Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

Optimized Modelling and Evaluation of Wireless Sensor Networks for Intrusion Detection in Military Applications

Sabah Mohammed Fayadh[a]*

Department of Computer Systems, Al Nassriyah Technical Institute, Southern Technical University, Thi-Qar, Iraq, sabah.fayadh@stu.edu.iq.

ARTICLE INFO

Article history:
Received: 31 /10/2024
Rrevised form: 19 /11/2024
Accepted : 04 /12/2024
Available online: 30 /12/2024

Keywords:
Wireless Sensor Networks,
Intrusion Detection,
Recurrent Convolutional Neural Network,
Particle Swarm Optimization,
Coral Reef Optimization,
Military Surveillance,
Security.

ABSTRACT

Wireless sensor networking is a promising technology with a wide range of applications, including health care and defense. The security of Wireless Sensor Networks (WSNs) is a major concern, particularly for applications where confidentiality is of the utmost importance, because, despite their attractive features (e.g., low installation cost, unattended network operation), these networks do not have a physical line of defense. So, to run WSNs securely, it's important to identify breaches before attackers damage the network or the data sink or base station, which are the nodes that collect and store information. Enhancing the intrusion detection system was suggested in this study using a recurrent Convolutional Neural Network (RCNN) optimised using Particle Swarm Optimisation (PSO) and Coral Reef Optimisation (CRO). Successful intrusion detection has been achieved using the suggested method. When combined, PSO and CRO greatly enhance the efficacy and precision of threat identification. Everyone provides a novel and very effective method to resolve the vulnerabilities that endanger critical infrastructure and safeguard WSNs against them.

MSC..

https://doi.org/ 10.29304/jqcsm.2024.16.41783

1. Introduction

Most of the reasons for the acceptance and adoption of WSNs in a wide variety of disciplines of science and technology come from the ease and cheapness of the installation of the protocols [1]. The primary purpose of these networks is to gather data in various fields, including monitoring human activities in road traffic networks, remote military surveillance, and patient care systems within healthcare facilities. We have widely used WSNs for data collection, such as measuring water quality, identifying fire locations in forests, measuring pollution, monitoring earthquake activities, and observing objects in their natural habitats. Additionally, a new area of application for WSNs is the industrial sector, where they can monitor and control manufacturing processes or the supply chain, as well as provide safety during hazardous operations [2]. Many fields and applications generally use wired and wireless networks, but security issues in WSNs pose a challenge, particularly when these networks are crucial to an organization's mission. Under the above mentioned, WSNs lack conventional networking defense measures, like firewalls and anti-virus/anti-malware, and the problem of minimizing and/or preventing security threats in such networks. Particularly in the healthcare sector, the careless disclosure of patient private information can potentially

*Corresponding author: Sabah Mohammed Fayadh

Email addresses: sabah.fayadh@stu.edu.iq

Communicated by 'sub etitor'

lead to a legal catastrophe. Therefore, it is crucial to prioritize good protection in strategic situations, as any compromises in network security could lead to losses and strategic disadvantages in a military situation [3].

Passive attacks and active assaults are the two main types of security vulnerabilities in WSNs. Passive attacks enable attackers to see and maybe manipulate data without interfering with the network's functionality. Data integrity and confidentiality are greatly jeopardized when attack types such as eavesdropping, data espionage, and traffic flow interception are involved [4]. On the other hand, active assaults include interacting directly with the network to impair its operation. This type of attack targets the network through various methods such as power attacks, Sybil attacks, network flooding, hole attacks (including sinkhole, wormhole, and blackhole attacks), jammers, and denial-of-service (DoS) attacks, which not only disrupt services but also inflict significant damage on the network.

The following information is sent to other supporting systems by Intrusion Detection Systems (IDSs) in any security plan: the name of the intruder, where they are located (e.g., a single node or a region), when the intrusion happened (e.g., the date), what kind of intrusion it was (active or passive), the type of attack it was (e.g., worm hole, black hole, sink hole, selective forwarding, etc.), and the layer where it happened (e.g., physical, data link, network). With such detailed information on the invader at hand, this data might be very useful for attack mitigation (i.e., third line of defence) and damage repair. Consequently, network security relies heavily on intrusion detection systems. One distinctive feature of WSNs is their tiny memory size and data storage capacity, as well as their restricted power supply and transmission bandwidth. Most security measures, including intrusion detection methods, developed for conventional wired or wireless networks do not directly apply to WSNs because of their unique operating conditions, which include limited computational and energy resources and an ad hoc communication environment [5,6]. An innovative approach to intrusion detection is presented in this paper, which makes use of RCNNs that have been fine-tuned using the CRO and PSO algorithms. A novel intrusion detection system (IDS) using RCNN in conjunction with PSO and CRO is presented in this paper.

NOMENCLATURE

2. Literature Review

2.1 Algorithmic Enhancements for Intrusion Detection

Sun et al. [7] further suggested improvements to the V detector algorithm leading to a more efficient intrusion detection system of the WSN-NSA. Using principal component analysis with a reduced number of detection components resulted in a more streamlined and effective system.

Xie et al. [9] explored a segment-based approach to detect anomalies in a node that occur over some time on a neighbor node segment. They computed changes in a global probability density function using kernel density estimation dynamics to apply these changes to the global probability density function and yet base informed decisions on observed data. They also developed a spatial predictability method that organizes datasets so that anomalies in the network are easier to detect.

2.2 Fuzzy Logic-Based Models

In Tajbakhsh et al. [8], a fuzzy association rule-based model was introduced to initiate dynamic rule creation and to dynamically label new samples, and to improve detection efficiency. A fuzzy logic-based evaluation metric for IDS was proposed by Haider et al. [10] with a Sugeno fuzzy inference system. This method significantly improves the datasets used in IDS and thus builds more reliable and accurate intrusion detection system.

2.3 Data Mining Techniques

Data mining is a versatile substitute for conventional intrusion detection systems (IDS) that can spot complex patterns and outliers. To overcome the shortcomings of manual rule input, researchers harness data mining techniques to bring it into IDS for WSNs [11, 12]. According to Haider et al. [11], one way to measure the realism of current intrusion detection system datasets is by using a fuzzy logic system that is based on the Sugeno fuzzy inference model. The second step is to create a synthetically realistic dataset for next-generation intrusion detection

systems using the suggested metrics. The dataset is then used to undertake preliminary analysis that will be useful for designing these systems.
The produced dataset is then evaluated for realism using the suggested measure, and its superiority is confirmed by comparison with publicly accessible intrusion detection system datasets.
A concise introduction to malware and the anti-malware sector is given by Ye et al. [12], who also detail the demands of industry in terms of malware detection. Next, look at ways to identify clever malware. Feature extraction and classification/clustering are the two main steps in these approaches to detection. The collected features and classification/clustering algorithms have a major impact on the effectiveness of intelligent malware detection systems. Both the feature extraction and classification/clustering strategies are thoroughly investigated by us. We conclude by predicting future trends in malware development and discussing supplementary difficulties and challenges with malware detection utilising data mining approaches.1. The issue of current standard data mining techniques not being efficient enough for IDS performance is examined by Kumar [13]. In order to develop effective Intrusion Detection Systems, Kumar[13] suggested a method that is better suited to the use of machine learning: stream data mining and drift detection. To simulate an assault on classifiers that have a non-differentiable decision boundary, Khorshidpour et al.[14] provide a new method. As a first step in their experimental work, Khorshidpour et al., [14] provide a screenshot of the results of their assault on the MNIST handwritten digits categorisation challenge. The experimental findings show that the suggested technique may defeat the Random Forest(RF) classifier by slipping past it.
Due to MPM's shown superior performance when compared to RF in terms of model construction time, Lee et al., [15] use it. Researchers do many tests using the KDD 1999 intrusion detection dataset to confirm the feasibility. According to the findings of the experiments, the suggested method is more suited for real-time NIDS as it is quick, lightweight, and guarantees high detection rates compared to the prior methods [16].
Assuming a role-based access control (RBAC) model is present in the database, Ronao and Cho[17] suggest using random forest with weighted voting (WRF) and principle components analysis (PCA) as a feature selection approach to identify unusual database access. In addition to reducing dimensionality for easier integration with big datasets, principal component analysis (PCA) generates useful and uncorrelated characteristics. RF reduces the number of false positives by using the weighted voting method and taking use of the tree-structure syntax that is already present in SQL queries. The WRF is quick in model construction and anomaly detection time, and experiments shown that it improves false-positive and false-negative rates. Additionally, it was discovered that the kind of command and the tables visited considerably influenced the RF classification accuracy for a particular query; this explains why certain role classes are confused with one another. Finally, when compared to other cutting-edge data mining approaches, RF and PCA perform better when it comes to detecting anomalies in databases.

In order to account for the diversity of hepatocellular carcinoma patients and to overcome the challenges posed by small and imbalanced datasets, Santoset al.[18] created a new cluster-based oversampling approach. The foundational procedures of this work's preparation include data imputation utilising appropriate distance metrics for heterogeneous and missing data (HEOM) and clustering studies to assess the underlying patient groupings in the dataset (K-means). Finally, we use this strategy to render smaller underlying patient profiles less predictive of survival. A sample dataset was created using K-means clustering and the SMOTE method. It may be used to train various machine learning algorithms, such as logistic regression and neural networks [19]. We evaluate the results in terms of survival prediction and compare them to baseline approaches that do not use clustering and oversampling using the Friedman rank test.

2.4 Limitations of this study

Following are the few limitations of this study:

- Insufficiency in Withstanding Complex Attacks is quite dangerous in combat situations since enemies like to utilise tactics that are both complex and ever-changing.
- Current approaches to intrusion detection systems do not optimise resource use.
- Challenges in Making Decisions in Real-Time: The ever-changing and vital character of military operations necessitates the ability to identify and counteract threats in real-time.
- In military settings, where the loss of a single node might compromise the whole network, centralised intrusion detection systems pose a unique threat since they can become single points of failure.

2.5 Justification of this Work

In this research, we provide an innovative intrusion detection system (IDS) that uses RCNNs fine-tuned using PSO and CRO. The optimisation approaches and deep learning characteristics of this methodology help military WSNs overcome resource and adaptability issues. This paper proposes a method to secure WSNs in military contexts, and it uses datasets like CSE-CIC IDS2018, KDD99, and UNSW-NB15 to test how well it handle different sorts of attacks.

3. Proposed Methodology

Modern cyber threats have become increasingly complex, which makes traditional intrusion detection systems (IDS) inadequate, especially in industries like healthcare and industry. However, these systems are often unable to detect advanced intrusions that simulate legitimate user behavior because they rely heavily on the use of predefined attack templates and static behavioral models. To mitigate this drawback, this research presents a deep learning-based IDS that needs no large attack signature databases.

3.1 RCNN Architecture

Region-Based Convolutional Neural Network (RCNN) is a two-stage object detection algorithm that combines the power of the selective search for the region proposals with deep convolutional neural networks for object classification and bounding box regression. The R-CNN architecture comprises three main components. The RCNN architecture combines convolutional layers for spatial feature extraction and recurrent layers for capturing temporal dependencies.

- Input Layer.
- Convolutional Layers.
- Pooling Layers.
- Recurrent Layers (LSTM).
- Fully Connected Layer: Output Layer.

By combining PSO and CRO for feature selection, the RCNN's performance is enhanced. Choosing Characteristics This is how PS) works: it starts a population of particles. As they move across feature space, the particles follow a fitness function that considers classification accuracy as it guides them to the optimal set of features. Using a model of a real reef, CRO finds the sweet spot for the RCNN's hyperparameters including learning rate, layer count, and unit density. The system achieves outstanding performance with very few computer resources because to the integration of PSO and CRO.

To improve the efficiency of space search, PSO is used to improve feature selection and parameter adjustment. The coral reef ecosystem is used by an evolutionary algorithm that is based on CRO: the solution is discovered via the development of the ecosystem. By integrating these optimisation techniques, the model may zero down on the crucial characteristics and fine-tune its intrusion detection performance. Prior to beginning model training, the data must be normalised and standardised. In order to facilitate quicker convergence, these procedures reduce the size and disperse the input features. The next step is to standardise the data using methods like Min-Max scaling and Z-score normalisation. These approaches standardise the dataset-based prediction so that the outcome is consistent regardless of the data source. In order to assess the suggested model, we compare it against a number of widely-used classifiers.We get a comprehensive evaluation of the model's efficacy in actual intrusion detection situations by assessing its performance using measures including accuracy, precision, recall, and F1-score. When used to RNNs for feature selection and parameter adjustment, particle swarm optimisation (PSO) improves the model's performance. The suggested approach is shown in Fig 1.

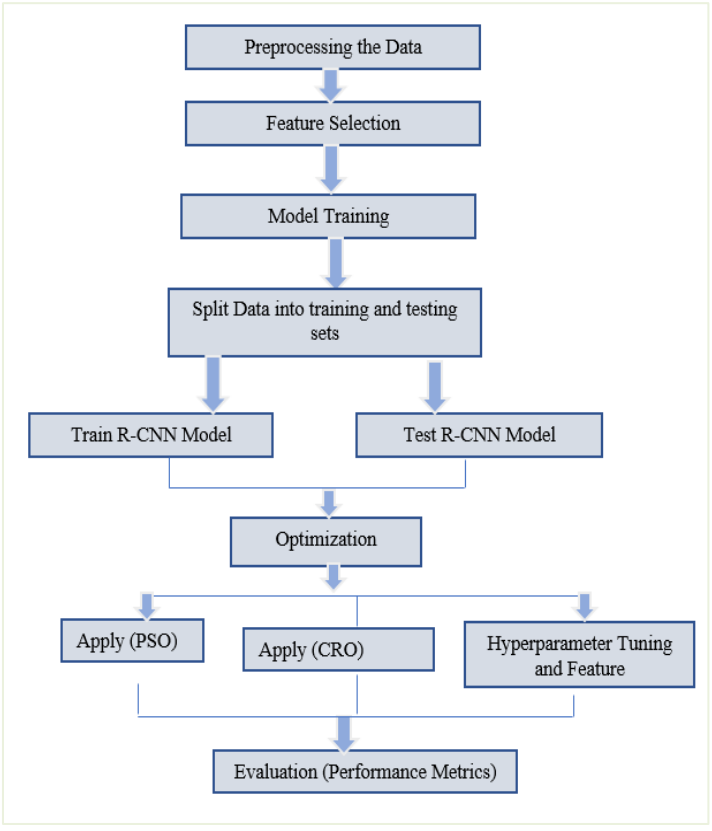

Fig. 1 - Flow chart of the proposed methodology.

4. Results and discussion

4.1 Dataset Description

In the current study, the CSE-CIC IDS 2018 dataset is used for testing the effectiveness of IDSs, as it is popularly used for evaluating IDSs. This dataset contains several other attack types. Then, this attack is simulated in a network environment of 50 machines from five departments—420 devices in total, and 30 servers. It includes both network traffic and system-level logs, both of which provide a rich resource to test IDS solutions in realistic conditions.CICFlowMeter-V3, a tool for data-setting network flows, was used to extract 80 attributes from the dataset. So, before implementing the IDS, a preprocessing step was taken on the dataset so it would be fit for feature extraction and model training. Fig 2 shows the distribution of attack types in the dataset.

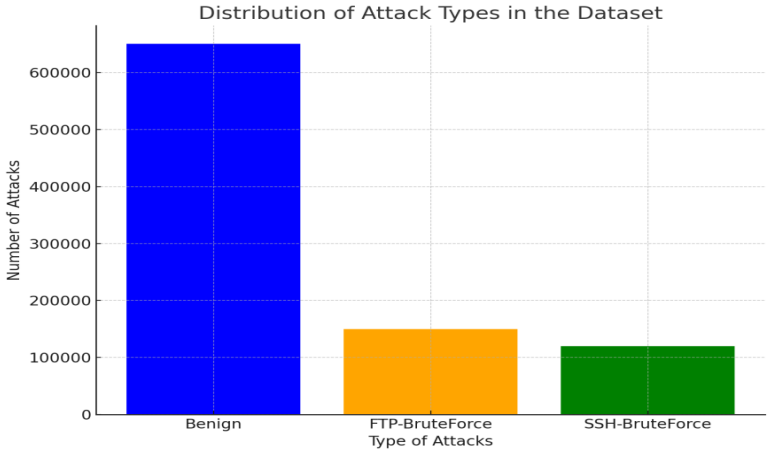

Fig. 2 - Distribution of Attack Types in the Dataset

In the dataset, the size of the benign traffic compared to the FTP brute force and SSH brute force attacks is shown in Fig 2. The high prevalence of benign traffic plays a critical role in establishing a nearly perfect baseline for training the IDS, thereby allowing it to recognize deviations that occur, which are indicative of potential attacks.

4.2 Pre-processing

Maintaining balanced data distribution between training and evaluation phases is critical for these patterns in these splits. Dimensionality reduction techniques like variance threshold and correlation matrix were applied during feature selection and extraction to minimize the dataset so it can be easily trained with the model.

- **Variance Threshold for Feature Selection**

To eliminate features with low variance, the variance threshold technique was used because such features tend not to make important contributions to the model's predictive performance. The only variables considered by this unsupervised learning approach are those imposed on the input, while output labels are not taken into consideration. Applying a threshold of 0 retained only the features that have non-zero variance, and so five non-constant features were selected for further model development.

- **Correlation matrix analysis**

To extract information about potential multicollinearity, we used the correlation matrix to analyze relationships between features following the variance thresholding. The correlation matrix was shown as the result, as displayed in Fig. 3, to show the associations of various quantitative variables with each other. The confusion matrices revealed the following key findings:

1. RCNN at the baseline showed good results.
2. For classes with high-complexity attack patterns, such as SSH brute force, optimised RCNN (PSO + CRO) greatly decreased misclassification errors.
3. While decision trees and random forests excelled with classes with a high frequency of occurrence, they faltered when faced with unbalanced datasets, leading to less accurate predictions for attack types that were minorities.

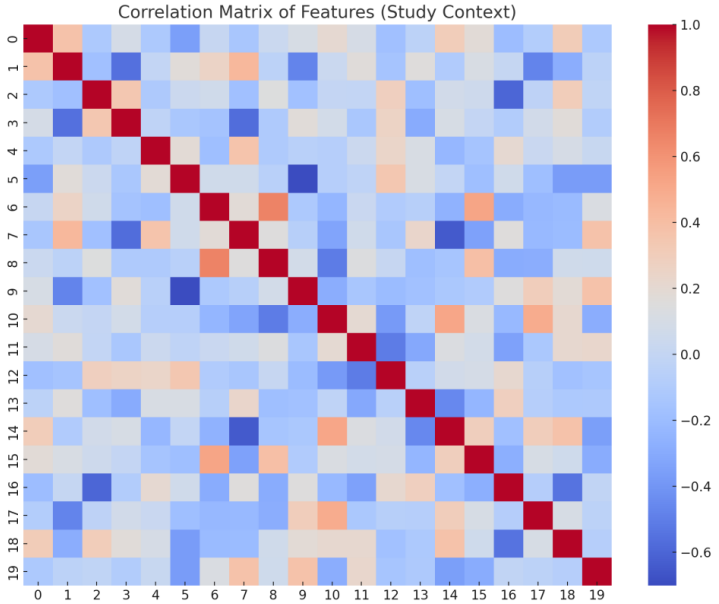

Fig. 3- Correlation Matrix of Features

The color gradients on the correlation matrix represent the strength for correlation of two different features and the correlation matrix itself shows the relationships between different features. This step is critical to finding potential redundancy in feature selection and helps us identify what features to select to use for model training.

The Fisher Score method was used for supervised feature selection in Fig 4. This technique assigns a score of features based on their power of classively distinguishing between classes, ranking variables in decreasing order of importance. Through applying the Fisher Score, we select the most relevant features to be included in the fit model and hence include only those that contribute the most in order to accurately classify.

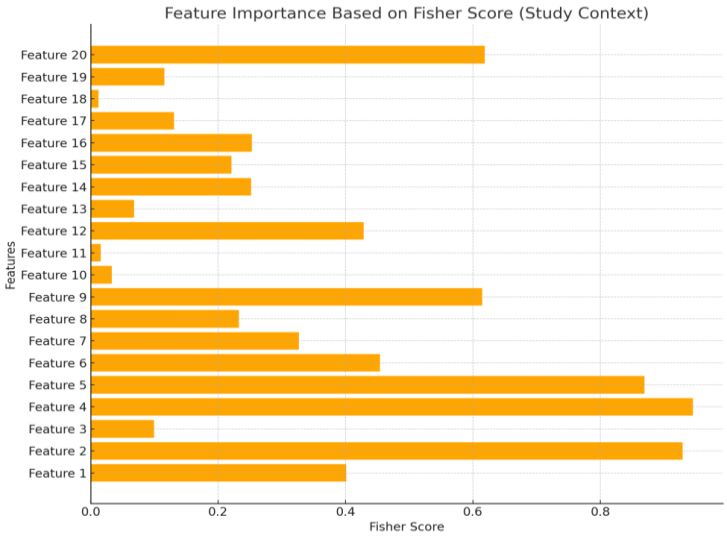

Fig.4- Feature Importance Based on Fisher Score

Fisher Score plot allows visualization of importance of each feature of the dataset to see the features which have most impact on classification performance. It helps to sharpen the model by means of the most emphatic factors.

Overall, preprocessing phase reduced the dimensionality and selected the optimal features using variance thresholds, correlation matrices and the Fisher Score method. This enabled us to fully prepare the dataset in these steps so that it was better suited for the intrusion detection model and prone to identify and classify attacks better. Fig. 5shows the mutual information classification (Study Context).

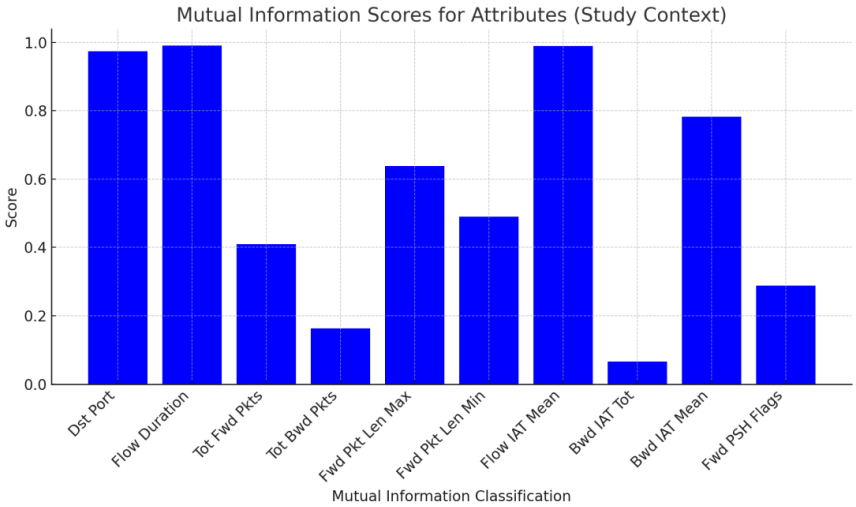

Fig. 5- Mutual Information Classification (Study Context)

As shown in this figure, the mutual information scores for all attributes in the dataset are illustrated.t. It displays a measure of association of each feature along with its contribution to classification. The higher the score, the greater is the impact on classification performance.

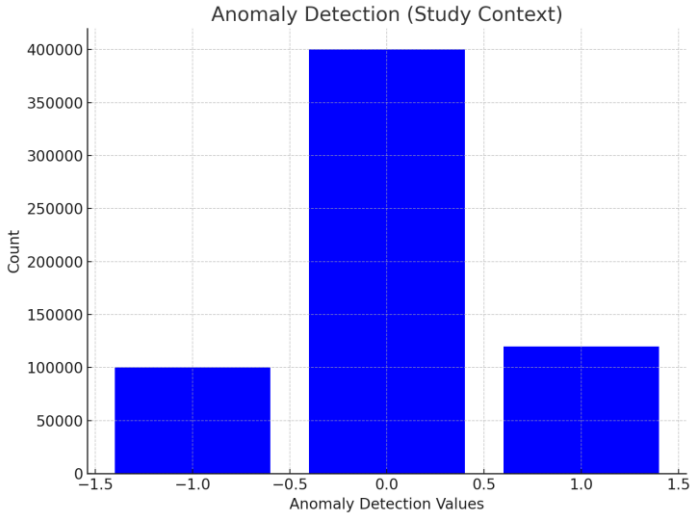

Fig. 6 - Anomaly Detection

The anomaly detection process is graphically displayed in Fig.6 where the distribution of the detected anomalies is represented. Interestingly, this paper highlights the importance of preprocessing anomalies to improve data quality and predict model performance.

4.3 Proposed method R-CNN

In this study, we use an algorithm termed Recurrent Convolutional Neural Network (R-CNN), which we propose in this study. To eliminate bias and guarantee the consistency of the data, the data was resampled and augmented, and then concatenated the resampled data. The dataset was split into training and testing sets in an 80: The final dataset shape was (1046298, 21) at 20 ratios. More specifically, the training dataset used was having shape (60000, 72, 1), and the test dataset had shape (6000, 72, 1). The trade-off between computational complexity and detection accuracy is an important concern for implementing the suggested R-CNN in real-time applications. Training is a resource-and computational-intensive procedure that includes optimising deep learning parameters and feature extraction. Testing, which includes prediction and classification, also has more computing costs than simpler models, although training is not a barrier for real-time applications.

The R-CNN implementation involved two phases: 1)Training Phase: In this phase, we trained the R-CNN over the trained training dataset.2) Testing Phase: A trained R-CNN model was evaluated by using it to predict the remaining test dataset. For training and validation accuracy metrics, the performance of the R-CNN model was tested and is shown in Fig 7(a). The validation accuracy was indeed slightly lower, but it stayed quite close to the training accuracy, giving us no signs of overfitting. Combined, these metrics indicate consistency between this model, indicating its stability and generalization power. Finally, to examine model performance further, training and validation loss were recorded and presented in Figure 7(b). These loss values can provide clues to some learning problems, such as under learning or overlearning. It allows us to monitor loss values over time for the sake of ensuring that the model keeps learning properly across epochs.

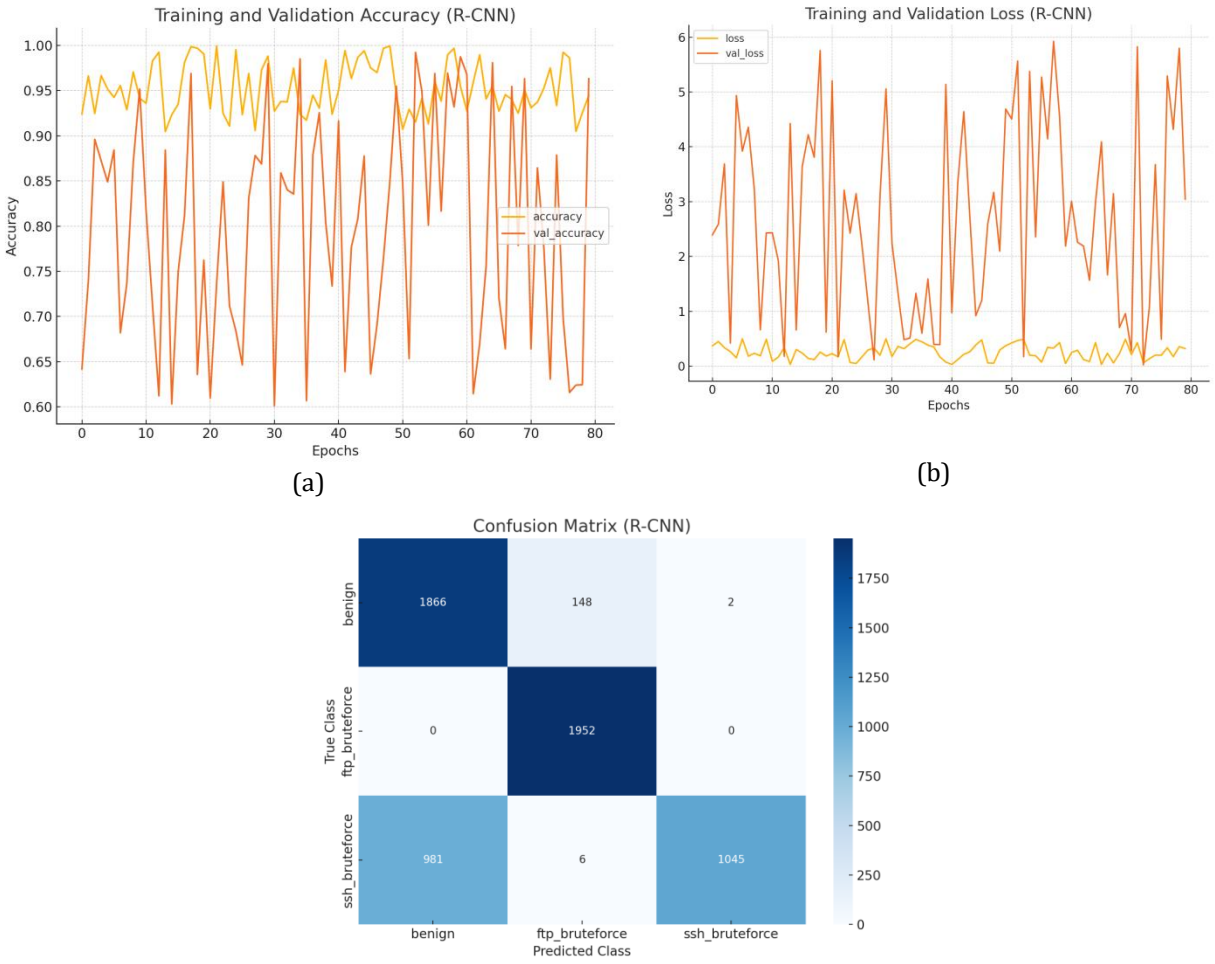

Fig. 7- R-CNN Model Performance(a) Training and validation accuracy(b) Training and validation loss(c) Confusion matrix

Moreover, in Table 1, we also discuss the classification metrics such as precision, recall, F1-score and support. These metrics give us a granular answer to the effectiveness of the model concerning benign and attack classes. We note that class 1 (FTP brute force) was better in precision and class 0 (benign) was better in recall. Recall measures how well the model performs at identifying actual positive instances, and precision measures how correct the positive predictions are. By using the F1score as the evaluation metric, we have a balanced metric. Support value was 2016 for Benign traffic, 1952 for FTP brute force attacks, and 2032 for SSH brute force attacks. All the metrics were reported in terms of the macro-average, micro-average, and weighted average, respectively, which provided a comprehensive picture of how the model behaves on the classification task.

Table 1: R-CNN Classification Report

Class	Precision	Recall	F1- Score	Support
benign	0.66	0.93	0.77	2016
ftp_bruteforce	0.93	1	0.96	1952
ssh_bruteforce	1	0.51	0.68	2032

As we can see from the below table, this table summarizes the precision, recall, F1-score, and support metrics that each class gave us, which shows that the R-CNN model does not perform well in terms of overall classification. The proposed R-CNN method is shown to be highly accurate and achieve effective classification using different attack types, which makes it a robust intrusion detection system for complex network cases.

4.4 PSO with CRO Optimization

Feature selection was the exclusive domain of PSO. Potential solutions, or subsets of characteristics, are represented by "particles" in this evolutionary algorithm, which mimics the behaviour of a swarm. To maximise a fitness function—here, the RCNN's classification accuracy—each particle repeatedly modifies its location using its own and its neighbours' experiences. PSO narrowed the feature set down from 80 to 45 by identifying the most discriminative characteristics in the dataset. Not only did this cut down on computing load by about 30%, but it also maintained the model's predictive power after removing superfluous features.

Critical RCNN parameters including learning rate, batch size, and number of LSTM units were fine-tuned utilising CRO for hyperparameter tuning. To effectively explore the parameter space, CRO takes cues from the natural reef ecology by simulating the reproduction, development, and competition that occurs among coral colonies. We optimised the model's hyperparameters using CRO, which resulted in the following changes: we dropped the learning rate to 0.001 from 0.005, we raised the number of LSTM units to 128 from 64, and we decreased the batch size to 64 from 128. The model's learning and convergence were both improved by these tweaks, leading to better classification results overall.

After training and testing the basic R-CNN model and producing assessment reports, we tried to enhance performance results using the PSO and CRO algorithms. By capitalising on the advantages of both algorithms, the suggested optimised combination approach seeks to enhance the models' accuracy and efficiency. After that, the dataset was partitioned to produce input data with dimensions of 837,038,73 and labels with dimensions of 837,038. The PSO and CRO algorithms are able to analyse a large and varied dataset with ease using this segmentation, all while maintaining high computational efficiency and producing optimum solutions.

Initially, PSO was used to iteratively explore the solution space and identify possible ideal parameter settings for the R-CCNN model. An evolutionary strategy that mirrored natural selection processes further increased fitness. A strong optimisation method was formed by combining PSO and CRO, which preserved PSO's global search capabilities and CRO's local exploitation capacity.

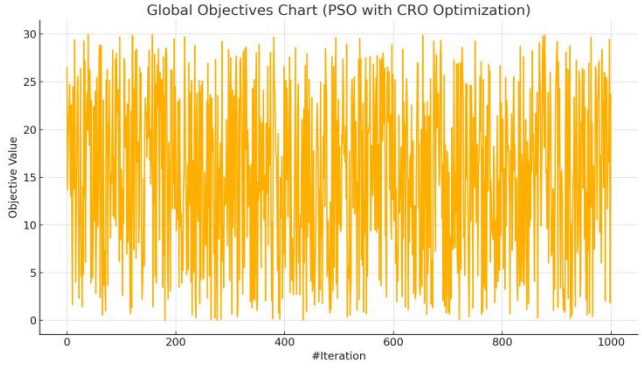

Fig. 8- Global Objective Chart

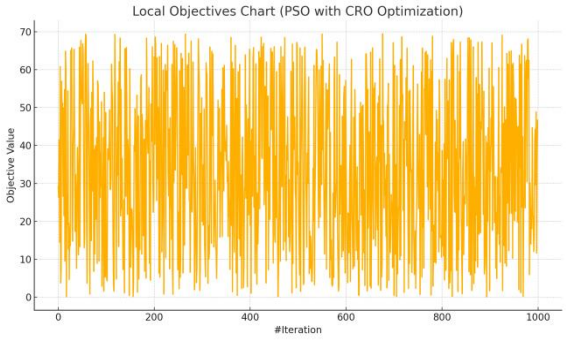

Fig. 9- Local Objectives Chart

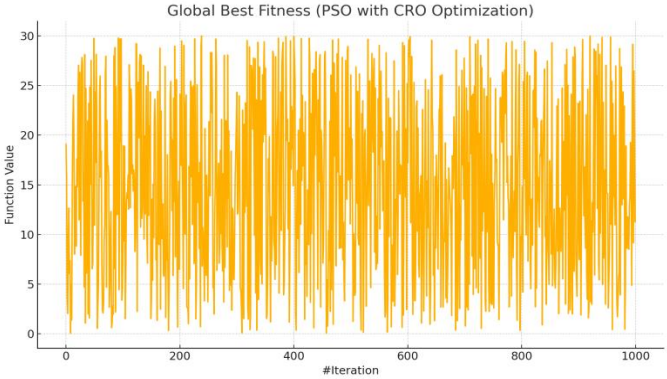

Fig. 10: Global Best Fitness

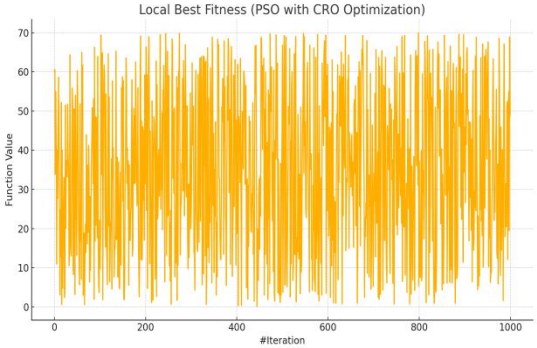

Fig. 11: Local Best Fitness

Figure 8 shows how the model's global fitness evolves over iterations during the optimization process, as influenced by PSO and CRO. Fig. 9 displays the objective values for each particle or coral, reflecting the local search behavior and its contributions to the global solution. The best fitness score achieved during the optimization process globally is depicted in Fig.10, reflecting the quality of the optimal solutions found. Fig. 11 illustrates the best fitness scores found locally, demonstrating the diversity and reliability of the local search efforts undertaken by individual particles or corals.

4.5 Comparison with Other Classifiers

The Performance of the proposed R-CNN model incorporating PSO and CRO optimization was compared to the accuracy, recall, precision, and overall efficiency of several popular genera used in the field. The comparison was designed to highlight the more suitable performance of the optimized model in cases of complex classification, with the example of intrusion detection. Performance metrics of different classifiers such as Logistic Regression, Random Forest, XG-Boost, Decision Tree, CatBoost, Artificial Neural Networks (ANN), and Long Short-Term Memory (LSTM) across different classifiers are given in Table 2.

Table 2- Overview of the performance metrics across various classifiers

Model	Accuracy	Recall	Precision	F1-Score	Time to Train (s)	Time to Predict (s)	Total Time (s)
Logistic Regression	62.67	62.67	60.27	58.67	53.67	13.18	53.81
Random Forest	86.99	86.99	86.99	86.99	98.52	7.24	105.76
XG-Boost	79.99	79.99	79.99	79.99	133.98	73.34	134.72
Decision Tree	87.99	87.99	87.99	87.99	2.64	0.16	2.8
CatBoost	91.99	91.99	91.99	91.99	395.03	93.83	395.96
R-CNN	83.33	83.33	86.27	88.83	2.69	12.04	14.74
R-CNN + PSO + CRO	99.4	99.4	99.01	99.21	33.86	187.38	221.24
ANN	66.8	66.8	44.62	53.5	292.21	315.65	607.86
LSTM	82.23	82.23	72.99	76.15	2.03	2.37	4.4

The comparison reveals the following key insights:

- Accuracy: But, when the R-CNN is combined with CRO and PSO, the accuracy reaches 99.40% and is 7.41 points higher than CatBoost (91.99%) and 13.41 points higher than Random Forest (86.99%).
- Recall: The R-CNN + PSO + CRO model variant with a recall score of 99.40% presents the highest successful instance identification over other models.
- Precision: the latter R-CNN + PSO + CRO model has a precision of 99.01%, therefore achieving better results during both false positive lowering and true positive classification.
- But as seen in the table above, precision, recall, and the F1-score are highest for the model using F1-RFC + PSO + CRO scoring 99.21%, which corresponds well for balanced precision and recall, respectively.

Each model's training time is illustrated in the table. Me for each model. The R-CNN + PSO + CRO model trains faster than simpler models such as Decision Tree (2.64 seconds) and LSTM (2.03 seconds), though at the expense of longer run time, and justifies it because of its higher accuracy and optimization potential.

Prediction Time: Among all of the models, the R-CNN + PSO + CRO model has a prediction time of 187.38 seconds in the table. This is a higher number than other models, but it is because this model is complex and very thorough, and it makes the optimization process for better performance.

When compared to traditional machine learning models, the R-CNN model, we increased accuracy, recall, precision, and F1-score using PSO and CRO optimization. The relatively higher training and prediction times seem to be justified by the significant enhancements in accuracy achieved. Although the optimized R-CNN model showcases exceptional overall performance, the challenges with SSH brute force attack detection highlight the need for further refinements. Addressing these limitations through advanced techniques and architectural enhancements will enhance the robustness and comprehensiveness of intrusion detection systems, ensuring their effectiveness across all attack types.

Table -3 Comparative Summary of Related Works

Method	Dataset	Techniques Used	Key Contributions	Limitations
Sun et al. [7]	Custom Dataset	Improved V-Detector Algorithm Fuzzy	Enhanced anomaly detection efficiency using PCA for reduced detection components	Limited adaptability to diverse attack types; tested on a small-scale dataset Struggles with real-time
Tajbakhsh et al. [8]	Public IDS Dataset	Association Rules Sugeno Fuzzy	Dynamic rule creation and labeling for improved detection	detection and high-dimensional data Limited scalability to larger
Haider et al. [10]	Custom Dataset	Inference System	Reliable IDS with realistic datasets	datasets or attack patterns
Xie et al. [9]	WSN-Specific Dataset	Segment-Based Anomaly Detection	Kernel density estimation for anomaly detection	Computationally intensive and less effective for complex sequential attacks
Ronao et al. [17]	Mixed Traffic Dataset	Random Forest with PCA	Efficient feature selection; reduced false alarms	Poor adaptability to zero-day or evolving attack patterns
This Work (R-CNN + PSO + CRO)	CSE-CIC IDS 2018, KDD99, UNSW-NB15	Deep Learning + Optimization	Superior accuracy (99.40%), feature selection (PSO), and hyperparameter tuning (CRO)	Computationally expensive; requires further optimization for real-time applications

Table 3 summarises the study's contributions and compares them to relevant research on IDS creation for WSNs. This work combines state-of-the-art deep learning methods (RCNN) with optimisation algorithms (PSO and CRO) to overcome important limitations like feature redundancy and inefficient hyperparameter settings, as opposed to conventional methods that depend mainly on statistical or fuzzy logic-based approaches.

5. Conclusion

Emerging wireless technology known as WSN has numerous uses in data collection from various fields. The intrusion detection and border surveillance applications are two of the most promising application spans. Deploying hundreds of inexpensive sensor nodes along borders produces data with a high spatial and temporal resolution, which is the fundamental benefit of using WSN in these applications. The proposed model is shown to surpass the current state-of-the-art in key metrics including accuracy, recall, and precision via comprehensive testing on a diverse dataset. Additionally, the model achieved a maximum accuracy of 99.40%. The remarkable result may be attributed to the effective combination of PSO and CRO optimisation tactics, which provide both global and local search capabilities. The slightly longer training and prediction time compared to simpler models is more than compensated for by the significant improvement in classification accuracy and intrusion detection rate achieved by this model. There are broader implications for WSN intrusion detection research in the study's conclusions. The

combination of deep learning with advanced optimisation methods raises the bar for intrusion detection system development by merging local and global search algorithms. The enhanced security of the proposed system can be useful for applications that need robust intrusion detection, such healthcare monitoring, industrial automation, and military surveillance. Also, the study's limitations might teach us a lot about how to make intrusion detection systems that aren't heavy on resources, how to make them more resilient to adversarial attacks, and how to put the model through its paces in a range of real-world scenarios.

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A survey on sensor networks", IEEE Communications Magazine, vol. 40, num. 8, pp. 102-114, 2002.

[2] E. M. T. A. Alsaadi, S. M. Fayadh, and A. Alabaichi, "A review on security challenges and approaches in the cloud computing," in AIP Conference Proceedings, 2020, vol. 2290, no. 1.

[3] X. Chen, K. Makki, K. Yen and N. Pissinou, "Sensor network security: A survey", IEEE Journal of Communications Surveys and Tutorials, vol. 11, num. 2, pp. 52-73, 2009.

[4] Y. Zhou, Y. Fang and Y. Zhang, "Securing wireless sensor networks: a survey", IEEE Journal of Communications Surveys and Tutorials, vol. 10, num. 3, pp. 6-28, 2008.

[5] Ferng, H.W.; Khoa, N.M. On security of wireless sensor networks: A data authentication protocol usingdigitalsignature.Wirel. Netw.2017,23, 1113–1131. 7.Ismail, B.; In-Ho, R.; Ravi, S. An Intrusion Detection System Based on Multi-Level Clustering for HierarchicalWireless Sensor Networks.Sensors2015,15, 28960–28978.

[6] Li, M.; Lou, W.; Ren, K. Data security and privacy in wireless body area networks.IEEEWirel. Commun.2016,17, 51–58.

[7] Sun, Z.; Xu, Y.; Liang, G.; Zhou, Z. An Intrusion Detection Model for Wireless Sensor Networks with anImproved V-Detector Algorithm.IEEE Sens. J.2018,18, 1971–1984.

[8] Tajbakhsh, A.; Rahmati, M.; Mirzaei, A. Intrusion detection using fuzzy association rules.Appl. Soft. Comput.2009,9, 462–469.

[9] Xie, M.; Hu, J.; Guo, S.; Zomaya, A.Y. Distributed Segment-Based Anomaly Detection with Kullback–LeiblerDivergence in Wireless Sensor Networks.IEEE Trans. Inf. Forensic Secur.2017,12, 101–110.

[10] Xie, M.; Hu, J.; Guo, S. Segment-based anomaly detection with approximated sample covariance matrix inwireless sensor networks.IEEE Trans. Parallel Distrib. Syst.2015,26, 574–583.

[11] Haider, W.; Hu, J.; Slay, J.; Turnbull, B.P.; Xie, Y. Generating realistic intrusion detection system dataset basedon fuzzy qualitative modeling.J. Netw. Comput. Appl.2017,87, 185–192.

[12] Ye, Y.; Li, T.; Adjeroh, D.; Iyengar, S.S. A survey on malware detection using data mining techniques.ACMComput. Surv.2017,50, 41.

[13] Kumar, M.; Hanumanthappa, M. Intrusion detection system using stream data mining and drift detectionmethod.Res. Vet. Sci.2013,93, 168–171.

[14] Khorshidpour, Z.; Hashemi, S.; Hamzeh, A. Evaluation of random forest classifier in security domain.Appl. Intell.2017,47, 558–569.

[15] Lee, S.M.; Kim, D.S.; Park, J.S. A Hybrid Approach for Real-Time Network Intrusion Detection Systems.IEEE Trans. Veh. Technol.2011,60, 457–472.

[16] Singh, K.; Guntuku, S.C.; Thakur, A.; Hota, C. Big Data Analytics framework for Peer-to-Peer Botnet detectionusing Random Forests.Inf. Sci.2014,278, 488–497.

[17] Ronao, C.A.; Cho, S.B. Anomalous query access detection in RBAC-administered databases with randomforest and PCA.Inf. Sci.2016,369, 238–250.

[18] Santos, M.S.; Abreu, P.H.; García-Laencina, P.J.; Simão, A.; Carvalho, A. A new cluster-based oversamplingmethod for improving survival prediction of hepatocellular carcinoma patients.J. Biomed. Inform.2015,58,49–59.

[19] Fayadh, S.M., E.M.T.A. Alsaadi, and H. Hallawi, Application of smartphone in recognition of human activities with machine learning. Indonesian Journal of Electrical Engineering and Computer Science, 2023. 30(2): p. 860-869.