Int. J. Sci. R. Tech., 2025 2(4)
A Multidisciplinary peer-reviewed Journal
www.ijsrtjournal.com [ISSN: 2394-7063]

Original Article

Intrusion Detection System for Smart Vehicles Using Machine Learning Classifiers

Lokesh Kumar, Siddabattula Maheswar, Sai Siddartha, Vasanth Reddy, K. SathiyaPriya

Computer Science and Engineering, Bharath institute of higher education and research, Tamilnadu, India

ABSTRACT

This paper presents the development of an Intrusion Detection System (IDS) for smart vehicles utilizing advanced machine learning algorithms. The system is designed to detect and classify various types of cyberattacks, such as Distributed Denial of Service (DDoS), Fuzzy, and Impersonation attacks.The dataset used for model training and evaluation is the CAN-intrusion-dataset, which contains crucial vehicle communication features, including Message_ID, Byte-level signals, and Target labels. The study employs a range of machine learning models, including Random Forest, Gradient Boosting, Adaboost, LSTM, and CatBoost classifiers, to identify and mitigate potential threats. By leveraging the power of these algorithms, the system aims to provide robust and real-time detection of anomalous behaviour in vehicular networks, enhancing the security and reliability of smart vehicle systems. The ultimate goal is to develop an efficient and scalable IDS capable of protecting smart vehicles from evolving cyber threats.

Keywords: Intrusion Detection System, smart vehicles

INTRODUCTION

The increasing integration of smart technologies in modern vehicles has significantly enhanced their functionality, safety, and convenience. However, this transformation has also introduced new cybersecurity risks, with vehicles becoming potential targets for various cyberattacks. The connected nature of smart vehicles, relying on communication protocols like the Controller Area Network (CAN), makes them vulnerable to attacks that can disrupt vehicle operations, compromise safety, or even expose sensitive data. As these threats evolve, there is an urgent need for advanced systems to detect and mitigate such intrusions in real-time. An Intrusion Detection System (IDS) serves as a critical defense mechanism to identify malicious activities and protect vehicle systems from cyber threats. Traditional IDS solutions have been tailored to general IT networks, but the unique characteristics of vehicular networks require specialized approaches to effectively address the challenges posed by these environments. This paper proposes the development of an IDS specifically designed for smart vehicles, leveraging machine learning algorithms to detect and classify various types of cyberattacks.

PROBLEM STATEMENT:

As smart vehicles become increasingly integrated with advanced communication networks, they are exposed to a wide array of cybersecurity threats, including Distributed Denial of Service (DDoS), Impersonation, and Fuzzy attacks. These cyberattacks pose significant risks to the safety, privacy, and reliability of vehicular systems. Current intrusion detection mechanisms are often inadequate in detecting sophisticated, evolving threats in real time. Therefore, there is a pressing need for an efficient and scalable Intrusion Detection System (IDS) that can accurately identify and classify these cyberattacks in vehicular networks. This study aims to address this gap by leveraging machine learning algorithms, such as Random Forest, Gradient Boosting, and LSTM, to develop a robust IDS capable of enhancing the security of smart vehicles.

OBJECTIVE OF THE PROJECT:

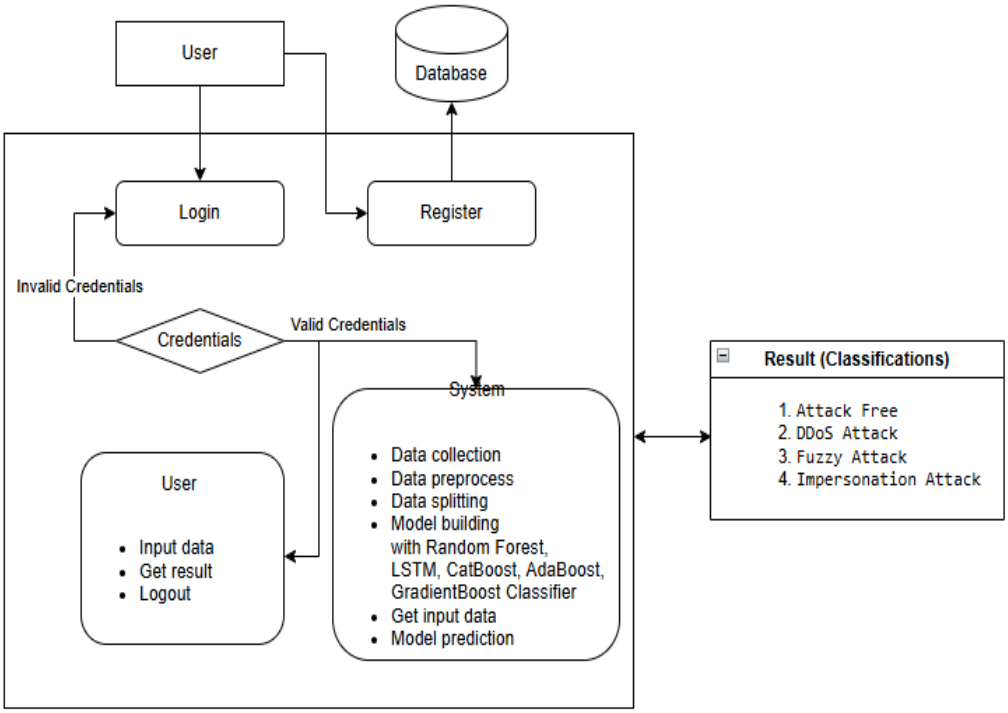
Relevant conflicts of interest/financial disclosures: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The objective of this project is to develop a robust Intrusion Detection System (IDS) tailored for smart vehicles, leveraging advanced machine learning techniques to identify and mitigate cyber threats in vehicular networks. By utilizing the CAN-intrusion-dataset, the system will classify a range of cyberattacks, including DDoS, Fuzzy, and Impersonation attacks, as well as distinguish between normal and malicious traffic. The project aims to implement a variety of machine learning models, such as Random Forest, Gradient Boosting, Adaboost, LSTM, and CatBoost classifiers, to ensure high accuracy and real-time detection of threats. The ultimate goal is to create an efficient, scalable, and reliable IDS that enhances the security and resilience of smart vehicle systems against emerging cyber risks.

SCOPE:

The scope of this study focuses on the development of an Intrusion Detection System (IDS) tailored for smart vehicles, leveraging advanced machine learning algorithms to identify and classify a variety of cyberattacks. The system aims to detect threats such as Distributed Denial of Service (DDoS), Fuzzy, and Impersonation attacks, as well as distinguish between normal and anomalous vehicle communication patterns. The research utilizes the CAN-intrusion-dataset, incorporating vehicle communication features like Message_ID, Byte-level signals, and Target labels, to train and evaluate various machine learning models. The primary objective is to create a robust, real-time IDS that enhances the security of vehicular networks, ensuring effective protection against dynamic cyber threats.

Architecture:



Feasibility Study

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. Three key considerations involved in the feasibility analysis are

- Economical Feasibility
- Technical Feasibility
- Social Feasibility

Economic feasibility:

This study is carried out to check the economic impact that the system will have on the organization. The

amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Technical feasibility:

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

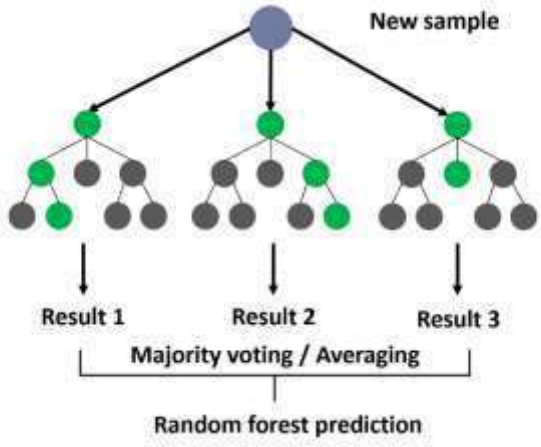
Social feasibility:

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

METHODOLOGY:

Random Forest:

Random Forest is an ensemble learning algorithm that operates by constructing multiple decision trees during training and combining their outputs for improved accuracy and robustness. Internally, it employs the bagging (Bootstrap Aggregating) technique to create diverse subsets of the training data by sampling with replacement. Each decision tree is trained independently on these subsets, reducing overfitting. Random Forest introduces additional randomness by selecting a random subset of features at each split in a tree, promoting varied decision boundaries across trees. This feature selection mitigates the dominance of strongly predictive variables, fostering model diversity. During prediction, the algorithm aggregates individual tree outputs: for classification, it employs majority voting, while for regression, it averages the predictions. The ensemble effect enhances generalization, minimizes variance, and addresses biases inherent in individual trees. Its parallel architecture ensures scalability, making it effective for high-dimensional data and complex tasks.

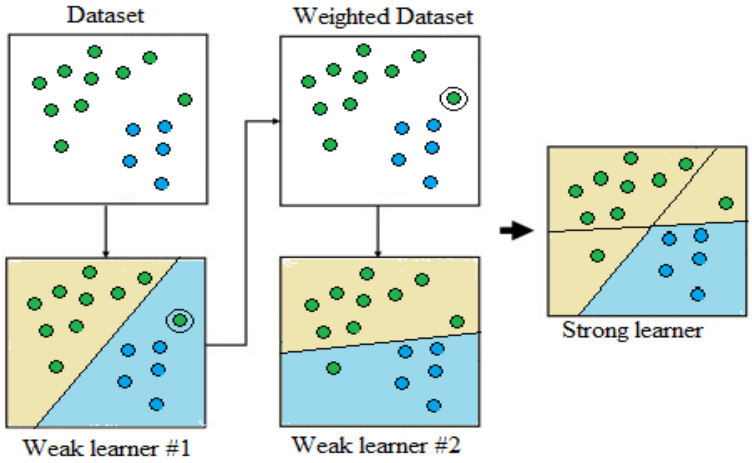


AdaBoost:

AdaBoost (Adaptive Boosting) is an ensemble learning technique that combines multiple weak classifiers to create a strong classifier. The core idea is to iteratively train weak models, typically decision trees with a single split (stumps), each focusing on the mistakes made by the previous model. The algorithm begins by assigning equal weights to all training samples. In each iteration, a weak classifier is trained, and the model's error rate is calculated. The weights of the misclassified samples are then increased, while

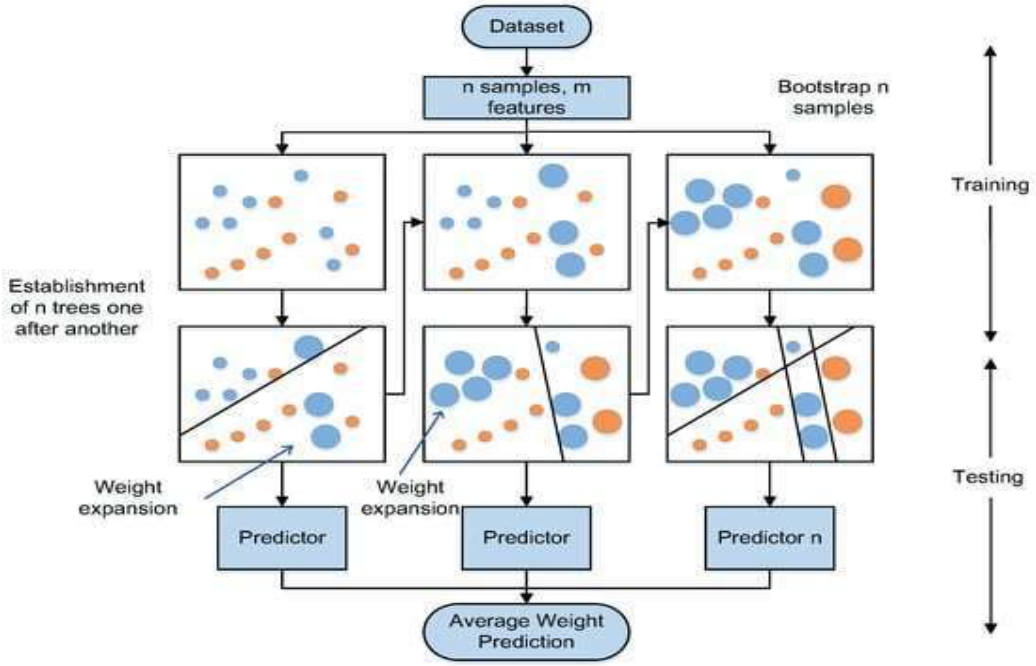
correctly classified samples are given less weight. This adjustment directs the next weak classifier to focus more on the harder-to-classify instances.

Each weak classifier's contribution to the final prediction is weighted based on its accuracy. The final strong classifier is a weighted combination of all individual weak classifiers. AdaBoost typically improves performance by reducing bias and variance, making it effective even when base models are simple and prone to underfitting.



CatBoost:

CatBoost Classifier is a gradient boosting algorithm designed to handle categorical features efficiently. It builds an ensemble of decision trees, each focusing on correcting errors made by previous trees. Unlike traditional gradient boosting methods, CatBoost uses an innovative technique called ordered boosting, which addresses overfitting and improves generalization by shuffling the training data in a specific way to avoid bias from the order of observations. CatBoost's strength lies in its ability to directly handle categorical variables without requiring one-hot encoding or extensive preprocessing. It applies a method called target-based encoding, where categorical features are encoded based on the target values, reducing the need for manual feature engineering. The model optimizes the objective function using gradient descent, minimizing the loss at each iteration. CatBoost incorporates symmetric trees, where splits are balanced across the tree, which reduces computation time and improves prediction accuracy. It also supports parallelization to speed up training.



SYSTEM DESIGN:

Input Design:

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc. Therefore, the quality of system input determines the quality of system output. Well-designed input forms and screens have following properties −

- It should serve specific purpose effectively such as storing, recording, and retrieving the information.
- It ensures proper completion with accuracy.
- It should be easy to fill and straightforward.
- It should focus on user's attention, consistency, and simplicity.
- All these objectives are obtained using the knowledge of basic design principles regarding −
  - What are the inputs needed for the system?
  - How end users respond to different elements of forms and screens.

Objectives for Input Design:

The objectives of input design are −

- To design data entry and input procedures
- To reduce input volume
- To design source documents for data capture or devise other data capture methods
- To design input data records, data entry screens, user interface screens, etc.
- To use validation checks and develop effective input controls.

Output Design:

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

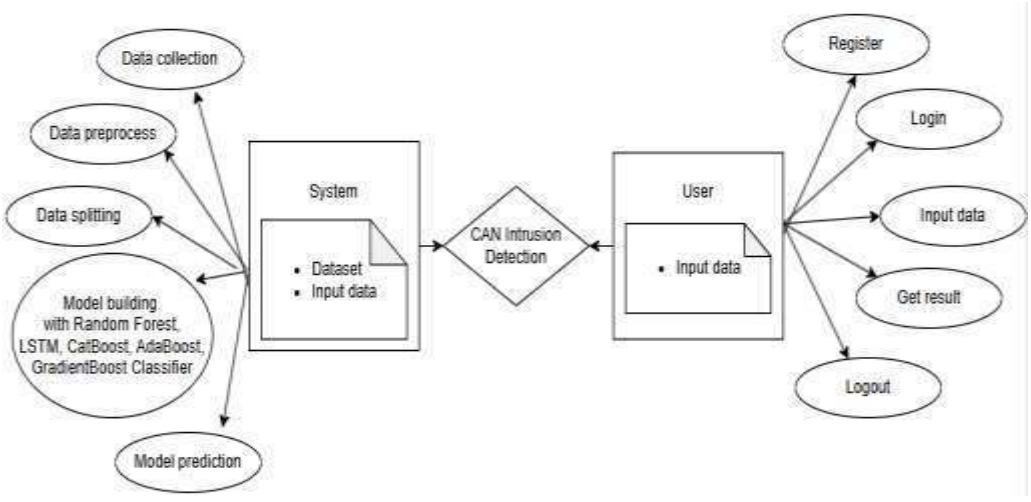
Objectives of Output Design:

The objectives of input design are:

- To develop output design that serves the intended purpose and eliminates the production of unwanted output.
- To develop the output design that meets the end user's requirements.
- To deliver the appropriate quantity of output.
- To form the output in appropriate format and direct it to the right person.

ER Diagram:

- An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of ER model are: entity set and relationship set.
- An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let's have a look at a simple ER diagram to understand this concept.



Test Cases:

Input	Output	Result
Input	Tested for different model given by user on the different model.	Success
Random Forest Classifier	Tested for different input given by the user on different models are created using the different algorithm and data.	Success
Prediction	Prediction will be performed using to build from the algorithm.	Success

Test cases Model building:

S.NO	Test cases	I/O	Expected O/T	Actual O/T	P/F
1	Read the datasets.	Dataset's path.	Datasets need to read successfully.	Datasets fetched successfully.	It produced P. If this not F will come in case the data is not in the form of .csv
2	Feature engineering	Need to check the dataset null values/categorical values	Dataset Preprocessed successfully	Data wrangled successfully	It produced P. If this not F will come
3	Modelling	Input with algorithms to get metrics	Algorithm accuracy will be in the form of percentage	We can get the accuracy of each and every model one by one	It produced P. If this is not, it will undergo F
4	Prediction	Need to enter the input values	Need to predict the output based on the user input	Result successfully predicted with particular algorithm	It produced P. If this is not, it will undergo F

CONCLUSION:

In conclusion, this paper demonstrates the successful implementation of an Intrusion Detection System (IDS) for smart vehicles using advanced machine learning algorithms. By leveraging models such as Random Forest, Gradient Boosting, Adaboost, LSTM, and CatBoost, the IDS effectively detect and classifies various cyberattacks, including DDoS, Fuzzy, and Impersonation attacks

REFERENCE

1. C. Chen, C. Wang, T. Qiu, M. Atiquzzaman, and D. O. Wu, ''Caching in vehicular named data networking: Architecture, schemes and future directions,'' IEEE Commun. Surveys Tuts., vol. 22, no. 4, pp. 2378–2407, 4th Quart., 2020, doi: 10.1109/COMST.2020.3005361.
2. Z. Lv, D. Chen, and Q. Wang, ''Diversified technologies in Internet of Vehicles under intelligent edge computing,'' IEEE Trans. Intell. Transp. Syst., vol. 22, no. 4, pp. 2048–2059, Apr. 2021, doi: 10.1109/TITS.2020.3019756.
3. H. Gao, D. Fang, J. Xiao, W. Hussain, and J. Y. Kim, ''CAMRL: A joint method of channel attention and multidimensional regression loss for 3D object detection in automated vehicles,'' IEEE Trans. Intell. Transp. Syst., vol. 24, no. 8, pp. 8831–8845, Aug. 2023, doi: 10.1109/TITS.2022.3219474.
4. L. Xing, P. Zhao, J. Gao, H. Wu, and H. Ma, ''A survey of the social Internet of Vehicles: Secure data issues, solutions, and federated learning,'' IEEE Intell. Transp. Syst. Mag., vol. 15, no. 2, pp. 70–84, Mar. 2023, doi: 10.1109/MITS.2022.3190036.
5. F. Lone, H. K. Verma, and K. P. Sharma, ''A systematic study on the challenges, characteristics and security issues in vehicular networks,'' Int. J. Pervasive Comput. Commun., vol. 20, no. 1, pp. 56–98, Jan. 2024, doi: 10.1108/ijpcc-04-2022-0164.
6. S. Xu, Y. Qian, and R. Q. Hu, ''Data-driven edge intelligence for robust network anomaly detection,'' IEEE Trans. Netw. Sci. Eng., vol. 7,

no. 3, pp. 1481–1492, Jul. 2020, doi: 10.1109/TNSE.2019.2936466.

7. F. Fakhfakh, M. Tounsi, and M. Mosbah, ‘‘Cybersecurity attacks on CAN bus-based vehicles: A review and open challenges,’’ Library Hi Tech, vol. 40, no. 5, pp. 1179–1203, Nov. 2022, doi: 10.1108/lht-01-2021- 0013.
8. L. Lihua, ‘‘Energy-aware intrusion detection model for Internet of Vehicles using machine learning methods,’’ Wireless Commun. Mobile Comput., vol. 2022, pp. 1–8, May 2022, doi: 10.1155/2022/9865549.

HOW TO CITE: Lokesh Kumar, Siddabattula Maheswar, Sai Siddartha, Vasanth Reddy, K. SathiyaPriya, Intrusion Detection System for Smart Vehicles Using Machine Learning Classifiers, Int. J. Sci. R. Tech., 2025, 2 (4), 197-203. https://doi.org/10.5281/zenodo.15191644