Library Progress International
Vol.44 No.3, Jul-Dec 2024: P.21003-21015

Print version ISSN 0970 1052
Online version ISSN 2320 317X

Original Article

Available online at www.bpasjournals.com

Enhancing Autonomous Vehicle Safety through LSTM-AE Based Anomaly Detection

[1]Ms.Keerthana B, [2]Dr.Jayanthi B

[1](Corresponding Author)
Assistant Professor, Department of Computer Applications,
Kongu Arts and Science College (Autonomous), Erode - 638107.
keerthikec@gmail.com,
[2]Associate Professor & Head, Department of Computer Science (PG),
Kongu Arts and Science College (Autonomous), Erode – 638107.
sjaihere@gmail.com

How to cite this article: Keerthana B, Jayanthi B (2024) Enhancing Autonomous Vehicle Safety through LSTM-AE Based Anomaly Detection. *Library Progress International,* 44(3), 21003-21015.

Abstract

Vehicles are essential in our lives, providing popular private transportation. Despite their comfort, they pose road safety risks. Autonomous vehicles, a technological advancement, mitigate these risks with sensors and algorithms, reducing human error. However, they introduce cybersecurity challenges, necessitating urgent research to detect and mitigate potential threats to ensure safety. This work suggested a novel anomaly detection technique, the Long Short-Term Memory based Autoencoder (LSTM - AE), to address the issue. First, use an autoencoder to extract features from the autonomous vehicle data and compress them into a latent representation in order to train the model. The LSTM network receives the compressed features and uses them to identify any linked relationships between the features. The output is a reconstruction of the features. Anomalies depending on the output's reconstruction loss are found using an anomaly score. To evaluate effectiveness of the proposed framework, various evaluation metrics such as accuracy, precision, recall, F1 score and ROC curve analysis are employed. The experimental findings are compared with existing models, such as Convolution Neural Networks (CNN), Recurrent Neural Networks (RNN) and Deep Reinforcement Learning (DRL), and Deep Belief Networks (DBN). With an accuracy of 92% and a larger area under the receiver operating characteristic curve than alternative approaches, the comparative findings show that the suggested model outperforms the others in anomaly detection, making it a useful tool for autonomous vehicle anomaly detection. Finally, it is determined that the model's output can be used to intelligently identify and prevent cyberattacks against autonomous cars.

Keywords: Autonomous vehicles, Cyber security attacks, Anomaly detection, Deep learning, Autoencoder, long short-term memory

1. Introduction

With population growth and increasing urbanization over the past few years, the global automobile rate has increased dramatically. Due to the high vehicle density on the roads, which produced an unstable, congested traffic environment with more complicated and dangerous driving conditions, this growth was undoubtedly one of the main reasons of the millions of traffic accidents that occur each year [1-5]. The market for automotive communication protocols is expected to expand by $574.57 million between 2020 and 2024, with a compound annual growth rate (CAGR) of 9% [3]. The market is driven by government programs promoting the use of telematics, growing government support for EVs, and the rising electrification of automobiles. Furthermore, it is projected that rising vehicle electrification will accelerate market expansion. A security researcher from the KU Leuven university in Belgium drove off in a Tesla Model X in 2020 by taking advantage of a Bluetooth weakness on the car's key fob [6]. Intelligent Transportation Systems (ITSs) [7] have been developed as clever technical solutions to enhance traffic safety and encourage smart mobility, among other reasons shown in Figure 1. Through the use of technologies and communication protocols known as vehicles to everything (V2X), such cars can be connected to other ones[8]. Smart traffic control and monitoring, improved safety services, user-oriented mobility

services, and more are just a few of the advantages that ITS provides. They are made up of a range of parts that integrate artificial intelligence with Internet of Things, lessens the need for human interaction. Thus, a self-driving system uses complex algorithms and machine learning models to evaluate data gathered from various onboard sensors and make real-time judgments on how to drive a car [9]. It enables vehicles to roam independently with minimal, if any, human guidance.

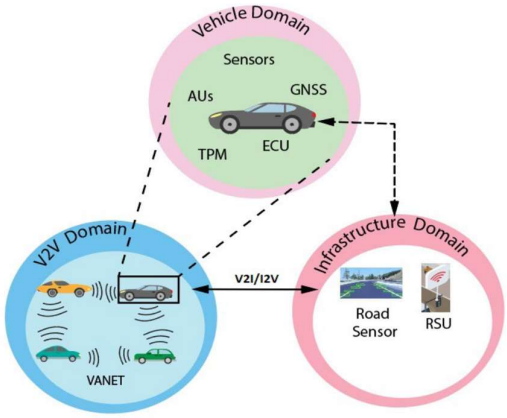

Figure.1 ITS architecture for self-driving cars

Road safety and transportation efficiency will benefit greatly from the introduction of autonomous vehicles (AVs), which usher in a new era of transportation [10-12]. However, autonomous driving systems need a sophisticated design to detect abnormalities in sensor data and eventually reduce their impact due to uncertainties in sensor data caused by weather, environmental anomalies, and cyberattacks, among other external factors. Notwithstanding the potential advantages of antivirus software, a significant obstacle is the prevalence of various cyber security risks, which exposes AVs to various forms of cyberattacks. As a result, it is critical to locate and identify these outliers in the gathered data since they could seriously harm autonomous driving systems [13-17].

This paper's main goal is to investigate this gap for autonomous driving. Using datasets of autonomous driving, the suggested LSTM-AE architecture makes use of explanatory power to comprehend AI decisions [18-20]. This system uses a variety of AI classification techniques along with distinct attack features to differentiate between anomalous and regular malicious actors. Each AV's raw data is processed in order to obtain the key characteristics that can be used to identify threats.

The following briefly describes the primary contributions of this paper:

- The study introduces a pioneering anomaly detection approach, LSTM-AE, which utilized for the early detection of anomaly detection in autonomous vehicle.
- To demonstrate the effectiveness of the model, compare the performance of the suggested framework with the current techniques utilizing univariate, multivariate time series data.
- Several metrics are used to quantitatively measure the effectiveness of the proposed method, including recall rate, accuracy, precision, and ROC.
- The results of the experiment demonstrated that the recommended model outperformed the others.

The remaining paper is structured as follows: Section 2 includes a representation of recent literature review. The research's autonomous vehicle simulation model is covered in Section 3. Experimental results explained in Section 4. Section 5 concludes with recommendations for further research on this topic.

2. Literature Review

Deep learning models have been used in a few studies to detect anomalies. The majority of research projects seek to identify fabricated Autonomous Vehicle (AV) trajectories through offline methods. A variety of techniques are used to classify trajectories following the collection of AV data (normal and faked trajectories).

2.1. Existing Related Models

A trajectory-embedding model was created by [21] using input from the natural language processing (NLP) community. To calculate the similarities between trajectories, the model produced a vector representation

of the CAVs' paths. The distance matrix between each pair of trajectories was then estimated using a hierarchical clustering approach, which also helped to identify trajectories that were fake [22]. A high detection rate (>97%) might be attained with the suggested approach. A multi-stage attention method utilizing a CNN model based on Long Short-Term Memory (LSTM) was proposed. The technique increased the F-score by up to 3.24%. An approach to identify anomalies called Adaptive Extended Kalman Filtering (AEDF) was presented. The movements of the platoon were the subject of this investigation rather than those of a single vehicle. They employed anomaly detection techniques by using the target vehicle's location and speed data from nearby traffic [23-27].

In [28], used information from a simulated environment in SUMO to develop a misbehavior detection algorithm (MDA). They used machine learning techniques to detect misbehavior and produced compromised data using six different attack methods. In order to investigate the effects of cooperative adaptive cruise control platoons on traffic flow and safety, [29] carried out simulated attacks on them. They discovered that there was a detrimental effect on traffic and a rise in the likelihood of collisions as the frequency and intensity of these strikes on the targeted vehicles increased [30]. In [31], examined a platoon of six vehicles using cooperative adaptive cruise control (CACC) against time-delay assaults and found that their CACC algorithm remained stable in the face of attacks involving neither cruising nor jerks. An unsupervised deep learning technique was created by [32]. The authors used an advanced autoencoder and an ANN to identify a pattern of behavior from typical sensor messages and then compared it with vehicle observations based on the notional behavior they had developed for vehicle anomaly detection.

In [33], introduced graph-based intrusion detection and classification system that addresses computational constraints by fusing a machine-learning-based classifier with a intrusion detection system based on thresholds [34]. H-IDFS, a framework for using charts for sorting and identifying attacks was introduced. To identify malicious traffic windows, a multi-class IDS classifier is used in conjunction with histograms of CAN packets arranged into windows. Using n-gram analysis, proposed DAGA, an anomaly detection system [35]. DAGA does not use the payload content or other CAN message fields; instead, it defines n-grams for identification based only on sequences of CAN message IDs.

2.2. Research Gap and Novelty

There are a number of restrictions on the field's current state of study. First off, most research has used AV from traditional human-driven cars to mimic hacks as real-world data is unavailable. Consequently, there is a deficiency in the literature about the evaluation of cyber hazards and the identification of unusual activities in authentic settings. The current research is difficult to implement in real time due to their high computing costs. Secondly, previous research failed to distinguish between distinct kinds of attacks, such as malfunctioning sensor behaviors. Autonomous vehicles are becoming more secure and safe because to the LSTM AE model that has been proposed. The gathered dataset is divided into two categories using the LSTM-AE method: normal and anomalous data which is discussed in next section.

3. Materials and Methods

The complete process flow of LSTM-AE approach is illustrated in Figure 3. Four tasks have to be performed for prediction: i) Data collection ii) Pre-processing the data; iii) Feature selection; iv) Data partition and v) Methodology.

3.1 Data Collection

Data collection for anomaly detection in autonomous vehicles involves gathering extensive and diverse datasets to identify unusual patterns or behaviors that deviate from the norm. Data is collected from various sensors which capture real-time information about the vehicle's surroundings, including distances, speeds, and obstacles. Recording the vehicle's internal states such as fast, acceleration, brake pressure, and steering angle. This helps in understanding the vehicle's operational dynamics. Gathering information on road conditions, weather, and traffic, which can influence vehicle behaviour. Logging incidents like sudden stops, collisions, or system malfunctions to identify potential anomalies.

LSTM-AE's performance in detecting anomalies is assessed using six open databases. Table 1 presents statistical data for these sediments. Assess model's performance on univariate time series even though it is intended for multivariate data. As a result, NAB and UCR were chosen as the two databases. The properties of

the time series that were chosen from these data span the specified period, and they can be utilized to direct multivariate statistics and evaluate the repeatability of the model. Because the main goal of the LSTM-AE design is to reduce distributional differences between several variables, datasets with a wide variety of variables and significant variances are essential. With values of 55, 25, and 123, respectively, the MSL, SMAP, and WADI dimensions are excellent options for a stable assessment of model performance. Figure 2 displays univariate and multivariate type data.

Table 1: Datasets

Type	Dataset	Test	Dimension	Anomalies
Univariate	NAB	4022	1	0.93
	UCR	5800	1	1.87
Multivariate	MBA	100000	2	0.13
	MSL	73724	54	10.73
	SMAP	427618	26	13.13
	WADI	172802	124	5.98

NAB (Numenta Anomaly Benchmark)

NAB provides a diverse set of real-world time-series data with labelled anomalies, facilitating the evaluation of anomaly detection algorithms. It covers various domains, including server metrics, network traffic, and environmental sensors, making it suitable for testing anomaly detection models in autonomous vehicles.

UCR (HexagonML)

The UCR Time Series Classification/Clustering Repository offers a collection of time-series datasets for classification, clustering, and anomaly detection tasks. It includes datasets with different characteristics and complexities, making it useful for benchmarking anomaly detection algorithms in autonomous vehicle systems.

MBA (Multi-source Background Autoencoders)

MBA is a dataset designed specifically for multi-source anomaly detection tasks, where data from multiple sensors/sources are combined. It provides labelled data for training and evaluation, enabling the development of robust anomaly detection systems for autonomous vehicles.

MSL (Mars Science Laboratory)

MSL dataset comprises telemetry data collected from the Mars Science Laboratory mission, including rover sensor readings and environmental data. It offers an opportunity to test anomaly detection algorithms in extreme and remote environments, simulating challenges faced by autonomous vehicles in unfamiliar terrains.

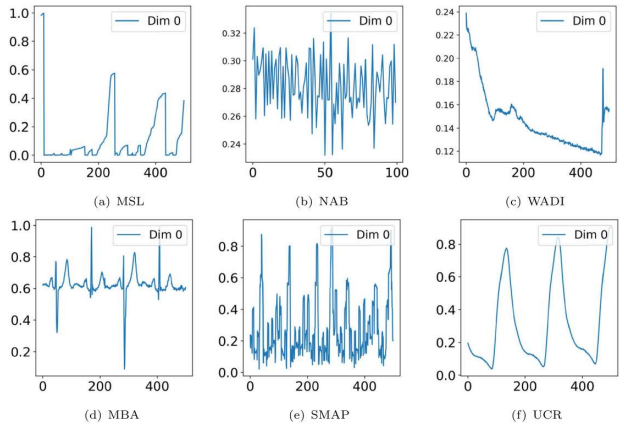

Figure 2: Samples of Datasets.

SMAP (Soil Moisture Active Passive)

SMAP dataset consists of soil moisture measurements collected from satellite sensors. While not directly related to autonomous vehicles, it provides valuable environmental data that can be used to detect anomalies affecting vehicle operation, such as changes in terrain conditions or weather patterns.

WADI

WADI dataset contains real-world sensor data from a water distribution system, including flow rates, pressure readings, and valve statuses. It offers a unique environment for testing anomaly detection algorithms relevant to autonomous vehicles operating in infrastructure networks, such as detecting leaks or malfunctions in the water distribution system.

3.2 Pre-processing the data

To improve the suggested model's robustness, normalize the data and turn it into time-series windows for testing and training. Normalizing time series data involves scaling the values within a consistent range, typically between 0 and 1, to make them more comparable and suitable for analysis or modelling. One common method for normalization is min-max scaling, which can be done using the following formula:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \qquad (1)$$

In the time series x is the original value, x_{min} is the minimum value, x_{max} is the maximum value and x_{norm} is the normalized value. This formula scales each data point proportionally based on its relationship to the minimum and maximum values in the time series. After normalization, the lowest value will be assigned to 0, and the highest value will be assigned to 1. Normalization helps to remove the scale of the data, making it easier to compare different time series or to use them as input for different methods without bias due to the magnitude of the values. It's a crucial preprocessing step in many time-series analysis and forecasting tasks.

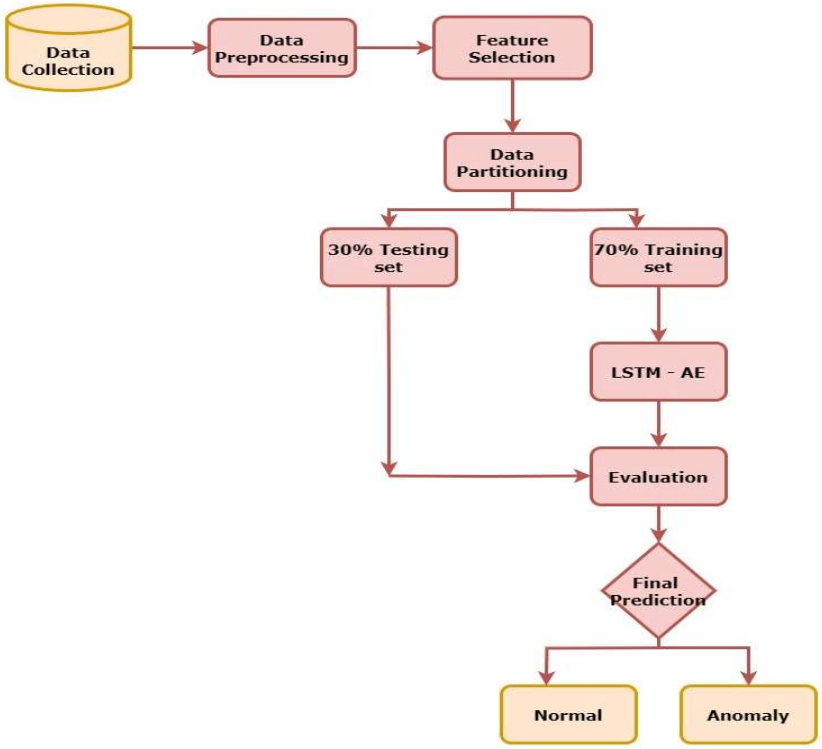

Figure 3. Flow diagram of Proposed model

3.3 Feature selection

Finding the most pertinent features in a dataset in order to enhance model performance and lessen overfitting is known as feature selection. One popular method for feature selection is the use of statistical metrics.

The following equation represents the chi-square (χ^2) statistic, which is used to assess the degree of independence within the goal variable and categorical characteristics:

$$x^2 = \frac{(O_i - E_i)^2}{E_i} \quad (2)$$

Where O_i denotes observed frequency, E_i represents expected frequency of independence between the feature and the target. When χ^2 value is high then it indicates a strong correlation between variable and good for selection.

3.4 Data Partition

Partition the dataset into two groups: 30% for testing, remaining 70% for training. For LSTM-AE model training, the Adam optimization approach is employed. Cross-Validation technique is used to assess or train procedure. This method can be applied as a validation scheme to address over-fitting issues. A contemporary deep learning variant of the stochastic gradient descent process, which has been applied to numerous applications, is the adaptive moment estimation (ADAM) optimization algorithm. It combines the benefits of RMSProp and AdaGrad, two additional optimization approaches. Uncentered variance in sparse gradients is handled by AdaGrad. These algorithms can be calculated according to the following Equations, Equations (3) and (4).

$$m_t = \beta_1 m_{t-1} - 1 + (1 - \beta_1) g_t \quad (3)$$

$$v_t = \beta_2 v_{t-1} - 1 + (1 - \beta_2) g_t^2 \quad (4)$$

where g_t is the gradient at time step t, and $\beta_1$and β_2 are the exponential decay rates for the moment estimates.

3.5 Methodology

3.5.1 Long Short-Term Memory (LSTM) Cell Structure

The LSTM network is an enhanced RNN variant. To control information flow, identify temporal correlations, and identify long-term dependencies in time-series data, the LSTM network offers a long-term memory cell. Logic gates, input, and output are all part of the structure of an LSTM cell, as shown in Figure 4. The LSTM cell begins with a forget gate, which is in charge of either forgetting or keeping the cell state data from the previous iteration, c_{t-1}. As illustrated in (5) and (6), the forget choice is determined by feeding input data, x_t, and prior hidden state, h_{t-1}, whose output value, f_t, lies between [0,1].

$$f_t = \sigma\left(W_{fx} x_t + W_{fh} h_{t-1} + b_f\right) (5)$$

$$\sigma(x) = \frac{e^x}{e^x + 1} (6)$$

where W and b stand for the gate's weight and bias, respectively.

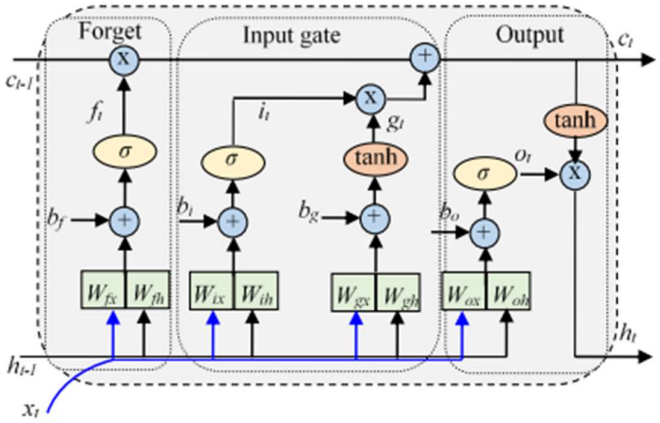

Figure 4. LSTM Cell Structure

Second, as shown in equations (7) and (8), the input gate feeds x_t and h_{t-1} into the tanh activation function to create a new memory state, g_t. By creating an input state, it, as in (9), input gate will simultaneously decide which gt sections will be neglected.

$$g_t = \tanh\left(W_{gx} x_t + W_{gh} h_{t-1} + b_g\right) (7)$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} (8)$$

$$i_t = \sigma\left(W_{ix} x_t + W_{ih} h_{t-1} + b_i\right) (9)$$

As in (10) and (11), output gate finally generates new hidden state, h_t, based on output state, o_t, and newly updated memory cell state.

$$o_t = \sigma (W_{ox} x_t + W_{oh} h_{t-1} + b_o) \quad (10)$$

$$h_t = \tanh(c_t) \odot o_t \quad (11)$$

3.5.2 Autoencoder (AE)

AE is a kind of neural network that uses training data to learn a latent representation. It then produces a fixed-sized representation, typically with fewer dimensions than the input. As shown in Figure 5, the three functions of an autoencoder are encoding, decoding, and reconstruction loss.

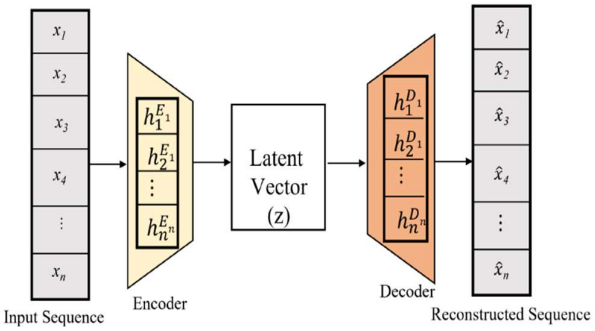

Figure 5. Autoencoder Architecture

Equation (12) illustrates how input data x, which is a m high dimensional vector in the encoding procedure, is transferred to a low dimensional layer representation (h) following the removal of any unimportant features. Equation (13), which illustrates the decoding process, produces the output $\hat{x}$ that maps back into reconstruction of x using the bottleneck layer representation of (h).

$$\mathbf{1.}\ \mathbf{h = f_1(w_i x + b_i)\ (12)}$$

$$\hat{x} = f_2 (w_j h + b_j) \quad (13)$$

where f_1 is activation function, b_i is a bias, and w_i is the weight matrix. The decoder's activation function is called f_2. The weight matrix is denoted by w_j, the reconstructed input sample is represented by $\hat{x}$, and b_j stands for a bias. The process of minimizing the difference between the input and the output in a typical autoencoder model is shown in Equation 14 by determining the reconstruction loss (L). In tasks involving anomaly identification, this reconstruction loss is commonly used.

$$L(x - \hat{x}) = \frac{1}{n} \sum_{n=1}^{n} |\hat{x}_t - x_t| \quad (14)$$

where x denotes input data, n is number of samples.

3.5.3 LSTM-AE

The LSTM-AE model's deployment procedure for anomaly detection is depicted in Figure 6. The LSTM-Autoencoder constructs LSTM networks on encoder and decoder schemes of autoencoder, utilizing characteristics of both autoencoder and LSTM neural network. The algorithm for proposed model is shown in Algorithm 1. The first dataset is composed of a succession of time intervals [x_1, x_2, x_3, ……, x_n].

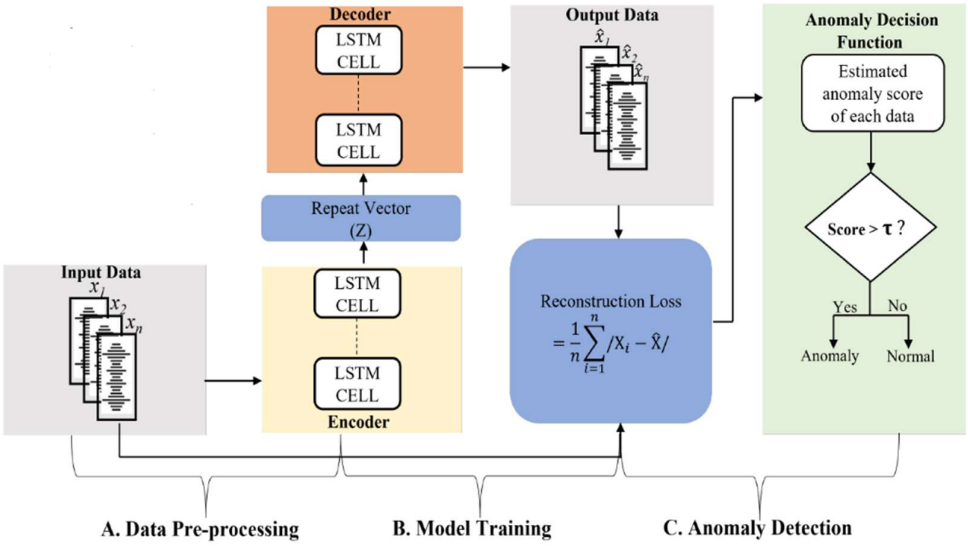

Figure 6. LSTM AE anomaly detection model

The LSTM encoder is to transform features into a batch of time-based feature sequences, much like a sequence folding layer. It is comparable to performing independent convolution operations on feature a sequence timesteps. The high-dimensional input data a sequence is obtained by the encoder as a fixed-size vector. The data handled by the encoder system maintains dependencies between several data points by a time-series a sequence by using the LSTM memory cells. After the sequence folds on timesteps, the LSTM decoder functions as an order unfolding layer, restoring the input data's a series structure. Reconstruction error rates are used to set a threshold, and it develops the fixed-size data a series from the minimized model of the input information in the space of latent information. It uses the threshold to find abnormalities.

3.5.4 Anomaly Detection

An anomaly is a significantly different observation from most data. A threshold determines how much deviation is acceptable. Any observation exceeding this threshold is considered an anomaly. Using this threshold-based anomaly detection method, the dataset is used to train the model. In doing so, the reconstruction error rates corresponding to the normal data points will be obtained. The maximum error rate is established as a threshold when training is complete and every possible reconstruction error has been calculated for every sample. Enter the testing dataset after deciding on the threshold. This sample is regarded as anomalous if reconstruction error rate exceeds cutoff.

Algorithm 1 Proposed Anomaly Detection

Input:

Training set $\{x_0, x_1, x_2, \ldots\ldots\ldots, x_{n-1}\}$,

Test set $\{\grave{x_0}, \grave{x_1}, \ldots\ldots\ldots, \grave{x_{m-1}}\}$,

Timesteps t

Output: Normal set (N_t), anomalies set (A_t)

begin

/ Step 1: To set everything up */*

X_i : training set

$\grave{X_i}$: testing set

for i ∈ $[0, n-t)$ do

$X_i = [x_i :: x_{i+t}]$

end

for i ∈ $[0, m-t)$ do

$\grave{x_i} = [\grave{x_i} :: \grave{x_{i+t}}]$

end

/ Step 2: training */*

Set proposed model's initial parameter (M).

for $X_i \in [X_0, X_1, X_2, \ldots\ldots, X_{n-t})$ do

$$\hat{X}_I = M(X_i)$$

$$L_{err} = \sum |X_i - \hat{X}_i|$$

Update LSTM-AE to minimize L_{err}

end

/ Step 3: Reconstruction loss computation */*

Function RLOSS(X):

for i ∈ (0; n - t) **do**

$$L(x - \hat{x}) = \frac{1}{n} \sum_{n=1}^{n} |\hat{x}_t - x_t|$$

end

return $L(x - \hat{x})$;

End Function

End

4. Experimental Results

After looking into a variety of anomaly detection methods for AVs that have been published in the literature, a novel method called LSTM-AE was proposed, and the analysis of its findings is covered in this part. The test dataset contained both typical and unusual information. Segment the dataset into training, validation, and test sets prior to training the model. To reduce the reconstruction loss, training is carried out utilizing the training and validation sets. Plot the training and validation error against the number of epochs to assess the effectiveness of the suggested model training, as shown in Figure 7.

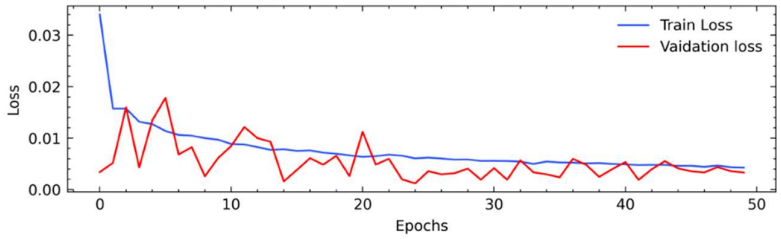

Figure 7. Training and validation loss

4.1 Performance Metrics for Evaluation

The efficiency of the suggested models was evaluated based on five criteria: receiver operating characteristic (ROC), accuracy, recall, precision, and precision. These metrics gauge the model's ability to detect anomalies and are derived from both detected anomalies and ground-truth labels.

4.2 Comparison with existing work

Figure 8 displays the evaluation metrics result for five algorithms. The LSTM-AE classifier achieved the highest accuracy at 92.1%, alongside leading F1-score (86.9%), recall (85.7%), and precision (88.7%). The proposed LSTM-AE method is compared against CNN, RNN, DRL and DBN.

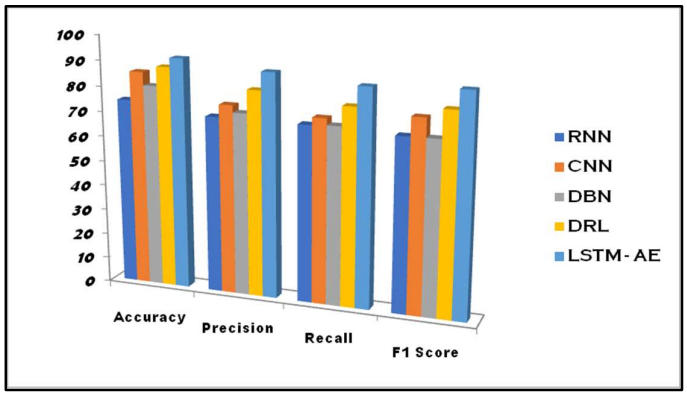

Figure 8: Comparison of RNN, CNN, DBN, DRL with LSTM – AE

The area under the ROC curve, or AUC-ROC, is a metric that expresses a classification model's performance across all possible classification criteria shown in figure 9. Plotted on the x and y axes, respectively, the classification threshold's variation from 0 to 1. In other words, superior detection ability is shown by a sharper ROC curve; a higher true positive rate corresponds to a lower false positive rate. It aids the model's ability to discriminate between various classes.

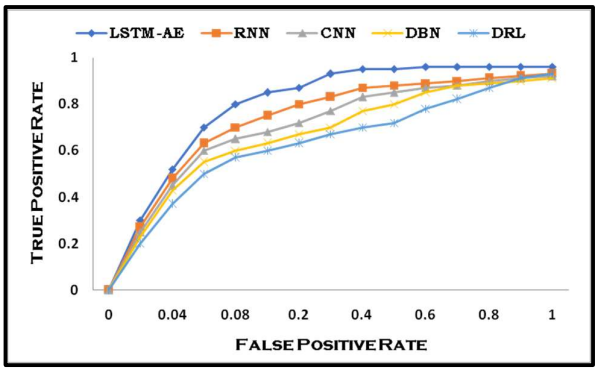

Figure 9: ROC curves of different models

Implementing anomaly detection in autonomous vehicles using LSTM Autoencoder (AE) offers significant implications. Firstly, it enhances safety by detecting deviations from normal driving patterns, alerting drivers or initiating corrective actions promptly. Secondly, it optimizes vehicle maintenance by identifying anomalous behavior in vehicle components, preventing potential failures and reducing downtime. Moreover, it contributes to cybersecurity by detecting abnormal network activities, thwarting potential cyberattacks. Additionally, it improves fleet management by providing insights into driver behavior and vehicle performance, enabling better decision-making. Lastly, it fosters trust in autonomous technology, showcasing its ability to adapt and respond to dynamic real-world scenarios effectively.

5. Conclusion and Future Work

The development of autonomous vehicles holds immense promise for revolutionizing road safety and transportation efficiency. However, challenges such as anomalies pose significant threats to the seamless functioning of these systems. This article proposed a novel data-driven approach using a symmetrical LSTM-AE to detect anomalies, particularly focusing on FDI attacks targeting the control system through compromised sensors. The evaluation of the proposed model through 5-fold cross-validation demonstrated its remarkable performance. With an average detection accuracy of 92.1%, precision of 88.7%, sensitivity (recall) of 85.7%, and F-Score of 86.9%, the model showcased superior capabilities compared to existing approaches. These outcomes highlight the effectiveness of the suggested model in accurately identifying anomalies, thereby enhancing the security and reliability of autonomous vehicle systems.

Future work in autonomous vehicles anomaly detection firstly, advancements in sensor technology will be crucial. This includes not only improving the resolution and range of existing sensors like LiDAR, radar, and

cameras but also integrating emerging technologies such as infrared sensors and advanced imaging techniques to provide a more comprehensive view of the vehicle's surroundings. Ultimately, by advancing the state-of-the-art in anomaly detection, future work in this area holds the promise of unlocking the full potential of autonomous vehicles, making transportation safer, more efficient, and more accessible for all.

Reference

1. Wang, Y.; Zhang, R.; Masoud, N.; Liu, H.X. Anomaly detection and string stability analysis in connected automated vehicular platoons. Transp. Res. Part C Emerg. Technol. 2023, 151, 104114.
2. Parvathala, Balakesava & Manikandan, A. & Vijayalakshmi, P. & Parvez, M. & Gopalan, S. & Ramalingam, S.. (2024). Bio-Inspired Metaheuristic Algorithm for Network Intrusion Detection System of Architecture. 10.4018/979-8-3693-5276-2.ch004
3. Ali, R., Manikandan, A., Lei, R. et al. A novel SpaSA based hyper-parameter optimized FCEDN with adaptive CNN classification for skin cancer detection. Sci Rep 14, 9336 (2024). https://doi.org/10.1038/s41598-024-57393-4.
4. Harihara Gopalan, S., Muzammil Parvez, M., Manikandan, A., & Ramalingam, S. (2024). Cognitive radio spectrum allocation using Nash equilibrium with multiple scheduling resource selection algorithm. Ain Shams Engineering Journal. https://doi.org/10.1016/j.asej.2024.102688.
5. Mahalakshmi, G., Ramalingam, S. & Manikandan, A. An energy efficient data fault prediction based clustering and routing protocol using hybrid ASSO with MERNN in wireless sensor network. Telecommun Syst (2024). https://doi.org/10.1007/s11235-024-01109-6
6. Abdulsaheb, Ghaidaa & Selvaraj, Sekaran & Manikandan, A & Palanisamy, Satheeshkumar & Uddin, Mueen & Khalaf, Ibrahim & Abdelhaq, Maha & Alsaqour, Raed & Khalaf, Osamah. (2023). Reverse polarity optical Orthogonal frequency Division Multiplexing for High-Speed visible light communications system. Egyptian Informatics Journal. 24. 100407. 10.1016/j.eij.2023.100407.
7. Gopalan, S. & Manikandan, A. & Dharani, N P & Sujatha, G.. (2024). Enhancing IoT Security: A Blockchain-Based Mitigation Framework for Deauthentication Attacks. International Journal of Networked and Distributed Computing. 10.1007/s44227-024-00029-w.
8. Parvathala, Balakesava & Manikandan, A. & Vijayalakshmi, P. & Parvez, M. & Gopalan, S. & Ramalingam, S.. (2024). Bio-Inspired Metaheuristic Algorithm for Network Intrusion Detection System of Architecture. 10.4018/979-8-3693-5276-2.ch004.
9. Kolli, Srinivas & V., Praveen & John, Ashok & Manikandan, A.. (2023). Internet of Things for Pervasive and Personalized Healthcare: Architecture, Technologies, Components, Applications, and Prototype Development. 10.4018/978-1-6684-8913-0.ch008.
10. Palaniappan, Mathiyalagan & Annamalai, Manikandan. (2019). Advances in Signal and Image Processing in Biomedical Applications. 10.5772/intechopen.88759.
11. V. A.R, S. David, E. Govinda, K. Ganapriya, R. Dhanapal and A. Manikandan, "An Automatic Brain Tumors Detection and Classification Using Deep Convolutional Neural Network with VGG-19," 2023 2nd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA), Coimbatore, India, 2023, pp. 1-5, doi: 10.1109/ICAECA56562.2023.10200949.
12. Manikandan, A., Madhu, G.C., Flora, G.D. et al. Hybrid Advisory Weight based dynamic scheduling framework to ensure effective communication using acknowledgement during Encounter strategy in Ad-hoc network. Int. j. inf. tecnol. (2023). https://doi.org/10.1007/s41870-023-01421-5.
13. P.Nagarajan,M.Renuga, A.Manikandan, S.Dhanasekaran, “Design and simulate a novel 16T SRAM cell for Low Power Memory Architecture”, Journal of Circuits, Systems and Computers, 2023, https://doi.org/10.1142/S0218126624500038.
14. Xue X, Palanisamy S, A M, Selvaraj D, Khalaf OI, Abdulsahib GM. A Novel partial sequence technique based Chaotic biogeography optimization for PAPR reduction in eneralized frequency division multiplexing waveform. Heliyon. 2023 Aug 24;9(9):e19451. doi: 10.1016/j.heliyon.2023.e19451. PMID: 37681146; PMCID: PMC10481292.

15. Ali, R., Manikandan, A. & Xu, J. A Novel framework of Adaptive fuzzy-GLCM Segmentation and Fuzzy with Capsules Network (F-CapsNet) Classification. Neural Comput & Applic (2023). https://doi.org/10.1007/s00521-023-08666-y
16. Sengolrajan, T. & Chandramohan, Kalaivani & John, Ashok & Manikandan, A.. (2023). A Novel Design of 9 Level Cascade Multi-level inverter for Decoupled Double Synchronous Reference Frame in State Delay Controller. Journal of Engineering Research. 100106. 10.1016/j.jer.2023.100106.
17. Chandramohan, K., Manikandan, A., Ramalingam, S., Dhanapal, R. (2023). Performance Evaluation of VANET using Directional Location Aided Routing (D-LAR) Protocol with Sleep Scheduling Algorithm. Ain Shams Engineering Journal. 102458. https://doi.org/10.1016/j.asej.2023.102458.
18. Gopalan, S.H., Ashok, J., Manikandan, A. et al. Data dissemination protocol for VANETs to optimize the routing path using hybrid particle swarm optimization with sequential variable neighbourhood search. Telecommun Syst (2023). https://doi.org/10.1007/s11235-023-01040-2
19. Reka, R., Manikandan, A., Venkataramanan, C. et al. An energy efficient clustering with enhanced chicken swarm optimization algorithm with adaptive position routing protocol in mobile adhoc network. Telecommun Syst (2023). https://doi.org/10.1007/s11235-023-01041-1
20. Dr.S. Vijayalakshmi. Early detection of breast cancer using robust back propagation neural network classifier. Rom Biotechnol Lett. 2022; 27(2): 3407-3415 DOI: 10.25083/rbl/27.2/3407.3415
21. S. Sadesh, Dinesh Valluru. A. Manikandan. Hybrid Approach for Human Diseases Prediction Using Air Quality Index. Rom Biotechnol Lett. 2022; 27(1): 3270-3281 DOI: 10.25083/rbl/27.1/3270-3281
22. Annamalai, Manikandan & Muthiah, Ponni. (2022). An Early Prediction of Tumor in Heart by Cardiac Masses Classification in Echocardiogram Images Using Robust Back Propagation Neural Network Classifier. Brazilian Archives of Biology and Technology. 65. 10.1590/1678-4324-2022210316.
23. Manikandan, Annamalai, M,Ponni Bala. (2023). Intracardiac Mass Detection and Classification Using Double Convolutional Neural Network Classifier. Journal of Engineering Research. 11(2A). 272-280. 10. 36909/jer.12237.
24. Venkataramanan, C. & Ramalingam, S. & Manikandan, A.. (2021). LWBA: Lévy-walk bat algorithm based data prediction for precision agriculture in wireless sensor networks. Journal of Intelligent & Fuzzy Systems. 41. 2891-2904. 10.3233/JIFS-202953.
25. Nilabar Nisha, U.,& Manikandan, A.& Venkataramanan, C.& Dhanapal R. (2023). A score based link delay aware routing protocol to improve energy optimization in wireless sensor network. Journal of Engineering Research. https://doi.org/10.1016/j.jer.2023.100115
26. Balamurugan, D. & Seshadri, s.Aravinth & Reddy, P. & Rupani, Ajay & Manikandan, A.. (2022). Multiview Objects Recognition Using Deep Learning-Based Wrap-CNN with Voting Scheme. Neural Processing Letters. 54. 1-27. 10.1007/s11063-021-10679-4.
27. Bommaraju, K., Manikandan, A., & Ramalingam, S. (2017). Aided System for Visually Impaired People in Bus Transport using Intel Galileo Gen-2: Technical Note. International Journal of Vehicle Structures and Systems, 9(2), 110–112. https://doi.org/10.4273/ijvss.9.2.09
28. Manikandan, A., Ramalingam, S., & Aathi bhagavan, V. (2016). Potholes Alert System for Riders. International Journal of Advances in Natural and Applied Sciences, 10(9), 440–444.
29. Amit Grover, A. Manikandan, Vivek Soi, Anu Sheetal, Mehtab Singh, Realisation of white LED using fiber based hybrid photonic structures, Optoelectronics and Advanced Materials - Rapid Communications, 15, 11-12, November-December 2021, pp.521-527 (2021).
30. Sheikdavood K, Surendar P, Manikandan A. Certain Investigation on Latent Fingerprint Improvement through Multi-Scale Patch Based Sparse Representation. Indian Journal of Engineering. 2016; 13(31):59-64.
31. Manikandan, A., & Nirmala, V. (2015). A Low Cost Thermoelectric Refrigerator. International Journal of Applied Engineering Research, 10(55), 3097–3101.
32. Nirmal Kumar, P., & Manikandan, A. (2017). Zigbee Based Online Air Pollution Monitor. Journal of Chemical and Pharmaceutical Sciences, 230–232.
33. Manikandan, A., & Pradeep, S. (2017). IoT Based Electricity Theft Detection. International Journal of Control Theory and Applications, 10(12), 211–215.

34. Manikandan, A., & Tamilselvan, S. (2017). RFID Based Voice Bank Alert System for Blind People. Journal of Chemical and Pharmaceutical Sciences, 308–310.
35. Manikandan, A., & Sakthivel, J. (2016). IoT Based Real Time Traffic Analyzer. Journal of Chemical and Pharmaceutical Sciences, 139–142.
36. Manikandan, A., & Sakthivel, J. (2017a). Recognizable Proof of Biometric System With Even Distorted And Rectification States. Journal of Advanced Research in Dynamical and Control Systems, 9(2), 1393–1398.