An Experimental Investigation on the Detection and Classification of Active Network Attacks

Dr. Karthikeyan Kaliyaperumal[1], Prof. Raja Sarath Kumar Boddu[2], Prof. Sai Kiran Oruganti [3]

[1]Post Doctoral Fellow Researcher, Lincoln University College, Malaysia
[2]Professor and Head of IT, School of Engineering, Malla Reddy University, Hyderabad, India.
[3]Professor in Faculty of Engineering & Built Science, Lincoln University College, KL – Malaysia.

Corresponding author Email id: pdf. kirithicraj@lincoln.edu.my

---

Abstract: As Internet traffic has risen exponentially and network technology has advanced quickly, network attacks have become more common. A network attack occurs when a person gains illicit access to a network. This includes any attempt to damage the network, which might have catastrophic implications. Organizations rely heavily on tried and reliable network infrastructure security measures such as firewalls, encryption, and antivirus software. These tactics do, however, offer some protection against viruses and increasingly complex attacks. Two key ideas in artificial intelligence that became well-known at the turn of the century are machine learning (ML) and deep learning (DL). By teaching computers to think like humans, these strategies' emphasis on statistical methodologies and data may significantly increase computing capacity. Therefore, computer scientists began using intelligent approaches in network security to solve the shortcomings of non-intelligent systems. Many deep learning and machine learning techniques for attack detection and classification are thoroughly examined in this article.

Keywords: Deep Q-Network Attacks; IDS; NIDS; DDoS detection; Machine learning; Deep learning;

Introduction

A network attack aims to gain unauthorized access to a company's network in order to steal information or carry out harmful activities. There are two possible sources of the threat: an external attack or an inside assault. An ongoing goal of networking systems has been to improve data circulation and transmission. Their dedication to ongoing development has made it easier to launch a number of cutting-edge services. Cloud computing, which allows the on-demand delivery of various applications, services, and processing and storage resources to numerous users via the Internet, has been made possible by recent developments in network technology. This paradigm offers several advantages, including enhanced accessibility, efficiency, and dependability; less administrative load; cost-effective resource utilization; and other additional benefits [1] [2].

A multitude of individuals that engage with networks benefit from the Internet's continual enhancement and extensive use from many perspectives. The significance of network security is increasing as network

use becomes more prevalent [3]. Network security encompasses computers, networks, software, data, and related components, with the objective of safeguarding against unauthorized access and modification. Cyberattacks provide a substantial risk and inflict considerable damage on the growing array of internet-connected equipment used in the banking industry, e-commerce, and the military [4]. Ten percent of active assaults are denial-of-service (DDoS) attacks. When offenders implement actions to incapacitate a tool or network, it is termed a Denial-of-Service attack. The first user may lose access to the device or network as a consequence of this. An assailant may render a device or network unusable or even incinerate it by inundating it with traffic. Services such as online banking, email, and websites are affected. A denial-of-service attack (DDoS) may be initiated from any location. Disrupting an ongoing conversation or data transfer is referred to as a man-in-the-middle attack, a kind of eavesdropping. The offenders assume the identities of two legitimate entities after positioning themselves in the intermediary role of the transfer [5-7].

An intrusion detection system may discover malicious activity by collecting and analyzing data from the network, its connected computers, and the security log. An intrusion detection system may protect a system via real-time responses by assessing anomalous behaviors against the security policy and signs of an attack. In traditional setups, an intrusion detection system (IDS) enhances a firewall—primarily a passive defence mechanism—in a rational, proactive, and efficient manner. Intrusion Detection Systems (IDSs) can identify cyberattacks that may jeopardize information systems [8]. Intrusion Detection Systems (IDS) perform their functions by examining two categories of data: one related to the operating system (HIDS) and the other related to the network (NIDS). The use of NIDSs has efficiently utilized data mining techniques, which are also applied in several other domains. Detection of cyberattacks may be improved by the use of these technologies, which may reveal complex data connections. Network data, however, resists uncomplicated use by commercially accessible data mining techniques. The intricate procedure of intrusion detection starts with the aggregation of network data and proceeds with its preparation and preprocessing. Machine Learning (ML) and Deep Learning (DL), two key artificial intelligence techniques in network security, underpin several innovative detection procedures designed to swiftly and efficiently identify attacks.

Literature Review

Churcher et al. proposed several machine learning techniques for use in intrusion detection systems, including K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Decision Tree (DT), Naive Bayes (NB), Random Forest (RF), Artificial Neural Network (ANN), and Logistic Regression (LR). The Bot-IoT dataset is used to evaluate machine learning techniques for binary and multi-class classification. The reliability of RF in mitigating HTTP Distributed Denial-of-Service (DDoS) assaults is 99%. Conversely, KNN attains a 99% accuracy rate in multi-class classification [9].

Sarumi et al. compared intrusion detection systems, specifically examining Apriori, which use data mining association rule techniques, and Support Vector Machine, which utilizes machine learning methodologies. We assess the two systems based on the UNSW-NB15 and NSL-KDD datasets, which represent the University of New South Wales – Knowledge Discovery and Data Mining [10].

Sahoo et al. assert that the centralized control capability of SDN may be used to detect attack traffic. The STN sector used several machine learning methods to preempt suspect traffic. Employing support vector machine (SVM) based kernel principal component analysis (KPCA), the dimensionality of feature vectors was reduced, and genetic algorithms (GA) were used to optimize different SVM parameters. An enhanced kernel function (N-RBF) was used to mitigate noise resulting from feature discrepancies. The experimental results indicated that the model surpassed a singular SVM regarding generalization and classification accuracy [11].

Tuan et al. suggested a detection method for botnet DDoS attacks using machine learning techniques. The UNBS-NB 15 and KDD99 publicity datasets, renowned for detecting Botnet DDoS attacks, were used to evaluate the methodology. We analyzed the dataset's sensitivity, accuracy, specificity, area under the curve (AUC), false positive rate (FPR), and used several machine learning techniques including support vector machine (SVM), naïve bayes (NB), unsupervised learning (USML), and decision tree (DT) [12]. Kim et al. developed the convolutional neural network (CNN) model for denial-of-service attacks. They created double types of invasion photographs: RGB and greyscale. In constructing their CNN model, they considered the kernel size and the number of convolutional layers. The CNN model exhibited superior results on the KDD dataset, attaining multiclass and binary classification accuracies of 99% or above. The RNN achieved an accuracy of 99% in binary categorization [13].
The objective of the deep learning model created by Yang et al. was to detect malicious traffic inside an encrypted network. The proposed model originated from a Residual Neural Network (ResNet). The adversarial sample of encrypted traffic was produced with Deep Convolution Generative Adversarial Networks (DCGAN) and Deep Q-Network (DQN) reinforcement learning. The problem of uneven and inadequate samples was solved. The accuracy of the model was 99.94%. indicating exceptional performance [14].

To monitor and recognize insider authentications, Hu et al. used deep learning methods to develop a paradigm for user authentication based on mouse activity characteristics. The open-source Balabit Mouse Dynamics challenge for the dataset and the CNN methodology were used. CNN exhibited robust efficacy in user authentication using mouse features, achieving a FAR of 2.94% and a FRR of 2.28% [15].
A technique for the early identification of distributed denial-of-service (DDoS) assaults executed via a botnet integrates real network data with deep convolutional neural networks (CNNs), as suggested by Hussain et al. [16]. To execute a coordinated distributed denial of service (DDoS) attack inside a cell that might impair CPS operations, the puppet device oscillates between quiet calls, SMS spamming, or a combination of these tactics aimed at disrupting calls, Internet access, SMS, signaling, or a blend thereof.

Liang et al. primarily focused on an intrusion detection system using a hybrid placement strategy that integrates multi-agent systems, blockchain technology, and deep learning algorithms. The system was meticulously created, deployed, and tested. The primary components of the system are data collection, data management, analysis, and response. The system is evaluated using the NSL-KDD dataset, which represents the National Security Lab Knowledge Discovery and Data Mining. The results demonstrate that deep learning systems are proficient at detecting transport layer attacks. The findings indicate that deep learning techniques are effective in identifying breaches inside IoT networks [17].
Research Methods:

Data Collection & Preparation

- NSL-KDD dataset.
- Its publicly available dataset
- Data preprocessing involved data cleaning and
- Encoding, transformation, standardization

Model Building & Tools used

- The development tools include Anaconda and Python, Jupiter
- Model performance evaluation using measuring matrices:-
  - ✓ Accuracy
  - ✓ Precision
  - ✓ Recall
  - ✓ F1 Score

Tools will use: Python,Anacoda Jupyter notebook, Tensorflow & Keras

Suitable algorithms for the data are selected for the Deep Learning algorithms

- ✓ Capable of extracting features automatically
- ✓ DNN
- ✓ CNN
- ✓ LSTM
- ✓ BiLSTM and **GRU**

Support Vector Machines (SVM) are used for classification, regression, and outlier detection. It is a supervised learning model. The data is split linearly by the hyperplane. Support vector machines (SVMs) split data into classes by using a hyperplane that maximizes the margin between class occurrences, after the mapping of data into feature space. This classifier can do both binary and multi-class classification. Support Vector Machines excel in the presence of nonlinear data. Several research using SVM to detect intrusions. The SVM concludes data categorization by identifying the largest classification margin. The SVM classification technique use a hyperplane to distinguish between positive and negative class variables, using the principle of structural risk minimization [18].

An exceptionally effective data mining technique is the Random Forests algorithm, which integrates ensemble approaches for classification and regression. A variety of applications have extensively used the random forests approach. It has been used for calculating probability and formulating forecasts. As its name suggests, RF constructs a forest comprised of several decision trees. The creation involves the amalgamation of several decision trees, with their average used for predictive purposes. Generally, it surpasses a single sign about precision. The apparent strength of a forest is directly proportionate to its tree density. Both classification and regression problems are suitable for its use. In terms of accuracy, random forests are unparalleled. In comparison to an individual decision tree, random forests have less variation. This indicates that it has more versatility than singular decision trees and can effectively manage a broader range of data inputs. Moreover, the input data is unnecessary for their functionality. Data scaling is superfluous. No accuracy is lost despite the significant absence of data [19].

Derived from the Shallow Neural Network (SNN), Deep Neural Networks (DNN) have lately been a primary focus of research in the field of intrusion detection. In the realm of simulating intricate models, DNN surpasses its competitors significantly. Thirimanne et al. assert that the capacity of DNNs to accurately characterize data and provide viable solutions is extensive. A variety of hyperparameters including the quantity of hidden layers, the number of neurons, the activation function, the learning rate, the regularization coefficient, and the optimizer—are pertinent to deep neural networks and must be established in advance. These hyperparameters have an immediate influence on the performance of the final model. The input layer and all hidden variables were activated using the Rectified Linear Unit (ReLU) function layers in the DNN model. The ReLU activation function, characterized as a piecewise linear function, outputs the input value when the input is positive; if not, it yields zero. The nodes triggered by this function are referred to as rectified linear activation units. The Sigmoid function was used to activate the output layer since it can convert any real number into a range between zero and one. This approach converts the output of the DNN network into a probability score [20].

Convolutional neural networks (CNNs) aim to effectively learn the representation of incoming input characteristics. This architecture employs a series of learnable filters applied to an image alongside a group of convolutional feature extractors in the first layers. The filters operate somewhat to a sliding window, traversing all areas of the input image, with the stride indicating the overlapping distance, and the feature maps serving as the outputs. Various convolutional kernels are used to produce a distinct feature map in each layer of the CNN. A neuron in the feature map of the succeeding layer is linked to a region of adjacent neurons. The kernel is uniformly applied across all spatial locations of the input to produce the feature map. Classification is completed by one or more completely connected layers subsequent to the convolution and pooling layers [21].

Experimental Performance Measurement Methods

Loss function

It is an approach to assess how effectively our machine learning algorithm predicts the highlighted data set. In other words, loss functions are a measure of how well the model can predict the desired result. For this study, we used Binary Cross-Entropy and Categorical cross entropy loss functions.

Binary cross-entropy

Binary Cross-Entropy (BCE) is a loss function commonly used in binary classification tasks, where the goal is to classify data into one of two possible categories (e.g., attack or no attack in network attack detection model)(Terven et al., 2024).

The formula for Binary Cross-Entropy is:

$$\mathrm{BCE} = -\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}[\mathrm{yi}\,\log(\mathrm{y\hat{}i}) + (1-\mathrm{yi})\,\log(1-\mathrm{y\hat{}i})]$$

Where:

- yi is the actual label (0 or 1).
- y^i is the predicted probability for the positive class (i.e., 1).
- N is the number of samples.

The Binary Cross-Entropy loss penalizes wrong predictions, especially when the predicted probability is far from the actual label. A model that minimizes this loss will produce predictions that closely match the true labels. It's particularly well-suited for tasks where the output is a probability between 0 and 1 (such as when using a sigmoid activation function in the output layer of a neural network). But, for multi classification we used categorical cross entropy.

Categorical cross entropy

When doing multi-class classification problems, categorical cross entropy is employed. In such situations, the model has to decide which category they should fit into out of a wide range of available categories. Multi-Class only classifies one object from multiples objects in one sample. Based on the idea that only one class, out of all those that may exist, is accurate, categorical cross-entropy is calculated.

In training classification models, sparse categorical cross-entropy loss functions are frequently employed. It is used when the target labels are integers, representing the index of the correct class label. For example, if we have five (5) types, target label for a network attack class would be 0, 1, 2, 3, and 4 respectively. Sparse categorical cross-entropy compares the predicted probabilities for each class with the corresponding integer target label.

The main difference between the two loss functions is in how they handle target label representation. Categorical cross-entropy is used with one-hot encoded target labels, while sparse categorical cross-entropy is used with integer target labels. In this study we used Categorical cross-entropy and binary cross entropy for multi and binary classification respectively.

With multi-class classification tasks, where the objective is to classify data into multiple categories, the loss function known as categorical cross-entropy (CCE) is frequently employed. It measures the performance of a classification model whose output is a probability distribution across multiple classes(Terven et al., 2024).

The formula for Categorical Cross-Entropy is:

$$\mathrm{CCE} = -\sum_{\mathrm{i}=1}^{\mathrm{N}} * \sum_{\mathrm{c}=1}^{\mathrm{C}} \mathrm{yi,c}\ \log(\mathrm{y\hat{}i,c})$$

Where:

N is the number of samples.

C is the number of classes.

yi,c is a binary indicator (0 or 1) if class label c is the correct classification for sample i.

y^i,c is the predicted probability that sample i belongs to class c.

Model Performance Evaluation Metrics

Model's performance on test dataset indicates at what extent it will perform in the real world. Classification metrics are commonly used to evaluate models on the dataset. The performance of the models in this study is evaluated using five evaluation metrics based on the literature review: confusions matrix, accuracy, precisions, recalls, and f1-score. To further validate the performance of the deep learning models, it will be compared with other deep learning models in detecting and classifying active network attacks on the collected dataset.

Evaluation techniques

The next step after the model has been trained is to access the model effectiveness and success rate using various metrics. The fundamental performance indicators we utilized for the network attack detection and classification model are listed below. Recall, F1 score, Confusion matrix, precision, Accuracy.

Confusion matrix

A table called a confusion matrix is employed to assess effectiveness of a machine learning classification model is doing. It is a matrix, with the actual classes and expected classes represented in the rows and

columns, respectively. It assesses the effectiveness of a classification model and identifies the specific types of errors that the model is making.

A multiclass classification model's performance may also be assessed using a confusion matrix. In this case, the matrix will have more than two rows and columns, where each of them represents a unique class label. The matrix's diagonal elements indicate the examples that were successfully identified for each class, whereas the off-diagonal components reflect the incorrectly classified instances.

True Positive (TP): The number of accurately predicted positive outcomes for a particular class.

False Positive (FP): The quantity of incorrectly predicted positive for a particular class.

True Negative (TN): The quantity of correctly predicted negative for a particular class.

False Negative (FN): The amount of incorrectly predicted negative for a specific class.

From a confusion matrix, the following metrics may be determined:

Overall Accuracy: how many out of all the occurrences were correctly categorized.

Overall accuracy = $\frac{\text{Sum of TP for all classes}}{\text{Sum of all instances}}$

Precision: among all positive predictions, the percentage of TP for each class.

Precision = $\frac{\text{TP}}{\text{TP+FP}}$

Recall: among all actual positives for each class, the percentage of true positives.

Recall = $\frac{\text{TP}}{\text{TP+FN}}$

F1-score: It is challenging to compare two models if one has high recall and low accuracy or vice versa. Consequently, for this reason, F-score can be used. For each class, it is the recall and precision harmonic mean. As a weighted average of the precision and recall, it may be interpreted,

F1-score = 2* $\frac{(\text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})}$

Support: The quantity of each class's instances(Liu & Lang, 2019).

A Classification Report

A classification report summarizes the extent to which machine learning model performed for a particular classification problem. It offers a number of measures, including accuracy, F1-score, recall, and support for every class in the dataset. The classification report offers a more thorough analysis of a model's performance than a simple accuracy score, and it can help identify which classes the model is performing well on and which classes need improvement.

Implementation Details

In this study we were explained how the implementation is going on. The first steps are acquiring the datasets and the data is preprocessed then string indexer is applied to in both train dataset and test datasets which convert the string data to the numeric which are 0 for normal and 1 for attack.

Expected research outcomes:

- In this research study of different machine learning and deep learning techniques for classification and detection of various network attacks and leading to significant financial and information losses.
- We analysed the best Deep learning approaches are selected because; it can significantly overcome the limitations of traditional machine learning.
- Many deep learning and machine learning techniques for attack detection and classification are thoroughly examined in this research work.

Conclusion

When someone gains unauthorized access to a network, it's called a network attack. This includes any effort to take down or interfere with the network, which might have devastating results. Organizations rely heavily on well-established network infrastructure security measures including antivirus software, firewalls, and encryption. These tactics do, however, provide some protection against viruses and more complex assaults. Two key ideas in artificial intelligence that became well-known around the turn of the century are machine learning (ML) and deep learning (DL). By teaching computers to think like humans, these strategies' emphasis on statistical procedures and data may significantly increase computing capacity. Therefore, computer scientists began using intelligent techniques in network security to solve the shortcomings of non-intelligent systems. Many deep learning and machine learning techniques for attack detection and classification are thoroughly examined in this article.

References

[1] Takashi Adachi and Kazumasa Omote. An approach to predict driveby-download attacks by vulnerability evaluation and opcode. In 2015 10th Asia Joint Conference on Information Security, pages 145–151. IEEE, 2015.

[2] Marion Olubunmi Adebiyi, Micheal Olaolu Arowolo, Goodnews Ime Archibong, Moses Damilola Mshelia, and Ayodele Ariyo Adebiyi. An sql injection detection model using chi-square with classification techniques. In 2021 International Conference on Electrical, Computer and Energy Technologies (ICECET), pages 1–8. IEEE, 2021.

[3] Sanket Agarkar and Soma Ghosh. Malware detection & classification using machine learning. In 2020 IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC), pages 1–6. IEEE, 2020.

[4] Mayank Agarwal, Dileep Pasumarthi, Santosh Biswas, and Sukumar Nandi. Machine learning approach for detection of flooding dos attacks in 802.11 networks and attacker localization. International Journal of Machine Learning and Cybernetics, 7:1035–1051, 2016.

[5] G Ajeetha and G Madhu Priya. Machine learning based ddos attack detection. In 2019 Innovations in Power and Advanced Computing Technologies (i-PACT), volume 1, pages 1–5. IEEE, 2019.

[6] Md Abdullah Al Ahasan, Mengjun Hu, and Nashid Shahriar. Ofmcdm/irf: A phishing website detection model based on optimized fuzzy multi-criteria decision-making and improved random forest. In 2023 Silicon Valley Cybersecurity Conference (SVCC), pages 1–8. IEEE, 2023.

[7] Mousa Al-Akhras, Mohammed Alawairdhi, Ali Alkoudari, and Samer Atawneh. Using machine learning to build a classification model for iot networks to detect attack signatures. Int. J. Comput. Netw. Commun.(IJCNC), 12:99–116, 2020.

[8] Sura Abdulmunem Mohammed Al-Juboori, Firas Hazzaa, Zinah Sattar Jabbar, Sinan Salih, and Hassan Muwafaq Gheni. Man-in-the-middle and denial of service attacks detection using machine learning algorithms. Bulletin of Electrical Engineering and Informatics, 12(1):418– 426, 2023.

[9] Churcher, A, Ullah, R, Ahmad, J, Ur Rehman, S, Masood, F, Gogate, M, Alqahtani, F, Nour, B & Buchanan, WJ 2021,An experimental analysis of attack classification using machine learning in IoT networks‘, Sensors, vol. 21, no. 2, p. 446.

[10] Sarumi, OA, Adetunmbi, AO & Adetoye, FA 2020, Discovering computer networks intrusion using data analytics and machine intelligence‘, Scientific African, vol. 9.

[11] Sahoo, KS, Tripathy, BK, Naik, K, Ramasubbareddy, S, Balusamy, B, Khari, M & Burgos, D 2020, An evolutionary SVM model for DDOS attack detection in software defined networks', IEEE Access, vol. 8, pp. 132502-132513.
[12] Tuan, TA, Long, HV, Son, LH, Kumar, R, Priyadarshini, I & Son, NTK 2020, Performance evaluation of botnet DDoS attack detection using machine learning', Evolutionary Intelligence, vol. 13, no. 2, pp. 283-294.
[13] Kim, A, Park, M & Lee, DH 2020, AI-IDS: Application of deep learning to real-time web intrusion detection', In IEEE Access, vol. 8, pp. 70245-70261.
[14] Yang, J, Liang, G, Li, B, Wen, G & Gao, T 2021, A deep-learning and reinforcement-learning-based system for encrypted network malicious traffic detection', Electronics Letters, vol. 57, no. 9, pp. 363-365.
[15] Hu, T, Niu, W, Zhang, X, Liu, X, Lu, J & Liu, Y 2019, An insider threat detection approach based on mouse dynamics and deep learning', Security and Communication Networks, vol. 12, no. 4, pp. 1-12.
[16] Hussain, B, Du, Q, Sun, B & Han, Z, 2021, Deep learning-based DDoS-attack detection for cyber-physical system over 5G network', IEEE Transactions on Industrial Informatics, vol. 17, no. 2, pp. 860-870.
[17] Liang, C, Shanmugam, B, Azam, S 2020,Intrusion detection system for the internet of things based on blockchain and multi-agent systems', Electron, vol. 9, no. 7, pp. 1-27.
[18] Ye, J, Cheng, X, Zhu, J, Feng, L & Song, L 2018, A DDoS attack detection method based on SVM in software defined network', Security and Communication Networks, vol. 2018, pp. 1-8.
[19] Li, X, Chen, W, Zhang, Q & Wu, L 2020, Building auto-encoder intrusion detection system based on random forest feature selection', Computers & Security, vol. 95, no. 1, p. 101851.
[20] Thirimanne, SP, Jayawardana, L, Yasakethu, L, Liyanaarachchi, P & Hewage, C 2022, Deep neural network based real-time intrusion detection system', SN Computer Science, vol. 3, no. 2.
[21] Zhang, Q, Zhang, M, Chen, T, Sun, Z, Ma, Y & Yu, B 2019, Recent advances in convolutional neural network acceleration', Neurocomputing, vol. 323, pp. 37-51.