Engineered Science
DOI: https://dx.doi.org/10.30919/es1891

Enhancing Safety in Autonomous Vehicles through Forensic Analysis and Intrusion Detection Using Transformer Model

Masira M.S Kulkarni,* Prashant Dhotre and Mohd Shafi Pathan

Abstract

Autonomous vehicles (AV) have been developed as a transformative innovation in the perspective of restructuring the transportation background. There vehicles are developing with innovative features like autonomous ability, which improve road safety and expand mobility. But Autonomous vehicles are vulnerable to cyberattacks due to their dependence on communication and computer processes. These processes contain flaws that hackers can employ to obtain unauthorized access, steal data and handle vehicle control, which result in accidents and severe damage. In Autonomous vehicles, there is need for Intrusion Detection System (IDS) mechanism to improve characteristics then detect cyber-threats. This research introduces a novel-based transformer method to enhance safety and predict intrusion detection in Autonomous vehicles. In this work, a publicly available Car-Hacking dataset contains injected and normal messages. Here, investigator extract related information from dataset based on time stamped sensor and event data. Initially, data are pre-processed by data cleaning and normalization methods that enhance quality of data which are structured into log formation to easy analysis. Here, the Cuckoo Hashing function was utilized to log creation and store based on their timestamps. Then improved transformer used to analysis log events and predict potential incidents. Lastly, Forensic examination includes different kinds of working processes to enhance safety in Autonomous vehicles. Forensic processes collect evidence from crime scenarios, and then the collected evidence is further investigated to generate detailed reports about Autonomous vehicles malicious activity. The experimental result displays proposed transformer model attains an accuracy of 99.69%. The proposed model enhances the safety of Autonomous vehicles data and provides better processes for describing and analyzing Autonomous vehicles incidents.

Keywords: Autonomous vehicles; Intrusion detection system; cyber-attack; Cuckoo hashing; Log creation; Transformer model and Forensic examination.
Received: 12 June 2025; Revised: 11 November 2025; Accepted: 19 November 2025
Article type: Research article.

1. Introduction

In the twenty-first century, digital forensics (DF) acts as the heart of providing justice, spanning the whole criminal justice process, from crime scene to courtroom. In the process of enabling the addressing of new and emerging threats, it also informs policy and is essential to achieving the mutual objectives of reducing crime and increasing public safety.[1] In the context of current criminal investigations, the significance of DF cannot be overestimated. Keeping up with modern technology and its potential for abuse is a continuing challenge for this discipline.[2] Nowadays, digital evidence is utilized in several criminal cases. Since there is no evidence that the propensity will turn around, the area of DF is required to support its tasks with an obvious, reliable methodology.[3] The objective of most research over the past 20 years has been to clarify and define the steps involved in the DF investigation process.[4] However, the models for the DF investigation process are already in practice and provide an effective basis. It still feels that even more research is required when the field has yet to produce one that precisely describes all modern industrial and investigative processes. Also, current methods frequently describe the DF investigation process at "task level," illustrating physical tasks that an expert could perform as part of their job.[5,6]

Research on how law enforcement uses digital forensics[7] containing standards and guidelines, how they receive, examine, and analyze digital data sources, and the extent to which digital forensics methodology and technologies are inadequate.[8] The investigative reports maintain chain of custody data to audit the digital forensic analysis carried out

Department of CSE, MIT School of Computing, MIT ADT University, Loni Kalbhor, Pune, Maharashtra, 412201, India
*Email: masiramskulkarni31@gmail.com (Masira M. S Kulkarni)

in each situation, which is not well known. The primary concern in cybersecurity continues to be a difficult topic for researchers, including cyberattacks.[9] Data, systems, resources, and the network as a whole could all be eliminated by these attacks, which processes including software, databases, the internet, and emails may all include computer forensic data.[10] Research revealing network information may be significant since the computer can interact and provide the required data.

Many standard techniques for recognizing and avoiding cyberattacks have been used in earlier days, but these are no longer sufficient for modern threats.[11] Nowadays, a wide range of fields interact with machine learning (ML)[12] and metaheuristic techniques to generate efficient computing and rapid processing of complex network data.[13] Reactive forensics was achieved by developing an approach for safely preserving digital evidence that captured pre-incident. Several factors were considered, such as two-factor authentication, secure encryption, environment sandboxing, integrity checks, and distinctive random file naming. A proof-of-concept tool was recognized to execute this approach and create its efficacy.[14]

Several tests were carried out to verify requirements, performance, and security enhancement.[15] By combining ML methods support vector machines (SVM) and isolation forests with distributed systems during processing, the framework can detect ransomware behaviors that were not previously identified with high precision.[16] Based on the Zero Trust Concept,[17] an innovative technique for digital forensics was developed in network security. The individual's attitude and concepts known as "Zero Trust" reduce the demand for trust in network components for dynamic validating network interactions.[18] Wireless or interfaces beyond passive forensics for wearable devices were used to develop a forensic model based on immediate interactions.[19] The forensic model is divided into logical and physical forensic techniques that were developed using the wearable device environment.[20]

1.1 Related works

A survey of existing techniques was described as follows.

Mohammad Aminual Hoque and Ragib Hasan *et al*.[21] developed a forensic investigation framework named AVGuard, which was utilized to gather and store autonomous driving logs. In order to protect the integrity of the logs acquired and stop collusion attacks from several dishonest actors, the framework can generate and validate evidence. Researchers can use the saved logs to identify the exact incident at a later time. The proof-of-concept employment is that their workflow is combining proficiently through autonomous driving components without any important outflows.

Andre Budel *et al*.[22] developed a smart contract-based data integrity and validation tool for autonomous vehicle incident investigation utilizing road trials. The Vehicle Incident Investigation Data System (VINCY), a proof-of-concept software solution, was designed to help Trial Organizations (TOs) manage data by maintaining its integrity throughout time and using it as factual evidence in investigations. VINCY securely stores data hashes and metadata by combining blockchain technology's immutability with the accessibility, efficacy, and dependability of a distributed cloud storage system.

Abin Oommen Philip and RA K Saravanaguru *et al*.[23] developed a conceptual evidence management system to analyze prospective accident investigations and forensics using linked vehicles. This structure demonstrates how smart contracts in a vehicle-to-everything connected system can be used to collect and arrange evidence from incident-related vehicles, as well as backup data from surrounding vehicles' CCTVs and other road users, using blockchain. Smart contracts are used to dynamically control access to files and evidence data based on the incident's location and the vehicles intricate. The cost of planning and carrying out transactions using smart contracts was evaluated on both the public and private Ethereum blockchains.

Autonomous vehicles face challenges in vehicle accident forensic conservation, breach of vehicle owner confidentiality, and identifying legal facilities. Yao *et al*.[24] suggested accident concern identification techniques for the Internet of Vehicles based on a lightweight blockchain solution to address the aforementioned concerns, in addition to collecting data from accident vehicles utilizing Car Forensics Master. At the same time, this model collects data from maintenance service providers, insurance companies, transportation management groups, and car manufacturers. Then, other parties involved in a vehicle collision store it in the preservation chain. Additionally, this study uses Vehicular Public Key Infrastructure (VPKI) to protect the privacy of autonomous vehicle IDs. The preservation chain and the accident identification chain collaborate to identify accident liability, increasing the model's efficacy and granting permitted access for connected entities. Furthermore, this framework shows that suggested model's provide protocol optimal security properties. But, Blockchain technology, particularly public blockchains such as Ethereum, might experience scalability challenges when processing a large number of transactions, especially at peak hours.

Chuka Oham *et al*.[25] developed a Witness-based Data Priority Mechanism (WIDE) for vehicles around an accident to help make accurate decisions. The WIDE developed a two-level integrity evaluation to achieve end-to-end integrity by evaluating data provided by roadside units (RSUs) via a practical byzantine fault tolerance (pBFT) protocol. Because of the implementation of a blockchain-based reputation management system (BRMS), only data from highly credible witnesses was used as proof to assist in determining obligations. Finally, the Automated Verification of Internet Security Protocols and Applications (AVISPA), along with the High-Level Protocol Specification Language (HLPSL), was used to verify the suggested protocol against data integrity.

Lee *et al*.[26] introduced Tesla Vehicle analysis, while Tesla vehicle contain numerous sensor and created under software-defined-vehicle (SDV) process to gather, store and occasionally transfer data to enthusiastic servers. Here, data record in and out of vehicle through manufacturer utilized to digital forensic analysis. Initially, data are used to detect location and collection of storage media and second phase described how information are obtained. Third phase, obtained information are analyzed by several examination. Finally, examined data associated with analyzed one to verify information are modified are not. This process improve reliability and accuracy on accident examination analysis.

Elmisery *et al*.[27] developed Collaborative Forensic Platform for Electronic Artefacts (CFEPA) which provide securely gather, store and share data from Internet of Vehicles (IoV) nature. This process make sure own and collection to manage information are gathered by intelligent vehicle and recorded in non-proprietary formation. This process permit crime analysis and law enforcement agencies to obtain access to real-time and related road accidents information. This suggested model provide well efficient for resolving accidents and leading through examinations.

Srivastava *et al*.[28] suggested deep learning based model for traffic calculating and anticipation based on utilize of forensic model on autonomous car information. Here, Dilated Long Short Term Memory (D-LSTM) model are optimized by improved Osprey Optimization Algorithm (OOA) are processed the restricted Boltzmann machine originate profound and weighted features. In experimental analysis, suggested model achieve effective performance analysis that compare with existing model to show their effectiveness and robustness.

Liu *et al*.[29] provided a new LSTM and deep belief network by binary encoding (LSTM-BiDBN) which are combined with Controller Area Network Identifier (CAN-ID) to extract related event sequence analysis and semantic information of CAN-ID itself. This model completely deliberate possible communication among electronic control units, whether detect vehicle are attacked by invaded through outside, to online control accountable gathering of accident. This model utilize car-intrusion-dataset to detect attack-free and abnormal situation, and in experimental analysis suggested model is compare existing model to show their higher performances analysis.

Chen *et al*.[30] offered digital forensic workflow for automotive intelligent networked terminal devices. Furthermore, terminal devices of intelligent connected vehicle contain several memory chips that store vehicle information at driving and manufacture it potential to discover digital forensic analysis. Here, model provide two different sets of forensic cases that are utilized to T-Box and analyze model applicability and validity. The existing model methodology and performance are detailed in Table 1.

Table 1: Existing model methodology and performance.

Author & Reference	Methodology	Dataset	Merits	Demerits
Hasan *et al*.[21]	AVGuard	Waymo open dataset	Confidentiality and integrity	Implement log collection function applies for all other AD model
Andre Budel *et al*.[22]	Smart contract and VINCY	Real time data	Data security and immutability	Improve performance for better service to stakeholders
Saravanaguru *et al*.[23]	Smart contract	Real time dataset	Cost efficiency	Immutable and decentralized
Yao *et al*.[24]	VPKI	Real time dataset	Handle large process	Enhance the detection process to find malicious contributors
Chuka Oham *et al*. [25]	WIDE, AVISPA and HLPSL	Real time dataset	Scalability and storage overheads	Improve workflow with incentive process
Lee *et al*.[26]	Tesla Log data analysis	Real time data	Reliability and increase performances	More analysis are required to improve forensic analysis
Elmisery *et al*.[27]	CFPEA	Real time data	Feasibility and robustness	This study require additional data to increase detection deliberate phase
Srivastava *et al*.[28]	D-LSTM, OOA	Real time data	Effectiveness	Low adaptability
Liu *et al*.[29]	LSTM-BiDBN	CAN-intrusion dataset	Low computational complexity	Model failed to analysis particular attack analysis
Chen *et al*.[30]	Digital forensic model	Real time dataset	Generalization ability	Limited amount of data are utilized

1.2 Research gap

This section discusses on recent models have significantly progress in autonomous vehicles, however it have some limitations in forensic methods. In particular, introducing standard implementation of log collection function applies to all other AD model that are further needed to develop their process. Improve performance for better stakeholders, immutable and decentralized. Development in the detection process is required to find malicious contributors successfully, and an incentive process could improve workflow. This process also contains a detailed report process for further analysis. For safety procedures, AVs depend on a wide range of sensors, actuators, and software elements. Any cyberattack or malfunction on these systems could result in errors with potentially dire impacts. Investigators can carefully examination digital evidence left by an AV incident, including sensor data, vehicle logs, and software code, by using a forensic investigation framework. This examination makes it possible to identify the exact cause, whether data is cyberattacked, software glitch, or hardware malfunction. There are legal and regulatory issues as AVs become more popular. It can be difficult to suggest blame and determine liability in an accident. A clear framework for forensic investigation provides a coherent method for gathering, storing, and evaluating evidence. The collected evidence is assured and validated for further investigation process ensuring that are digital evidence are acceptable in court. Addressing the safety, legal, and technological complexities surrounding autonomous vehicles requires a forensic investigation framework. It helps guarantee accountability, permits efficient remediation, and ultimately aids in the safe and responsible deployment of this game-changing technology by offering an organized method for incident investigation. To address these challenges and enhance forensic function, this research process introduces a cuckoo hash function for secure and effective way to provide data storage and retrieve these logs, based on their timestamped. Further, a transformer model is applied to predict the injected method and reduce dimensionalities data. Lastly, the forensic method is introduced to provide a complete analysis of the incident information to determine if attack is caused through cyberattacks or not, generate detailed report analysis to make decision making process. Here, main objective of proposed model is described in as follows:

- To design and implement a log creation and storage system using Cuckoo hashing.
- To train and deploy a Transformer-based AI model to predict potential incidents or anomalies.
- To apply digital forensic examination and analysis techniques to the logs to summarize findings with visualizations and timelines.

2. Proposed methodology

A robust methodology is still required to investigate Autonomous Vehicle (AV) incidents effectively. Here, AVs offer numerous benefits, they are not without their shortcomings. Autonomous driving decisions can be fallible, leading to accidents on the road. A significant challenge is the absence of a comprehensive forensics investigation framework for AVs. This approach is critical for a variety of reasons, including determining insurance disputes, researching potential cyberattacks, declaring compliance with autonomous driving safety rules, and, most crucially, analyzing the core causes of events involving AV. By determining the precise causes of these instances, the research may work toward designing more resilient and secure AV systems. To accomplish this, it is critical to collect useful logs from multiple autonomous driving modules and store them securely and tamper-proof. Fig. 1 depicts the overall design of the suggested model.

This research, Enhancing Safety through Forensic Analysis

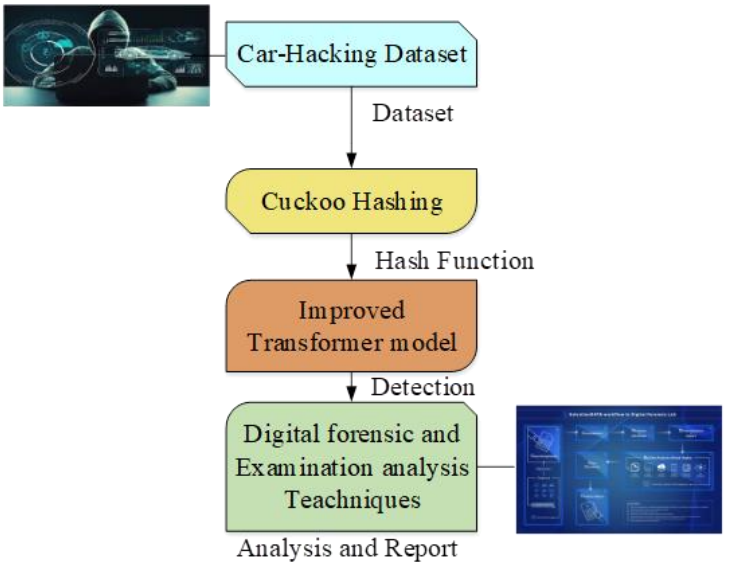

Fig. 1: An overall architecture for the proposed model.

of Autonomous Vehicles, focuses on the forensic examination of AV. The Car-Hacking Dataset typically consists of a DoS attack, fuzzy attack, spoofing drive gear, and RPM gauge. By extracting relevant data from the Car-Hacking Dataset and pre-processing it, investigators can create structured logs containing timestamped sensor data, annotations, and event information Cuckoo hashing provides an efficient way to store and retrieve these logs, enabling rapid access to specific events or periods. This is crucial for analyzing accident scenarios and identifying potential anomalies. Log creation focuses on event-based organization for easier analysis by pre-processing datasets to reduce the mis-detection process. Once the data is organized, an AI based transformer model can be applied to analyze sequential log events and detect potential incidents or irregularities. This predictive ability assists in practical safety events and incident prevention in AV. Following, digital forensic investigation and analysis techniques are functional to the logs. Data integrity is confirmed through hash verification while relevant data is filtered based on specific events or timeframes. Anomalies are detected by unusual behavior or patterns in the data, and complete reports are generated to assists further investigation and decision making.

2.1 Pre-processing and log creation

The pre-processing process is important approach for preparing datasets to provide optimal compatibility through ML and DL methods. In this proposed model, Car-Hacking Dataset is used to analyze and detect malicious activities, while dataset contain noisy and missing data. This research employ two preprocess methods namely, data cleaning and normalization, which is the adaptation of categorical features into numerical formats. Extraordinary dissimilarity variation originates among extreme and normal ranges of the dataset.

2.1.1 Data cleaning

The data cleaning[31] method contains two processes. In the first process, occurrences are reduced through unclear and 'Null' values from the dataset. In the second process, it may find duplication in the dataset by engaging two processes. Primarily, this process eliminates duplication in similar classes by dropping the duplicates function.

2.1.2 Min-max normalization

Normalization contains cleaning data to provide constant in particular range, trailed through compressed feature for classifiers which contain features of different ranges. The cleaned dataset incorporates characteristics by different scales, demanding normalization. This study established min-max normalization[32] methods to regularize characteristics in the range of 0 to 1, which are represented in Eq. (1).

$$N = \frac{c - c_{MIN}}{c_{MAX} - c_{MIN}} \tag{1}$$

where, N indicates output value, then term c denotes actual value, c_{MAX} and c_{MIN} represent maximum and minimum values of the input variable c, individually. Here both pre-processed methods produce well-defined data that increase model detection performance and then, data are structured into log format based on time-stamped and senor data.

2.2 Cuckoo hashing

Cuckoo Hashing is the dynamization of a static dictionary. The dictionary applies two hash tables, namely $T1$ and $T2$, both containing p arguments. The two hash operations are denoted as $H1$ and $H2$: $V \to \{0, \ldots, p-1\}$. Every key $y \in A$ is record in the cell $H1(y)T1$ or in the cell $H2(y)T2$, then not ever in together. The lookup operation is as follows

$$functionlookup(y)$$
$$returnT1[H1(y)] = y \vee T2[H2(y)] = y$$
$$end$$

The two hash tables retrieved for the lookup are in detail ideal between each dictionary by linear space, excluding different cases.

If $p \geq (1+\rho)m$ for particular continuous $\rho > 0$, $H1$ and $H1$ are selected consistently at random after $\big(R(1), R(log\, m)\big)$ the universal family. The possibility that there is no chance of arranging keys T according to $H1$ and $H2$ is $R(1/m)$. A right procedure of keys designates assessable in predictable linear time by decrease $2 - SAT$. Then consider normal dynamization of overhead till supposing $p \geq (1+\rho)m$ apply for certain constant $\rho > 0$. For insertion, it turned out that the "cuckoo process", activating extra keys missing while every key contains its individual "nest", works exactly. Particularly, if y are to be introduced, then if the cell $H1(y)$ of T_1 reemployed. Then it $T1[H1(y)] \leftarrow y$ well, therefore it creating existing occupant "nestles". This key is implanted in $T2$ a similar way and into view repetition.

This process ensure that total number of repetitions is controlled through value "MaxLoop". If the total number of repetitions is extended, then rehash keys in tables by hash operations and attempt again to put up a nestles key. There is no essential requirement to assign tables at rehashing; as an alternative previous table basically run complete tables to altered and accomplish the traditional addition technique on each key found that is not selected in planned position. Fig. 2 displays the architecture for the cuckoo hashing method using two hash tables.

By using representation $y \leftrightarrow x$ to rapid that values of the variable y and x are exchanged, subsequent code recapitulates supplement processes.

$$procedureinsert(y)$$
$$iflookup(y)thenreturns$$
$$loopMaxLooptimes$$
$$y \leftrightarrow T1[H1(y)]$$
$$ify = \perp\ thenreturn$$
$$y \leftrightarrow T2[H2(y)]$$
$$ify = \perp\ thenreturn$$
$$endloop$$
$$rehash(\,); insert(y)$$
$$end$$

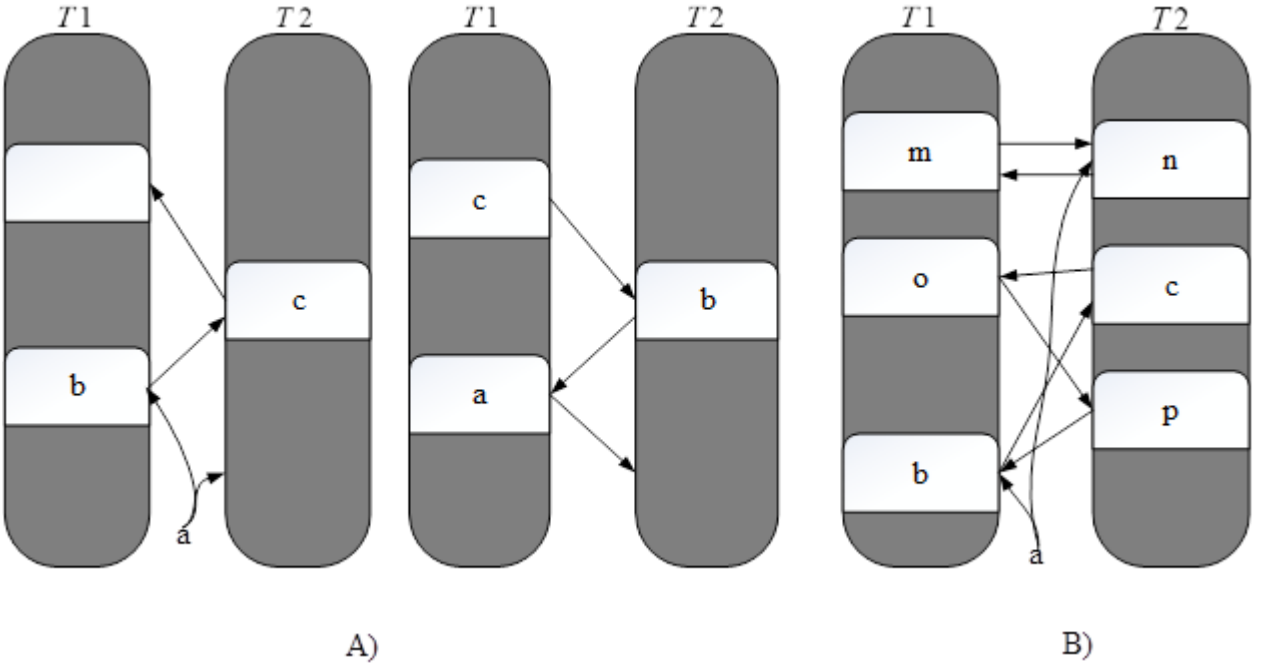

Fig. 2: Architecture for cuckoo hashing method.

The process adopted that every table is superior to $(1+\rho)m$ cells. While there are no certain requirements acknowledged, investigation requirements must be completed to identify them, while rehash to superior tables is required. Resizing of tables might be finished in remunerated predictable, continuous time per modification by using normal replication or splitting methodology. If hash tables consume size p, then implement not further than p^2 supplement are achieved lacking altering hash functions. Further, especially if p^2 supplements are achieved during the establishment of the final rehash, then strength novel rehash.

2.2.1 Hash function

It might be to build a hash process family that, while limited to some set of p^2, are $(1, m^\gamma)$ universal, apply for certain continuous $\gamma > 0$, by possibility $1 - R(1/m^2)$. Then, selected from the family chance process $H1$ and $H2$ consuming continuous calculation time and explanation of $r(m)$ words. Since there were at most p^2 keys introduced by a specific pair of hash operations, this denotes that:

- By possibility, $y \leftrightarrow x$ hash operations consume certain indeterminate performance.
- Then, the hash function performs precisely as if it is selected from $y \leftrightarrow x$ the complete family.

For m superior to certain continuous, it takes MaxLoop$< m^\gamma$, for example, by great possibility family are (1, MaxLoop)-universal. It represents $H1$ and $H2$ will act like an arbitrary process on every set of keys handled at the supplement loop. This method provides an efficient process to store hash values and produce rapid use of time stamps or 0.00 events.

2.3 Transformer model

The transformer method is related to the supreme viable neural structure transmission process, and it similarly applies to an encoder-decoder process. The encoder plans to input representative arrangement $(y^1, y^2, \ldots . y^m)$ into the constant representative arrangement $s = (s^1, s^2, \ldots s^m)$, and the decoder outputs certain constant representative arrangement as $(x^1, x^2 \ldots x^n)$. Both process in this technique is auto-reversible, that represent output from one phase serves as input for following encoder or decoder, except for the initial encoder's that are process input at the bottom layer. The improved transformer approach applied to turn input structures into conforming vector drawings. The transformer and RNN differ primarily in that the former uses attention matrices instead of recurrent processes. An improved transformer model[39] is applied to recognize automotive hacking, managing tough hacking information on AV while revealing sensitive information. The improved transformer approach consists of three types of processes: input embedding, encoder and decoder, and Softmax layer. This method then renders each log formation data as an equal-length vector using input embedding. Encoder and Decoder function vectors are generated using a masked-head self-attention workflow to incorporate input data and improve the way of identifying injected messages. Finally, the Softmax layer is used to assess malevolent probability.

2.3.1 Input embedding

Initially, this model utilizes an embedded method to collect log formatted data as inputs. This approach contains more positional encoding data at the base of the encoder stack to reap the benefits of input structure position data. Meanwhile, positional embedding (PE) magnitudes match input embedding (IE) proportions, and self-attention layer's final

input are contour of embedding as well as PE.

$$PE(position,2j) = \sin\left(position/1000^{2j/e_{model}}\right) \quad (2)$$

$$PE(position,2j+1) = \cos\left(position/1000^{2j/e_{model}}\right) \quad (3)$$

Furthermore, PE implements positional encoding by performing sine and cosine operations on certain occurrences. Here, $position$ denotes as location index in input order, j denotes index of dimension of PE vector and then $position, 2j$ express positional encoding value at $position$ for even dimension index $2j$ and $position, 2j+1$indicates as positional encoding $position$ for odd dimension index $2j+1$,e_{model}represent total number of dimension of model embedding that are described in Eqs. (2) and (3).

2.3.2 Encoder and decoder stack

2.3.2.1 Encoder stack

This PE information is fed to the encoder stack, which calculates the outcomes of each process. Here, there are six encoder stacks, and every encoder contains a multi-head self-attention network (MHSA) and point-wise feed-forward network (FFN). Measurements of FFN layers are parameters which are tuned at training. To achieve best presentation, allocate 1024 neurons in FFN layers and collection 32 padding widths for inserting. Furthermore, residual assembly and layer regularization are employed to evaluate output of both sub-layer, which are accepted by following encoder in the stack. Fig. 3 demonstrates the architecture of improved transformer model.

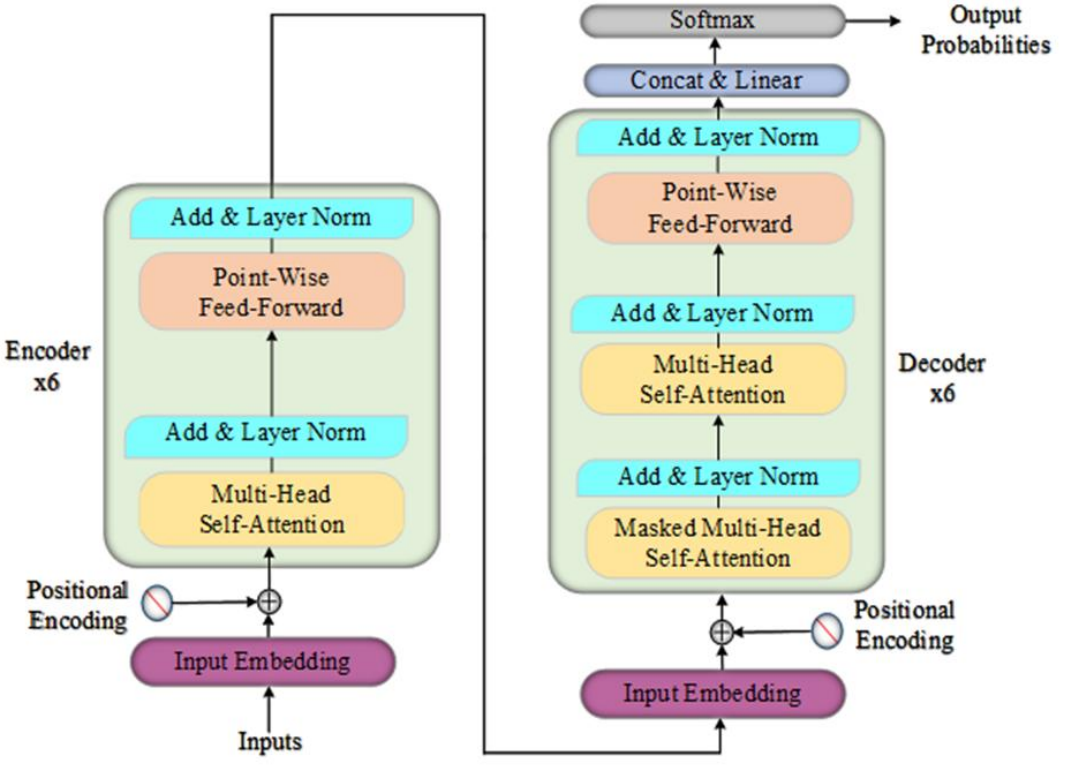

Fig. 3: Structure of improved transformer model.

2.3.2.2 Decoder stack

After the encoder phase, the outputs are served to the decoder process, which is further enhanced by the masked multi-head self-attention sub layer in every decoder in comparison to the upgraded transformer model. This model enhances the robustness of the detection process, and the mask portion of features is randomly detected vector by other unmasked structures. Furthermore, to retain the method's enhanced organization, six decoders can be used in decoder stack. Afterward that, the encoder and decoder stacks were rebuilt, and softmax layer was used as the last result layer for detection. There is residual construction among the input self-attention sub-section and the point-wise FFN, which is monitored by layer normalization.

$$FFN(y) = \chi(max(0, yQ1 + c1)\, Q2 + c2) \quad (4)$$

The determination of the production method can be enhanced by identifying possible issues, such as fading grade and covariate alteration. In addition to the attention sub-section, both the encoder and decoder that consist of a fully associated sub-section known as FNN. It comprises of two linear transformations attained by the ReLU process χ in Eq. (4) for illustration. Furthermore, similar weights are added to both row of attention matrices that are employed as convolutional on both row of the attention-transformed background. It might be argued that this approach creates embeddings using new data.

2.3.3 Masked self-attention workflow

The masked self-attention workflow is utilized by the head g of the decoder. It consumes input as a group of queries P, keys L and values U. It can be calculated in Eq. (5); however e_l are denoted as a scaling factor and N is known as a masking matrix.

$$Attention(P, L, U) = soft\,max\left(N + \frac{PL^W}{\sqrt{e_L}}\right)Z \quad (5)$$

where, $\sqrt{e_L}$ signifies improved grades at training. Afterwards, $N \in S^{L\times L}$ are utilized to preserve the presence of the following places, which makes sure that prediction for place j depends on the individual on recognized outcomes at the place, not more than j. This workflow is able to assist in challenging overfitting issues. Additionally, the covering process is accomplished in the softmax layer by adding $-\infty$. Then, both rows of the matrix are regularized into possibility scattering by the softmax activation layer. Lastly, novel input illustrations are fabricated through the dot product of a regularized matrix U. The performance of the self-attention process is additionally enhanced by multi-head attention; both attentions autonomously preserve their own $P/L/U$ weight matrix.

$$Multihead(P, L, U) = Concat\left(head_{1,\ldots}, head_c\right)Q^G \quad (6)$$

$$head_j = attention\left(PQ_j^P, LQ_j^L, UQ_j^U\right) \quad (7)$$

where, $Multihead(P, L, U)$ represent multi-head attention outcomes that integrate several $head_j$ and concate of result from each attention heads, $head_j$ denote result of j^{th} attention head, Q^G indicates last prediction matrix to map concatenate head posterior to model dimension in Eq. (6,7). Furthermore, the dimensions of the restriction matrix are $Q_j^P \in X^{e_{model\times e_L}}$, $Q_j^L \in X^{e_{model\times e_L}}$, $Q_j^U \in X^{d_{model_U}}$ and $Q_j^G \in X^{e_{model\times fe_u}}$ separately. The outcomes acquired by both heads are concatenated to build the last outcomes. This work assists the method in contributing to various places and delivers numerous illustration sub-spaces applied to the attention layer. The persistence of stable training and fast convergence, layer standardization are applied for on both samples $y \in X^d$ in Eq. (8),

$$LN(y) = \frac{y - \phi}{\lambda} \cdot \omega + \varphi \quad (8)$$

where, $\phi \in X$ and $\lambda \in X$ are signified as mean and standard deviation (SD) of input structures. Then $\cdot$ are known as element-wise don't operators, $\omega \in X^e$ and $\varphi \in X^e$ are trainable affine transform parameters.

2.4 Digital forensic examination and analysis techniques

Digital Forensic Examination and Analysis Techniques (DFEAT) is a structured process that contains preparation, identification, evidence collection, preservation, examination, analysis, reporting and review, as well as closure. These techniques are utilized in autonomous vehicles, smartphones and Internet of Things (IoT) systems. In this method, Data are gathered, reserved, unbothered and safe to present in court instances or support upcoming investigations conducted by law enforcement agencies. Fig. 4 reveals the overall architecture for Digital Forensic Examination and Analysis Investigation.

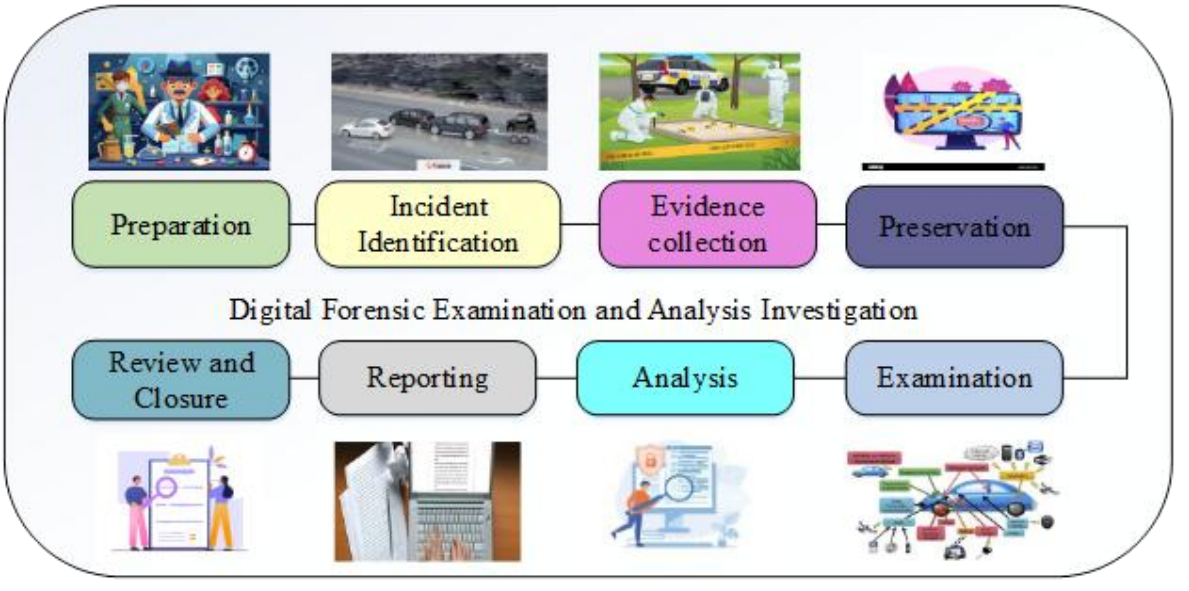

Fig. 4: Overall flow for Digital Forensic Examination and Analysis Investigation.

2.4.1 Preparation

DFEAT is established in science that is related to biology, environments, transformation, and their control relative to the environment of Earth. Forensic examination is initially interrelated to the examination of death or, in another way, is applied to outline event location, time of occurrence, and hold the event process. DFEAT contains some tools used in the examination and analysis stage. In customary writing tools, it

is likely Encase, FTK, and Autopsy do not fully analyze a range of evidentiary interests. However, the present DFEAT contains a tool that can extract specific metadata, namely GPS data, MOV files, and MP4 video. Usually, it depends on particularly manufactured characters and processes. It assists forecasters in extracting metadata from video data. These tools deliver output as word-based outcomes, which are necessary for each analysis in CSV or an extra mapping folder, as shown by the mapping tool. The DFEAT process includes several new devices, which devices are individually trained by the forensic team and should learn knowledge about devices. Mapping tool results are converted into an understandable format, which may lead to enhanced further processes.

2.4.2 Incident identification

Initially, the DFEAT model, which describes the identification of any autonomous vehicle incident, is employed as a valuable detection workflow. In cybercrime cases, crucial articles frequently comprise convoluted data logs, data breaches, or malware activity. Optimizing the DFEAT method to observe manuscript-based logs and connected files facilitates the differentiation and identification of possible or ongoing autonomous vehicle incidents in the surrounding area. In network-message-related processes, anomaly detection plays a vital part in initiating vehicle incidents. The data was extracted from the running vehicle for about 10 minutes. The DFEAT model accesses an autonomous vehicle incident and provides other processes to describe in the manuscript report.

2.4.3 Evidence collection

In gathering evidence, it includes physical processes with human contact, while humans are at risk of losing evidence. DFEAT may play a vital part in categorizing and cataloguing prospective fragments of proof in crime scenarios. These methods are proficient in handling data contained by pictures and producing text-based results, assisting in the explanation and grouping of photographic information. While this process may be humble and within the abilities of human representatives, proficiency turn into predominantly evident when distributing by huge-scale examination, including thousands of pictures of gathered objects and collecting autonomous vehicle data. Employing DFEAT for the beginning process can expressively save time for human representatives who can then aim at the critical task of proof and validation. After collecting evidence, every process of investigation is managed by documentation to preserve guardianship.

2.4.4 Preservation

After collecting evidence, preserving evidence is placed on upholding reliability. To achieve this process, the hash function is applied to preserve data. In the framework of preservative disk evidence, it becomes practical for the investigator to express their supplies in natural language and hash function to produce secure code personalized to specific requirements. The cuckoo algorithm focuses on code generation, such as hash function, log creation, and optimized evidence processing. Then, the process is linked to prevent disk evidence by tailored code generation. Then, in some occurrences, collecting real data for forensic investigators becomes difficult, specifically gathered data at crime scenes. This hash method provides flexibility and proficiency in the investigative process. Then, evidence is secured and duplicated by DFEAT for additional purposes and to keep the original data.

2.4.5 Examination

In this phase, collected data are examined by a technical person. The data are secured, preserved and duplicated by the forensic team. Here, an improved transformer model is pre-trained with a car-hacking dataset to predict various types of attack. Then, collected data are analyzed to determine if data are injected or not, and then it can predict various types of injected IDS. If any injected data are found in the investigation, then it can reconstruct sequence-based related information and create a timestamped process. These processes improve efficiency and effectiveness in the examination stage of digital forensics. This process allows investigators to contribute to large-level analysis and make decisions about outcomes. The examination outcomes are related to autonomous vehicles; then, the results are analyzed by experts to generate detailed information.

2.4.6 Analysis

The analysis stage involves understanding the occurrence and arriving at convincing conclusions based on gathered data at the investigation stage. Integrating substances such as event logs, timestamps, and network attack seizures additionally permits active recreation of incidents by associating both datasets and helping the hash function. Then, automated agents can allocate the investigation workload successfully. This approach provides a conclusion from evidence and examination of the performance of the results. After analyzing outcomes, whether injected messages are retrieved from autonomous vehicles, DFEAT will be analyzed to provide detailed information about retrieved data like timestamps, Event ID, IP address, and Sensor data.

2.4.7 Reporting

After analysis, the reporting part generates a digital report based on detailed analyses of retrieval data. The dominance and validity of evidence, along with the thoroughness of investigation, are summarized in the final report document. The reporting stage embraces important weight, as complete judgment may be a turning point in a critical stage. Detailed forensic investigation reports are provided in Fig. 5.

Particularly, DFEAT ensures sensitive examination of reports quality that highlighting the significance of exactitude and clarity at report stage. Then, to provide help and improve inspection, integrating the DFEAT method report formation is

```
Forensic Investigation Report for Vehicle ID: 11
================================================

The investigation into the incident involving Vehicle ID: 11 has been documented as follows:
- Timestamp: The incident occurred on 2016-11-03 17:08:06.
- Event: The event identified was a RPM attack.
- Event ID: The corresponding Event ID for this incident is 2016-11-03 17:08:06_RPM.
- IP Address: The source of the event was traced to the IP address 192.168.1.11.
- Sensor Data: The sensor data received at the time of the event included:
   - CAN ID: 043f
   - DLC (Data Length Code): 8
   - Data: ['00', '40', '60', 'ff', '7a', 'c8', '08', '00']

================================================
```

Fig. 5: Report Generation from the DFEAT model.

a viable solution. Then this model also assists in automating forensic reports and different formats such as HTML or Latex.

2.4.8 Review and closure

Here, this model provides better performance in the forensic process. It may generate a final report in digital form, and it will provide an improved process to secure forensic data. This model provides new digital equipment to collect evidence in video format; collected data is stored with a hash function to generate a secure process. The transformer model examines the secure data to check whether the data is an injected message or not, and then the DFEAT model provides a report about the investigation process. This approach utilizes new techniques to discover evidence and investigate it well. Then, it contains hashes method to store evidence in a secure way. This method effectively generates digital evidence that is suitable for evidence as final evidence.

3. Result and discussion

This section discusses the hardware and software settings used to evaluate experimental analysis and exhibits the dataset used in the study. The proposed method is next evaluated using performance measures, followed by an experimental evaluation and analysis. The suggested model uses 80% of the data for the training phase and 20% for testing.

3.1 Hardware and software configuration

This sub-section provides configuration details that are applied to the proposed model. System configurations are detailed and mentioned in Table 2. Parameter values are described in Table 3.

Table 2: System Configuration.

Parameters	Configuration
Processor	Intel® Core ™ i5-9500
Installed RAM	16.0 GB
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

Table 3: Hyperparameter Values.

Hyperparameter	Details
Learning rate	0.001
Epoch	300
Loss function	Categorical_crossentropy
Batch Size	16
Optimizer	Adam

3.2 Dataset description

This area of the car-hacking datasets[33,38] covers DoS assaults, fuzzy attacks, faking driving gear, and spoofing the RPM gauge. Datasets are built by logging Controller Area Network (CAN) activity from the vehicle via the OBD-II connection while message injection attacks are carried out. The datasets contain 300 instances of message injection. Every intrusion lasts 3 to 5 seconds, and both datasets contain 30 to 40 minutes worth of CAN traffic. Table 4 presents an overview of the dataset.

Table 4: Overview of Dataset.

Attack Type	#Message	Normal Message	Injected Message
DoS Attack	3,665,771	3,078,250	587,521
Fuzzy Attack	3,838,860	3,347,013	491,847
Spoofing Attack (Gear)	4,443,142	3,845,890	597,252
Spoofing Attack RPM	4,621,702	3,966,805	654,897
Normal	988,987	988,872	-

Table 5: Mathematical expression.

Performance	Derivation
Accuracy (%)	$Accuracy = \frac{T.pos + T.Neg}{T.pos + F.pos + T.neg + F.neg}$
Precision (%)	$precision = \frac{T.pos}{T.pos + F.pos}$
Recall (%)	$recall = \frac{T.pos}{T.pos + F.neg}$
F1-score (%)	$F1 - score = \frac{2 * precision * recall}{precision + recall}$
Specificity (%)	$Specificity = \frac{T.neg}{T.neg + F.pos}$
FPR (%)	$FPR = \frac{F.pos}{T.neg + F.pos}$
TPR (%)	$TPR = \frac{T.pos}{T.pos + F.neg}$
MSE	$MSE = \frac{\sum_{n=1}^{m}(predicted(n) - actual(n))^2}{m}$
RMSE	$RMSE = \sqrt{\frac{\sum_{n=1}^{m}(predicted(n) - actual(n))^2}{m}}$

1. Dos attack: Injecting message of '0000' CAN ID every 0.3 milliseconds. '0000' are peak prevailing.
2. Fuzzy attack: Injecting message of entirely accidental CAN ID and DATA values each o.5 milliseconds.
3. Spoofing attack (RPM): Injecting messages of some CAN ID associated with RPM data each 1 millisecond.
4. Spoofing attack (gear): Injecting messages of some CAN ID, which are associated with gear data each 1 millisecond.
5. Normal data: It does not contain any injection messages.

3.3 Performance metrics

This section provides performance metrics Accuracy, precision, recall, F1-score, specificity, False Positive Rate (FPR), True Positive Rate (TPR)[34] and Mean Square Error (MSE), Root Mean Square Root (RMSE)[35] are implemented to calculate effective of improved transformer method-based attention network for car-hacking detection function. Mathematical expressions are derived in Table 5.

From Table 5, changed pixels are referred to as positive and unchanged pixels as negative samples. The relationship between parameter $T.Pos$, $T.Neg$ $F.Neg$ and $F.Pos$ are known as True Positive, True Negative, False Negative and False Positive and also $predicted(n)$denotes the predicted value for $n$data point $m$denotes the number of observations.

3.4 Experiment analysis

In assessment baselines for car-hacking injecting message prediction employs traditional DL methodologies such as Recurrent Neural Network (RNN), Long-Short Term Memory (LSTM), Convolutional Neural Network (CNN-LSTM) and Deep Neural Network (DNN). Then, the existing model is compared with the suggested model.

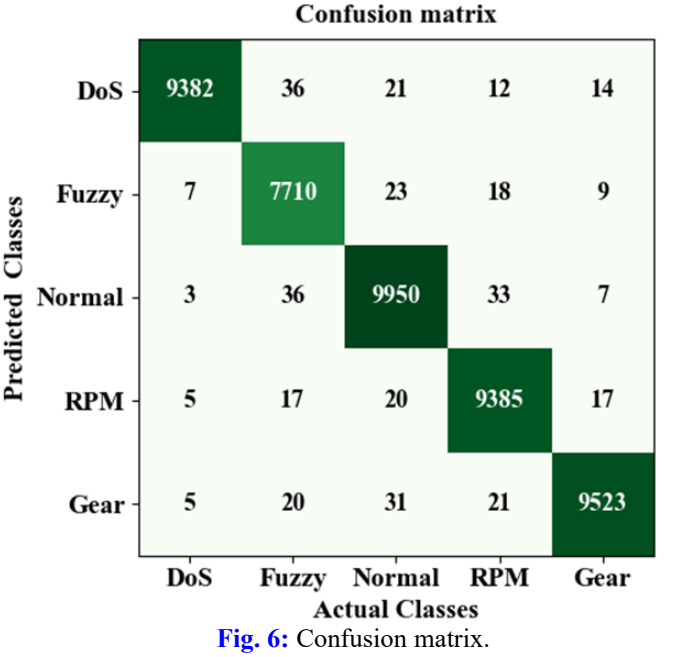

Fig. 6: Confusion matrix.

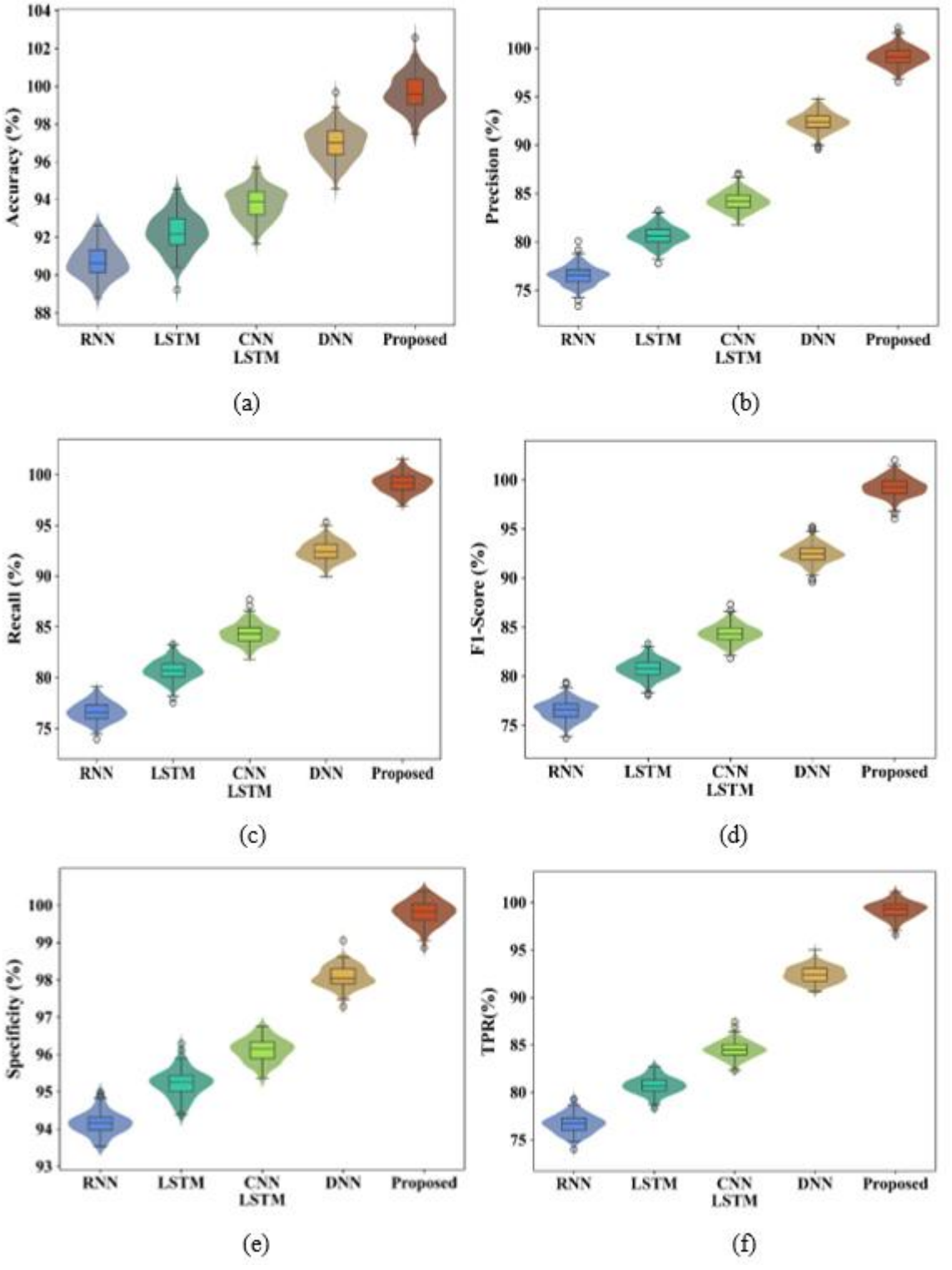

Fig. 7: Overall performance of the suggested model.

Fig. 6 explores the confusion matrix for the car-hacking intrusion detection dataset. The dataset contains five classes, namely DoS, Fuzzy, Normal, RPM and gear message. The proposed model utilizes 46,305 messages for testing process. Here, the proposed model correctly predicts 45,950 labeled data, which includes 9382 messages as DoS class, 7710 messages as Fuzzy class, 9950 messages as Normal class, 9385 messages as RPM class and 9523 messages as Gear class. Therefore, it wrongly predicted 335 labeled data. The proposed model provides potential improvement, differentiates among each class, and will detect injected messages.

Fig. 7 demonstrates the overall performance of the suggested methodology. This model provide enhance robustness analysis through a masked MSHA layer to predict unmasked features, while softmax layer is applied to classify

different types of injected messages. Fig. 7(a) explores the performance of transformer model accuracy of 99.69%; Fig. 7(b-c) provides precision and recall values of 99.21% and 99.23%, which are slightly higher than existing models. Fig. 7(d) shows an F1-score value of 99.22%. Fig. 7(e-f) demonstrates specificity and TPR values of 99.80% and 99.23%. However, the proposed model obtains higher performance analysis than other existing models in terms of practical implementation approaches. The current existing model is facing some issues in accurately detecting injected messages that generate poor performance. The proposed model improves performance and also enhances model speed to detect injected datasets. Comparison of proposed model with existing model is expressed in Table 6.

Table 6: Comparison of proposed model with existing model.

Methods	RNN	LSTM	CNN-LSTM	DNN	Proposed
Accuracy	90.66	92.31	93.75	96.98	99.69
Precision	76.54	80.66	84.37	92.41	99.21
Recall	76.65	80.76	84.28	92.45	99.23
F1-score	76.59	80.71	84.33	92.43	99.22
TPR	76.65	80.76	84.37	92.4	99.23
Specificity	94.16	95.19	96.07	98.11	99.80

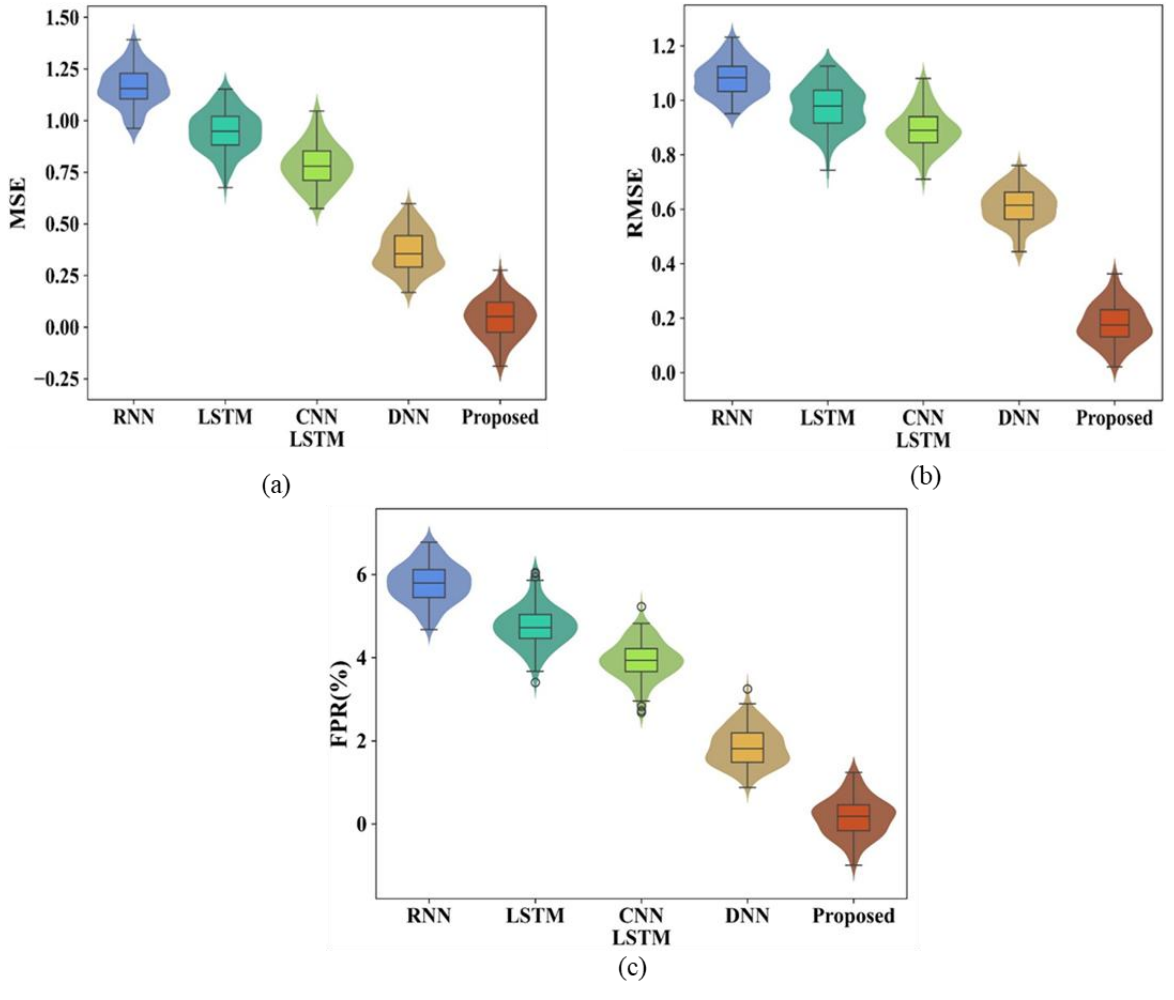

Fig. 8: Error analysis.

Fig. 8 reveals the error analysis for the suggested model. Initially, data cleaning and normalization remove noise and error, which produces accurate results and eliminates error values. This model optimizes features and reduces overfitting issues. Each feature is normalized into possibility scattering by the softmax activation process. Fig. 8(a) represents an MSE error value of 0.028. Fig. 8(b) presents an RMSE value of 0.167. Then, Fig. 8(c) displays an FPR rate of 0.0019. The outcomes contain the proposed model that enhance safety by predicting the common injected messages and reducing false alarms. This method generates low error performance compare to other prevailing models. Comparison of proposed model with existing model is explained in Table 7.

Fig. 9 demonstrates ROC for the suggested model. The

curve denotes two key metrics: the TPR and the FPR. The ROC works as a key metric for calculating detection performance of injected labeled messages. The suggested model explores an AUC value of 0.997, which is compared to other prevailing models, and it indicates reliability and good performance.

Table 7: Comparison of error analysis for proposed model with existing model

Methods	RNN	LSTM	CNN-LSTM	DNN	Proposed
MSE	1.175	0.974	0.792	0.3742	0.028
RMSE	1.084	0.987	0.890	0.611	0.167
FPR	0.059	0.048	0.039	0.0188	0.0019

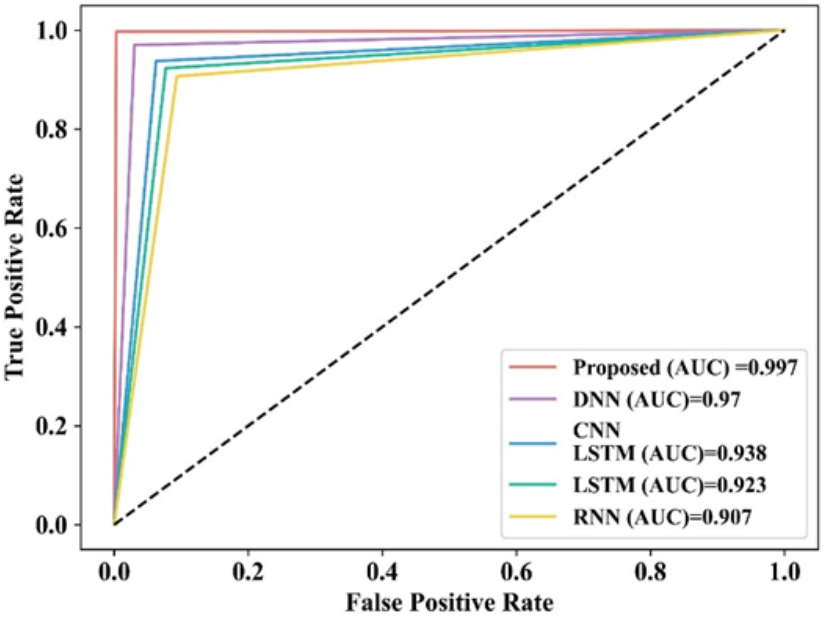

Fig. 9: ROC analysis.

3.5 Discussion

In this section, this research introduces a transformer model for detecting different types of injected messages and forensic methods to investigate the incident location and collect data from an autonomous vehicle. Moreover, the car-hacking dataset includes different types of injected messages, such as DoS, spoofing gear, spoofing RPM, and fuzzy and normal injected messages. The dataset might contain some noise and error, which are reduced with the data cleaning process. Then, the min-max method enhances data in the standard numerical range of 0 to 1; this may provide better results and reduce time complexity. Pre-processed data are securely stored through using a hash function, through each entry producing a log creation based on its timestamp. Then, an improved transformer model is applied to detect different types of intrusion messages from stored data. Encoder parts include the multi-self-attention layer and FFN, which are optimized parameter values at training. The decoder phase further integrated masked the MSHA to contrast the actual transformer method and enhance the robustness of injected messages IDS against injected message. This method ensures that the model contributes to various locations and provides many sub-sections for attention layers. Then, the softmax layer is utilized to predict different types of injected IDS. These methods increase the model process by finding potential issues, such as vanishing gradient and covariate shift. This model provides a stable training process, layered normalization, and quicker convergence. Lastly, the DFEAT model includes several different sub-section processes such as preparation, incident identification, evidence collection, preservation, examination, analysis, reporting and review, and closure. Each stage process are employs a unique process that provides a better investigation and enables the generation of comprehensive reports for autonomous vehicle accidents. This workflow delivers a consistent function and accurately detects injected messages by combining them with the transformer model.

Then, the experimental result analysis is provided in a graphical representation for transformer models. The suggested model achieves an accuracy of 99.69%, which is better than other prevailing models, as represented in graphical format. This section highlights limitation of existing models that lack classical forensic methods for implementing log collection function applied to all other AD models,[21] ensuring immutable and decentralized,[23] and requiring to enhance the detection process to find malicious contributors,[24] less forensic analysis,[26] require more data to increase forensic analysis,[27] Low adaptability,[28] Model failed to analysis particular attack analysis that led to low forensic analysis[29] and Limited amount of data are used, so model attained low

performances in investigation.[30] Here, DFEAT model forensic method provides an analysis of the obtained incident, examining whether attacks are injected or not and producing a detailed report process. Hence, this process is a centralized and secure process; then, the model generates a digital report process for further investigation.

Fig. 10 represent comparison analysis of proposed model with existing models. Here, proposed model utilize Car-hacking dataset to achieve high performances analysis which are compare with existing model to show their superior performances analysis. Comparison analysis of the suggested model and prevailing models is described in Table 8.

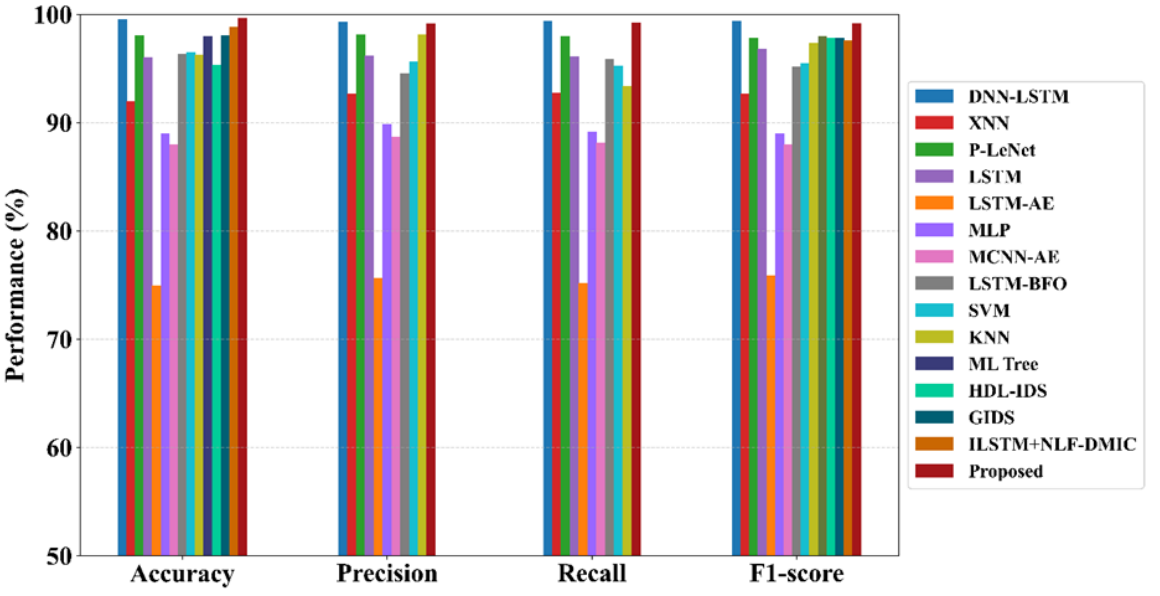

Fig. 10: Comparison analysis of proposed model with existing models.

Table 8: Comparison of proposed model and existing models using Car-Hacking dataset.

Methodology	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
DNN-LSTM[36]	99.55	99.36	99.44	99.42
XNN[36]	92	92.7	92.8	92.7
P-LeNet[37]	98.10	98.14	98.04	97.83
LSTM[37]	96.03	96.18	96.17	96.82
LSTM-AE[40]	75	75.7	75.2	75.9
MLP[40]	89	89.9	89.3	89.5
MCNN-AE[40]	88	88.7	88.2	88.5
LSTM-BFO[40]	96.3	94.6	95.9	95.2
SVM[41]	96.50	95.70	98.30	93.30
KNN[41]	96.30	98.20	93.40	97.40
ML based Fine Tree model[42]	98	-	-	-
DCNN[42]	-	-	-	98
HDL-IDS[42]	95.37	-	-	97.83
GIDS[42]	98.1	-	-	97.83
ILSTM +NLF-DMIC[42]	98.87	-	-	98.63
Proposed model	99.69	99.21	99.23	99.22

4. Conclusion

This research proposed an improved transformer model as an injected message ID process for CAN in the autonomous vehicle environment. This research aim to deliver a new method for detecting well-injected message ID detection that surpasses the limitations of existing models. Enchanting

benefits of associations among mechanisms in CAN data: the proposed model delivers dominant detection abilities, achieves better processes on extensive consecutive CAN IDs, collects pre-processed data, and outperforms other existing models in experimental results. The proposed model attains great performance when calculated by consecutive data. The outcome of the proposed model brings safety maintenance to the vehicle, and it achieves an accuracy of 99.69%. Moreover, the transformer model provides knowledge from the target method to enhance performance in intrusion detection. In current scenarios, development of intelligence systems that are led to improve DFEAT models which are employed to solve secret crimes. This DFEAT model is applied to analyze autonomous vehicle accidents, which are analyses of black box data from the vehicle and whether or not they are modified by any injected intrusion message. Here, the DFEAT model secures data collected from vehicles, examines data for injected ID messages formerly, this process provides a final report to enhance the process of forensic operation and make it easy to identify the culprit. The limitation of the proposed model is that it is not applied in the real-time vehicle operating process. In future processes, a robust model will be combined with transformer to detect unknown injected messages from real-time autonomous vehicles. Therefore, the increased performance of the transformer model in quickly recognizing injected messages led to decreased damage caused by anomalous incidents.

Acknowledgments

I would like to express my sincere gratitude to Department of Computer Science and Engineering, School of Computing, MIT ADT University, Pune, for providing the research facilities and continuous support throughout the research. Special thanks to the Department of Research and Development for their technical assistance and resources that enabled successful completion of this work.

Conflict of Interest

I declare that there is no conflict of interest regarding the publication of this research work.

Supporting Information

Not applicable.

CRediT Statement

Masira M. S Kulkarni: Conceptualization, Software / Implementation, Validation & Testing, Writing – Original draft. **Masira M. S Kulkarni, Prashant Dhotre** and **Mohd.Shafi Pathan:** Methodology, Writing – Review & editing. **Prashant Dhotre** and **Mohd Shafi Pathan:** Supervision.

References

[1] G. Horsman, N. Sunde, Unboxing the digital forensic investigation process, *Science & Justice*, 2022, **62**, 171-180, doi: 10.1016/j.scijus.2022.01.002.

[2] K. U. Maheswari, G. Shobana, The State of the art tools and techniques for remote digital forensic investigations, *3rd International Conference on Signal Processing and Communication (ICPSC)*, Coimbatore, India, May 13-14, 2021, 464-468, doi: 10.1109/icspc51351.2021.9451718.

[3] A. Al-Dhaqm, R. A. Ikuesan, V. R. Kebande, S. A. Razak, G. Grispos, K. R. Choo, B. Ali Saleh Al-Rimy, A. A. Alsewari, Digital forensics subdomains: the state of the art and future directions, *IEEE Access*, 2021, **9**, 152476-152502.

[4] A. R. Javed, W. Ahmed, M. Alazab, Z. Jalil, K. Kifayat, T. R. Gadekallu, A comprehensive survey on computer forensics: state-of-the-art, tools, techniques, challenges, and future directions, *IEEE Access*, 2022, **10**, 11065-11089.

[5] V. Prakash, A. Williams, L. Garg, P. Barik, R. K. Dhanaraj, Cloud-based framework for performing digital forensic investigations, *International Journal of Wireless Information Networks*, 2022, **29**, 419-441, doi: 10.1007/s10776-022-00560-z.

[6] Z. Shahbazi, Y.-C. Byun, NLP-based digital forensic analysis for online social network based on system security, *International Journal of Environmental Research and Public Health*, 2022, **19**, 7027, doi: 10.3390/ijerph19127027.

[7] M. M. S. Kulkarni, P. Dhotre, M. S. Pathan, Identifying intruder in artificial intelligence of things using digital forensic framework: a review, *ICT for Intelligent Systems*, Springer Nature, Singapore, 2024, 485-501, doi: 10.1007/978-981-97-6678-9_43.

[8] J. A. Yaacoub, H. N. Noura, O. Salman, A. Chehab, Advanced digital forensics and anti-digital forensics for IoT systems: Techniques, limitations and recommendations, *Internet of Things*, 2022, **19**, 100544, doi: 10.1016/j.iot.2022.100544.

[9] X. Fernández-Fuentes, T. F. Pena, J. C. Cabaleiro, Digital forensic analysis methodology for private browsing: Firefox and Chrome on Linux as a case study, *Computers & Security*, 2022, **115**, 102626, doi: 10.1016/j.cose.2022.102626.

[10] A. A. Khan, A. A. Shaikh, A. Ali Laghari, IoT with multimedia investigation: a secure process of digital forensics chain-of-custody using blockchain hyperledger sawtooth, *Arabian Journal for Science and Engineering*, 2023, **48**, 10173-10188, doi: 10.1007/s13369-022-07555-1.

[11] S. Sachdeva, A. Ali, Machine learning with digital forensics for attack classification in cloud network environment, *International Journal of System Assurance Engineering and Management*, 2022, **13**, 156-165, doi: 10.1007/s13198-021-01323-4.

[12] K. Yun, H. Yun, S. Lee, J. Oh, M. Kim, M. Lim, J. Lee, C. Kim, J. Seo, J. Choi, A study on machine learning-enhanced roadside unit-based detection of abnormal driving in autonomous vehicles, *Electronics*, 2024, **13**, 288, doi: 10.3390/electronics13020288.

[13] Y. Shin, S. Kim, W. Jo, T. Shon, Digital forensic case studies for in-vehicle infotainment systems using Android auto and apple CarPlay, *Sensors*, 2022, **22**, 7196, doi:

10.3390/s22197196.
[14] A. Singh, R. A. Ikuesan, H. Venter, Secure storage model for digital forensic readiness, *IEEE Access*, 2022, **10**, 19469-19480.
[15] P. M. Shafi, J. Ingale, A. Bahattab, V. Kimbahune, Security enhancement in route once and cross-connect many (ROACM) protocol, *Proceeding of First Doctoral Symposium on Natural Computing Research*, Springer, Singapore, 2021, 339-354, doi: 10.1007/978-981-33-4073-2_32.
[16] Jones, Rual, and Hana Davies, High-performance digital forensic framework for anomalous ransomware detection in file system log data, *TechRxiv,* 2024, doi:10.36227/techrxiv.172599923.38750111/v1.
[17] Ashfaq Shaikh, Sonali A. Patil, Santosh Borde, Pankaj Chandre, Pathan Mohd Shafi, and Anjali Jadhav, Zero Trust Security Paradigm: A Comprehensive Survey and Research Analysis, *Journal of Electrical Systems*, 2023, **19**, doi: 10.52783/jes.688.
[18] C. Neale, I. Kennedy, B. Price, Y. Yu, B. Nuseibeh, The case for zero trust digital forensics, *Forensic Science International*: *Digital Investigation*, 2022, **40**, 301352, doi: 10.1016/j.fsidi.2022.301352.
[19] M. Kim, Y. Shin, W. Jo, T. Shon, Digital forensic analysis of intelligent and smart IoT devices, *The Journal of Supercomputing*, 2023, **79**, 973-997, doi: 10.1007/s11227-022-04639-5.
[20] Skouby, Knud Erik, Idongesit Williams, Prashant Dhotre, and Kamal Kant Hiran, Introduction: Wireless networks and privacy in developing countries, *5G, Cybersecurity and Privacy in Developing Countries*, River Publishers, 2022, 1-18, doi: 10.1201/9781003374664.
[21] M. A. Hoque, R. Hasan, AVGuard: a forensic investigation framework for autonomous vehicles, *ICC 2021 - IEEE International Conference on Communications*, June 14-23, 2021, Montreal, Canada, 1-6, doi: 10.1109/icc42927.2021.9500652.
[22] Budel, André, Reem Alhabib, Mark Nicholson, and Poonam Yadav, Vincy: A smart-contract based data integrity and validation tooling for automated vehicle incident investigation, *arXiv*, 2023, doi:10.48550/arXiv.2311.13728.
[23] A. O. Philip, R. K. Saravanaguru, Smart contract based digital evidence management framework over blockchain for vehicle accident investigation in IoV era, *Journal of King Saud University - Computer and Information Sciences*, 2022, **34**, 4031-4046, doi: 10.1016/j.jksuci.2022.06.001.
[24] Q. Yao, T. Li, C. Yan, Z. Deng, Accident responsibility identification model for Internet of Vehicles based on lightweight blockchain, *Computational Intelligence*, 2023, **39**, 58-81, doi: 10.1111/coin.12529.
[25] C. Oham, R. A. Michelin, R. Jurdak, S. S. Kanhere, S. Jha, WIDE: a witness-based dat *a priori* ty mechanism for vehicular forensics, *Blockchain*: *Research and Applications*, 2022, **3**, 100050, doi: 10.1016/j.bcra.2021.100050.
[26] J.-H. Lee, S. H. Lim, B. Hyeon, O.-Y. Jeon, J. J. Park, N. I. Park, Tesla log data analysis approach from a digital forensics perspective, *World Electric Vehicle Journal*, 2024, **15**, 590, doi: 10.3390/wevj15120590.
[27] A. M. Elmisery, M. Sertovic, Enabling collaborative forensic by design for the Internet of vehicles, *Information*, 2025, **16**, 354, doi: 10.3390/info16050354.
[28] V. Srivastava, S. Mishra, N. Gupta, E. Albalawi, S. Basheer, Autonomous vehicle forensics: investigating data streams for traffic prediction and incident mitigation, *IEEE Transactions on Consumer Electronics*, 2025, **71**, 1211-1218, doi: 10.1109/TCE.2025.3564924.
[29] W. Liu, J. Xu, G. Yang, Y. Chen, Online vehicle forensics method of responsible party for accidents based on LSTM-BiDBN external intrusion detection, *Journal of Shanghai Jiaotong University (Science)*, 2024, **29**, 1161-1168, doi: 10.1007/s12204-022-2549-8.
[30] Z. Chen, Q. Mu, W. Luo, X. Yang, D. Li, X. Shao, Y. Liu, H. Zhu, Digital forensics for automotive intelligent networked terminal devices, *IEEE Transactions on Vehicular Technology*, 2024, **73**, 5128-5138, doi: 10.1109/TVT.2023.3334754.
[31] M. S. Alshehri, O. Saidani, F. S. Alrayes, S. F. Abbasi, J. Ahmad, A self-attention-based deep convolutional neural networks for IIoT networks intrusion detection, *IEEE Access*, 2024, **12**, 45762-45772.
[32] H. M. Song, J. Woo, H. K. Kim, In-vehicle network intrusion detection using deep convolutional neural network, *Vehicular Communications*, 2020, **21**, 100198, doi: 10.1016/j.vehcom.2019.100198.
[33] Y. N. Rao, K. Suresh Babu, An imbalanced generative adversarial network-based approach for network intrusion detection in an imbalanced dataset, *Sensors*, 2023, **23**, 550, doi: 10.3390/s23010550.
[34] M. A. Talukder, S. Sharmin, M. A. Uddin, M. M. Islam, S. Aryal, MLSTL-WSN: machine learning-based intrusion detection using SMOTETomek in WSNs, *International Journal of Information Security*, 2024, **23**, 2139-2158, doi: 10.1007/s10207-024-00833-z.
[35] M. Al Razib, D. Javeed, M. T. Khan, R. Alkanhel, M. S. Ali Muthanna, Cyber threats detection in smart environments using SDN-enabled DNN-LSTM hybrid framework, *IEEE Access*, 2022, **10**, 53015-53026.
[36] S. Aziz, M. T. Faiz, A. M. Adeniyi, K.-H. Loo, K. N. Hasan, L. Xu, M. Irshad, Anomaly detection in the Internet of vehicular networks using explainable neural networks (xNN), *Mathematics*, 2022, **10**, 1267, doi: 10.3390/math10081267.
[37] S. T. Mehedi, A. Anwar, Z. Rahman, K. Ahmed, Deep transfer learning based intrusion detection system for electric vehicular networks, *Sensors*, 2021, **21**, 4736, doi: 10.3390/s21144736.
[38] https://ocslab.hksecurity.net/Datasets/car-hacking-dataset.
[39] Z. Wu, H. Zhang, P. Wang, Z. Sun, RTIDS: a robust transformer-based approach for intrusion detection system, *IEEE Access*, 2022, **10**, 64375-64387.
[40] Dennyson, W. Beniel, and C. Jothikumar, Securing

Automotive Networks from DoS and Fuzzy Attacks with Optimized LSTM Models, *International Journal of Computational Intelligence Systems*, 2025, **18**, 1-22, doi:10.1007/s44196-025-00782-y.
[41] A. Alshammari, M. A. Zohdy, D. Debnath, G. Corser, Classification approach for intrusion detection in vehicle systems, *Wireless Engineering and Technology*, 2018, **9**, 79-94, doi: 10.4236/wet.2018.94007.
[42] Dasari, Deepthi Reddy, and G. Hima Bindu, An Intelligent Intrusion Detection System in IoV Using Machine Learning and Deep Learning Models, *International Journal of Communication Systems*, 2025, **38**, e70131, doi:10.1002/dac.70131.