectivity was minimal, and cybersecurity threats were largely nonexistent. As a result, the protocol lacks essential security mechanisms such as message authentication, encryption, and source verification. With the emergence of connected, autonomous, and software-defined vehicles, this absence of intrinsic security poses a significant challenge. Newer vehicles now integrate internet connectivity, telematics systems, Wi-Fi, Bluetooth, and Vehicle-to-Everything (V2X) communication, which collectively broaden the attack surface available to cyber adversaries. Thus, malicious individuals can exploit these external interfaces to infiltrate the vehicle's internal network and inject unauthorized CAN messages that may influence critical functions including steering, braking, and engine control. The increasing frequency and sophistication of such attacks emphasize the urgent need for robust intrusion detection systems (IDS) specifically tailored for automotive networks.

## 1.1 Need for Automotive Intrusion Detection

The vulnerability of the CAN protocol stems primarily from its lack of built-in security features. Since the protocol does not authenticate message sources, any compromised ECU or external device with access to the network can transmit arbitrary messages posing as legitimate components. This allows attackers to introduce falsified engine speed readings, spoof gear positions, override sensor values, or flood the bus to suppress legitimate communication, thereby creating potentially dangerous driving scenarios. Additionally, the widespread adoption of internet-enabled infotainment systems and wireless communication modules has resulted in vehicles being constantly exposed to external networks. Every wireless interface—from telematics units to keyless entry systems—introduces a possible entry point for cyberattacks. Unlike traditional computing environments, vehicular systems require strict real-time operation and cannot afford delays, making it impractical to rely on heavy cryptographic operations or conventional IT security techniques. The safety-critical nature of automotive systems further heightens the importance of continuous monitoring of CAN traffic to identify anomalies indicative of cyberattacks. Regulatory frameworks such as ISO/SAE 21434 and UNECE WP.29 have recognized these risks and now mandate that manufacturers implement vehicle cybersecurity measures, including detection and mitigation of malicious intrusions. These factors collectively underline the necessity for dedicated IDS solutions capable of analyzing CAN-bus

behavior and ensuring the integrity of communication within the vehicle.

## 1.2 Scope of the Study

The scope of this introductory study is focused on examining the cybersecurity vulnerabilities of CAN-bus networks and understanding the motivation, challenges, and existing research approaches related to in-vehicle intrusion detection. This report does not propose a new detection model; instead, it serves as a foundational analysis of the problem domain. It outlines why automotive IDS solutions are essential, the limitations of the CAN protocol, and the specific characteristics of cyberattacks that threaten in-vehicle systems. Additionally, the study explores the evolution of in-vehicle networking and discusses how increasing vehicle connectivity has simultaneously increased exposure to cyber threats. The literature reviewed in this report spans reinforcement learning-based IDS models, generative models for data augmentation, adversarial multi-agent IDS frameworks, and deep learning techniques tailored for CAN-bus environments. By summarizing existing research and identifying common gaps, the report establishes a strong contextual basis for further exploration of advanced IDS methodologies for securing modern vehicles.

## 1.3 Evolution of In-Vehicle Network Security

Automotive communication systems have undergone substantial evolution over the past three decades, each phase bringing enhanced functionality but also new cybersecurity concerns. In the early years, vehicles contained only a handful of ECUs, typically operating in isolation and responsible for basic mechanical functions. During this era, external connectivity was almost nonexistent, and the risk of cyber intrusion was negligible. As vehicle functionality grew more complex, manufacturers introduced the CAN bus to interconnect ECUs responsible for powertrain, braking, transmission, and comfort systems. Although CAN efficiently supported real-time communication, its designers did not incorporate security mechanisms because the system was intended to be closed and physically inaccessible to attackers. The next phase of evolution introduced connected car technologies. Infotainment systems capable of internet access, telematics units, GPS navigation, and Bluetooth interfaces created communication pathways that extended beyond the physical boundary of the vehicle. Academic and industry researchers soon demonstrated that these external interfaces could be exploited to gain remote access to the internal CAN bus. The widely publicized Jeep Cherokee hack exemplified how an attacker could manipulate critical driving functions remotely by exploiting vulnerabilities in the vehicle's wireless systems. The move toward software-defined and autonomous vehicles further amplified these challenges. Autonomous systems depend on a massive network of sensors, cameras, LiDAR, and high-performance ECUs that must communicate seamlessly to ensure safe decision-making. This dense digital ecosystem creates

numerous vectors for cyberattacks targeting both perception and control mechanisms. Consequently, the evolution of in-vehicle networks has revealed an urgent need for sophisticated intrusion detection approaches capable of monitoring CAN traffic, detecting deviations from normal patterns, and responding to malicious activity in real time.

# 2 Literature Survey

## 2.1 MAFSIDS: A Reinforcement Learning-Based Intrusion Detection Model

### 2.1.1 Overview

[1] is an intrusion detection framework that combines graph-based deep learning with reinforcement learning to achieve efficient feature selection and high classification performance on large-scale intrusion datasets. The central idea of this work is to reduce redundant features through intelligent selection while maintaining strong detection accuracy across diverse network attack types.

### 2.1.2 Methodology

The approach begins with the use of a Graph Convolutional Network (GCN) to understand relational dependencies among dataset features. This graph-based embedding helps identify correlations and provides a structured representation of the data. Feature selection is performed through a Multi-Agent Feature Selection (MAFS) framework, where reinforcement learning agents collaboratively determine which features should be retained or discarded. The final classification stage employs a Deep Q-Network (DQN) trained on the optimized feature set to categorize traffic instances as normal or malicious.

### 2.1.3 Results

The model demonstrates strong performance on both CSE-CIC-IDS2018 and NSL-KDD datasets, achieving approximately 96.8% and 99.1% accuracy respectively. The reinforcement-learning-driven feature selection mechanism reduces redundant attributes by nearly 80%, leading to significant computational savings without degrading detection accuracy.

### 2.1.4 Strengths

The principal advantage of MAFSIDS lies in its combination of structural feature learning and dynamic RL-based optimization, enabling effective dimensionality reduction and improving system efficiency. The use of GCN allows the system to capture complex inter-feature relationships that traditional feature selection methods often miss.

### 2.1.5 Limitations

Despite its strengths, the model is computationally expensive due to the involvement of multiple RL agents and GCN training. Its reliance on predefined reward signals makes it sensitive to design choices, and the framework may require extensive tuning when applied to datasets with different feature distributions.

## 2.2 ID-RDRL: Recursive Deep Reinforcement Learning for Intrusion Detection

### 2.2.1 Overview

[2] aims to enhance intrusion detection by framing feature selection as a learning problem, enabling the system to recursively identify the most informative subset of attributes. This is intended to improve classification quality while reducing computational overhead in handling large feature spaces.

### 2.2.2 Methodology

The system integrates Recursive Feature Elimination (RFE) with a Deep Q-Network (DQN). RFE provides an initial estimate of feature importance, while the DQN agent treats each possible selection as a state and learns through trial-and-error which features should be retained. The agent receives reward signals based on classification accuracy and iteratively converges toward an optimal set of features. A final classifier is trained on the refined dataset to detect different types of attacks.

### 2.2.3 Results

Experiments conducted on NSL-KDD and UNSW-NB15 datasets show high accuracy, with values reaching 98.2% and 96.5% respectively. F1-scores also remain consistently high, demonstrating strong performance across both balanced and imbalanced datasets.

### 2.2.4 Strengths

The integration of RL enables ID-RDRL to discover non-linear feature interactions and adapt automatically to the dataset's structure. Its recursive elimination strategy ensures robustness and prevents over-reliance on noisy or redundant features.

### 2.2.5 Limitations

A key limitation lies in the model's high training cost, as multiple evaluations are required during the RL-driven elimination process. Additionally, the final performance heavily depends on the reward formulation and the base learner used in the RFE.

## 2.3 Deep Reinforcement Learning IDS Using CIC-IDS2017

### 2.3.1 Overview

[3] employs deep reinforcement learning to build an adaptive intrusion detection system capable of handling dynamic and evolving network threats. The study uses the CIC-IDS2017 dataset, which offers realistic and diverse network traffic patterns.

### 2.3.2 Methodology

The model encodes network flow characteristics into state representations for a Deep Q-Network (DQN). The agent learns classification policies through reward-driven optimization, where correct classifications earn positive rewards and incorrect decisions produce penalties. Over time, the system refines its internal decision boundaries without the need for explicit retraining.

### 2.3.3 Results

The DRL-based system achieves approximately 94.5% accuracy and maintains an F1-score of around 93%. These results suggest that reinforcement learning can adapt well to shifting traffic patterns and detect several attack categories, including DDoS and brute force attacks.

### 2.3.4 Strengths

The primary strength of this model is its ability to adjust dynamically to changes in traffic behavior, making it suitable for real-time environments. Reinforcement learning reduces the need for frequent retraining, which is beneficial in large-scale networks..

### 2.3.5 Limitations

A significant limitation is that RL models can exhibit unstable learning patterns if the reward structure is not carefully designed. In addition, performance relies heavily on effective state representation, which may be challenging to construct for complex traffic scenarios.

## 2.4 XGBoost-Based Feature Selection for UNSW-NB15

### 2.4.1 Overview

[4] investigates how feature selection based on XGBoost can improve the efficiency and performance of traditional machine learning models on the UNSW-NB15 dataset.

### 2.4.2 Methodology

XGBoost is used to compute feature importance scores, which serve as criteria for pruning low-value features. After reducing dimensionality, several classifiers—including decision trees, ANN, logistic regression, SVM, and kNN—are trained on the optimized dataset and evaluated on both binary and multiclass tasks.

### 2.4.3 Results

The study reports improved performance for certain models, most notably a rise in decision tree accuracy from approximately 88% to over 90%. Feature selection also reduces computation time.

### 2.4.4 Strengths

The method is simple yet effective, demonstrating that feature-selection can enhance accuracy and efficiency, especially in resource-limited environments.

### 2.4.5 Limitations

Reliance on XGBoost's importance scoring may lead to the removal of subtle features that contribute to non-linear patterns. The approach may not generalize equally well across all classifiers.

## 2.5 Zero-Day Detection Using Autoencoder-Based Hybrid Models

### 2.5.1 Overview

[5] focuses on identifying unknown attacks using anomaly detection techniques that rely on autoencoder reconstruction behavior, combined with supervised learning classifiers

### 2.5.2 Methodology

An autoencoder is trained exclusively on benign data to learn its underlying structure. When anomalous inputs are processed, their reconstruction error is significantly higher. These errors are then used as input features for Random Forest and XGBoost models, forming hybrid classifiers capable of detecting zero-day attacks.

### 2.5.3 Results

The model demonstrates exceptional accuracy on the CIC-MalMem-2022 dataset, with Random Forest-AE achieving 99.9892% accuracy and XGBoost-AE achieving 99.9741% accuracy.

### 2.5.4 Strengths

The approach excels at detecting previously unseen threats because it relies on anomaly-based modeling instead of signature-based classification

### 2.5.5 Limitations

Autoencoders can inadvertently reconstruct certain malicious patterns too well, reducing anomaly sensitivity. Performance also depends on how clean the benign training data is.

## 2.6 Attention-CNN-LSTM IDS for In-Vehicle Networks

### 2.6.1 Overview

[6] develops an IDS optimized for automotive environments by integrating convolutional networks, recurrent sequence modeling, and attention mechanisms.

### 2.6.2 Methodology

CAN traffic is transformed into temporal sequences. CNN layers extract spatial features from each timestep, LSTM layers model sequential dependencies, and the attention mechanism highlights the most relevant segments contributing to anomalies. The model is trained using labeled in-vehicle datasets.

### 2.6.3 Results

The system achieves high performance, including 99.43% accuracy and near-perfect ROC-AUC scores, demonstrating suitability for real-time automotive IDS applications.

### 2.6.4 Strengths

Its hybrid architecture effectively captures both local and temporal patterns, while attention improves interpretability and precision.

### 2.6.5 Limitations

The model demands considerable computational power and large labeled datasets, which may limit deployment on embedded systems.

## 2.7 Attention-CNN-LSTM IDS for In-Vehicle Networks

### 2.7.1 Overview

[7] proposes a novel, lightweight, unsupervised Intrusion Detection System (IDS) designed specifically to secure in-vehicle Controller Area Networks (CAN). Because CAN

buses lack built-in security features, they are vulnerable to various cyberattacks (e.g., spoofing, DoS). The authors designed an autoencoder-based model intended for real-time, on-device implementation. Unlike many complex deep learning models, this system is optimized not just as software, but specifically for hardware deployment on resource-constrained vehicle electrical control units (ECUs) via an FPGA (Field Programmable Gate Array).

### 2.7.2 Methodology

Unsupervised Learning:* The autoencoder model was trained *exclusively on normal CAN data without requiring labeled attack data during the primary training phase. Instead of using multiple thresholds for different attack types, the researchers used a Gaussian Kernel Density Estimation (KDE) function. They fed a small portion of attack data through the trained autoencoder to analyze reconstruction errors and calculate a single, optimal detection threshold.

### 2.7.3 Results

The system achieved an impressive average accuracy of 99.2%, precision of 99.2%, recall of 99.1%, and an F1-score of 99.2% across four distinct attack scenarios (Denial of Service, Fuzzy, Gear spoofing, and RPM spoofing).

### 2.7.4 Strengths

The proposed model is designed with efficiency and practicality at its core. By leveraging a single anomaly threshold, it eliminates the need for multiple parallel detection rules or complex multi-class decision systems, significantly reducing computational overhead. Its lightweight architecture minimizes power consumption and logic gate usage, making it highly suitable for deployment on resource-constrained automotive ECUs. The hardware-optimized design enables real-time processing of CAN frames at wire speed, ensuring that intrusion detection does not interfere with critical vehicle functions. Additionally, since the autoencoder learns only normal traffic behavior, the system remains robust against zero-day or previously unseen attacks by detecting deviations from established patterns.

### 2.7.5 Limitations

While the training of the autoencoder is fully unsupervised (using only normal data), the mathematical calculation of the single threshold using Gaussian KDE still required a subset of known attack data. This means true ”zero-day” performance might depend heavily on how well that threshold generalizes to entirely novel attack vectors not used in the KDE calculation.

## 2.8 Comparitive Study

Table 1: Comparison of Research Papers on Intrusion Detection Systems

| Publisher and Year | Author | Paper Title | Overview | Accuracy/Results |
|---|---|---|---|---|
| Springer 2023 | Kezhou Ren, Yifan Zeng, Zhiqin Cao, Yingchao Zhang | MAFSIDS: Reinforcement Learning-Based IDS | 80% feature reduction; efficient hybrid IDS using GCN + Multi-Agent Feature Selection + DQN | Accuracy: 99.82% |
| Nature Portfolio 2022 | Kezhou Ren, Yifan Zeng, Zhiqin Cao, Yingchao Zhang | ID-RDRL: Recursive Deep Reinforcement Learning IDS | Dynamic RL-driven feature selection using RFE + DQN | Accuracy: 99.7% |
| MDPI 2024 | Hooman Alavizadeh / V.K. Javvaji | Deep Reinforcement Learning IDS | Adaptive real-time learning using Deep Q-Network (DQN) | Accuracy = 94.5%, F1 score: 93% |
| Springer 2020 | Sydney M. Kasongo, Yanxia Sun | XGBoost-Based Feature Selection IDS | Feature reduction improved performance using XGBoost feature ranking + ML models | Accuracy: 90.85% |
| PLOS ONE 2024 | Zhen Dai, et al. | Zero-Day Detection Hybrid AE Models | Zero-day anomaly detection using Autoencoder + RF / XGBoost | Accuracy: 98% |
| Springer 2024 | A. Taneja, G. Kumar | ACL-IDS for In-Vehicle Networks | Lightweight real-time IDS using CNN + LSTM + Attention | Accuracy: 99.63% |
| MDPI | Donghyeon Kim, Hyungchul Im, Seongsoo Lee | Adaptive Autoencoder-Based Intrusion Detection System with Single Threshold for CAN Networks | A lightweight, unsupervised IDS designed for real-time CAN networks. It uses an autoencoder trained exclusively on normal data to detect attack data. | Accuracy: 99.2%, precision: 99.2%, recall: 99.1%, F1 score: 99.2% |

# 3 Motivation

## 3.1 Growth of Intelligent Transportation Systems

Modern vehicles are rapidly transitioning into intelligent and automated systems equipped with advanced Electronic Control Units (ECUs), sensors, and communication modules. This transformation has increased dependency on in-vehicle networks such as the Controller Area Network (CAN) bus for real-time data exchange and cooperative decision-making. As vehicles continue to evolve into connected and autonomous platforms, ensuring secure internal communication has become a critical necessity to safeguard passenger safety and system reliability.

## 3.2 Vulnerabilities in the CAN Communication Protocol

The CAN bus protocol, despite being the backbone of intra-vehicle communication, lacks essential cybersecurity features such as message authentication and encryption. It operates on a broadcast mechanism where all ECUs receive messages without verifying their source. Consequently, attackers can easily inject malicious messages, modify existing data, or flood the network, triggering safety-critical malfunctions like disabling brakes, altering RPM readings, or manipulating gear signals. These vulnerabilities highlight the urgent need for intelligent and robust anomaly detection approaches.

## 3.3 Limitations of Existing Intrusion Detection Approaches

Traditional Intrusion Detection Systems (IDS) based on rule-based techniques, signature matching, and statistical analysis fail to detect new or evolving attack patterns. Similarly, supervised machine learning methods require large labeled attack datasets, which are costly, time-consuming to collect, and often unavailable. Many existing models struggle with:

- High false-positive rates due to complex CAN traffic dynamics
- Poor adaptability to unseen attacks
- Manual feature engineering requirements
- Inability to operate effectively in real-time environments

These limitations strongly motivate the need for an adaptive and scalable detection framework.

## 3.4 Need for Advanced AI-Based Intrusion Detection

Deep learning-based anomaly detection has shown significant potential in identifying subtle deviations in CAN traffic. However, models trained only on normal or limited attack data lack generalization capabilities. Generative Adversarial Networks (GANs) enable learning normal traffic distributions and generating synthetic attack samples, whereas Reinforcement Learning can optimize feature selection automatically, enhancing model performance. A hybrid approach combining RL and GAN can:

- Learn complex internal structures of CAN data
- Detect previously unseen attacks efficiently
- Reduce dependency on large labeled datasets
- Improve robustness and detection accuracy

# 4 Objectives

## 4.1 To Strengthen Security in Modern Intelligent Vehicles

As automotive systems evolve toward connected and autonomous architectures, ensuring secure in-vehicle communication has become essential for passenger safety and operational reliability. A key objective of this research is to support the development of secure transportation ecosystems by enabling early detection of malicious behaviors that may compromise critical vehicular control systems.

## 4.2 To Address Vulnerabilities in CAN Bus Communication

The CAN bus protocol, despite its efficiency and reliability, lacks built-in cybersecurity mechanisms such as encryption and message authentication. This exposes the communication network to cyber-attacks including message injection, spoofing, and DoS attacks. Therefore, an important objective is to investigate existing weaknesses in CAN communication and propose effective mechanisms for detecting abnormal traffic patterns that could pose safety risks.

## 4.3 To Improve the Effectiveness of Intrusion Detection Systems

Traditional IDS approaches often struggle to detect evolving attack types and produce high false alarm rates due to the dynamic nature of CAN traffic. This research aims to develop a more accurate and adaptive intrusion detection strategy capable of identifying known and unknown attack patterns with improved precision and reliability, overcoming limitations of existing rule-based and statistical detection methods.

## 4.4 To Utilize Data-Driven Techniques for Enhanced Anomaly Detection

With the increasing availability of vehicle CAN datasets, machine learning and deep learning techniques provide new opportunities for intelligent cybersecurity solutions. This research seeks to explore advanced data-driven methodologies for analyzing CAN traffic behavior and identifying anomalies by modeling real-world communication patterns rather than relying solely on handcrafted rules.

## 4.5 To Facilitate Real-Time and Scalable Implementation

An additional objective is to ensure that the proposed detection method is computationally efficient and capable of operating in real time without interrupting vehicular

communication. The research aims to design a scalable and practical intrusion detection solution suitable for deployment in real automotive environments and adaptable to various vehicle models and attack scenarios.

## 4.6 To Contribute to Future Advancement of Automotive Cybersecurity

Finally, this research intends to generate meaningful insights and contribute to ongoing development efforts in automotive security. By evaluating the effectiveness of advanced anomaly detection methods on real datasets, the study aims to provide guidance for future improvements and encourage further innovation in the protection of intelligent transportation systems.

# 5 Methodology

The proposed intrusion detection framework is designed as a hybrid deep learning architecture that integrates supervised temporal classification with unsupervised anomaly detection. The objective of this methodology is to accurately detect known attack categories while maintaining the capability to identify previously unseen malicious behaviors in Controller Area Network (CAN) traffic. The overall pipeline consists of data preprocessing, sequence construction, supervised LSTM-based classification, Autoencoder-based anomaly modeling, and a hybrid decision mechanism for final prediction.

## 5.1 DATASET

We used car-hacking datasets which include DoS attack, fuzzy attack, spoofing the drive gear, and spoofing the RPM gauge. Datasets were constructed by logging CAN traffic via the OBD-II port from a real vehicle while message injection attacks were performing. Datasets contain each 300 intrusions of message injection. Each intrusion performed for 3 to 5 seconds, and each dataset has total 30 to 40 minutes of the CAN traffic.

- DoS Attack : Injecting messages of '0000' CAN ID every 0.3 milliseconds. '0000' is the most dominant.
- Fuzzy Attack : Injecting messages of totally random CAN ID and DATA values every 0.5 milliseconds.
- Spoofing Attack (RPM/gear) : Injecting messages of certain CAN ID related to RPM/gear information every 1 millisecond.

### 5.1.1 Data attributes

Timestamp, CAN ID, DLC, DATA[0], DATA[1], DATA[2], DATA[3], DATA[4], DATA[5], DATA[6], DATA[7], Flag

- Timestamp : recorded time (s)
- CAN ID : identifier of CAN message in HEX (ex. 043f)
- DLC : number of data bytes, from 0 to 8
- DATA[0 7] : data value (byte)
- Flag : T or R, T represents injected message while R represents normal message

Table 2: CAN dataset statistics (per attack trace): total messages, normal messages, and injected messages

| **Attack Type / Dataset** | **of Messages** | **of Normal Messages** | **of Injected Messages** |
|---|---|---|---|
| DoS Attack | 3,665,771 | 3,078,250 | 587,521 |
| Fuzzy Attack | 3,838,860 | 3,347,013 | 491,847 |
| Spoofing the drive gear | 4,443,142 | 3,845,890 | 597,252 |
| Spoofing the RPM gauge | 4,621,702 | 3,966,805 | 654,897 |
| Attack-free (normal) | 988,987 | 988,872 | – |

### 5.1.2 Data Preprocessing and Normalization

To ensure robust learning and avoid statistical bias, feature normalization is performed using Min-Max scaling. Importantly, the scaler is fitted exclusively on normal training data to prevent data leakage from attack samples. Each feature is transformed using:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

where $x_{min}$ and $x_{max}$ are computed only from benign training frames. This approach ensures that attack distributions do not influence normalization parameters.

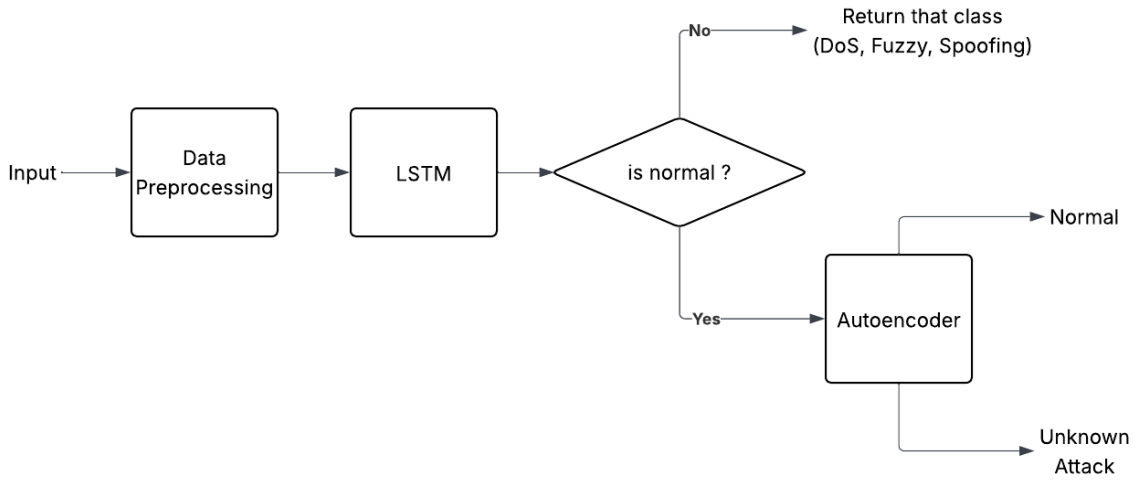
## 5.2 Model Architecture

The proposed framework adopts a two-stage hybrid architecture for robust CAN intrusion detection. In the first stage, a Long Short-Term Memory (LSTM) network performs supervised multi-class classification on fixed-length sliding windows of CAN traffic. Each

window ($20 \times 10$) captures temporal dependencies across consecutive frames, enabling the model to identify known attack types such as DoS, Fuzzy, Gear, and RPM.

To enhance detection of unseen or zero-day attacks, the second stage incorporates an Autoencoder-based anomaly detection module. The Autoencoder is trained exclusively on normal traffic to learn its intrinsic structure. Windows classified as Normal by the LSTM are further evaluated using reconstruction error. A statistically derived threshold using Kernel Density Estimation (KDE) determines whether the sample is truly normal or represents an unknown anomaly.

This hybrid design combines the strengths of supervised classification and unsupervised anomaly detection, ensuring both high accuracy for known attacks and robustness against emerging threats.




### 5.2.1 Sliding Window Construction

Since CAN traffic is sequential in nature, individual frames do not sufficiently represent temporal behavior. Therefore, consecutive frames are grouped into fixed-length sliding windows of size $W = 20$. Each window forms a two-dimensional tensor of shape $(20, 10)$, where 20 represents the number of frames and 10 corresponds to the extracted numerical features.

Window construction is performed independently within each dataset file to preserve temporal continuity and avoid boundary mixing across different traffic traces. A window is labeled as an attack window if at least one frame inside it corresponds to injected traffic; otherwise, it is labeled as normal. This strategy enables the model to capture temporal attack bursts such as flooding and spoofing sequences.

### 5.2.2 Long Short-Term Memory (LSTM) Model

The first stage of the hybrid system is a Long Short-Term Memory (LSTM) network that performs supervised multi-class classification. LSTM is a specialized recurrent neural network capable of modeling long-term temporal dependencies through gated memory cells. Unlike traditional feed-forward networks, LSTM processes sequences step-by-step while maintaining an internal memory state.

At each timestep $t$, the LSTM updates its cell state using three gating mechanisms: the forget gate, input gate, and output gate. The cell state update is governed by:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$$

where $f_t$ determines what information to discard, $i_t$ decides what new information to retain, and $\tilde{C}_t$ represents candidate information derived from the current input. The hidden state $h_t$ is then computed from the updated cell state and passed to the next timestep.

In this implementation, each input window of shape $(20, 10)$ is processed by an LSTM layer containing 128 hidden units. Batch normalization and dropout are applied to improve generalization and reduce overfitting. The LSTM output is passed through fully connected dense layers before a softmax layer produces probabilities for the five classes: Normal, DoS, Fuzzy, Gear, and RPM. The model is trained using categorical cross-entropy loss with class weighting to handle imbalance among traffic categories.

### 5.2.3 Autoencoder-Based Anomaly Detection

While the LSTM model effectively detects known attack patterns, supervised models may fail when encountering unseen attack behaviors. To enhance robustness, an Autoencoder is introduced as the second stage of the framework.

The Autoencoder is trained exclusively on normal windows to learn the intrinsic structure of legitimate CAN traffic. Each input window is flattened into a 200-dimensional vector and passed through an encoder network that compresses it into a lower-dimensional latent representation. The decoder reconstructs the original window from this compressed representation.

The training objective is to minimize the Mean Squared Error (MSE) between input and reconstruction:

$$\mathcal{L}_{MSE} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{x}_i)^2$$

Since the Autoencoder is exposed only to normal traffic, it reconstructs benign windows accurately while producing higher reconstruction errors for anomalous windows. This reconstruction error serves as an anomaly score.

### 5.2.4 KDE-Based Threshold Estimation

To determine a statistically reliable anomaly threshold, Kernel Density Estimation (KDE) is applied to the reconstruction loss distributions of normal and attack windows. Let $f_N(x)$ and $f_A(x)$ denote the probability density functions of normal and attack losses respectively. The optimal threshold $\tau$ is selected by minimizing the absolute difference:

$$\tau = \arg\min_x |f_N(x) - f_A(x)|$$

This non-parametric method provides a more stable and data-driven separation boundary compared to heuristic approaches such as mean plus standard deviation.

## 5.3 Hybrid Decision Mechanism

The final prediction process combines outputs from both stages. First, each window is classified by the LSTM model. If the predicted class is one of the known attack categories, the decision is directly accepted. However, if the LSTM predicts a window as Normal, the same window is passed through the Autoencoder to compute reconstruction error. If the error exceeds the KDE-derived threshold, the window is labeled as an Unknown Attack; otherwise, it is confirmed as Normal.

This two-stage mechanism enables precise classification of known attacks while simultaneously providing anomaly detection capability for zero-day or unseen threats.

# 6 Conclusion

The increasing connectivity and automation in modern vehicles have exposed Controller Area Network (CAN) systems to significant cybersecurity risks. Due to the absence of built-in authentication and encryption mechanisms, CAN traffic is vulnerable to attacks such as message injection, spoofing, and denial-of-service. These vulnerabilities necessitate intelligent and adaptive intrusion detection mechanisms capable of monitoring vehicular communication in real time.

In this study, a hybrid deep learning-based intrusion detection system was developed by integrating a supervised Long Short-Term Memory (LSTM) classifier with an unsupervised Autoencoder-based anomaly detector. The LSTM model captures temporal dependencies in CAN traffic and effectively classifies known attack categories. To enhance robustness against unseen threats, the Autoencoder models normal traffic behavior and detects anomalies using reconstruction error, with threshold selection performed through Kernel Density Estimation (KDE).

Experimental evaluation on real vehicular datasets demonstrates that the hybrid framework achieves reliable detection performance while maintaining scalability for prac-

tical deployment. By combining supervised classification with anomaly detection, the proposed system improves generalization capability and strengthens vehicular cybersecurity. Future work may focus on optimizing computational efficiency for embedded implementation and evaluating performance against additional zero-day attack scenarios.

# References

[1] K. Ren, Y. Zeng, Z. Cao and Y. Zhang, "MAFSIDS: A reinforcement learning-based intrusion detection model for multi-agent feature selection networks," *Journal of Big Data*, vol. 10, Article 98, 2023.

[2] K. Ren, Y. Zeng, Z. Cao and Y. Zhang, "ID-RDRL: A deep reinforcement learning-based feature selection intrusion detection model," *Scientific Reports*, vol. 12, 2022.

[3] H. Alavizadeh and V. K. Javvaji, "Deep reinforcement learning-based intrusion detection system for adaptive network security," *Electronics*, vol. 13, 2024.

[4] S. M. Kasongo and Y. Sun, "Performance analysis of intrusion detection systems using machine learning techniques and feature selection on UNSW-NB15," *Journal of Big Data*, vol. 7, Article 99, 2020.

[5] Z. Dai, Y. Yang, H. Zhang, et al., "An intrusion detection model to detect zero-day attacks in unseen data using machine learning," *PLOS ONE*, vol. 19, 2024.

[6] A. Taneja and G. Kumar, "Attention-CNN-LSTM based intrusion detection system (ACL-IDS) for in-vehicle networks," *Soft Computing*, 2024.

[7] D. Kim, H. Im and S. Lee, "Adaptive autoencoder-based intrusion detection system with single threshold for CAN networks," *Sensors*, vol. 25, no. 13, Article 4174, 2025.