NextGen Scientific Publication
American International Journal of Computer Science and Technology
Volume 4 Issue 1, Pg. No. 65-76, AIJCST-V4I1P107, 2022
https://doi.org/10.63282/3117-5481/AIJCST-V4I1P107

Original Article

Machine Learning-Driven Threat Analysis for Enhancing Security in Data Center Networking

***Venkata Teja Nagumotu[1], Harsha Vardhan Reddy Kavuluri[2], Akhil Kumar Pathani[3], Ajay Dasari[4], Venkata Kishore Chilakapati[5], Srikanth Reddy Keshireddy[6]**

[1]Sr Network Engineer, Techno-bytes Inc.
[2]Lead database administrator, Wissen infotech.
[3]Network Engineer, Ebay.
[4]Senior Support Engineer, Microsoft.
[5]Technical Advisor, Microsoft.
[6]Senior Software Engineer, Keen Info Tek Inc.

Abstract:

Cyber threats constitute ill-purposed practices aimed at hacking into computer systems and stealing classified data with individuals and with more frequency, a broad spectrum of organizations as possible targets. This study offers a solution by exploring various machine learning algorithms for predictively analyzing and evaluating cyber threats. This research presents a cutting-edge approach to threat analysis in data center networking using the UNSW-NB15 data, with the goal of improving security. Data cleaning, normalization, feature selection, and balancing were carried out to ensure the model achieved its maximum potential. There is a single binary output feature and forty-three input features in the data. The suggested Deep Neural Network (DNN), along with several DL and ML models like BKP, C-Support Vector Machine (C-SVM), and K-Nearest Neighbors (KNN), were tested. The DNN one is head and shoulders above the competition when it comes to threat recognition and categorization, boasting an amazing ACC of 97.93, PRE of 97, REC of 97, and F1-score (F1) of 97.

Keywords:

Cybersecurity, Cyber Threat Detection, Intrusion, Artificial Intelligence (AI), Critical Infrastructure Security, Network Security, Threat Intelligence.

Article History:

Received: 04.12.2021

Revised: 23.12.2021

Accepted: 03.01.2022

Published: 20.01.2022

1. Introduction

The Internet is also altering how people learn and work as it is further becoming part of the social life of the people, but it is also placing us into greater levels of security threats [1]. Issues relating to the identification of various forms of network attacks, and those that have never been witnessed should be dealt with as soon as possible. The study of cyber security is vital due to the sensitivity of networks to the modern society. The common cyber defences are the firewalls, intrusion detection systems (IDSs) and antivirus programs [2]. Utilising these techniques can guarantee that networks are protected from both internal and external dangers. Important for cyber defence, intrusion detection systems keep tabs on a network's software and hardware.

Multi-tenant and multi-objective applications have rising communication and computing requirements that are hosted on data center networks (DCNs) [3]. A large number of applications rely on the tens of thousands of components that make up modern data centres, including servers, switches, routers, and related hardware [4]. The blistering development of the digital technologies has altered the nature of activities of individuals, organizations, and governments, creating unprecedented connectivity in terms of networks and the Internet. This interconnectivity has led to innovation, efficiency and collaboration worldwide but there have been numerous cybersecurity challenges as a result of the interconnectivity.

Ransomware, phishing, and DDoS attacks are only some examples of the ever-expanding range of cybersecurity attacks of DCNs [5]. Criminals exploit such threats to exploit the scale, complexity of DCNs and pose a desperate challenge to data confidentiality, integrity and accessibility. Intrusion detection systems (IDS) in use since the 1980s are mainly rule based. Although they used to work, such solutions cannot detect and eliminate new and advanced cyberattacks in real-time [6]. The drawbacks of the ineffective nature of the static, signature-based mechanisms have mandated the urgency of effective threat detection systems that are both adaptive, scalable, and intelligent in nature.

ML and DL are emerging trends in cybersecurity in DCNs because they have strong feature extraction, nonlinear mapping properties, and capabilities to handle very large, high-dimensional data sets [7]. The application of ML to threat analysis enables the identification of the anomalies, behavioural divergences, and malignant patterns that might be missed with the help of traditional systems. Proactive defines methods, including anomaly detection, behavioural analytics, and predictive modelling can increase proactive defines by enabling attack detection of the techniques before they can establish a meaningful impact [8]. Artificial intelligence (AI) has also been used to automate incident response, live intrusion prevention, and threat detection in cybersecurity, which has reduced the need for humans to physically analyse situations and take manual actions [9]. DL and ML application to improve DCN security by the means of smart threat analysis. It is focused on creating adaptive, scalable solutions capable of meeting performance, ACC and reliability needs of modern data centers [10]. In addition, the security intelligence modelling and automated decision-making are also investigated by the research as key aspects that can lead to the enhanced resilience of an organization. It is aimed at achieving a clear image of how AI and ML are transforming the cybersecurity of DCNs and propose new research paths that result in robust preventive and real-time defence measures.

1.1. Motivation and Contribution

Data safety, confidentiality, and service accessibility are also at serious risk when it comes to advanced and extensive cyber-attacks of data center networks. The legacy security analytics are not capable of handling the uneven and multi-dimensional network traffic data or real-time the emerging threats. Alongside this challenge comes the need to have automated intrusion detection systems (IDS) which able to think and identify hostile activity. The strategy is especially relevant to the existing data center protection as it raises detection ACC, reduces false alarms, and provides a better security level with the help of the UNSW-NB15 dataset, novel feature selection strategies, and neural networks such as DNN. The research contributes several important pieces of information to the topic of data center networking:

- Created a more refined IDS architecture of the data center networking based on the UNSW-NB15 data.
- Applied comprehensive data pre-processing, including cleaning, handling missing values, removing duplicates, and eliminating outliers.
- Implemented z-score normalization to standardize features and improve model training stability.
- Addressed class imbalance in the dataset through random oversampling to improve classification performance.
- In order to improve the detection of threats a dataset-specific DNN was developed and trained.
- Popular measures such as F1, REC, ACC, and PRE were used to compare model performance and contrast it.

1.2. Justification and Novelty

The research is motivated by the fact that current intrusion detection techniques and methods have their shortcomings. Data centre environments typically exhibit large-scale, imbalanced, and high-dimensional network traffic patterns, which these solutions struggle to manage. The technique enhances detection and performance through z-score normalisation, features selection, and strict pre-processing to select and utilize only the most significant information. The originality of the current study is the combination of highly advanced feature engineering and a DNN that is fine-tuned to the UNSW-NB15 dataset to offer a more accurate, scalable and reliable threat analysis solution compared to conventional IDS methods.

1.3. Structure of the Paper

The following is the structure of the paper In Section I, lay out the background and purpose of the study. In Section II, survey the existing literature. In Section III, detail the research methods. In Section IV, present and discuss the results. Finally, in Section V, offer some recommendations for future research.

2. Literature Review

Extensive research studies on threat analysis in data centre networking have been reviewed and critically analyzed to guide and strengthen the development of this work.

Khan et al. (2019) DL is a cutting-edge technique that can automatically take samples and extract their attributes. This paper introduces a CNN-based network IDS model that aims to solve the problem of insufficient ACC in traditional ML intrusion detection systems. In order to properly classify incursion samples, the model may automatically extract their useful properties. Experimental findings using KDD99 datasets demonstrate that the suggested model may considerably increase the ACC of IDS [11].

Vinayakumar et al. (2019) A comprehensive analysis of the DNN and other typical studies on the use of the ML classifier is provided on several popular malware datasets that are publicly available. According to the KDDCup 99 dataset, the following approaches to the selection of hyperparameters are applied in order to select the best network parameters and topologies of DNNs. Every DNN experiment involves the learning rate of between 0.01% and 0.55% and 1,000 epochs. In order to achieve the benchmark with the DNN model that has been effective on KDDCup 99%, a number of datasets are utilised. These datasets include NSL-KDD, UNSW-NB15, Kyoto, WSN-DS, and CICIDS 2017 [12].

Singh, Mehtre and Sangeetha (2019) offer a mixed-methods ensemble ML approach that uses convolutional neural networks (CNNs) and multi-state long short-term memory (MSLSTM) to identify additive behaviour patterns that deviate from the norm. Compared to single-state LSTM, multiple-state LSTM performs better. Areas under the curve (AUC) in both the train and test data sets of 0.9042 and 0.9047, respectively, indicate that the Multistate LSTM technique can detect insider threats when trained using publically available data [13].

Kumar, Viinikainen and Hamalainen (2018) This model was developed considering potential input from adversaries. The model makes use of forty time-based network flow characteristics that were taken from both benign and malicious apps' real-time data. By initiating the retraining phase, the suggested approach addresses the evolving behaviour of the attackers in addition to identifying known and undiscovered mobile threats. Mobile providers can protect their customers by implementing the suggested approach. We used a few supervised ML approaches to train the model, and its average ACC is 99.8% [14].

Vigneswaran et al. (2018) NIDS assault prediction has made use of DNNs. The network is trained and benchmarked using the KDDCup-'99' dataset, and it runs for 1000 epochs with a 0.1 learning rate. Train DNNs with 1–5 layers on the same dataset as many other conventional ML methods so can compare their performance. After comparing the outcomes, it was determined that among all the conventional ML algorithms, a 3-layer DNN performed the best [15].

Kumar, Viinikainen and Hamalainen (2017) Ensemble ML approaches handle problems associated with attackers' shifting behaviour and prevent the model from being over-fit. This research focusses on Android-based mobile malwares because of their market prevalence and uses ensemble methods to integrate the work of five trained ML models: RF, PART, JRIP, J.48, and Rider. The assessment findings showed that the suggested model had an ACC of 98.2% and was effective at recognising both known and unknown threats [16].

He, Zhang and Lee (2017) propose an ML-based method for tracing the origin of distributed denial of service attacks in the cloud. In order to prevent network packages from being sent to the external network, this solution utilises statistical data obtained from the hypervisor and virtual machines of the cloud server. It is necessary to thoroughly assess and compare nine ML approaches. We were able to identify almost 99.7 percent of the four distinct types of denial-of-service attacks after conducting the testing. method is easily extensible to larger denial-of-service attacks and has no impact on performance [17].

Kumar, Viinikainen and Hamalainen (2016) Data sets with labelled examples of network traffic variables produced by both good and bad apps were used to train supervised ML classifiers. This study zeroed in on Android-based malware due to Android's pervasiveness and the prevalence of its usage in mobile malware. The study found that the model may obtain an ACC of up to 99.4% for both known and unknown threats. Additionally, conventional intrusion detection systems can benefit from integrating this ML model to better identify sophisticated threats while simultaneously decreasing the number of FP [18].

The Table I presents a summary of recent studies on threat analysis in data center networking, highlighting innovative models, datasets used, key findings, limitations and future work.

Table 1. Comparative Analysis of Studies On Predictive Modeling of Threat Analysis In Data Centre Networking Using Machine Learning

Author(s)	Proposed Work	Dataset	Key Findings	Limitations & Future Work
Khan et al. (2019)	A CNN based intrusion detection system that can autonomously extract discriminative characteristics from network data.	KDD99	Achieved significantly higher intrusion detection accuracy compared to traditional ML models.	Limited to KDD99 dataset; future work may include testing on modern, diverse datasets and improving generalization for real-time traffic.
Vinayakumar et al. (2019)	Comprehensive evaluation of DNN architectures and ML classifiers using hyperparameter tuning (LR=0.01–0.5; 1000 epochs).	KDDCup99, NSL-KDD, UNSW-NB15, Kyoto, WSN-DS, CICIDS 2017	Identified optimal network architectures; DNN performed well across multiple benchmark datasets.	High computational cost for training 1000-epoch DNNs; future work should focus on reducing training time and optimizing architectures for deployment.
Singh, Mehtre & Sangeetha (2019)	Hybrid ensemble model using Multi-State LSTM + CNN for spatial–temporal anomaly detection.	Insider Threat Public Dataset	Achieved AUC ≈ 0.904 on both train and test sets; Multistate LSTM outperformed basic LSTM.	Focused only on insider threats; future research may consider broader threat categories and real-time implementation.
Kumar, Viinikainen & Hamalainen (2018)	Adversarial-aware ML model using 40 time-based flow features to detect known & unknown mobile threats.	Real-time network flow data (malicious + benign apps)	Achieved up to 99.8% accuracy; model adapts to attacker behavior changes through retraining triggers.	Limited to mobile traffic; retraining phase may increase overhead; future work should optimize retraining frequency and explore cross-platform applicability.
Vigneswaran et al. (2018)	DNN model (learning rate 0.1, 1000 epochs) for intrusion detection with varying layer depths (1–5 layers).	KDDCup99	DNN with 3 layers outperformed traditional ML algorithms and other DNN variants.	Overfitting risk with deeper layers; KDD dataset is outdated; future work should apply model to modern datasets like CICIDS2017.
Kumar, Viinikainen & Hamalainen (2017)	Ensemble ML model combining RF, PART, JRIP, J48, and Ridor for Android malware detection.	Android malware datasets	Achieved 98.2% accuracy; robust against evolving attacker behavior.	Focus restricted to Android malware; future work should evaluate model on cross-platform threats and newer malware variants.
He, Zhang & Lee (2017)	Source-side DOS detection system using cloud server hypervisor + VM statistics analyzed via ML algorithms.	Cloud environment datasets (four DOS attack types)	Over 99.7% detection accuracy; approach prevents outgoing attack traffic without degrading performance.	Limited to DOS attacks; future work should explore extension to DDoS and application-layer attacks.

3. Research Methodology

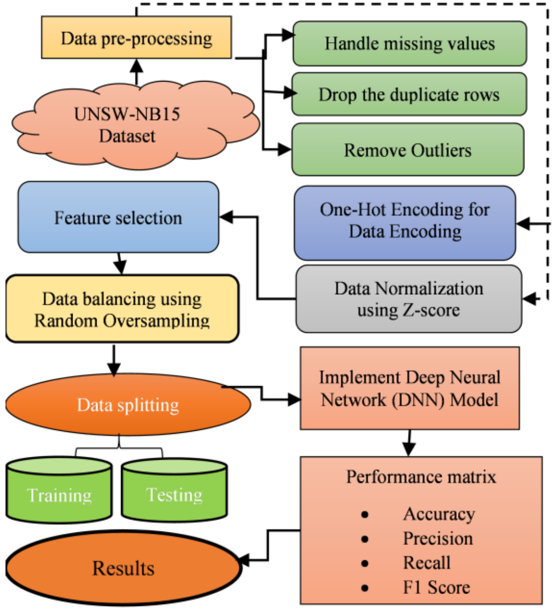

Figure 1. Proposed Flowchart for Threat Analysis in Data Center Networking

The methodology for this work involves enhancing an IDS using the UNSW-NB15 dataset. It performs pre-processing, handles missing values, eliminates duplicates, eliminates outliers, and transforms categorical variables using label encoding and one-hot encoding. Data is normalised using Z-score normalisation to ensure that feature scales and feature choices are consistent. An approach to addressing class imbalance is to randomly oversample the dataset and then divide it into testing and training sets. Finally, a DNN model is trained using the balanced and processed data to improve the effectiveness of data centre networking threat detection. In Figure 1, the entire process workflow is depicted.

This document provides an in-depth review of the proposed flowchart for threat analysis in data centre networking, with the objective of enhancing security.

3.1. Data collection

The UNSW-NB15 dataset is a standard for evaluating detection systems for network intrusions. It includes both innocuous country traffic and contemporary attack instances, such as fuzzes, backdoors, exploits, and DoS. Using the Argus and Bro tools, 49 distinct characteristics that cover the realistic behaviour of networks have been retrieved from the dataset. The UNSW-NB15 dataset is often used in cybersecurity to build and assess ML and DL models. Data visualizations such as bar plots and heatmaps were used to examine attack distribution, feature correlations, etc., are given below:

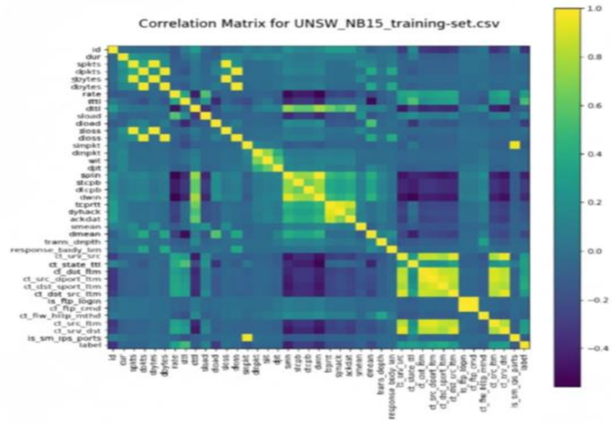

Figure 2. Correlation Matrix

The correlation matrix for the UNSW-NB15 training dataset is shown in Figure. 2, which visually represents the pairwise correlation coefficients between all numerical features. A color gradient ranging from dark purple (strong negative correlation) to brilliant yellow (high positive correlation) gives an intuitive comprehension of the linear correlations between the variables. Features along the diagonal exhibit a perfect correlation of 1.0 with themselves, while off-diagonal values reveal varying degrees of correlation, indicating potential multicollinearity among certain features. This analysis is important in feature selection, dimensionality reduction and enhancing interpretability of the model in future tasks in ML.

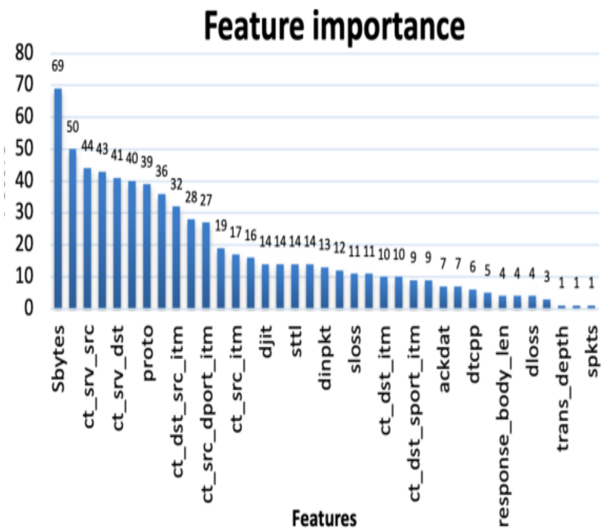

Figure 3. Feature Importance Score

Figure 3 shows the ratings of features of the UNSW-NB15 dataset. These ratings show the relative significance of each of the features to the execution of the prediction model. The feature Sbytes has the most significant importance score of 69, followed by ct_srv_src (50), ct_srv_dst (44) and proto (43), and they are the most influential features in classification. On the contrary, all the other features trans depth, dloss, spkts have an inconsequential effect with a score of less than 5. This ranking allows intelligent decision making when it comes to feature selection as it favors attributes that increase model ACC and it may also decrease computational complexity by removing less significant features.

3.2. Data pre-processing

The data preparation was associated with downloading the UNSW-NB15 dataset, merging, and cleaning it and selecting the relevant features. The pre-processing steps undertaken were addressing missing values, dropping of duplicate rows, and removal of outliers. After this, normalization and data transformation were carried out. The fine- Questionnaires on pre-processing are elaborated below:

- Handle missing value: In a data set with missing values, these may be dealt with by deleting rows or columns, by filling in with statistical estimates of the missing value, or by more sophisticated methods such as ML algorithms [19]. The amount of missing data, the kind of data, and the analysis's goal all influence the best course of action.
- Drop the duplicate rows: Removal of duplicates in a row is an ordinary and considerable task in information pre-processing, particularly when an individual is handling information that contains duplicates. The task of this procedure is to ensure that the data is of the desired quality and there is no bias or inefficiencies in further analysis or training of models.
- Remove Outliers: Pre-cleansing or removal of outliers may dramatically influence the quality and reliability of ML models and data analysis by eliminating data points that have little or nothing in common with the rest of the data.

3.3. One-Hot EncodingfFor Label Encoding

Label Encoding is a processing method that is applied in ML transforming data that is of the type kind into a numerical representation of the said type of data type. A distinct number is assigned to each category of a categorical feature in this process. An often-performed step before ML, one-hot encoding transforms categorical data into a numerical representation.

3.4. Data Normalization with z-score

Data is transformed or adjusted through normalization to get a more consistent distribution. Normalisation via rescaling, min-max, or z-score is a common way to standardize data. A mean of 0 and a standard deviation of 1 were produced by this study's usage of Z-score normalisation. The values cantered around the average value are transformed using the unit standard deviation in this scaling procedure. Equation (1) provides the definition of the z-score normalization.

$$E' = \frac{E-\bar{M}}{\sigma_M} \qquad (1)$$

Where,
$\bar{M}$ is the mean, σ_M is the standard deviation, and E' and E are new and old for every data entry.

3.5. Feature Selection

Feature selection involves picking characteristics that are most similar to and have an effect on the target variable in the dataset [20]. Some of the many feature selection approaches available include information gain characteristics, principal component analysis, and correlation attributes. The system does, however, make use of a second tree classifier. The extra tree classifier is one feature selection approach that might help and find the characteristics that are really important. To get the data ready for training a DNN model, the feature selection procedure is employed.

3.6. Data Balancing using Random Oversampling

The term "data balancing" describes methods for modifying the distribution of classes in a dataset, especially when working with unbalanced datasets in ML. In ML classification issues, where one class (the minority class) contains substantially less instances than another (the majority class), random oversampling is a data balancing approach used to resolve class imbalance in datasets.

3.7. Data Splitting

The model's efficacy was evaluated by partitioning the dataset into training and testing subsets. To begin with, 70% of the data was utilised for parameter estimation and model creation; the remaining 30% was reserved for testing and performance evaluation.

3.8. Proposed Deep Neural Network (DNN) Model

DL methods include learning information hierarchies in patterns that mimic how neurons function in the human brain. The threshold logic unit (TLU) is the simplest artificial neurone. The TLU determines the weighted sum of the inputs $(z = w_1x_1 + w_2x_2 + \cdots + w_nx_n = X^TW)$ by applying weights to each input. After applying a step function to that sum, the outcome is is $h_wX = step(z) where\ z = X^TW$. Dense or fully linked networks are those in which each neurone is attached to every other neurone in the layer above it [9]. A neural network's input layer consists of all the input neurones in addition to a bias neurone. Using Equation. (2), the results of a dense layer of synthetic neurones are computed for several cases at once.

$$h_{w,b}(X) = \emptyset(XW + b) \qquad (2)$$

W is a weight matrix that includes all of the connection weights minus those in the bias neuron; b is a bias vector that includes the weights of the connections between artificial neurons and bias; X is a matrix of input features; and $\emptyset$ is an activation function.

The six hidden dense layers in the created model all used the to provide non-linearity, use ReLU activation (where ReLU(x) = max (0, x). As opposed to Sigmoid, SoftMax was the output layer's activation function because are working with six classes, not just two. The SoftMax function, as defined by Equation (3), normalizes a proportionate distribution of K probability from a vector (z) of K real values.

$$S(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_i}} \qquad (3)$$

The loss function used in the process of minimization was the Sparse Categorical Cross-Entropy (LSCCE). It saves time and computing memory by representing classes using integers rather than vectors.

3.9. Evaluation Metrics

Through the utilisation of ACC computations and data tables, the model's performance, F1, PRE, and REC were assessed [21]. False positives (FPs) are the opposite of true positives (TPs), which are categorised as assaults based on real attack data. When valid attack data is mistakenly labelled as normal, get false negatives (FN), and when valid normal data is wrongly labelled as normal, and get true negatives (TN).

- Accuracy: Accuracy is the rate at which data are accurately categorized, i.e., when authentic attack data is categorized as assaults and regular data as normal. It is expressed as Equation (4)-

$$Accuracy = \frac{\text{TP+TN}}{\text{TP+Fp+TN+FN}} \quad (4)$$

- Precision: The percentage of favourably anticipated cases that are really positive is called PRE. It specifically captures the percentage of malicious packets that are appropriately recognized. It is expressed in Equation (5)-

$$Precision = \frac{\text{TP}}{\text{TP+FP}} \quad (5)$$

- Recall: Recall calculates the proportion of accurate predictions among all positive occurrences. In mathematical form it is given as Equation (6)-

$$Recall = \frac{\text{TP}}{TP+FN} \quad (6)$$

- F1 score: Integrating ACC and memory into a harmonic mean, it aids in achieving a balance between REC and PRE. Any value between zero and one is acceptable. To put it mathematically, it is Equation (7).

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (7)$$

4. Results and Discussion

The standard Python 3.11.1 libraries, constructed using Jupiter Notebook 6.5.2, are utilized in the experiments conducted in this paper. A minimum of 16 GB of RAM and an Intel Core i5 CPU are required to run Windows 11 x64. The Scikit Learn ML Python framework is used for building, testing, and evaluating the ML models. In Table II evaluated the DNN model's capability for data center networking threat assessments using the UNSW-NB15 dataset. The model regularly achieves PRE, REC, and F1 at 97%, and it has a high ACC of 97.93%, according to the experimental results. These results show that the model successfully reduces the amount of false positives and negatives while also effectively differentiating between harmless and dangerous events. The powerful and stable measures of all measurement parameters indicate the strength and dependability of the DNN in identifying and categorizing network-based threats, and hence verify the applicability of the DNN to real-time intrusion detection and prevention in multifaceted data centre frames.

Table 2. Experiment Results of Proposed Models for Threat Analysis in Data Center Networking On Unsw-Nb15 Dataset

Performance Matrix	Deep Neural Network (DNN)
Accuracy	97.93
Precision	97
Recall	97
F1-score	97

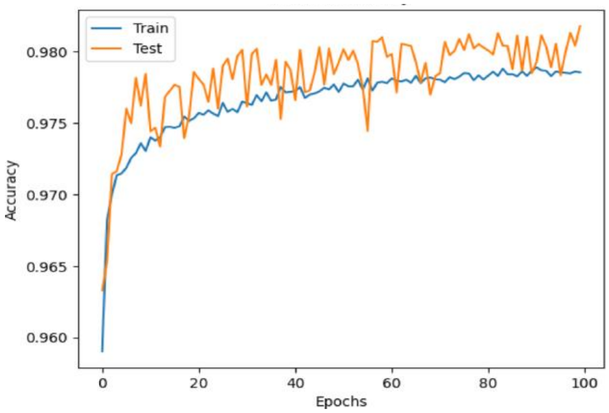

Figure 4. Accuracy Curves for the DNN Model

The ACC performance of the suggested model is displayed in Figure 4, in training and testing after 100 epochs. Both plots indicate the rapid growth of ACC at the beginning of the epochs, with the testing ACC being somewhat higher in most parts of the training process than the training ACC. The training ACC starts at around 96.0% and steadily rises, reaching approximately 97.8% by the final epoch. The testing ACC initially starts at the same level as, but soon overtakes the training ACC, oscillating slightly (meaning with values greater than 97.5) and as a result reaching the highest value of approximately 98.0%. This trend indicates that the model generalizes well, avoids overfitting, and consistently achieves high performance on unseen data.

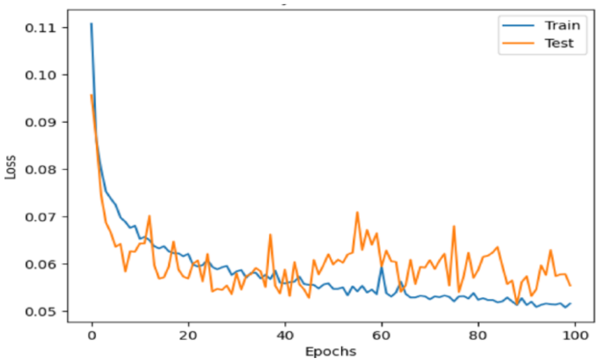

Figure 5. Loss Curves for the DNN Model

Figure 5 illustrates the model loss over 100 epochs for a binary classification model that compares testing and training results in NIDS. Both training and testing losses are initially large, with the testing loss being close to 0.095 and the training loss beginning at 0.11. Loss values drop sharply during the first 10 epochs, indicating rapid model learning, and then gradually stabilize. The training loss steadily declines to around 0.05, while the testing loss fluctuates slightly more due to validation variability but remains close to the training loss, indicating low overfitting and adequate generalization.

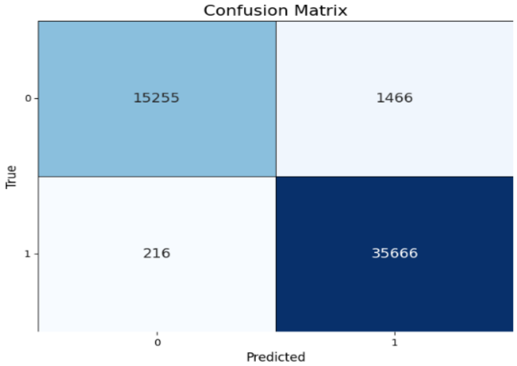

Figure 6. Confusion Matrix for DNN Model

Figure 6 the confusion matrix displays the distribution of predicted against real values, so visualizing a model's classification performance. 35,666 TP (bottom-right) and 15,255 TN (top-left) were accurately detected by the model, indicating strong ACC in detecting both classes. It made 1,466 false positive errors (top-right), where normal instances were misclassified as attacks, and 216 false negative errors (bottom-left), where attacks were missed. Misclassifications as a percentage of true predictions show how well the model can differentiate between the two groups.

4.1. Comparative Analysis

In order to assess the efficacy of the suggested DNN model, Table III presents a comparative ACC study with other models already in use. The classical ML algorithms such as C-SVM have a low ACC of 79.9 percent, which means that they are weak at dealing with complex and high dimensional threat patterns. The models of BKP and KNN are proven to be more efficient, with the results of 94.16% and 92.17 correspondingly, as they are more competent to reveal the non-linear relationships in network behavior. Nonetheless, DNN has an impressive ACC of 97.93, the approach surpasses all the other methods and indicates its better capability in learning complex threat patterns and improving protection in the contemporary data center setting.

Table 3. Accuracy Comparison of Different Predictive Models of Threat Analysis in Data Center Networking for Enhancing Security

Models	Accuracy
C—SVM[22]	79.9
BKP[23]	94.16
KNN[24]	92.17
DNN	97.93

DNN model suggested to be used in the threat analysis of data center networking has shown a great potential in the process, with an ACC of 97.93% along with other high levels of PRE, REC, and F1 of 97%. Its DL structure allows it to learn complicated patterns within the UNSW-NB15 data set automatically and decreases the dependence on the manual feature engineering and improves the flexibility to various kinds of threats. By virtue of its superior PRE and overall performance in the various parameters, the DNN is especially useful in protecting the network against threats in real-time and enhancing security and resilience of data centers-oriented networks so that few unjustified or incorrect alarms transpire.

5. Conclusion and Future Study

Threat Analysis in Data Center Networking Enhancing Security presents a new DNN framework that applies the UNSW-NB15 dataset to deliver optimal IDS result. In Improving Security in Data Center Networks by Threat Analysis the authors present a system capable of an ACC of 97.93 and a PRE, REC, and F1 of 97%. This is compared to those of other algorithms such as C-SVM, BKP and KNN. Threat Analysis to Enhance Security of Data Center Networking is a system which displayed a sufficient generalization, less over-fitting and reasonable performance following benign and malicious traffic.

Further research will involve a number of directions to enhance the proposed system even further. With regard to Threat Analysis to improve security of data center networking, advanced DL-based models can be presented to characterize the spatial and temporal dependencies of the traffic data with more strength such as CNN models, GNN models, and Transformer models. Related to Threat Analysis for Enhancing Security in Data Center Networking, hybrid detection frameworks combining signature-based and it is possible to enhance the detection of zero-day attacks by developing anomaly-based approaches. Related to Threat Analysis for Enhancing Security in Data Center Networking, federated learning and edge AI can be employed to enable distributed, privacy-preserving IDS deployment across multiple data center infrastructures.

References

[1] Y. Xin *et al.*, "Machine Learning and Deep Learning Methods for Cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018, doi: 10.1109/ACCESS.2018.2836950.

[2] H. Liu and B. Lang, "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey," *Appl. Sci.*, vol. 9, no. 20, p. 4396, Oct. 2019, doi: 10.3390/app9204396.

[3] A. Kushwaha, P. Pathak, and S. Gupta, "Review of Optimize Load Balancing Algorithms in Cloud.," *Int. J. Distrib. Cloud Comput.*, vol. 4, no. 2, p. 1, 2016.

[4] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, 2019, doi: 10.1186/s42400-019-0038-7.

[5] R. Boutaba *et al.*, "A comprehensive survey on machine learning for networking: evolution, applications and research opportunities," *J. Internet Serv. Appl.*, vol. 9, no. 1, p. 16, Dec. 2018, doi: 10.1186/s13174-018-0087-2.

[6] J. Gao and R. Jamidar, "Machine Learning Applications for Data Center Optimization," *Google White Pap.*, pp. 1–13, 2014.

[7] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya, and R. Ranjan, "A Hybrid Deep Learning-Based Model for Anomaly Detection in Cloud Datacenter Networks," *IEEE Trans. Netw. Serv. Manag.*, vol. 16, no. 3, pp. 924–935, 2019, doi: 10.1109/TNSM.2019.2927886.

[8] M. Barreno, B. Nelson, A. D. Joseph, and J. D. Tygar, "The security of machine learning," *Mach. Learn.*, vol. 81, no. 2, pp. 121–148, 2010, doi: 10.1007/s10994-010-5188-5.

[9] N. Sultana, N. Chilamkurti, W. Peng, and R. Alhadad, "Survey on SDN based network intrusion detection system using machine learning approaches," *Peer-to-Peer Netw. Appl.*, 2019, doi: 10.1007/s12083-017-0630-0.

[10] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. M. Leung, "A survey on security threats and defensive techniques of machine learning: A data driven view," *IEEE Access*, vol. 6, 2018, doi: 10.1109/ACCESS.2018.2805680.

[11] R. U. Khan, X. Zhang, M. Alazab, and R. Kumar, "An Improved Convolutional Neural Network Model for Intrusion Detection in Networks," in *2019 Cybersecurity and Cyberforensics Conference (CCC)*, IEEE, May 2019, pp. 74–77. doi: 10.1109/CCC.2019.000-6.

[12] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep Learning Approach for Intelligent

Intrusion Detection System," *IEEE Access*, vol. 7, pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

[13] M. Singh, B. M. Mehtre, and S. Sangeetha, "User Behavior Profiling using Ensemble Approach for Insider Threat Detection," in *ISBA 2019 - 5th IEEE International Conference on Identity, Security and Behavior Analysis*, 2019. doi: 10.1109/ISBA.2019.8778466.

[14] S. Kumar, A. Viinikainen, and T. Hamalainen, "A Network-Based Framework for Mobile Threat Detection," in *2018 1st International Conference on Data Intelligence and Security (ICDIS)*, IEEE, Apr. 2018, pp. 227–233. doi: 10.1109/ICDIS.2018.00044.

[15] R. K. Vigneswaran, R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Evaluating Shallow and Deep Neural Networks for Network Intrusion Detection Systems in Cyber Security," in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, Jul. 2018, pp. 1–6. doi: 10.1109/ICCCNT.2018.8494096.

[16] S. Kumar, A. Viinikainen, and T. Hamalainen, "Evaluation of ensemble machine learning methods in mobile threat detection," in *2017 12th International Conference for Internet Technology and Secured Transactions, ICITST 2017*, 2017. doi: 10.23919/ICITST.2017.8356396.

[17] Z. He, T. Zhang, and R. B. Lee, "Machine Learning Based DDoS Attack Detection from Source Side in Cloud," in *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*, IEEE, Jun. 2017, pp. 114–120. doi: 10.1109/CSCloud.2017.58.

[18] S. Kumar, A. Viinikainen, and T. Hamalainen, "Machine learning classification model for Network based Intrusion Detection System," in *2016 11th International Conference for Internet Technology and Secured Transactions (ICITST)*, 2016, pp. 242–249. doi: 10.1109/ICITST.2016.7856705.

[19] S. Garg, "Predictive Analytics and Auto Remediation using Artificial Inteligence and Machine learning in Cloud Computing Operations," *Int. J. Innov. Res. Eng. Multidiscip. Phys. Sci.*, vol. 7, no. 2, 2019, doi: 10.5281/zenodo.15362327.

[20] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, "On the effectiveness of machine and deep learning for cyber security," in *International Conference on Cyber Conflict, CYCON*, 2018. doi: 10.23919/CYCON.2018.8405026.

[21] J. Kim, N. Shin, S. Y. Jo, and S. H. Kim, "Method of intrusion detection using deep neural network," in *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, IEEE, Feb. 2017, pp. 313–316. doi: 10.1109/BIGCOMP.2017.7881684.

[22] M. Yan and Z. Liu, "A new method of transductive SVM-based network intrusion detection," in *IFIP Advances in Information and Communication Technology*, 2011. doi: 10.1007/978-3-642-18333-1_12.

[23] S. Vishwakarma, V. Sharma, and A. Tiwari, "An Intrusion Detection System using KNN-ACO Algorithm," *Int. J. Comput. Appl.*, vol. 171, no. 10, pp. 18–23, Aug. 2017, doi: 10.5120/ijca2017914079.

[24] A. A. Kumar and K. Parasuraman, "An Hybrid Intrusion Detection Approach based on SVM Classification and k-NN," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol. © 2018 IJSRCSEIT |*, vol. 5, no. 3, pp. 2456–3307, 2018.

[25] Polu, A. R., Buddula, D. V. K. R., Narra, B., Gupta, A., Vattikonda, N., & Patchipulusu, H. (2021). Evolution of AI in Software Development and Cybersecurity: Unifying Automation, Innovation, and Protection in the Digital Age. *Available at SSRN 5266517*.

[26] Padur, S. K. R. (2020). From centralized control to democratized insights: Migrating enterprise reporting from IBM Cognos to Microsoft Power BI. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol*, *6*(1), 218-225.

[27] Bitkuri, V., Kendyala, R., Kurma, J., Mamidala, V., Enokkaren, S. J., & Attipalli, A. (2021). Systematic Review of Artificial Intelligence Techniques for Enhancing Financial Reporting and Regulatory Compliance. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(4), 73-80.

[28] Padur, S. K. R. (2019). Machine learning for predictive capacity planning: Evolution from analytical modeling to autonomous infrastructure. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *5*(5), 285-293.

[29] Attipalli, A., Enokkaren, S., BITKURI, V., Kendyala, R., KURMA, J., & Mamidala, J. V. (2021). Enhancing Cloud Infrastructure Security Through AI-Powered Big Data Anomaly Detection. *Available at SSRN 5741305*.

[30] Singh, A. A. S., Tamilmani, V., Maniar, V., Kothamaram, R. R., Rajendran, D., & Namburi, V. D. (2021). Predictive Modeling for Classification of SMS Spam Using NLP and ML Techniques. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *2*(4), 60-69.

[31] Padur, S. K. R. (2020). AI augmented disaster recovery simulations: From chaos engineering to autonomous resilience orchestration. *International Journal of Scientific Research in Science, Engineering and Technology*, *7*(6), 367-378.

[32] Reddy Padur, S. K. (2021). From Scripts to Platforms-as-Code: The Role of Terraform and Ansible in Declarative Infrastructure Rollouts. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 621-628.

[33] Kothamaram, R. R., Rajendran, D., Namburi, V. D., Singh, A. A. S., Tamilmani, V., & Maniar, V. (2021). A Survey of Adoption Challenges and Barriers in Implementing Digital Payroll Management Systems in Across Organizations. *International Journal of Emerging Research in Engineering and Technology*, *2*(2), 64-72.

[34] Padur, S. K. R. (2018). Autonomous cloud economics: AI driven right sizing and cost optimization in hybrid infrastructures. *International Journal of Scientific Research in Science and Technology*, *4*(5), 2090-2097.

[35] Rajendran, D., Namburi, V. D., Singh, A. A. S., Tamilmani, V., Maniar, V., & Kothamaram, R. R. (2021). Anomaly Identification in IoT-Networks Using Artificial Intelligence-Based Data-Driven Techniques in Cloud Environmen. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(2), 83-91.

[36] Padur, S. K. R. (2021). Bridging Human, System, and Cloud Integration through RESTful Automation and Governance. *the International Journal of Science, Engineering and Technology*, *9*(6).

[37] Attipalli, A., BITKURI, V., KURMA, J., Enokkaren, S., Kendyala, R., & Mamidala, J. V. (2021). A Survey of Artificial Intelligence Methods in Liquidity Risk Management: Challenges and Future Directions. *Available at SSRN 5741342*.

[38] Padur, S. K. R. (2021). From Control to Code: Governance Models for Multi-Cloud ERP Modernization. *International Journal of Scientific Research & Engineering Trends*, *7*(3).

[39] Routhu, K. K. (2021). Harnessing AI Dashboards in Oracle Cloud HCM: Advancing Predictive Workforce Intelligence and Managerial Agility. *International Journal of Scientific Research & Engineering Trends*, *7*(6).

[40] Padur, S. K. R. (2021). Deep learning and process mining for ERP anomaly detection: Toward predictive and self-monitoring enterprise platforms. *Available at SSRN 5605531*.