Journal of Autonomous Intelligence (2024) Volume 7 Issue 1
doi: 10.32629/jai.v7i1.751

ORIGINAL RESEARCH ARTICLE

Mitigation of network security attacks in wireless multimedia sensors networks using intrusion detection system

M. A. Matheen, S. Sundar*

Vellore Institute of Technology, Vellore 632014, India

*** Corresponding author:** S. Sundar, sundar.s@vit.ac.in

ABSTRACT

In this work, research propose a deep transfer learning-based Lecun network P-Lenet technique to move the knowledge contained in source domain data to the target domain and integrate to produce an efficient intrusion detection system (IDS) that would enhance the detection accuracy for any wireless multimedia sensors networks (WMSN) ecosystems. To prevent attacks on the software defined network (SDN) platform in real time, the proposed method placed a strong emphasis on anomaly detection as the primary mechanism. The Lecun network (LeNet) was investigated in this work, and a new variant of the network called the Lenet was proposed. In addition, research make use of techniques like as feature normalization to increase the accuracy of the algorithm predictions and to optimize the training process in such a way that it consumes the least amount of time and resources. The performance of the proposed model that was recommended was superior to that of existing network intrusion detection system (NIDS) algorithms.

Keywords: network security; wireless multimedia networks; security attacks; intrusion detection system

ARTICLE INFO

Received: 19 June 2023
Accepted: 28 July 2023
Available online: 8 October 2023

1. Introduction

The function of a multimedia communication service is the transfer of audio, video, and data files in real time between two or more recipients. Only over a private network is it possible to deploy multimedia services, and there is currently no way for services that are running on different networks to connect with one another. Only a private network can support the use of multimedia services. A growing number of multimedia services are becoming available as the technology that underpins sensor networks continues to undergo rapid advancement, and packet switching is increasingly being utilized as the foundation for the carrier network that allows the transmission of multimedia data[1].

An intrusion detection system, often known as an IDS, is responsible for monitoring a computer network or the logs of a computer system to look for possibly hostile activity. However, it is feasible for IDS systems to be compromised if the authorized users act in a way that puts the system security at risk. Because there are so many kinds of IoT networks, developing an intrusion detection system (IDS) that is up to the task of safeguarding them all can be a difficult task. IDS, also referred to as IDSs for short, are pieces of hardware that can recognize malicious actions. This term refers to anything that is done to detect illegal conduct that is taking place on computers and other devices that are connected to a network. The software that was

developed especially for this objective is able to recognize and terminate any conduct that is deemed to be abnormal[2].

Network intrusion detection system (NIDS) is a type of intrusion detection that operates covertly and in real time. The host level and the network level are both simultaneously included in its scope of detection at the same time. A controlled network architecture is the only type of architecture that can be used for NIDS successfully. NIDS has several benefits, including lower overall costs and quicker response times, and one of those benefits is affordability. This is because it is not necessary to keep the programming for sensors updated on the host computer, which is the reason for this result. By utilizing NIDS, it is feasible to recognize attacks in real time, and the performance of the traffic monitoring comes incredibly near to that of real time[2].

It has limited access to the workings of the host machine and hence cannot assess whether the attacks were effective. There is also no reliable method for analysing encrypted network traffic that may be used for the purpose of carrying out this duty. In addition, NIDS may have difficulties catching all the packets in a network that is either very busy or very vast. This issue may be caused by any of these factors. It is susceptible to being misled by an attack that is carried out while it is being utilized by many people[3].

A significant effort on the part of researchers to explore ways in which NIDS can be used to its fullest over the course of several years. This type of software is also possible, but it's essentially analysing the traffic on a network, looking for violations of the rules governing, and finding potential vulnerabilities. In either case, the software or hardware analyses the traffic on a network[3].

An efficient NIDS can successfully incorporate network anomalies as a component of the solution to the security problem. This is possible because the NIDS primary objective is to discover unexpected network traffic or behaviour. NIDS ensures the early identification and mitigation of threats and attacks, which is extremely crucial considering how difficult it is to stop them from happening in the first place.

To identify any potentially harmful activities, the NIDS analyses a duplicate of the traffic that travels over the network. Because of their adaptability and aptitude to manage sophisticated logic without interrupting current systems, hybrid approaches are becoming an increasingly attractive choice for the purpose of constructing IDS. While their training, agents that have been instructed to make use of hybrid methods are instructed to learn how to use classification rules. The dataset that is produced by network activity is commonly utilized in its entirety by the existing network IDS. This includes all the properties of the dataset. There are some components of the system that are unnecessary and do not contribute to the detection of intrusions in any way. The optimization of features is an absolute requirement to reduce the amount of time required for detection, bring the false alarm rate (FAR) down, and increase the percentage of incidents that are detected.

This paper highlights the work related to NIDS algorithms based on this new algorithm is proposed deep learning-based P-LeNet. The objective of the paper is to mitigate the network security attacks in wireless sensors networks using the proposed algorithms. Here the papers are the following section's introduction, related work, proposed method, results of the proposed methods. Later it concludes with summary of results.

2. Related works

An anomaly can be detected by looking at the typical activity of sensor nodes and then searching for any variations from that pattern. This type of detection can be used to determine whether an intrusion has taken place. In contrast to abuse detection and detection based on specifications, abnormal detection is more accessible to the public. People find it simpler to obtain abnormal detection than the other two types. Researchers make use of this methodology as their primary way of intrusion detection, and the following strategies are utilized in doing:

Onat and Miri[4] made a presentation about their idea for an intrusion detection system that might potentially identify resource depletion attacks. To construct a statistical model, every node keeps a running

count of the average receiving rate as well as the average arrival rate of packets from other nodes in its local region. This information is used to compare the two rates. Only the data packets that originate from nodes that are geographically close to the one being analysed are taken into consideration during statistical analysis. The subsequent packet won't be acknowledged until it checked to see if it matches the demographic profile of this neighbour node, and only then will it be acknowledged. The statistical model has been oversimplified, which means that it is incapable of defending against complex threats like selective forward attacks, worm holes, and so on. Another key deficiency is that it does not consider the amount of time and effort that is spent on computations, which is an extremely important factor.

Chen et al.[5] proposed for three-layer WSNs the use of an isolation table for the purpose of implementing an intrusion detection technique. It is the responsibility of the detection agent to isolate any nodes that look suspicious based on the information included in the isolation table. On a cluster, isolation tables may be constructed at any given time by any node that is a part of the cluster. These two heads collaborate with one another to make sure that everything functions well. When the isolation table is eventually sent to the base station, it will be feasible to send copies of it to each of the nodes in the network. The results of the simulations indicate that the quantity of energy that is used by the model increases in direct proportion to the number of sensor nodes that are included in the network. Furthermore, the model does not consider any nodes that have been successfully seized or that have been unsuccessfully captured.

To investigate how a sensor responds to both internal and external threats, Shi et al.[6] created a state transition model that was based on the continuous-time Markov chain. The Markov chain with continuous time (MCCT) was used in the construction of this model. The model accounted for the existing circumstances of WSN in terms of its viability, availability, and energy usage. In addition to this, the model linked the model with the detection model for an inside attack. This was done to achieve a satisfactory equilibrium between the capabilities of the network and the degree of protection it offers.

Cheng[7] came up with the idea of a differential game model that may be utilized with wireless sensor networks. This model incorporates both an adversary and an intrusion detection system into its composition. The model first achieves Nash equilibrium, and then it implements a solution that produces a happy medium between the load that the security system imposes and the threat that is there. In other words, the model finds a balance between the two. The predictive power of the model decreases in proportion to the growing number of sensors in the network, which results in fewer precise predictions.

Fu et al.[8] proposed a plan was expanded to include the implementation of a decentralized structure. There is no necessity for each node to have its own separate intrusion detection system, as this is not a requirement. Only the detection of potential danger is the responsibility of the peripheral node, while the central node oversees storing a repertory of antibodies and receptors that can be employed to keep an eye on potential intruders.

Xiao and Zhang[9] devised a model for real-time distributed IDS and identifies the potential dangers more immediately. Research on the creation of an intrusion detection technology for wireless sensor networks is still in its early stages, according to the findings of an examination of the several methodologies that are now being deployed for the purpose of detecting intrusions. Many of the detection systems that are now being used rely on intrusion detection technologies that have been adapted from traditional computer networks[10,11]. There is a school of thought that contends the characteristics of wireless sensor networks are insufficient, and that detection systems ought to be developed in a way that is consistent with this belief. The vulnerability of the intrusion detection system was only evaluated virtually, and not in any actual wireless sensor networks. Despite the benefits of theoretical research and computer simulations, a successful intrusion detection system must first have its practicability examined in the actual environment in which it will be used before it can be deemed to have achieved its goals[12].

3. Proposed method

A large number of people have developed an interest in IDS as a result of the ease and effectiveness with which these systems can identify malicious intrusion attempts. An IDS is a type of monitoring system that watches for suspicious activity in a computer system, either on the network or on the host machine itself, and then sounds an alarm when it finds it. Those individuals who are located within the components of the targeted system and have permitted network access are referred to as internal intruders. On the other hand, external intruders are those who come from a location that is not part of the network and with the intention of gaining unauthorized access. Proposed IDS architecture is shown in **Figure 1**.

3.1. Pre-processing

When attempting to learn and cut down on the size of the loss function, the features dataset should have values on a variety of scales so that the function can be reduced. While the procedure for gradient optimization is being carried out, these scales have a propensity to have an impact on the optimization of the learning rate. The reason for this is because research want the model to converge to the global or local minimum in the most expedient and accurate manner feasible. Adopting min-max normalization rather than the regular scaling methods because it has a number of advantages that are not shared by the standard scaling methods.

The min-max scaling method performs exceptionally well on our Network Security Laboratory-Knowledge Discovery in Databases (NSL-KDD) dataset. This is because applications for anomaly detection are not needed to conform to a certain distribution. This is the reason why this is the case. It is possible to avoid the gradient by using the Min-Max normalization approach, which also optimizes the loss function along the non-smoothed path leading to the global minimum. The equation that is about to be presented takes into consideration both the value that is the lowest and the value that is the highest in the column. This is done so that the resulting values will be between 0 and 1.

$$X' = (X - X_{min})/(X_{max} - X_{min}) \tag{1}$$

where X_{min} is smallest number, X_{max} is largest number, and X is original sample.

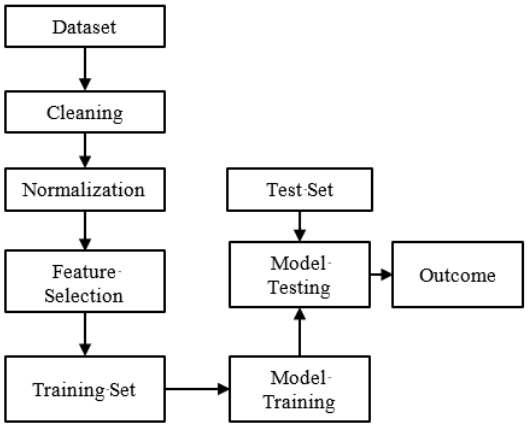

Figure 1. Proposed IDS architecture.

The data that comes from the source domain (Ds: As, Bs) consists of the pairs (As1, Bs1), (As2, Bs2), ..., (Asn, Bsm); on the other hand, the data that comes from the target domain (Dt: At, Bt) consists of the pairings (At1, Bt1), (At2, Bt2), ..., (Atn, Bsm). Within the context of the data class (Atn, Btm), the values 0 and 1 denote, respectively, the normal scenario and the attack scenario. These values are also shared by the source domain label data (Bs) and the target domain label data (Bt).

The Bs and Bt labels are part of the same feature space, the relative benefits, and drawbacks that each of these labels offers in a variety of domains are somewhat distinct from those that the other offers. Research

used a formula known as the Maximum Mean Discrepancy using Equation (1) in order to determine the separating distance between the two locations. The formula is as follows:

$$D(A_s, A_t) = \left\| \frac{1}{n}\sum_{i=0}^{n}\phi\left(A_{s_i}\right) - \frac{1}{m}\sum_{i=0}^{m}\phi\left(A_{t_i}\right) \right\|^2 \quad (2)$$

According to the dependency of DL models, the detection model that was trained on data from the source domain (Ds) does not perform well on data from the target domain (Dt), as the experiment that follows this statement indicates. This is a consequence of the fact that the detection model was trained on data from the source domain (Ds).

3.2. Architecture

Both the component for the training of models and the component for the detection of intrusions are included in the proposed block diagram for the P-LeNet model. Research have made use of the data after it has been cleansed and processed for the aim of training the model that research have proposed. Following this, empirical study was conducted, which ultimately resulted in the identification of the model parameters that are regarded as being of the utmost significance. The P-LeNet model that was suggested was first trained on a dataset that was picked at random, and it was afterwards evaluated on a dataset that was different from the one it was trained on. The ultimate IDS model was chosen by employing the model that demonstrated the highest level of accuracy in its predictions when tested on the validation dataset.

There are a total of 12,052 different variables that can be altered in the design that has been suggested for P-LeNet. These settings are spread across seven different tiers. A flattening layer, a completely connected layer, a fully connected output layer, and two convolutional layers are included in the development of this layer. A flattening layer, a fully connected layer, a fully connected output layer, and two convolutional layers are included in the development of this layer.

Nonlinear transformations are performed to the outputs of layers that come before it in order to construct a multivariate series. This allows for the generation of a multivariate series. The number of filters that are contained within that layer is the primary factor that decides the size of this series. The LeCun Network was investigated in this study, and a new variant of the network called the P-LeNet was proposed. **Figure 2** illustrates the P-LeNet general architecture.

It is impossible to consider the input layer of a network to be the first learning layer within that network because the input layer does not learn. The dataset is loaded into the system at the input layer, where it is subjected to processing before being passed on to the layer that comes after it. The dataset now has a total of four features thanks to the inclusion of the label feature.

At the first and third layers, a one-dimensional convolutional layer is responsible for performing a transformation on the dataset. This transformation takes place at both levels. (Conv1D). There are a total of five feature maps produced by the initial Conv1D layer, which has a kernel size of five as well. The second Conv1D layer, on the other hand, is responsible for the generation of a total of twenty feature maps. The activation function known as the Rectified Linear Unit (ReLU) is applied all during the process of convolution, which includes both layers. The trainable parameter range is from 30 all the way up to 520 inside of the two Conv1D layers.

Figure 2 demonstrates that the first Conv1D layer is followed by the first MaxPooling1D subsampling layer and that the second Conv1D subsampling layer is followed by the second MaxPooling1D subsampling layer. Additionally, the third Conv1D layer is shown to be followed by the third MaxPooling1D subsampling layer. In addition, the third MaxPooling1D subsampling layer comes immediately after the third Conv1D layer in this process.

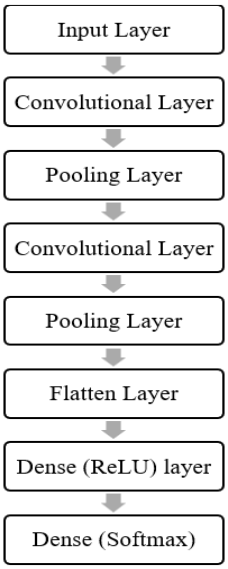

Figure 2. Proposed P-LeNet

Each of the two subsampling layers uses a down-sampling factor of 2 on the feature maps that it obtains from the layer that is directly above it. This is referred to as a down-sampling factor. Each of the feature maps that are created by the two subsampling layers corresponds to the feature maps that are provided as input from the layer that is below it in the sampling hierarchy. Five feature maps and twenty feature maps are generated, respectively, by the two subsampling layers. The pooled feature map is going to be turned into a single column thanks to the flatten layer, which is the fifth and final layer in the model that research have proposed. The next layer is going to be a dense one, and it will have a total of 10,500 trainable characteristics after all of the connections have been finished.

A considerable reduction in the total number of trainable parameters in a deep model is achieved by the use of this approach, which also enables the application of a class activation map. This map provides a means of comprehending the traits that have been acquired during training. The final step of the neural network is called the output layer, and it has the same number of neurons as there are distinct types of information represented in the training data. In this layer, the activation function utilized for making probabilistic predictions regarding the normal and attack states is the SoftMax function. This allows for the maximum amount of accuracy possible.

It is now possible for us to build a compiled version of the model that research have to determine how far off the network predictions are from the actual values that are contained in the training data while research is in the process of training the network.

Pseudocode:

Step 1: Initialize the wireless multimedia sensors network.

Step 2: Define the intrusion detection system (IDS) with a set of rules and parameters.

Step 3: Define the types of attacks to detect.

Step 4: Start monitoring the network traffic.

Step 5: Collect and analyze network traffic data.

Step 6: Apply data mining and machine learning algorithms to identify patterns and anomalies.

Step 7: If an attack is detected, generate an alarm, and notify the network administrator.

Step 8: Log the detected attack and take appropriate action to mitigate the attack.

Step 9: Update the IDS rules and parameters based on the detected attacks and their characteristics.

Step 10: Continuously monitor the network traffic and update the IDS rules and parameters accordingly.

By utilizing a technique for optimization and making use of the network loss values, it is possible to make many changes to the weights that are used in the network. However, to provide a more concrete verification, the performance of the recommended model was confirmed by utilizing the test dataset. This was done to provide a more concrete verification.

4. Results and discussions

In this section, research will detail the experimental setup to evaluate the model and it is compared with existing methods.

4.1. Experimental setup

The performance testing for the indicated model was carried out on a DELL workstation that featured 12 GB of RAM, the Microsoft Windows 10 Professional operating system, and a CPU from Intel that operated at 2.40 GHz and 2401 MHz and featured 2 cores and 4 logical processors. The workstation was equipped with an Intel Core i7-5500U, where python programming is used to model the proposed LeNet classifier.

4.2. Evaluation measures

In the process of determining how effective our solutions are, research make use of a number of performance indicators, some of which include precision in identifying malicious activity, recall of malicious behavior, the F-measure, and accuracy. When evaluating the performance, the following criteria are taken into consideration:

$$\text{Precision} = 100 * \text{TP}/(\text{TP} + \text{FP}) \quad (3)$$

$$\text{Recall} = 100 * \text{TP}/(\text{TP} + \text{FN}) \quad (4)$$

$$\text{F Measure} = (2 * \text{Precision} * \text{Recall})/(\text{Precision} + \text{Recall}) \quad (5)$$

$$\text{Accuracy} = 100 * (\text{TN} + \text{TP})/(\text{TN} + \text{FP} + \text{TP} + \text{FN}) \quad (6)$$

where TP is True Positive, TN is True Negative, FN is False Negative, and FP is False Positive.

For this evaluation, the accuracy measure will be utilized since it will make it possible to make a direct comparison between the results achieved by the proposed technique, when the binary classification issue was applied to them both. The degree to which the model can determine whether an attack has been carried out in a particular network environment can be gleaned from how well it performed on the binary classification test. Accuracy, which is defined as the degree to which the actual values of a classification are comparable to those that were predicted, is the metric that is used to determine how accurate a classification.

Figures 3–7 presented further information, which showed that the proposed model achieves a high level of accuracy (92%), which is an impressive accomplishment. There is a correlation between increased accuracy and reproducibility and repeatability, which refers to the consistency with which the results of the classification task can be measured multiple times under the same conditions. There is a positive association between increased accuracy and reproducibility and repeatability. Considering these discoveries, researchers can arrive at the conclusion that the proposed method, when applied to the binary classification job, generates results that are both satisfactory and dependable.

Both R2L and U2RL had detection score distributions that were considerably comparable (92% precision and 98% recall). There was only a small difference in terms of performance between DOS and Probe; even though both had recall and precision ratings of 90.1% and 90%, respectively, there was not much of a difference between the two. The F1-scores lead us to the conclusion that the recommended model performance against the four distinct types of attacks varies by no more than 2%, although the model precision and recall both vary by around 7%. The F1-scores lead us to the conclusion that the model performance against the four different types of attacks is consistent. This shows the robustness, efficacy, and vitality of the paradigm that was indicated, as well as its flexibility to various types of attack without lessening its efficiency.

When evaluating the proposed model against the four distinct types of attacks, research made use of a metric known as the receiver operating characteristic (ROC), which measures the trade-off between true positive rate (TPR) and false positive rate (FPR). This metric plot and evaluates the trade-off between the probability of detection and the chance of false alarm. This served to further ensure that the proposed method was both efficient and reliable.

In comparison with the F-score, the area under the curve (AUC) is not dependent on the threshold; rather, it evaluates the classifier in terms of both the threshold and the entire range of potential values. This contrasts with the F-score, which is dependent on the threshold. **Figure 3** demonstrates that the AUC for each of the four different classes is 98%. By employing an alternative strategy for IDS, it has been demonstrated that the proposed method can perform as expected and meets the requirements for attack classification.

Research have reached the conclusion that the method that was offered makes use of a variety of ways to get a result that can be independently confirmed. This was reached after coming to the previous conclusion. The NSL-KDD data set, which is widely recognized to be one of the most powerful benchmark datasets, was purposely selected for the purpose of our in-depth research and comparisons. It is vital to conduct several data statistics, cleaning, and verification methods on the dataset to deliver a learning experience that is free of difficulties caused by over- or under-fitting. This can be accomplished by comparing the dataset to a reference dataset that has been validated. This phase ensures that the model that was provided has unified data while also increasing the value of the data that was provided. To determine which of these three deep learning algorithms CNN, RNN, and LeNet is most suited to our endeavour, research conduct additional research and make a variety of comparisons based on these three machine learning algorithms.

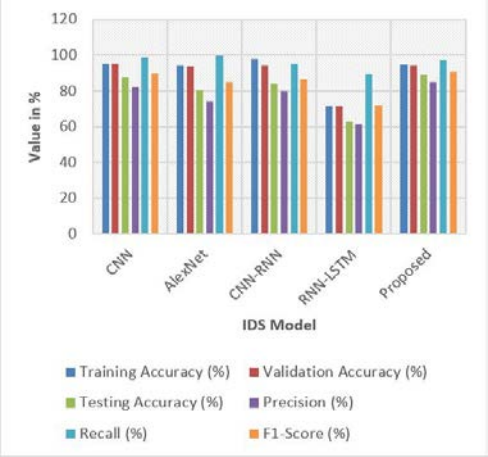

Figure 3. Performance evaluation of DARPA 1998 and DARPA 1999.

Figure 4. Performance evaluation of KDD CUP 1999.

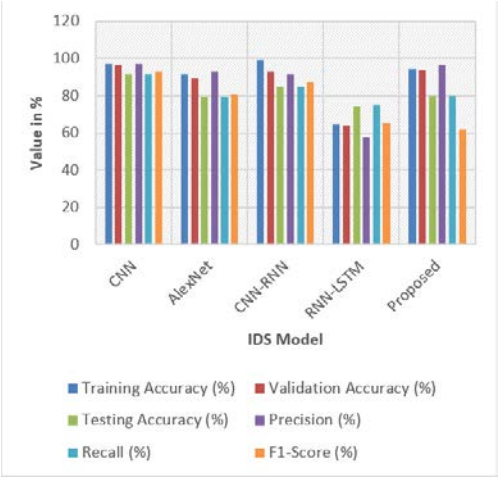

Figure 5. Performance evaluation of NSL-KDD.

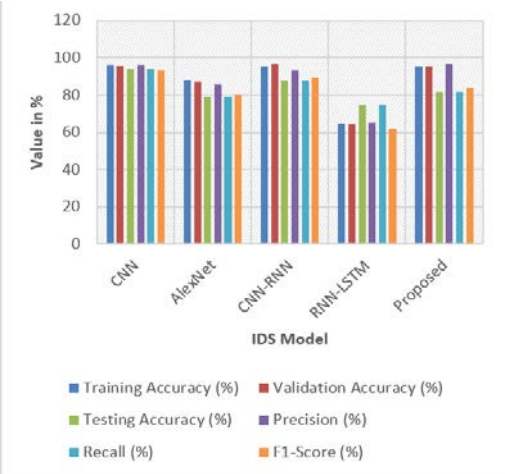

Figure 6. Performance evaluation of UNSW-NB15.

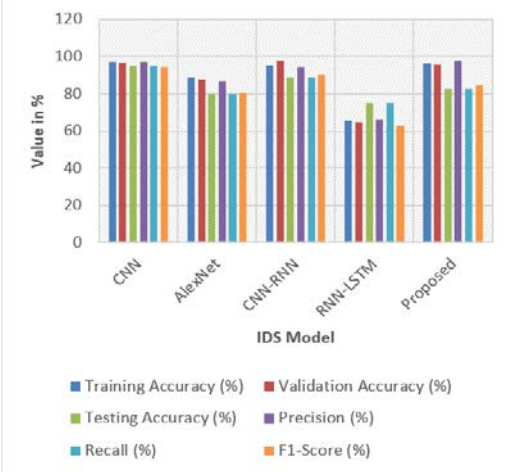

Figure 7. Performance evaluation of KYOTO2006+.

5. Conclusions

In this paper, we researched, trained, and evaluated our models using the reference dataset known as NSL-KDD, which is open to the general public. We made use of techniques such as feature normalization, feature selection, and data pretreatment to increase the accuracy of the algorithm predictions and to optimize the training process in such a way that it consumes the least amount of time and resources possible. Our goal was to boost the prediction ability of the algorithm as much as possible. The performance of the recommended

LeNet model was superior to that of four of the NIDS algorithms that were already available. To prevent attacks on the Internet of Medical Things (IoMT) platform in real-time, our proposed method placed a strong emphasis on anomaly detection as the primary mechanism. The proposed methods were able to assess whether or not an attack had taken place and determine the specific aspects of the attack simultaneously.

Author contributions

Conceptualization, MAM; methodology, MAM; software, MAM; validation, MAM; formal analysis, MAM; investigation, SS; writing—original draft preparation, MAM; writing—review and editing, MAM; supervision, SS. All authors have read and agreed to the published version of the manuscript.

Conflict of interest

The authors declare no conflict of interest.

References

1. Matheen MA, Sundar S. IoT multimedia sensors for energy efficiency and security: A review of QoS aware and methods in wireless multimedia sensor networks. *International Journal of Wireless Information Networks* 2022; 29: 407–418. doi: 10.1007/s10776-022-00567-6
2. Raveendranadh B, Tamilselvan S. An accurate attack detection framework based on exponential polynomial kernel-centered deep neural networks in the wireless sensor network. *Transactions on Emerging Telecommunications Technologies* 2023; 34(3): e4726. doi: 10.1002/ett.4726
3. Santhosh Kumar SVN, Selvi M, Kannan A. A comprehensive survey on machine learning-based intrusion detection systems for secure communication in internet of things. *Computational Intelligence and Neuroscience* 2023; 2023: 8981988. doi: 10.1155/2023/8981988
4. Onat I, Miri A. An intrusion detection system for wireless sensor networks. In: Proceedings of the WiMob'2005), IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, 2005. 22–24 August 2005; Montreal, QC, Canada. pp. 253–259.
5. Chen RC, Hsieh CF, Huang YF. A new method for intrusion detection on hierarchical wireless sensor networks. In Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication; 15–16 January 2009; Suwon, Korea. pp. 238–245.
6. Shi Q, Qin L, Song L, et al. A dynamic programming model for internal attack detection in wireless sensor networks. *Discrete Dynamics in Nature and Society* 2017; 2017: 5743801. doi: 10.1155/2017/5743801
7. Cheng ZM. A differential game model between intrusion detection system and attackers for wireless sensor networks. *Wireless Personal Communications* 2016; 90: 1211–1219. doi: 10.1007/s11277-016-3386-6
8. Fu RR, Zheng KF, Lu TL, Yang YX. Danger theory inspired intrusion detection model for wireless sensor networks. *Journal of China Institute of Communications* 2012; 33(9): 31–37. doi: 1000-436X(2012)09-0031-07
9. Xiao X, Zhang R. Study of immune-based intrusion detection technology in wireless sensor networks. *Arabian Journal for Science and Engineering* 2017; 42: 3159–3174. doi: 10.1007/s13369-017-2426-1
10. Saravanan V, Rajeshwari S, Jayashree P. Security issues in protecting computers and maintenance. *Journal of Global Research in Computer Science* 2013; 4(1): 55–58.
11. Gobinathan B, Mukunthan MA, Surendran S, et al. A novel method to solve real time security issues in software industry using advanced cryptographic techniques. *Scientific Programming* 2021; 2021: 3611182. doi: 10.1155/2021/3611182
12. Matheen MA, Sundar S. A novel technique to mitigate the data redundancy and to improvise network lifetime using fuzzy criminal search ebola optimization for WMSN. *Sensors* 2023; 23(4): 2218. doi: 10.3390/s23042218