Yugoslav Journal of Operations Research
35 (2025), Number 3, 619-632
DOI: https://doi.org/10.2298/YJOR240315036K

Research Article

A MACHINE LEARNING BASED CIDS MODEL FOR INTRUSION DETECTION TO ENSURE SECURITY WITHIN CLOUD NETWORK

Santosh Kumar MAURYA
Department of Computer Science and Engineering, IIMT University, Meerut, India
santoshcse200@gmail.com, ORCID: 0000-0002-4780-073X

Suraj MALIK
Department of Computer Science and Engineering, IIMT University, Meerut, India
surajmalik@iimtindia.net, ORCID: 0000-0003-3374-384X

Neeraj KUMAR*
Department of Science Technology & Technical Education, Patna, Government Engineering College Kishanganj, Bihar Engineering University, Patna, Bihar, India
javaneeraj@gmail.com, ORCID: 0000-0001-8922-4059

Hare Ram SINGH
Department of Computer Science and Engineering, G.N.I.O.T,
Greater Noida, U. P. India
hrsingh2000.2000@gmail.com, ORCID: 0000-0002-0771-1636

Received: March 2024 / Accepted: July 2024

Abstract: Cloud computing has changed quickly in recent years, and security issues have become more prevalent. Through the Internet, cloud computing has the amazing ability to deliver powerful, versatile, adaptable, affordable, and manageable assets that are always on the go. The potential of hardware resources is maximized by cloud computing through efficient and shared use. Data and service availability are challenges raised by a number of cloud computing difficulties. To improve cloud security for consumers and service providers, a multitude of security services are required. In order to provide security within the cloud and prevent load imbalance scenarios, this study suggests using Cloud Intrusion Detection Systems (CIDS). Cloud security vulnerabilities include denial of service, scanning, malware code injection, viruses, worms, and password cracking. If these attacks are not discovered in time, they might harm the company's finances and

* Corresponding author

image. Our idea through this work aims to protect the cloud from these types of attacks and to accurately detect and anticipate them early on. Previous research projects have noted that when dimensionality reduction is used in conjunction with data mining (DM), machine learning (ML) techniques have been proven to perform better. The authors proposed a CIDS by choosing relevant characteristics using relevant feature reduction techniques, then feeding this subset of data through the ML tool in order to develop such a robust model to assure a cloud network. Python and the Scikit-Learn program are used to simulate the proposed model. Using the KDDcup99 dataset as a benchmark, the simulation experiment's results were assessed using a variety of performance assessment criteria, including precision, recall, F-Score, detection ratio, RoC curve, etc. Our recommended methodology produced simulation results that were more efficient and on par with a number of other approaches. It has been noted that the ML-based proposed model was found to be sufficiently capable of safeguarding cloud-based data by identifying potentially suspicious user behavior, protecting the cloud network from threats, and demonstrating superior performance in true prediction and early intrusion detection, which led to a reduction in computational costs.

Keywords: Cloud computing, intrusion detection, machine learning, cloud security, cloud migration, digital transformation.

MSC: 68T05, 93C40.

1. INTRODUCTION

The phrase “cloud” or, especially, “cloud computing” refers to accessing resources, software, and databases via the Internet and outside local hardware's limits. Because of this technology, businesses can grow their operations more simply by OUTSOURCING a portion of their infrastructure management to third-party hosting providers. This may be done for the bulk of their infrastructure administration. The largest difficulty associated with cloud services is protecting users' privacy and security. Detecting attacks in the cloud is the core focus of this effort, primarily motivated by security concerns. Cloud security refers to a collection of processes and technology designed to respond to threats to business security that originate from both the outside and the inside. As they integrate cloud-based tools and services into their infrastructure and carry out their digital transformation plans, businesses need to ensure cloud security. Over the past few years, the terms "cloud migration" and "digital transformation" have grown more and more common in business contexts. These statements may have different meanings for different organizations, but they are all related in that change is necessary. As businesses adopt these ideas and work to improve their operational strategy, new challenges arise in striking a balance between security and efficiency. The transition to cloud-based settings can have various repercussions if it is not done safely, even though technological advancements that are more recent help organizations expand their capabilities beyond the limitations of on-premise infrastructure.

One of the many cases of cloud security that has been reported is Code Spaces, a code hosting service that was subjected to a distributed denial of service assault in 2014, which resulted in the service being rendered inaccessible for an indefinite period [1]. 2017 saw the takedown of some well-known French news websites due to a distributed denial of service attack against Cedexis, a cloud-based enterprise. In 2013, Yahoo suffered a data breach that exposed about three billion customer accounts. In 2017, a start-up cloud-based business named One Login had a hack. [1] After gaining access to AWS keys, a hacker exploited them to access the AWS API and move data across intermediary sites.

GitHub.com was taken offline briefly in February 2018 due to a distributed denial of service attack. According to NVD, network vulnerabilities are expanding annually, significantly increasing the security risks associated with cloud computing. Cloud security offers a variety of benefits, including protection from assaults, data security, increased accessibility, enhanced dependability, improved scalability, and compliance with laws and regulations [2]. These benefits are provided to address the difficulties that have been discussed above. Cloud security has several primary objectives, including safeguarding enterprises from hackers and distributed denial of service (DDoS) attacks, including R2L, U2R, and probing attacks [3].

In addition to addressing issues regarding security, it provides live monitoring and assistance, which not only boosts availability but also protects sensitive data by using protections such as encryption to prevent information from falling into the wrong hands. A reasonable cloud security approach includes redundancy, resulting in a more trustworthy experience. Secure cloud architecture makes it possible to restore the most vital resources and data more quickly in the case of an attack. It might be difficult to ensure that a complex cloud infrastructure conforms to the regulatory criteria that are in place regarding the sector. Providing security and help is one of the ways in which cloud service providers contribute to the compliance process [2]. ML technologies guarantee network security and address some of these problems. Additionally, this technique provides considerable advantages regarding early and accurate prediction and adequate assault detection accuracy.

The probability of complex cyber-attacks, such as advanced persistent threats and zero-day vulnerabilities, grows in proportion to cloud environments' increasing complexity and interconnection. The cyber threat environment is dynamic; malicious individuals always develop new strategies to exploit vulnerabilities. Organizations need strong cloud security measures to comply with regulations emphasizing data protection and privacy. To guarantee the reliability and accessibility of cloud services, a preventative security strategy is urgently required, one that can detect and counteract attacks in real-time. The goal is to construct a prediction model that can identify malicious connections (also known as intrusions or attacks) in a network and trustworthy, secure connections. Using ML, a system can be made that can detect and classify intrusions in computer networks. The objective is to create and deploy a system that can identify and promptly notify network managers of any intrusion, whether it's a known or unknown attack.

A "Network Attack" is a term that describes an attempt to obtain unauthorized access to a company's network to steal data or engage in other criminal acts. An assault on a corporate network should have the primary goal of breaching the perimeter and gaining access to the systems within the network. After gaining access to the system, attackers often employ various attack techniques, including causing damage to an endpoint, spreading malware, or taking advantage of vulnerability in the network system. Many different types of assaults may be classified as either active or passive. It is termed an active attack when hackers attempt to change or interfere with the content of communications or information. Hackers are deemed to be performing active attacks. An intruder collects data packets during a passive attack as the packets move over the network. There is a correlation between these risks and the integrity of the system and its availability. These assaults have an impact on the entirety of the system, and as a consequence, the data is transformed. These network assaults have the potential to be disastrous not only for a single nation or organization but for the whole globe as a whole. As a result, it is essential to take control of them early, and ML can do so for specific types of assaults.

The main contribution of the paper is

- Creating resilient Cloud Intrusion Detection Systems (CIDS) to provide cloud security and prevent load imbalance scenarios.
- Guarding the cloud against various dangers, as well as identifying and accurately predicting them early on.
- Scikit-Learn, a Python program, is used to simulate the proposed model. The KDDcup99 dataset and a variety of performance assessment criteria, including precision, recall, F-Score, detection ratio, RoC curve, etc., were used to gauge the simulation experiment's success.

The document is structured in the manner that is described below. In part 2, we have conducted an exhaustive analysis of all the relevant material. The third part describes the proposed security solutions that use efficient cloud-based strategies. As part of the experimental analysis, the simulation findings are provided and extensively explained in Section 4, which also contains the simulation results. Within our investigation into cloud parallelization approaches, the use of ML algorithms for intrusion detection is brought to the forefront. The conclusion and several other suggestions for future research were discussed in Section 5.

2. ANALYSIS OF LITERATURE

Many research projects have been offered to solve the intrusion detection issue in cloud computing systems. H. Hamad and colleagues (2012) contributed to the cloud intrusion detection services research. Supplying clients with web interfaces that are straightforward to use enables customers to pick the protection settings that they like to use. Intrusion detection framework has to meet a few requirements to adhere to the conventional SaaS service models. These include giving users the option to join or leave the service, changing the prerequisites, pricing according to the volume and complexity of the subscription database, and streamlining the service's functionality. To safeguard one network and get the best detection ratio for fifty networks, they use a basic snort security system. The required signatures might range from 200 to 1000 respectively. A drop in the detection ratio was shown to occur in proportion to the number of rules that were implemented [4].

In 2021, Sachdeva, Shaweta, and Aleem Ali contributed their work utilizing cloud network environments using ML and digital forensics for threat classification. Their cooperation allowed this study to be conducted. There exist other categories of service attacks, such as the User Datagram Protocol Attack, the Transmission Control Protocol Sync Attack, and the Internet Control Message Protocol Attack. They created a deep learning-based digital forensics method called the fusion algorithm. This method is very useful as a detective for data categorization; A broad digital forensic system that runs on the cloud. Techniques for collecting and analyzing dead or live forensic evidence were demonstrated both inside and outside of the TCP Sync, UDP, and ICMP attacks. A digital forensic triage system was also developed with a high accuracy of 99.36% to inspect and evaluate in-cloud computing systems. The Multilayer Perceptron (MLP), Random Forest, and Naive Bayes models obtained perception accuracy scores of 98.63%, 98.02%, and 96.91%, respectively [5].

Later, researchers observed and suggested that ML techniques are becoming increasingly popular in security applications because they can process information quickly and make predictions in real-time. The term "cloud computing" refers to a paradigm that provides a wide range of services via the Internet at a cheap cost and with a great degree of flexibility. In light of this, another option that uses ML techniques is Data Augmentation for Intrusion Detection and Classification in Cloud Networks. This idea was presented by Zina Chkirbene et al. (2021), and they encountered significant challenges, such as a reduced sample size of the training dataset. The application of these tactics is the training data accessible for each new category of attacks that might be viable. The Generative Adversarial Network (GAN) architecture was utilized to propose a ML-based model for identifying intrusions. The model's primary objective is to ensure that minority classes can learn more effectively. The GAN algorithm creates highly informative "like real" instances tied to the original data. This allows for improved class classification with less training data. Overall classification performance and detection accuracy for both the UNSW and NSL-KDD datasets [6], particularly for the classes that are only observed seldom [7].

An additional researcher named Onyema, E. M. (2022) based their ensemble intrusion strategy on the architecture of "Cyborg Intelligence," which combines biological and machine learning. The goal of this method was to improve the security of networks enabled by the Internet of Things (IoT) that are utilized for network traffic in smart cities. The KDDcup99 dataset was employed to examine the efficacy of several implemented methods. Among these algorithms were Artificial Neural Network (ANN), C5.0, Random Forest, Bayesian network (BN), and CART. This investigation's goal was to find cyborg intelligence-based dangers and assaults in Internet of Things networks. They adopted the AdaBoost approach because it makes use of tiny decision trees. According to the results, the AdaBoost ensemble learning framework—which is based on the Cyborg Intelligence Intrusion Detection framework—is able to support various network properties and effectively identify a range of botnet attacks due to its quick recognition capability. Redistributing the training dataset produces results that focus more on training cases that previous models were unable to maximize predictions for [8].

In their study [9], Varadharajan and colleagues presented an intrusion detection system with several distinct components. These components include an analyzer, a shared packet buffer in a virtual environment, an operating system load balancing (OSLR), a packet differentiator, and an intrusion detection engine (IDE). The model is included in the host operating system or virtual machine manager. Packets are transmitted from virtual computers to the packet differentiator to be processed. OSLR stores the information that pertains to each entity, including the process, application, virtual machine, and operating system. Checking the legitimacy of the source address is the responsibility of this component. The integrated development environment (IDE) makes a signature-matching comparison on the packet and then uses the anomaly module to check for valid patterns. Using ML techniques on top of OSLR, the Anomaly module modifies the behaviour of virtual machines.

Modi et al. demonstrated an integrated anomaly detection strategy that utilized a signature matching technique within cloud computing. Snort uses a database containing known attack signatures to function as signature-IDS. While SNORT cannot locate invasions, the decision tree technique is utilized to classify them into different categories. If Snort identifies permissible conduct, the packets are sent to the anomaly detection module [10].

Tien et al. suggested deploying NIDS architecture in the VMM privilege domain in their study. Each virtual machine's detection rules are chosen based on the services and operating system that are currently installed on the VM. The regulations that remain in force are no longer in force. Nmap, Xprobe2 P0f, Source Fire, and RNA (Real-time Network awareness) are some of the tools that may be used to obtain information on the operating system and network services running in virtual machines. The operating system kernel map that is contained in the virtual machine may also be used to retrieve information about the system [11].

Lee et al. proposed a multi-level anomaly score-based intrusion detection method for cloud environments. This tactic was created by the researchers. When a user wishes to access a cloud system using the AAA (Authentication, Authorization, and Accounting) module, an appropriate intrusion detection system (IDS) is selected for them based on the anomaly score. One database contains all of the user profile logs. AAA retrieves information from the database to expedite the handling of user transactions [9].

Christopher initially presented the Divided Data Parallel (DDP) approach, a content-matching method. This technology's fundamental idea is the parallel processing of a packet payload by n virtual computers. [12] To spread out among n processors and lower the total latency, a packet payload is split up into *n* subpackets. This lowers the total latency by enabling the comparison of signatures in parallel.

Vokorokos and colleagues proposed the GNORT algorithm. It is a modified version of the SNORT algorithm that uses the parallel processing capabilities of GPU computers. A cluster of nodes known as GNORT sensors is used by the GNORT NIDS architecture to capture network traffic. Each node receives traffic from a gateway so that it can perform additional processing. Coordination of other nodes' involvement is one of the duties of the coordinator node, also known as the mentor. The Network Activity Capturing Layer, Intrusion Detection Layer, and Synchronization Layer are the three levels that comprise architecture [13].

Bayesian networks have been employed in hybrid system decision-making in recent years because they provide a more sophisticated method of handling this than an RBS. According to Kruegel et al. [14], most hybrid systems have significant false alarm rates because of their overly simplistic methodologies. They, therefore, developed a hybrid host-based anomaly detection system that uses four different detection techniques. A Bayesian network determines the final output classification for the DARPA99 data set[15].

Hybrid Particle Swarm Optimization (HPSO) was proposed by Payam Chiniforooshan and Dragan Marinkovic [18] for scheduling single machines with setup times that depend on sequence and learning effects. The suggested HPSO method employs Particle Swarm Optimization (PSO) to address scheduling issues by encoding solutions using a random key representation; this allows for the conversion of task sequences to continuous position values. The HPSO also incorporates the local search technique to make the most of the algorithm. We compare the results of the proposed HPSO to the best solution provided by LINGO to verify its performance for small and medium-sized issues. The author produces 120 cases and compares them to the results of the Random Key Genetic approach (RKGA) to see whether the suggested approach can handle large-sized issues. The results demonstrate that the suggested model and algorithm are successful.

Intrusion Detection Systems should use Adversarial Machine Learning (AML), according to Afnan Alotaibi and Murad A. Rassam [19]. By deceiving intrusion detection systems (IDS) into incorrectly classifying network packets, AML creates many cyber security risks across a wide range of industries that rely on ML-based classification systems. Therefore, this article provides an overview of malicious machine-learning techniques and countermeasures. It begins by outlining the several hostile attack types that might compromise the IDS, and then it lays out the defensive measures to mitigate or eradicate such threats. Lastly, the author outlines the areas where current material is lacking and provides potential avenues for further study.

The ML methods for intrusion detection systems were suggested by Pierpaolo Dini et al. [20]. The main goal was to provide a classification system for supervised ML algorithms and networked intrusion detection systems. The datasets must be carefully selected to ensure the model is suitable for IDS use. The author tested the chosen ML algorithms for the dataset using binary and multi-class classification to ensure they were consistent. Based on the results of the experiments conducted on three common datasets, it can be concluded that supervised ML algorithms display good and promising classification performance. The accuracy rates for binary classification were 100% and 99.4%, respectively.

The reliable hybrid ML model for network intrusion detection was presented by Alamin Talukder et al. [21]. The study suggests a novel hybrid model integrating deep learning with ML to improve detection rates and ensure reliability. The suggested approach combines SMOTE for data balance and XGBoost for feature selection to guarantee efficient pre-processing. The author compared our devised technique to other deep learning and ML methods to determine which algorithm would be more effective to include into the pipeline. In addition, the author used a set of performance analysis benchmarks to choose the best model for network infiltration.

To compare and contrast adversarial learning versus deep neural networks in CV and NIDS, He et al. [22] surveyed the current research on network defenses, adversarial assaults, and network-based intrusion detection systems (NIDS) from 2015 onwards. An in-depth analysis of DL-based NIDS, rival attacks and defenses, and current research trends are provided to the reader. First, the author talks about how taxonomy affects adversarial learning and then provides DL-based NIDS taxonomy. After that, the author will review the current adversarial attacks on DNNs, both white-box and black-box and how they might be used in the NIDS domain. Lastly, the author looks at the features and current defense mechanisms in place to counter hostile instances.

The author reviewed several previous research ideas to attain our research goal of developing more secure service assessment within cloud networks. This part discusses the thorough investigation in light of our research domain of interest. It is observed that the AI-based hybrid technique is more efficient in both feature selection and classification and detection accuracy. Therefore, this analytical work has recommended an intelligent feature selection technique to suggest an optimal subset of features from big data: With this optimal subset of features for the same dataset. Following a thorough review of relevant literature, our effort was inspired to fill some of the research gaps:

- To create a more precise and effective modelfor detecting intrusions within the cloud and guaranteeing the safety of services for users.

- The objective is to evaluate the effectiveness of the ML methods currently used for cloud network intrusion detection.
- More reliable in true prediction even for fewer numbers of samples for training.
- To prevent the dependence of dimension reduction on the classifier.
- Attain maximum classification accuracy.
- Minimizes the False Alarm rate.
- High detection rate for any classifier technique.
- Low computation cost.
- Well-perform for big data.
- Avoid Load Imbalance situations within the cloud.

3. PROPOSED METHODOLOGY

The elimination of the need for local information resources is often accomplished through using cloud computing, which is extensively used. We examine the issue of intrusion detection in cloud-based systems in this paper, as well as the potential for offering intrusion detection as a service to clients. The purpose of this article is to discuss the Cloud Intrusion Detection System (CIDS), a service-based architecture intrusion detection web service that is meant to be made available to cloud clients.

3.1. About the Dataset

Mostintrusion detection research has been undertaken utilizing the few public datasets already accessible, such as '10% KddCup99'. '10% KddCup99' is the only dataset having labels for training and test sets since our model is based on supervised learning approaches. The '10% KddCup99' dataset has interesting qualities despite its limitations, and it is anticipated to represent a typical difficulty for the intrusion detection problem.

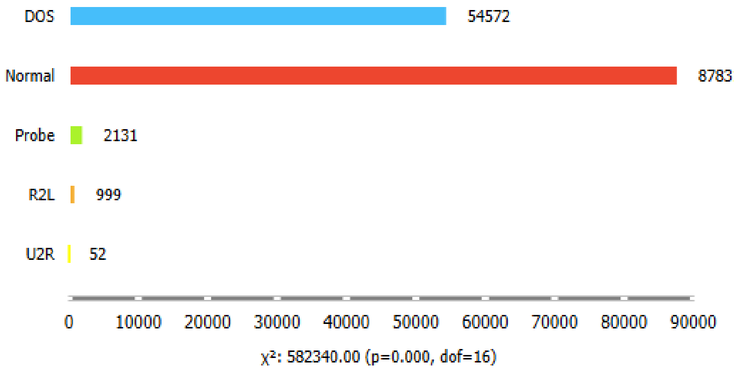

Figure 1: '10% KddCup99' UCI repository dataset

Since it is the most complete dataset and is still often used to compare, contrast, and assess the efficacy of intrusion detection, we made the decision to include it in our study. In contrast to the test data, which contains 38 different types of attacks, the training data only has 24 different attack categories. Each of these types of attacks is classified as belonging to one of the four fundamental attack classes, which are as follows: "Normal," "DoS," "Probe," "R2L," or "U2R" [16]. The total number of characteristics in each connection record is 42, including seven discrete attributes and 34 continuous features. The dataset known as '10% KddCup99' from the UCI repository was selected as the testbed. To illustrate the frequency of assaults that occurred within the dataset, a box plot is presented in Figure 1, which may be seen above.

3.2. Normalization and Dimensionality Reduction

This is another facet that must be examined because, in most cases, not all of the dimensions of a dataset are required or needed to arrive at the conclusion that must be reached. This may be accomplished in two different ways. The first approach, referred to as the feature selection technique, is deciding which feature is relevant to the demand, feeding it into the system in accordance with the specifications, and removing the other features. Our simulation, which used Principal Component Analysis (PCA), only took into account the five features of the '10% KddCup99' dataset. We would apply the Min-Max equation (Eq. (1)) to make sure the dataset was constant throughout [17]. We may remove redundancy in relational tables and prevent undesirable anomalies from happening in the database after the inputs for five characteristics—"duration", "protocol_type", "flag", "diff_srv_rate", and "dst_host_rerror_rate" [6]—of the thirty-two features have been finalized.

$$x_{normalized} = \frac{(x - x_{min})}{(x_{max} - x_{min})}. \tag{1}$$

3.3. Proposed Model CIDS Model

To eliminate the above research gap, we are motivated to propose a robust classification and best predictive model that is reliable on any dataset, best performing as far as accuracy is concerned, keeping low computational cost. Figure 2 is a proposed CIDS Model as depicted below, where services request data packets flowing over cloud network traffic were protected against threats like DDoS, U2R, Probe, R2L, etc.

Proposed Algorithm (Ensure secure service within Cloud Network)

```
Step 1: BEGIN
Step 2: Refinement of Dataset (Normalization).
Step 3: Operate ML classifiers.
Step 4: Train the Predictive Model.
Step 5: Perform testing on the Model.
Step 6: Performance Analysis using Evaluation Metric by Confusion
Matrix, Precision, Recall,
Step 7: Predictive Outcomes to Ensure Secure Services.
Step 8: STOP.
```

Early detection and prediction are essential for ensuring secure service on a cloud network. As shown below, the CIDS model (Figure 2) with the proposed algorithm, where datasets were analyzed and categorized using various ML techniques. The model is trained using the administrator's sample data trends. In the following sections, the proposed model has been realized and justified using simulation results.

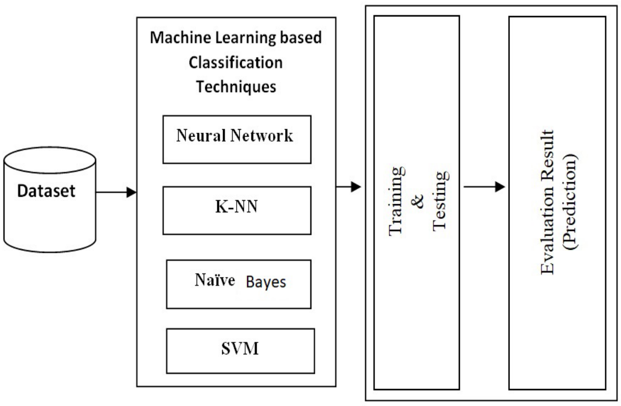
Dataset
Machine Learning based Classification Techniques
Neural Network
K-NN
Naïve Bayes
SVM
Training & Testing
Evaluation Result (Prediction)

Figure 2: Proposed CIDS Model

4. SIMULATION WORK

We use our proposed predictive model and follow the steps suggested per the proposed algorithm. Simulation work was carried out on Python, and testing training was done using the 10% KddCup99 dataset. The suggested intrusion detection paradigm uses the Python-based YAFS (Yet Another Fog Simulator) simulator. It incorporates sensors, actuators, and mobility into its models, allowing for execution-time modifications using JSON files. Modelling network failures, dynamic module allocation, service placement difficulties, and resilient network architecture are all made easy with this simulator. A cloud node with 10 GB of RAM and a 16 GHz CPU, 4 edge nodes with 2 GB of RAM and a 3 GHz CPU, and 50 to 400 Internet of Things devices with 500 MB of RAM and a 1 GHz CPU power the simulation hardware. Each instruction uses 100X108 packets, the connection bandwidth ranges from 3-10 Mbits, and each packet is 200 bytes in size. The performance of the suggested CIDS model has been analyzed based on metrics such as ROC curve, AUC, precision, recall, F1-score and confusion matrices.

4.1. Experimental Outcomes

K-fold cross-validation is helpful when we want to save as much data as possible for the training stage without losing valuable data to the validation dataset. For this particular instance, the dataset is folded into k folds, with one fold as the test dataset and the remaining as the training dataset. The entire process is then repeated n times per the prescribed parameters. A regression analysis will be performed, and the final result will be determined by taking the average of all obtained outcomes. In a classification case, the final result is the average of the outcomes, which may include things like accuracy, true positive rate (TPR), F1-score, and more parameters described in Table 1.

Sampling type: Stratified 5-fold Cross-validation
Target class: None, show average over classes

Table 1: Performance of Proposed CIDS Model on various ML classifiers

Model	AUC (%)	DR (%)	F1 (%)	Precision (%)	Recall (%)
K-NN	91.77	96.53	96.26	96.37	96.54
Neural Network	98.03	97.17	96.90	97.01	97.18
Naïve Bayes	96.37	96.11	95.63	95.32	96.11
SVM	66.98	14.15	18.22	31.31	14.16

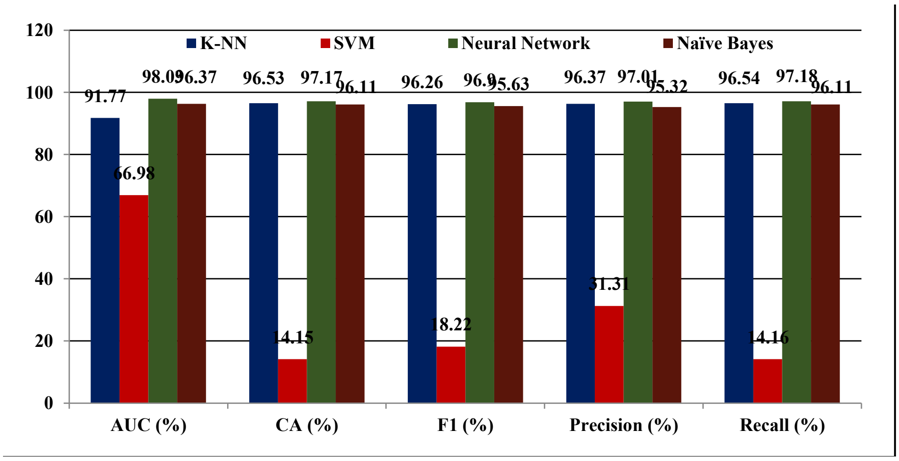

Figure 3: Performance of predictive proposed model using ML techniques

Confusion Matrix:

The confusion matrix, illustrated in Table 2, inferred that it performed well in predicting all types of attacks, including those with a small amount of train data samples.

Table 2: Confusion Matrix

		Prediction					
		DOS	**Normal**	**Probe**	**R2L**	**U2R**	**Σ**
Actual	**DOS**	97.3%	1.9%	1.3%	0.0%	0.0%	54572
	Normal	2.6%	96.2%	9.3%	20.7%	100.0%	87831
	Probe	0.1%	0.9%	88.9%	2.1%	0.0%	2131
	R2L	0.0%	0.9%	0.5%	77.1%	0.0%	999
	U2R	0.0%	0.1%	0.0%	0.0%	0.0%	52
	Σ	54367	89601	1426	188	3	145585

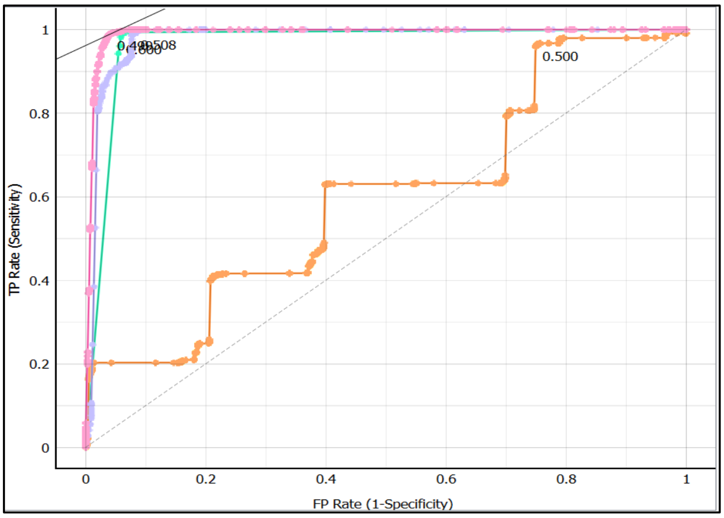

Figure 4: RoC Curve

The ROC curve is closer to the upper left corner of the graph, where sensitivity = 1 and false positive rate = 0 (specificity = 1). As a result, the optimal ROC curve, AUC, is 1.0. As per Table 2 and Figure 4, the outcome of all ML methodologies, AUC attains nearer to 1. Hence, it is clear that our proposal meets the objective through CIDS.

4.2. Observation of Proposed Work

- The majority of ML classifiers achieve excellent prediction accuracy.
- Neural Networks performed well among several ML classifying strategies, with an Area Under Curve (AUC) of 98.03 and Classification Accuracy (CA) of 97.17%, as shown in Table 1 and Figure 3.
- We obtained 66.98% AUC with 14.15% CA using the Support Vector Machine (SVM). Based on this finding, we may conclude that the SVM model could have previously provided results after feature reduction. Thus, after dimension reduction, the SVM classifier may classify up to a certain amount.
- According to Table 1, Table 2, and Figure 3, most of the classifiers performed well and achieved more than 95% CA.
- Table 2 and Figure 4 show that the AUC is getting closer to one, which reduces load imbalance situations in cloud load balancing.

5. CONCLUSION AND FUTURE DIRECTION

This paper aims to secure the cloud network when accessing resources from virtual machines linked with the cloud using Cloud Intrusion Detection Systems (CIDS). This study aims to safeguard the cloud against these attacks and provide early detection and accurate prediction. The experimental results of our proposed Cloud Intrusion Detection Systems (CIDS) prediction model were determined to be worthwhile in realizing the role of ML in providing security within a cloud computer network. The experimental result of the proposed simulation, as discussed within the observations of the proposed work, clearly shows that the proposed model is more efficient in all respects, with a high rate of true prediction, maximum CA, and most importantly, all outcomes of simulation AUC is closer to one, ensuring the cloud network is immune to attack. The attractive part of this idea is that, compared to competing methods, CIDS offered better attack detection rates above 96% in almost all approaches.

The study advocated this concept as a future scope to feed use in cloud load balancing. This predictive algorithm can forecast time series in a wide variety of domains. This paper contributes to protecting cloud infrastructures by laying the foundation for future studies that will enable intrusion detection systems to adapt to changing cyber threats. By using a data-fusion method, the future scope of attack prediction is not limited to the cloud; it may also include complex cloud environments with large data networks.

Funding: This research received no external funding.

REFERENCES

[1] D. Kadam, R. Patil and C. Modi, “An enhanced approach for intrusion detection in virtual network of cloud computing”, *In 2018 Tenth International Conference on Advanced Computing (ICoAC)*, pp. 80-87, 2018. doi: 10.1109/ICoAC44903.2018.8939107

[2] M. Carroll, A. Van Der Merwe and P. Kotze, “Secure cloud computing: Benefits, risks and controls”, *In 2011 information security for South Africa*, pp. 1-9, 2011.

[3] K. Aparna, G.R. Kumar, S. Ishar, N. Santhosh and D. Sreeja, “CaseStudy On DDoS Attacks And Attack TrendsIn Cloud Computing Environments”, *International Yournal of Techo-Engineering,* vol. 13, no. 3, pp. 378-383, 2021.

[4] A.K. Muhammed Kunju, S. Baskar, S. Zafar and B. AR, “A transformer based real-time photo captioning framework for visually impaired people with visual attention”, *Multimedia Tools and Applications*, pp. 1-20, 2024.

[5] S. Sachdeva and A. Ali, “Machine learning with digital forensics for attack classification in cloud network environment”, *International Journal of System Assurance Engineering and Management*, vol. 13, no. 1, pp.156-165, 2022. doi: 10.1007/s13198-021-01323-4

[6] C. Prajitha, K.P. Sridhar and S. Baskar, “Variance approximation and probabilistic decomposition noise removal framework for arrhythmia detection and classification on internet of medical things environment”, *Wireless Personal Communications*, vol. 125, no. 1, pp. 965-985, 2020.

[7] M. Arun, D. Barik and S.S. Chandran, “Exploration of material recovery framework from waste–A revolutionary move towards clean environment”, *Chemical Engineering Journal Advances,* vol. 18, p. 100589, 2024.

[8] E.M. Onyema, S. Dalal, C.A.T. Romero, B. Seth, P. Young and M.A. Wajid, “Design of intrusion detection system based on cyborg intelligence for security of cloud network traffic of smart cities”, *Journal of Cloud Computing*, vol. 11, no. 1, pp. 1-26, 2022. doi: 10.1186/s13677-022-00305-6

[9] K. Gangadharan, G.R.N. Kumari, D. Dhanasekaran and K. Malathi, “Detection and classification of various pest attacks and infection on plants using RBPN with GA based PSO algorithm”, Indonesian Journal of Electrical Engineering and Computer Science (IJEECS), vol. 20, no. 3, pp. 1278-1288, 2020. doi: 10.11591/ijeecs.v20.i3.pp1278-1288

[10] D. Singh, D. Patel, B. Borisaniya and C. Modi, “Collaborative ids framework for cloud”, *International Journal of Network Security*, pp. 1-63, 2013.

[11] B.B. Gupta and O.P. Badve, “Taxonomy of DoS and DDoS attacks and desirable defense mechanism in a cloud computing environment”, *Neural Computing and Applications*, vol. 28, pp. 3655-3682, 2017.

[12] C.V. Kopek, E.W. Fulp and P.S. Wheeler, "Distributed data parallel techniques for content-matching intrusion detection systems", *In MILCOM 2007-IEEE Military Communications Conference,* pp. 1-7, 2007. doi: 10.1109/ MILCOM.2007.4454922

[13] L. Vokorokos, M. Ennert, M. Čajkovský and A. Turinska, "A distributed network intrusion detection system architecture based on computer stations using GPGPU", *In 2013 IEEE 17th International Conference on Intelligent Engineering Systems (INES),* pp. 323-326, 2013. doi: 10.1109/INES.2013.6632834

[14] S. Stafford and J. Li, "Behavior-based worm detectors compared", In International Workshop on Recent Advances in Intrusion Detection, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 38-57, 2010.

[15] J. Kim and P.J. Bentley, "An evaluation of negative selection in an artificial immune system for network intrusion detection", *In Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation,* pp. 1330-1337, 2001.

[16] M.K. Siddiqui and S. Naahid, "Analysis of KDD CUP 99 dataset using clustering based data mining", *International Journal of Database Theory and Application*, vol. 6, no. 5, pp. 23-34, 2013. doi: 10.14257/ijdta.2013.6.5.03

[17] M. Mazziotta and A. Pareto, "Normalization methods for spatio-temporal analysis of environmental performance: Revisiting the Min–Max method*", Environmetrics*, vol. 33, no. 5, pp. 1-12, 2022.

[18] P. Chiniforooshan and D. Marinkovic, "A hybrid particle swarm optimization algorithm for single machine scheduling with sequence-dependent setup times and learning effects", *Computational algorithms and numerical dimensions*, vol. 2, no. 2, pp. 74-86, 2023.

[19] A. Alotaibi and M.A. Rassam, "Adversarial machine learning attacks against intrusion detection systems: A survey on strategies and defense", *Future Internet*, vol. 15, no. 2, pp. 1-62, 2023.

[20] P. Dini, A. Elhanashi, A. Begni, S. Saponara, Q. Zheng and K. Gasmi, "Overview on intrusion detection systems design exploiting machine learning for networking cybersecurity", *Applied Sciences*, vol. 13, no. 13, p.7507, 2023.

[21] M.A. Talukder, K.F. Hasan, M.M. Islam, M.A. Uddin, A. Akhter, M.A. Yousuf, F. Alharbi and M.A. Moni, "A dependable hybrid machine learning model for network intrusion detection", *Journal of Information Security and Applications*, vol. 72, pp. 1-12, 2023. doi: 10.1016/j.jisa.2022.103405

[22] K. He, D.D. Kim and M.R. Asghar, "Adversarial machine learning for network intrusion detection systems: A comprehensive survey", IEEE Communications Surveys & Tutorials, vol. 25, no. 1, pp. 538-566, 2023.