International Journal of Computer Trends and Technology
ISSN: 2231–2803 / https://doi.org/10.14445/22312803/IJCTT-V72I4P101

Volume 72 Issue 4, 1-14, April 2024

Original Article

Machine Learning-Enhanced IDS: RFE-LSTM-Based Model for Cloud Security

Karthik Rajashekaran[1], Rafaqat Kazmi[2], Rahul Jain[3]

[1]*Department of Computer Science, New York University.*
[2]*Dept. of Software Engineering, The Islamia University of Bahawalpur, Pakistan.*
[3]*Department of Computer Science and Engineering, Texas A&M University.*

[1]*Corresponding Author : karthik.rajashekaran.academic@gmail.com*

Received: 11 February 2024 Revised: 15 March 2024 Accepted: 03 April 2024 Published: 15 April 2024

Abstract *- Cloud computing impacts huge information science due to its services as infrastructure, software services, and platforms. The widespread use of cloud computing presents challenges such as security, privacy, and trust. The main threats are the susceptibility of the cloud infrastructure to various attacks, including address resolution protocol, IP spoofing, and denial of service. The classical intrusion detection techniques are insufficient to mitigate these new threats. The research proposes the REF-LSTM-IDS model, a novel technique that combines Recursive Feature Elimination (RFE) for optimised feature selection with a Long Short-Term Memory (LSTM) network used to identify dynamic threat pattern recognition. The proposed model's performance was assessed on the NSL-KDD and BoT-IoT datasets for feature selection reduction capability, and it was found that the model performs reasonably well on the evaluation criteria of accuracy and precision. The model performed 91.50% and 92.21% for accuracy measures for the datasets provided. The precision measure performance was 47.54%, and the recall measure was 82.31% for the datasets provided for the Matthews Correlation Coefficient (MCC) across the whole dataset. The proposed model improves cloud security and provides new insights for the integrated IDS model with machine learning capabilities. The integrated models reduce emerging security threats with embedded intelligence.*

Keywords *- Cloud computing, Data engineering, Intrusion detection, Machine learning.*

1. Introduction

Cloud computing represents a technological leap, delivering information technology services like facilities, platforms, and software over the internet [1]. It fulfills the longstanding vision of "Computing for Use" and is gradually adopted by organizations in the forms of private, public, or hybrid Clouds [2]. The primary goal is to empower users to access and pay for the services they need, ensuring on-demand solutions for software or infrastructure requirements [3]. While Cloud computing stands as a significant and positive shift in IT infrastructure, addressing its security shortcomings remains a priority. The Cloud's reliance on data centers to store personal and corporate information necessitates the identification and prevention of security issues and vulnerabilities. Operating through standard Internet protocols and employing virtualization techniques, Cloud infrastructure becomes susceptible to various attacks, including traditional ones like Address Resolution Protocol, IP spoofing, and Denial of Service (DoS) [4]. Additionally, emerging threats like zero-day attacks, characterized as unknown attacks, pose a considerable challenge in the cyber security domain [5]. Traditional detection and prevention techniques struggle to efficiently handle these attacks, especially when dealing with substantial data flows.

IDS is facing many challenges due to its distributed infrastructure, data volumes, black box types of encryption, and the evolving nature of intrusion attacks. IDS comprises many techniques to enhance the security and safety of the cloud infrastructure. The main techniques are signature-based detection, anomaly-based detection, policy-based detection, hybrid detection, machine learning and AI-enabled intrusion detection, network behaviour analysis (NBA), honeypots, cloud access security brokers, stateful protocol analysis, and deep packet inspections. In this context, trust management plays a pivotal role in enhancing security within cloud computing operations, acting as a reliable guardian to navigate complexities and foster trust in the cloud computing domain [6]. There is a shared recognition that continuous improvements are essential. In this landscape, intrusion detection emerges as a vital set of advanced technologies designed to identify malicious activities [7]. Within the domain of intrusion detection systems (IDSs), three primary types stand out: misuse detection, anomaly detection, and hybrid detection [8]. Anomaly IDS, a prevalent form of

intrusion detection, is crafted to spot attacks based on previously recorded normal behavior [9]. While widely used for its ability to detect new attacks by comparing current traffic with historical patterns, this approach also faces certain challenges. Notably, it tends to generate a significant number of false-positive alarms, suggesting that many regular packets may be mistakenly identified as potential threats. Ongoing advancements in anomaly IDS are crucial to striking a more refined balance between sensitivity and accuracy in detecting potential security breaches.

Machine learning has emerged as a valuable tool for both known and unknown threat detection. ML techniques can autonomously discern predictive patterns through data analysis, avoiding reliance on human-defined heuristics. By training on historical network metadata over time, ML approaches can gradually recognize anomalies without manual signature updates. This self-learning capacity enhances intrusion detection capabilities, especially around previously unknown "zero-day" threats [10]. By integrating techniques from computer science and statistics, machine learning provides three fundamental categories for modeling data [11].

Supervised learning algorithms rely on labeled datasets to inform their training procedures, capitalizing on metadata to estimate class boundaries during optimization—unsupervised techniques, conversely, cluster data into subgroups without guidance from pre-classified examples. A selection of methods have been developed adhering to these distinct paradigms, including k-Nearest Neighbors, Naive Bayes classification, decision tree induction, linear regression, and support vector machines for supervised tasks. K-means clustering exemplifies an unsupervised algorithm. Meanwhile, deep learning architectures, enabling multilayered computational models, have demonstrated significant potential through applications such as computer vision, natural language processing, and speech recognition. Specifically, deep neural networks can learn intricate statistical patterns from vast amounts of raw, unlabeled training examples to achieve human-level performance on complex prediction challenges that elude shallow approaches [12].

In this research study, we propose a REF-LSTM-IDS model that combines the features of machine learning, classical intrusion detection, and trust management on cloud services. This holistic combination provides a novelty in intrusion detection methods so far reported in the literature. The proposed model tries to enhance accuracy without compromising the efficiency of IDS by utilising recursive feature recognition (RFE) for optimal feature selection and long short-term memory (LSTM) networks for dynamic pattern recognition. The proposed model detects threats with improved accuracy and ensures robust defence against known and unknown vulnerabilities.

The research study proposed a model for intrusion detection using RFE-LSTM-IDS in a cloud computing environment with feature reduction and LSTM for preprocessing data. The rest of the research study was organized as follows: Section 2 covers the discussion about related work carried out in cloud computing; Section 3 represents the methodology of the experiment carried out for the assessment and evaluation of the proposed model; Section 3 presents the results of the proposed model; Section 4 outlines the discussion on the research outcomes; and Section 5 presents the conclusion of the research study.

However, this cloud computing type is not without its limitations, as concerns persist regarding maintenance and lower security levels [13]. To tackle these challenges, companies often opt for the private cloud model, located on-premises, offering enhanced protection [14]. Despite the numerous advantages of cloud computing (CC), security challenges impede its rapid adoption, encompassing regulatory hurdles, the risk of data loss, and privacy concerns [15]. These challenges weigh heavily on the minds of users and organizations. Various strategies, including the implementation of firewalls and anti-virus measures, have been devised and put into practice to safeguard applications, data, and cloud environments against potential threats [16].

2. Related Work

In the realm of cloud security, researchers have made strides in adapting traditional Intrusion Detection Systems (IDS) to operate effectively within cloud environments. One notable example is the VM-Integrated IDS [17], built on the Snort architecture to detect anomalies. A dual approach [18], combining Snort with machine learning classifiers to identify irregularities in network traffic between Virtual Machines (VMs), was proposed. A research [19] leveraged the "Bag of System Calls" technique, a classic but impactful means for identifying abnormal sequences in operating system calls generated by executing programs.[20] An immediate sequence of system call-based approaches, reminiscent of traditional methods, was introduced. [21] Several studies explored the application of artificial intelligence for cloud security. One investigation implemented Artificial Neural Networks for cloud attack detection. [22] applied a Fuzzy C-Mean Clustering based ANN for intrusion detection. In these novel techniques, the IDS operates in the typical fashion, generally stationed on end host cloud servers or tenant virtual machines. However, incorporating traditional IDS into cloud environments requires thoughtful deliberation of the specific components involved, their placement architecture, and the access authorizations assigned to each. Careful system integration is crucial to ensure effective and secure IDS operation within cloud platforms.

As threats to security continue to advance, conventional Host-based Intrusion Detection Systems can struggle to confront modern malware that is adept at bypassing signature matching and static analysis through obfuscation and encryption techniques. The evolving nature of these attacks poses challenges for traditional HIDS approaches [23]. Traditional intrusion detection approaches relying upon dynamic analysis techniques may be evaded through the inspection of memory contents on monitored virtual machines or host systems for signs of security monitoring processes, as the analytic agent is co-resident within the scrutinized environment. Furthermore, malware actors have demonstrated capabilities to probe virtualized platforms by scrutinizing registry keys and drivers unique to computer virtualization technologies. Periodic behavioral profiling of embedded security tools, as well as the checking of processor register value alterations, have additionally been observed as tactics leveraged by adversarial threats seeking to circumvent detection. The integrated deployment of intrusion monitoring agents with target systems thus enables sophisticated malware variants opportunities to profile defense solutions and enact obfuscation methods accordingly potentially. Therefore, research into mechanisms fostering more robust isolation between analytic and target environments may be warranted to curtail such technique neutralization approaches [24]. Certain advanced malware variants, like VM-rootkit attacks, have demonstrated capabilities to further evade detection by modifying the guest operating system kernel of monitored systems, thereby circumventing traditionally deployed intrusion detection solutions. Detecting such sophisticated threats at early stages is paramount to preempting subsequent risks, such as side-channel attacks that could compromise cloud security and privacy. Academic research into techniques capable of resiliently identifying these stealthy attacks prior to kernel infiltration merits exploration to bolster cloud resilience against agile cyber adversaries [25]. Conventional host-based intrusion detection approaches, however, contend with constraints in competently recognizing and remediating such sophisticated attacks operating within virtualized cloud platforms. As threats to security persistently progress in technical sophistication, continuous academic exploration and advancement in intrusion monitoring techniques are paramount to strengthening defensive measures in the ever-changing and complicated domain of cloud computing. The dynamic nature of cloud systems underscores the vital need for ongoing research into state-of-the-art intrusion detection strategies attuned to emerging vulnerabilities and evolving adversarial tactics in virtualized deployment models.

In the domain of cloud security, several approaches have been proposed that incorporate the capabilities of conventional Network Intrusion Detection Systems. By examining network traffic patterns for known signature matches or anomalous behavior, these solutions seek to recognize intrusions traversing cloud infrastructure networks. Specifically, they aim to bolster protections for inter-virtual machine communications and external connections to tenant virtual systems. Continuous advancement of such methods integrating established intrusion monitoring paradigms with modern cloud-native architectures remains an important area of ongoing research focused on enhancing incident detection capabilities across virtualized network perimeters in heterogeneous cloud environments [26]. These types of frameworks leverage signature-based techniques to identify network assaults aimed at tenant virtual machines. However, inherent issues exist with this methodology, as signatures necessitate consistent upkeeping due to their vulnerability against permutations in malicious schemes, permitting security bypass of the monitoring solution. Reliance on fixed definitions of unwanted behaviors leaves such approaches prone to apt evasion by determined or innovative threats. Continuous advancements such as through machine learning and behavior analytics, could help address this challenge by enabling detection models to dynamically learn and respond to new threats over time based on observed traffic rather than predefined rules alone. To strengthen intrusion detection in these frameworks [18], the proposal integrated the conventional NIDS tool Snort with an anomaly-based machine learning approach to analyze network traffic. In this two-tiered model, the ML module screened only benign traffic previously cleared by Snort. However, this method did not significantly reduce false positives for malicious traffic nor verify Snort alerts. While the additional analysis enhanced performance for legitimate traffic, it failed to curb false alarms from either component reliably within cloud environments wherein multiple VMs connect over virtual switches to form internal networks—traditional NIDS struggle to detect attacks between co-located VMs since their traffic bypasses physical networking. Likewise, host-based solutions confront limitations since they were designed principally for physical architectures rather than virtualized settings with diverse attack vectors. The continued evolution of detection paradigms attuned to cloud networking dynamics and multi-tier attack surfaces is needed to strengthen defensive capabilities in modern virtualized deployment models.

Detecting misuse in computer systems can be accomplished using supervised classification algorithms [27] like Back Propagation Artificial Neural Network (BP-ANN), Decision Tree (DT), and Multi-class Support Vector Machine (SVM), all well-established machine learning approaches. These methods provide a system that learns to distinguish different attack types by analyzing patterns in both normal and malicious behavior. The primary goal of utilizing machine learning for misuse detection is to generate a general model of

malicious behavior, with characteristic signs of attacks learned automatically from data rather than predefined manually. These algorithms can play an important role in identifying inherent relationships or anomalies within network traffic datasets, as demonstrated in past research[28]. Their strength lies in the ability to derive comprehension of likely threats based on analysis of patterns and attributes across large amounts of historical information rather than dependence on human-defined signatures. This capacity for autonomous representation learning may help address issues with rules-based approaches becoming outdated as adversaries evolve their tactics over time. The research [29] introduced an enhancement involving a feature selection technique combined with support vector machines for network intrusion classification to improve accuracy. Their proposed Modified Mutual Information-based Feature Selection approach (MMIFS) involved choosing attributes demonstrating maximum mutual information sharing with labels, shown to be an effective preprocessing step. By focusing model training on only the most pertinent traffic characteristics as determined by the MMIFS method, the overfitting risk was reduced, and intrusion detection performance was bolstered versus using all available features. This highlighted the potential for dimensionality reduction practices to optimize machine learning-based analysis of network security data. In [30] proposed using fuzzy association rules for intrusion detection. A fuzzy c-means clustering membership function transformed network features into fuzzy items. This allowed association rule learning to discover contextual relationships indicative of normal operation or anomalies by encoding features into categorical items with graduated membership across groups. Rather than crisp classification, this fuzzy approach could better represent mixed or gradual behaviors in real-world traffic.

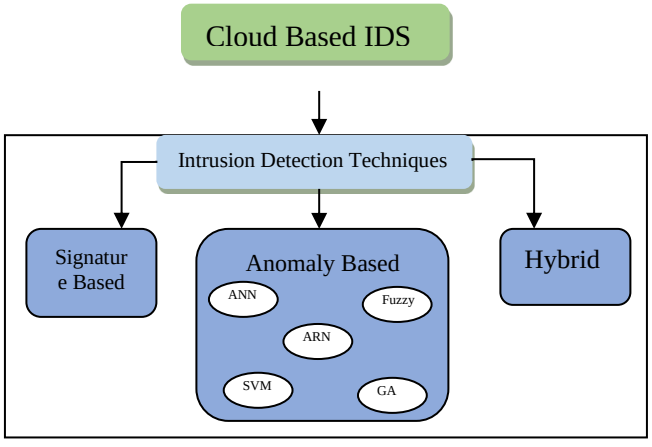

Fig. 1 Cloud-based IDS

Cloud Computing (CC) has grown significantly, now providing a wide range of public and private services through a cohesive Internet-based platform. Central to the CC environment are three core service models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), each serving as a basic element within different cloud architectures like community, private, hybrid, and public clouds tailored to diverse user needs [31]. IaaS forms the foundation of cloud computing, providing on-demand access to virtualized computing resources such as servers, storage, and networking capabilities over the internet. This enables a flexible and scalable environment for application development and management [15]. PaaS builds upon IaaS by including key tools and software frameworks. This allows developers to focus on developing applications without the burden of managing the underlying infrastructure [32]. SaaS, in contrast, stands out for offering complete cloud-hosted software solutions, eliminating the need for local installation for users. However, despite their advantages, each service model comes with its own set of challenges. IaaS may face constraints in virtualization that could reduce its long-term usefulness. PaaS grapples with issues like interoperability, sensitivity to the host environment, confidentiality, authorization, reliability and scalability. Meanwhile, SaaS is confronted with important security issues surrounding authorization, authentication, data integrity, reliability and network monitoring [32]. Addressing these security issues is crucial for cloud service providers, as it ensures the integrity and reliability of their offerings [33].

As threats in the cyber domain persist in their development, malicious actors aiming to compromise security protections within cloud environments routinely refine their technical arsenals and modes of operation. Due to the progressively mutable nature of these challenges, cloud service providers must correspondingly advance prevention, detection, and response capabilities to ensure defenses remain dynamically resilient against emerging risks. Vigilant and proactive management of security posture is thus necessitated to match the responsive agility exhibited by those preying on network vulnerabilities [34]. Conventional Intrusion Detection Systems may face challenges in accurately identifying changes in network traffic patterns. Consequently, academics emphasize the importance of adopting Machine Learning and Deep Learning techniques to bolster IDS performance. ML and DL have risen in prominence across diverse fields such as finance, government, scientific research, and cybersecurity, demonstrating their potential to strengthen anomaly detection and predictive analysis capabilities. These data-driven approaches may allow IDSes to continuously learn from network behaviors and gain specialized knowledge that could complement traditional signature-based methods[35]. In the field of cybersecurity, the efficiency of machine learning in data clustering and classification plays a vital role [36]. In the ever-changing landscape of cloud security, adopting advanced technologies such as ML and DL becomes crucial to fortify defenses against evolving threats and ensure the continued resilience of cloud environments.

Intrusion Detection Systems (IDSs) act as digital guards, spotting malicious activities and files. They split into two

kinds: misuse-based, which looks for known threats, and anomaly-based, which alerts on unusual behavior, potentially catching new threats [37]. Anomaly-based IDS acts like a keen observer, constantly measuring what is happening on the network against what it knows to be normal. This vigilance is fantastic for catching threats no one has seen before. However, it is a bit overeager at times, occasionally mistaking everyday activities for something sinister, which can lead to some unnecessary alarm bells. [9]. Conversely, misuse-based IDS uses a library of known attack patterns to identify threats, which helps in lowering false alerts. However, this method might overlook new attacks that do not match any of the existing signatures. [38]. The advancement of IDSs enriched with Machine Learning (ML) and Deep Learning (DL) technologies is gaining momentum across different industries, focusing on bolstering security in cloud landscapes and protecting against new and evolving threats.

Table 1. A comparative analysis of previous work

Ref No#	Title	Methodology	Accuracy
[39]	Leveraging Edge Machine Learning and OneM2M for Comprehensive IoT Network Security and Interoperability	DT, J48	92
[40]	Developing a Custom Deep Learning Model for Anomaly Detection and Enhanced Security in IoT Networks	CNN	--
[41]	A Software-Defined Networking Framework for Deep Learning-Based Intrusion Detection in IoT Environments	LSTM	99
[42]	An Ensemble Learning Approach to Intrusion Detection at the IoT Edge for Industrial Applications	Ensemble learning	93
[43]	Toward Intelligent Cloud Infrastructure Protection: Identifying Malicious Activity Through Artificial Intelligence	RF	97

Machine learning, deep learning, and ensemble methods have recently played a key role in enhancing intrusion detection system capabilities for identifying attacks [44]. Various studies have contributed to improving IDSs for cloud environments, as summarized in Table 1 [44]. In 2023, researchers proposed an intrusion detection system feature selection by applying K-nearest neighbors combined with PCA, UST, and GA (Mohy-eddine, Guezzaz et al., 2023).

Evaluated on the Bot-IoT dataset, their approach achieved a high detection accuracy of 99.99%. An earlier 2016 study put forth a collaborative hybrid approach for securing cloud computing. M. Douiba et al. developed an optimized intrusion detection solution integrating gradient boosting and decision tree techniques to strengthen the protection of Internet of Things systems.

The authors of [45] evaluated LSTM and RNN for multichannel intrusion detection systems and found them to be effective options, achieving estimated classification accuracies of 99.23% for LSTM and 98.94% for RNN. Research [44] tested various machine learning algorithms for intrusion detection, including Artificial Neural Networks, K-Nearest Neighbors, Decision Trees, Support Vector Machines, Naive Bayes, and Random Forests. The models were applied to verify data integrity, with Random Forest found to outperform techniques like Naive Bayes, Support Vector Machine, and K-Nearest Neighbors. Furthermore, this study employed three machine learning algorithms, K-Nearest Neighbors, Random Forest, and Naive Bayes, to detect DDoS attacks in cloud computing environments, achieving an impressive accuracy of 99.76%. In a study published in 2022, [35] proposed an intrusion detection system model utilizing ensemble learning techniques to safeguard edge computing environments within the Industrial Internet of Things.

In a separate study, authors developed an intrusion detection approach applying a genetic algorithm-based feature selection method alongside a random forest classifier [46]. The research [43] proposed an intrusion detection system for cloud security that combined graphical visualization techniques with a random forest classifier to help improve anomaly identification.

3. Materials and Methods

The RFE-LSTM-IDS model that our study proposed is described in detail in this section. To increase prediction accuracy and shorten processing times, we have outlined every model-building technique, including feature reduction. The two main steps in the suggested method for cloud security are preprocessing and intrusion detection, which use RFE and LSTM, respectively, as shown in Figure 1 and Algorithm 1.

3.1. Our Purpose IDS

In contrast to earlier studies using comparable datasets and methodologies, we have incorporated several optimisations into our research to improve the effectiveness of our suggested RFE-LSTM-IDS method:

3.1.1. Parallel Processing

To speed up computation, our implementation makes use of contemporary multi-core processors by leveraging parallel processing techniques. We greatly reduce processing time by dividing the workload across several cores, especially when dealing with large-scale datasets, and thereby improve overall efficiency.

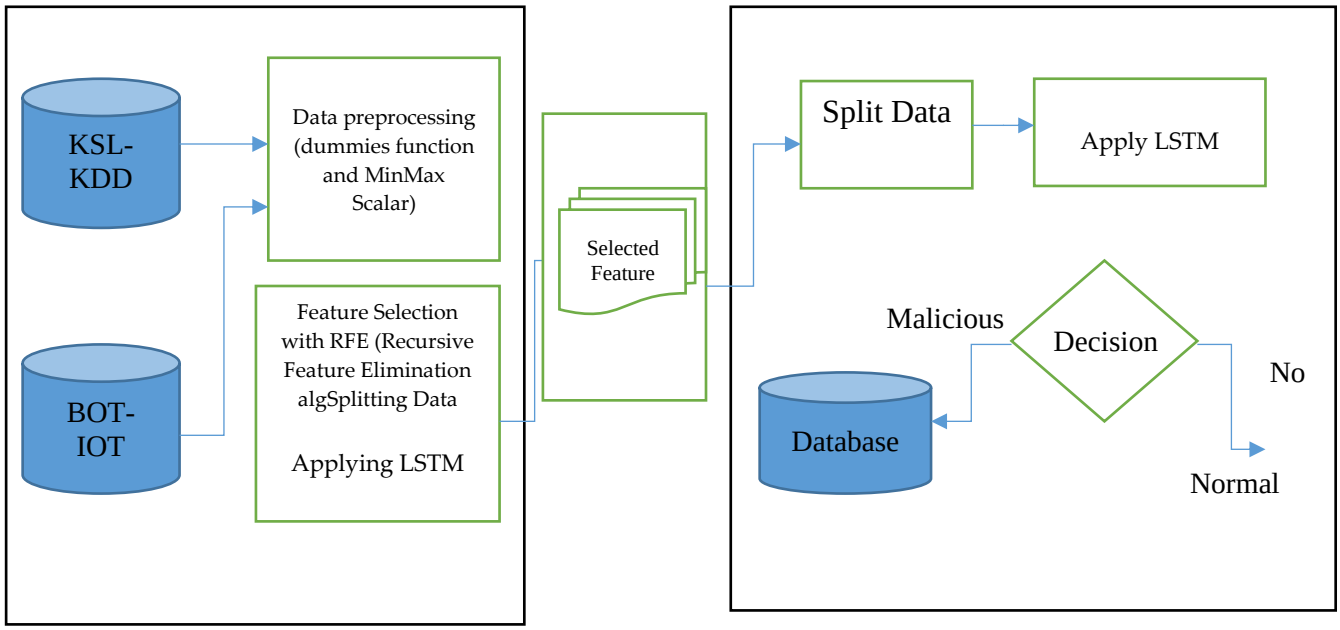

Fig. 2 Scheme of RFE-LSTM-IDS

3.1.2. Optimised Data Structures

During the training and testing stages, we have embraced effective data structures for dataset access and storage, which enable quick retrieval. This optimisation increases computational efficiency while reducing memory usage.

3.1.3. Normalization and Data Preprocessing

Detailed data preprocessing entails normalizing feature values and transforming categorical attributes into numerical values.

By securing steady scaling and accelerating convergence during training lowers the computational load and speeds up convergence.

3.1.4. Feature Reduction using RFE

RFE helps us find the most important features for detecting intrusions. By keeping only these key features and discarding unnecessary ones it reduces the complexity of the data dimensionality and allows the model to work faster and more efficiently.

3.1.5. Smart Batching

To optimize data batching during LSTM model training and minimise memory consumption while accelerating learning, intelligent batching techniques are implemented.

3.1.6. Optimized LSTM Hyperparameters

To guarantee accurate and successful intrusion detection, the LSTM model's hyperparameters are carefully adjusted.

The goal of this optimisation is to maximise overall efficiency by avoiding needless computational overhead and striking a balance between complexity and performance.

Table 2. RFE Algorithm

Algorithm 1: Feature Reduction Algorithm

Input:

- IntrusionDataset
- IoTDataset
- FeatureSelection algorithm
- RNN model

Output:

- MetricsTable1
- MetricsTable2

Variables:

- StandardizedData1
- StandardizedData2
- SelectedFeatures1
- SelectedFeatures2

Begin:

```
StandardizedData1 = Standardize(IntrusionDataset)
SelectedFeatures1 = FeatureSelection(StandardizedData1)
StandardizedData2 = Standardize(IoTDataset)
SelectedFeatures2 = FeatureSelection(StandardizedData2)
Model = RNN(SelectedFeatures1)
Model.fit(TrainData1)
MetricsTable1 = Evaluate(Model.predict(TestData1))
Model = RNN(SelectedFeatures2)
Model.fit(TrainData2)
MetricsTable2 = Evaluate(Model.predict(TestData2))
Display(MetricsTable1, MetricsTable2)
```

End.

By incorporating these efficiency-boosting strategies, we hope to demonstrate how our suggested approach performs better than previous research using comparable datasets and methodologies. With thorough comparative analysis and rigorous experimentation, we hope

to provide verifiable proof of the efficiency gains our method achieves. Through the examination of multiple metrics and benchmarks, such as computational time, memory usage, and detection precision, we aim to highlight the effectiveness and real-world relevance of our refined RFE-LSTM-IDS system. This empirical validation aims to improve the state-of-the-art intrusion detection systems for cloud computing, ultimately leading to increased resilience against ever-evolving security threats.

3.1. Data Preprocessing

To improve data quality and shorten processing times, our IDS preprocesses datasets that include both numerical and categorical features. Numerical values were assigned to categorical attributes, and character-based value systems were changed by applying suitable encoding techniques, like one-hot encoding. This conversion removes bias during analysis and allows for the effective processing of categorical data.

The process of normalizing features was carried out to guarantee uniform scaling throughout the dataset. Normalization makes the data analysis faster and fairer (reduces bias) and helps the intrusion detection system (IDS) learn and improve (converge) more efficiently. This is because normalization brings all data points to a similar range, leading to better training and, ultimately, a more effective IDS. Reducing features is essential for increasing model efficiency and processing speed. The RFE-LSTM is used to identify the most important features for classification. The approach aids in choosing the best features and removing any that were not helpful, resulting in a more focused and efficient model. We determined the most useful features required for efficient intrusion detection in cloud computing environments by using LSTM networks for feature selection. To summarize, our approach to IDS preprocessing, which involves feature transformation, normalization, and feature reduction through the application of the RFE-LSTM algorithm, maximizes processing time and improves intrusion detection effectiveness in cloud computing settings.

3.2. Intrusion Detection

After the feature selection phase, we used LSTM networks for network intrusion detection because of their propensity to identify patterns and temporal dependencies in data. Recurrent neural networks (RNNs) of the LSTM network type are specifically made to handle sequence learning tasks, which makes them ideal for classification issues that arise in intrusion detection. It uses a three-layer structure: input, hidden, and output layers, as shown in Figure 2. This structure effectively tackles problems with long-term dependencies.

LSTM networks are structured with three primary layers: the input layer, the LSTM layer, and the output layer. The input layer processes input data sequences before forwarding them to the LSTM layer. Within the LSTM layer, memory cells and gates facilitate the retention of information over time, enabling the model to capture long-term dependencies in the data. These memory cells are like tiny storage units with three controllers: an "input gate" that decides what new information to remember, a "forget gate" that cleans out old information, and an "output gate" that controls what information gets used at each point in time, shown in equations 1, 2, and 3.

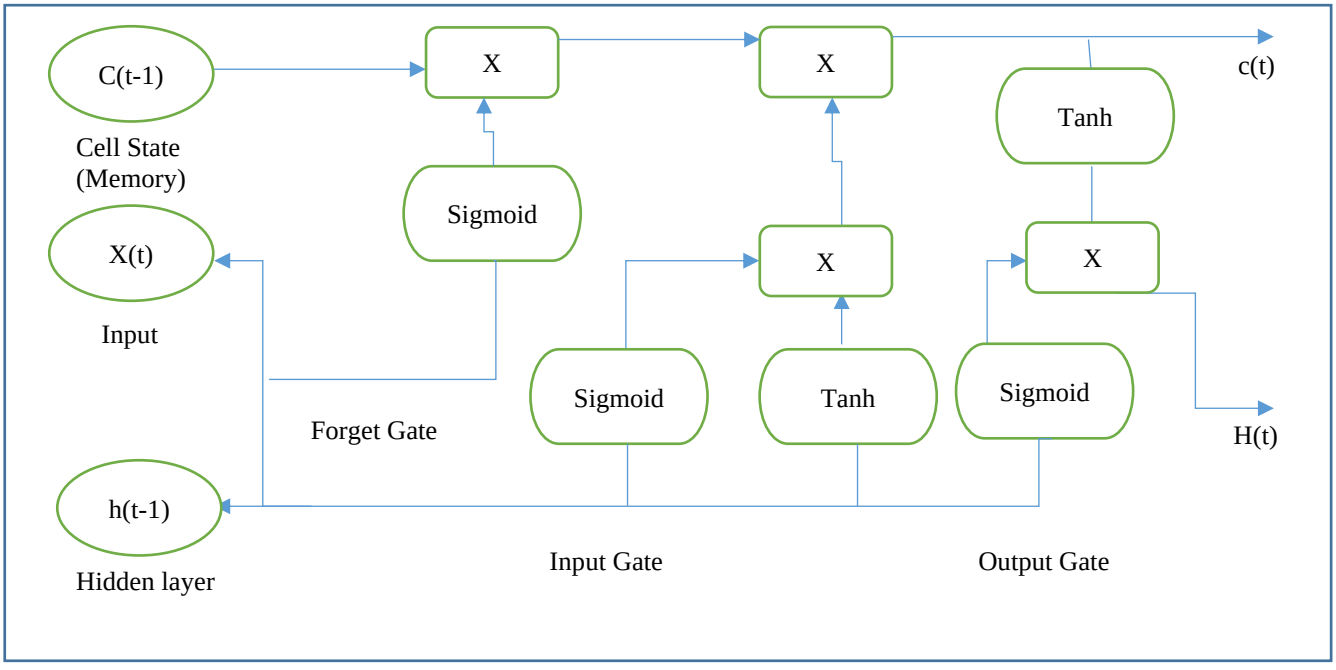

Fig. 3 Scheme of LSTM Model

$$i_t = \sigma(w_i\,[h_{t-1}, x_t] + b_i) \quad (1)$$

$$f_t = \sigma\big(w_f\,[h_{t-1}, x_t] + b_f\big) \quad (2)$$

$$O_t = \sigma(w_o\,[h_{t-1}, x_t] + b_o) \quad (3)$$

Whereas i_t represents the input gate and h_{t-1} represents the output of the previous LSTM block (at timestamp t – 1) and x_t represents the input at the current timestamp and b_x biases for the respective gates, f_t represents the forget gate and w_x represents the weight for the respective gate.

The hidden layer's LSTM units can decide which information to keep or discard depending on how pertinent it is to the input at hand. LSTM networks are useful for modelling complex sequences because of their dynamic nature, which also makes them useful for identifying intrusions in cloud computing environments. Moreover, LSTM networks' natural flexibility allows them to discover and adjust to underlying patterns in the data independently. With this capability, the network can model complex relationships between various features and quickly detect anomalies or intrusions in real time, eliminating the need for explicit feature engineering or selection. Our intrusion detection system's accuracy and efficiency in cloud computing environments are increased when we use LSTM networks for intrusion detection because of their innate capacity to capture temporal dynamics and dependencies within the data.

4. Results and Discussion

We implemented and evaluated the proposed framework in a controlled experiment using a Windows 10 Professional 64-bit desktop computer. The computer was equipped with an Intel Core TM-i5 8250U 1.8GHz CPU. The modeling was done in Python 3. Classification performance was assessed using confusion matrices. Two datasets were employed: the Bot-IoT dataset and the NSL-KDD dataset. The latter addresses issues in the original KDD 1999 data by removing redundancies and providing sufficient records, like in the eKDDTrain+ 20Percent file. It includes 41 features from KDD'99. The Bot-IoT data offers insights into IoT traffic types, including malware, regular traffic and IoT applications. As shown in Table 2, both datasets encompass numerous attributes, with 44 and 46 features, respectively.

Table 3. Dataset description

Dataset	Number of Features	Class
NSL-KDD 125,973	44	Normal, DoS, Probe, Neptune, Perl
Bot-IoT 73,370,443	46	Normal, DoS, DDoS, Information

The presented research explores two distinct datasets: the NSL-KDD and the Bot-IoT. Each dataset offers unique insights and challenges for study. The NSL-KDD dataset comes in two versions. The first version contains 10 features, such as "dst_bytes", "src_bytes", and "count", while the second version is condensed to only 4 key features: "flag", "logged_in", "same_srv_rate", "protocol_type", and "class". On the other hand, the Bot-IoT dataset also provides two variations; the first includes 10 features encompassing attributes like " daddr ", "TnP_PerProto", "TnP_PSrcIP", and others, and the second version reduces the feature set to just 3 key attributes - "daddr", "TnP_PerProto", and "TnP_PSrcIP", while also introducing the "attack" label. These datasets, with their varying feature dimensions, are integral to our research. They enable us to tailor our analyses and machine learning approaches to address different aspects of our research objectives, ultimately contributing valuable insights to our study.

Table 4. Used features

Dataset	Number of Features	Features
NSL-KDD	10	“dst_bytes”, “src_bytes”, “count”,
NSL-KDD	04	“flag”, “logged_in”, “same_srv_rate”, “protocol_type”, “class”.
Bot-IoT	10	“daddr”, “TnP_PerProto”, “TnP_PSrcIP”, “saddr”, “TnP_PDstIP”, “TnBPSrcIP”, “bytes”, “stime”, “TnP_Per_Dport”, “TnBPDstIP”, “attack”.
Bot-IoT	3	“daddr”, “TnP_PerProto”, “TnP_PSrcIP”, “attack”

4.1. Evaluation Metric

This subsection provides a summary of the effectiveness metrics that validate the suggested approach. The response of each metric to the suggested model is then further discussed in the ensuing subsection. To assess the effectiveness of the algorithm, a confusion matrix was created, as shown in Table 4. As a result, the metrics are calculated, which include Matthews Correlation Coefficient (MCC), Accuracy (ACC), Precision, and Recall.

Table 5. The confusion matrix

	Actually Positive	Actually Negative
Predict positive	True positive (TP)	False positive (FP)
Predict negative	False negative (FN)	True negative (TN)

It is crucial to comprehend the following entries (TP, FP, FN, and TN) in a confusion matrix:

- TP stands for True Positives or instances that the model correctly identified as attacks.
- TN: True Negatives: Cases that the model accurately identified as normal.
- False Positives (FP) are instances that the model mistakenly identified as attacks.
- False Negatives (FN) are instances that the model mistakenly identified as normal.

The four categories in the confusion matrix and the imbalance in the dataset are taken into account when evaluating the classifier's performance using the Matthews Correlation Coefficient (MCC). It offers a thorough evaluation of the model's ability to manage both attack and typical cases. Furthermore, the metrics that are utilised are explained as follows.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

$$Precsion = \frac{TP + TN}{TP + FP} \quad (5)$$

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

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + TN)(TP + FP)(TN + FP)(TN + FN)}} \quad (7)$$

Intrusion detection datasets often exhibit class imbalance, with common events outweighing rare intrusions. To address the challenge and ensure the presented model's accuracy and credibility, comprehensive measures were implemented. First, resampling techniques were used to balance the training data distribution by over- or under-sampling the minority class, ensuring equal representation of positive and negative instances during learning. Additionally, the Synthetic Minority Over-sampling Technique (SMOTE) was utilized to supplement the minority class through generating synthetic samples, amplifying its presence. The costs of different errors when training model is considered and made it a priority to accurately identify rare events (intrusions) even if it meant sometimes mistakenly labeling a normal activity as suspicious. The ensemble learning is applied to improve the model's overall performance. The approach combines multiple decision-making classifiers into one. The approach leads to better results, particularly for less common situations, such as identifying rare intrusions.

The F1-score and Matthews Correlation Coefficient (MCC), defined by Equation (7), serve as our chosen evaluation metrics. By combining these powerful measures, the proposed approach effectively tackles the problems of unbalanced data in intrusion detection, leading to accurate and reliable results.

The effectiveness of the proposed model on the NSL-KDD and Bot-IoT datasets is explored in this section.

4.2. NSL-KDD Dataset

A variety of metrics for assessing our model on the NSL-KDD dataset are shown in Table 5 and Figure 3. 91.50% Accuracy (ACC), 92.21% Precision, 47.54% Recall, and 82.31% Matthews Correlation Coefficient (MCC) were attained by the entire dataset. Our feature selection model significantly reduces the number of features while maintaining exceptional performance, even with these high scores. The following metrics were obtained from the ten and four selected features as proof:

- For ten specific features: 85.32% MCC, 45.21% Recall, 91.12% Precision, and 93.45% ACC.
- For four features that have been chosen: 88.49% MCC, 90.51% Precision, 42.81% Recall, and 96.62% ACC.

Table 6. Metrics used to evaluate the NSL-KDD dataset performance.

Features	ACC (%)	Precision (%)	Recall(%)	MCC (%)
Complete Dataset	**91.50**	**92.21**	**47.54**	**82.31**
10 Features	**93.45**	**91.12**	**45.21**	**85.32**
4 Features	**96.62**	**90.51**	**42.81**	**88.49**

The four chosen features successfully distinguished between normal occurrences and attacks, resulting in increased MCC scores. We concentrated on discussing the outcomes using ten features from the NSL-KDD dataset. Finding the minimal features from this subset that are needed to improve our model is the main goal of this study. Although we experimented with utilising one, two, or three features, we found that utilising four carefully chosen features produced the best results.

Figure 4 presents a bar graph analysis of the NSL-KDD dataset's scores for TP, TN, FP, and FN performance metrics using the proposed technique. When the figure is examined closely, our feature selection strategy greatly enhances the model's ability to identify negative cases and detect positive ones. Our methodology specifically reveals that The NSL-KDD entire dataset's feature set attained an 86% TN rate. The TN rate was raised to 89% by the ten features that were chosen. Additionally, by utilising just four selected components, the TN rate increased to a remarkable 93%.

Using our methodology, Figures 5, 6 and 7 show the four critical confusion matrix measures: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). These measures are obtained from the NSL-KDD dataset analysis over different feature sets. Figure 5 illustrates the confusion matrix for the total NSL-KDD dataset. A remarkable 89% TN rate was attained by the model, with FN contributing 12%, FP 7%, and TP 86%.

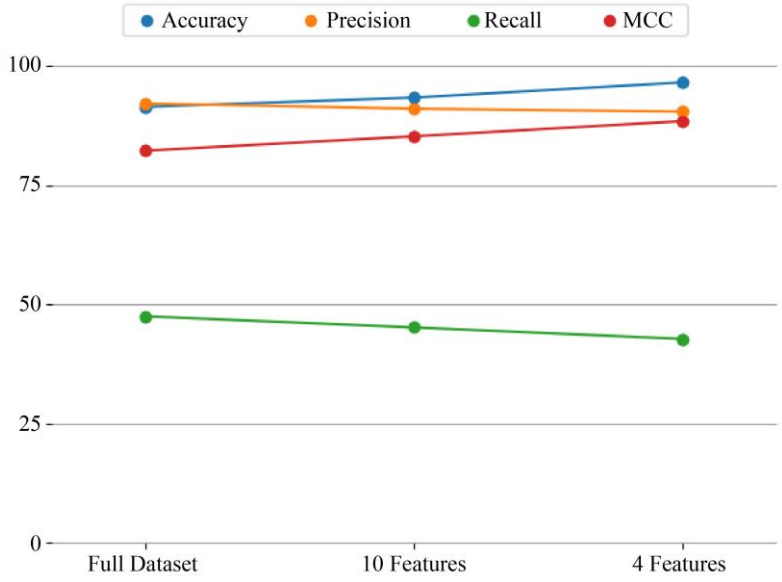

Fig. 4 Various measures evaluate the model's effectiveness on the NSL-KDD dataset

The confusion matrix in Figure 6 was produced using feature selections from the NSL-KDD dataset. The model notably enhanced the detection of negative cases, with an 87% true negative rate and only 5% false negative rate. Additionally, with 8% false positives and 89% true positives, it maintained the strong ability of the full dataset to identify positive samples. For the selected features in the NSL-KDD dataset, the confusion matrix is shown in Figure 7. The model performed extremely well in detecting negative cases, achieving a minimum 3% false negative rate and an excellent 87% true negative rate. It also confirmed the outstanding results observed previously using the full dataset and ten chosen features for differentiating positive samples, with 9% false positives and 93% true positives.

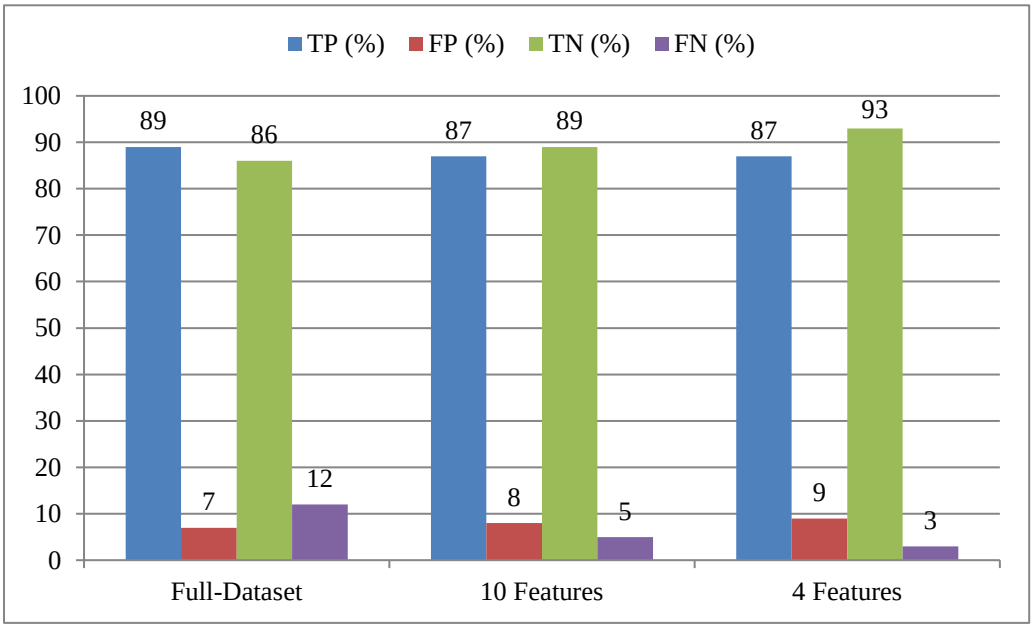

Fig. 5 The confusion matrix of the NSL-KDD dataset

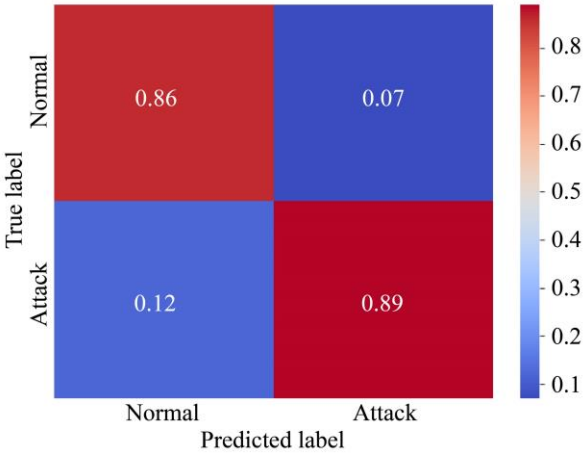

Fig. 6 The confusion matrix of the complete NSL-KDD dataset

The confusion matrix indicates that the model is highly accurate in classifying both "Normal" and "Attack" cases, with a slight tendency to misclassify "Attack" cases as "Normal" more often than misclassifying "Normal" cases as "Attack".

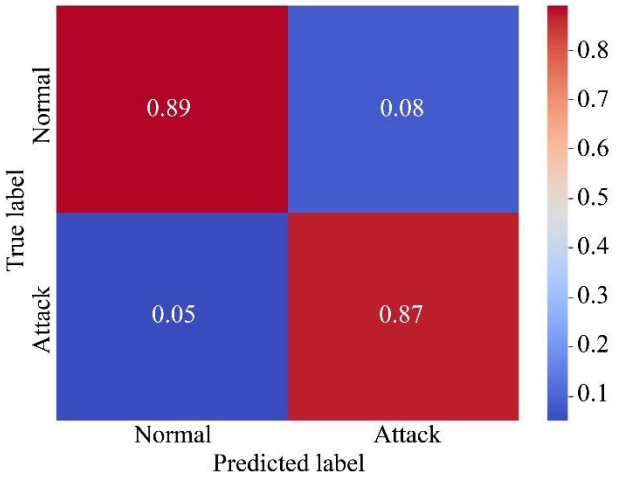

Fig. 7 The confusion matrix of ten specified attributes from the NSL-KDD dataset

Figure 7 shows that for ten features, the model correctly identifies "Normal" behavior with a high accuracy of 89% and "Attack" behavior with 87% accuracy.

The misclassification rate is low, with 8% of "Normal" behavior incorrectly labeled as "Attack" and only 5% of "Attack" behavior mislabeled as "Normal". This suggests an overall strong performance with balanced precision and recall across both classes.

Figure 8 shows the model's ability to correctly classify "Normal" behavior, with a high accuracy of 93%, while "Attack" behavior is classified correctly, with an accuracy of 87%. There is a very low rate of misclassification for "Attack" behavior as "Normal" (3%) and "Normal" behavior as "Attack" (9%).

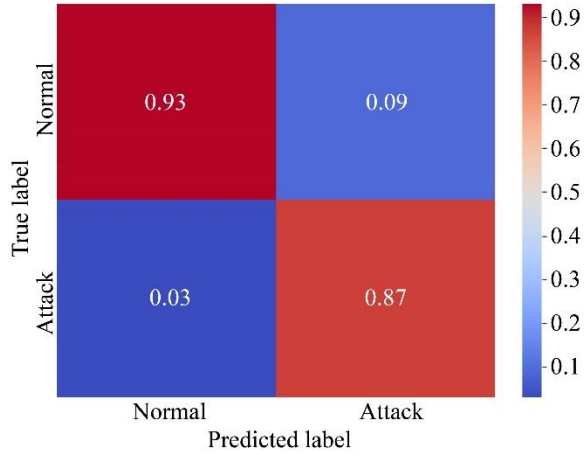

Fig. 8 The confusion matrix of four specified attributes from the NSL-KDD dataset

5. Conclusion

The research study proposed a machine-learning-enhanced intrusion detection system. The primary focus of this research was on recursive feature elimination and long-short-term memory-based techniques for enhanced cloud security. The analysis concludes that cloud computing security challenges the IDS in safeguarding cloud infrastructure versus broad-scale security threats, including traditional and zero-day attacks. The proposed RFE-LSTM-IDS model shows reasonable accuracy in intrusion detection by combining machine learning with traditional IDS techniques and also contributes to trust management without affecting efficiency. The presented model's performance was assessed on the NSL-KDD and BoT-IoT datasets for feature selection reduction capability, and it was found that the model performs reasonably well on the evaluation criteria of accuracy and precision. The model performed 91.50% and 92.21% for accuracy measures for the datasets provided. The precision measure performance was 47.54%, and the recall measure was 82.31% for the datasets provided for the Matthews Correlation Coefficient (MCC) across the whole dataset.

The second evaluation was performed by reducing the features to ten and four. The model again presents reasonable value; the four-feature model shows 96.62% accuracy and 88.49% MCC evaluation. The proposed model feature selection ability improved with the identification of true negatives; the increase rate was 86% to 93% with four feature settings. The model's classification capability was assessed by confusion matrix analysis, which represents “normal” and "attack" cases. The high true negative rates and low false negative rates were presented for assessment of the classification capabilities of the model.

In conclusion, the research contributes valuable findings for efficient and accurate intrusion detection systems. The proposed model provides an integrated method to identify the

threats termed zero-day threats and also performs accurately with classical threats within a cloud computing environment. The machine learning capabilities that enhance the traditional IDS, the integration of REF and LSTIM, and the proposed model open new ways to investigate intrusion detection systems.

The study advances cloud security as well as sets the direction for new integrated IDS models with artificial intelligence, machine learning, and deep learning-equipped models. This integration is very much required due to the ever-evolving security issues and security threats of cloud computing.

Funding Statement

Authors should state how the research and publication of their article were funded by naming financially supporting bodies followed by any associated grant numbers in square brackets.

Acknowledgments

An Acknowledgements section is optional and may recognise those individuals who provided help during the research and preparation of the manuscript. Other references to the title/authors can also appear here, such as "Author 1 and Author 2 contributed equally to this work.

References

[1] Md Tanzim Khorshed, A.B.M. Shawkat Ali, and Saleh A. Wasimi, "Trust Issues that Create Threats for Cyber Attacks in Cloud Computing," *2011 IEEE 17th International Conference on Parallel and Distributed Systems*, pp. 900-905, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[2] Talal Halabi, and Martine Bellaiche, "Towards Quantification and Evaluation of Security of Cloud Service Providers," *Journal of Information Security and Applications*, vol. 33, pp. 55-65, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[3] Talal Halabi, and Martine Bellaiche, "A Broker-based Framework for Standardization and Management of Cloud Security-SLAs," *Computers & Security*, vol. 75, pp. 59-71, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[4] Rafael Moreno-Vozmediano et al., "Efficient Resource Provisioning for Elastic Cloud Services Based on Machine Learning Techniques," *Journal of Cloud Computing*, vol. 8, pp. 1-18, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[5] Ahmed AlEroud, and George Karabatis, "A Contextual Anomaly Detection Approach to Discover Zero-day Attacks," *2012 International Conference on Cyber Security*, pp. 40-45, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[6] Diogo A.B. Fernandes et al., "Security Issues in Cloud Environments: A Survey," *International Journal of Information Security*, vol. 13, pp. 113-170, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[7] Maryam Douiba et al., "Anomaly Detection Model based on Gradient Boosting and Decision Tree for IoT Environments Security," *Journal of Reliable Intelligent Environments*, vol. 9, pp. 421-432, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Ansam Khraisat et al., "Survey of Intrusion Detection Systems: Techniques, Datasets and Challenges," *Cybersecurity*, vol. 2, no. 20, pp. 1-22, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[9] Mohammad Almseidin et al., "Evaluation of Machine Learning Algorithms for Intrusion Detection System," *2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY)*, Subotica, Serbia, pp. 277- 282, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[10] Nanak Chand et al., "A Comparative Analysis of SVM and Its Stacking with Other Classification Algorithm for Intrusion Detection," *2016 International Conference on Advances in Computing, Communication, & Automation (ICACCA)(Spring)*, Dehradun, India, pp. 1-6, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[11] Donghwoon Kwon et al., "A Survey of Deep Learning-based Network Anomaly Detection," *Cluster Computing*, vol. 22, pp. 949-961, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[12] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep Learning," *Nature*, vol. 521, pp. 436-444, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[13] Mohamad Mulham Belal, and Divya Meena Sundaram, "Comprehensive Review on Intelligent Security Defences in Cloud: Taxonomy, Security Issues, ML/DL Techniques, Challenges and Future Trends," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 10, pp. 9102-9131, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Ayman M. El-Zoghby, and Marianne A. Azer, "Cloud Computing Privacy Issues, Challenges and Solutions," *2017 12th International Conference on Computer Engineering and Systems (ICCES)*, Cairo, Egypt, pp. 154-160, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[15] Nabeel Mohammad Abdullah Al-Jaser, "A Survey on Cloud Computing Security–Challenges and Trust Issues," *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 18, no. 5, pp. 1-6, 2020. [Google Scholar] [Publisher Link]

[16] Z. Chiba et al., "A Cooperative and Hybrid Network Intrusion Detection Framework in Cloud Computing based on Snort and Optimized back Propagation Neural Network," *Procedia Computer Science*, vol. 83, pp. 1200-1206, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[17] Sebastian Roschke, Feng Cheng, and Christoph Meinel, "Intrusion Detection in the Cloud," *2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing*, Chengdu, China, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[18] Chirag Modi et al., "A Novel Framework for Intrusion Detection in Cloud," *Proceedings of the Fifth International Conference on Security of Information and Networks*, pp. 67-74, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[19] Suaad Alarifi, and Stephen Wolthusen, "Anomaly Detection for Ephemeral Cloud IaaS Virtual Machines," *Network and System Security: 7th International Conference*, NSS 2013, pp. 321-335, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[20] Sanchika Gupta, and Padam Kumar, "System cum Program-wide Lightweight Malicious Program Execution Detection Scheme for Cloud," *Information Security Journal: A Global Perspective*, vol. 23, no. 3, pp. 86-99, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[21] Hojoon Lee et al., "Ki-mon Arm: A Hardware-Assisted Event-Triggered Monitoring Platform for Mutable Kernel Object," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, no. 2, pp. 287-300, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[22] N. Pandeeswari, and Ganesh Kumar, "Anomaly Detection System in Cloud Environment Using Fuzzy Clustering Based ANN," *Mobile Networks and Applications*, vol. 21, pp. 494-505, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[23] Chris Benninger et al., "Maitland: Lighter-Weight VM Introspection to Support Cyber-security in the Cloud," *2012 IEEE Fifth International Conference on Cloud Computing*, Honolulu, HI, USA, pp. 471-478, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[24] Jessey Bullock, and Jeff T. Parker, *Wireshark for Security Professionals: Using Wireshark and the Metasploit Framework*, John Wiley & Sons, pp. 1-288, 2017. [Google Scholar] [Publisher Link]

[25] Michael Pearce, Sherali Zeadally, and Ray Hunt, "Virtualization: Issues, Security Threats, and Solutions," *ACM Computing Surveys (CSUR)*, vol. 45, no. 2, pp. 1-39, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[26] Hisham A. Kholidy et al., "HA-CIDS: A Hierarchical and Autonomous IDS for Cloud Systems," *2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks*, Madrid, Spain, pp. 179-184, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[27] Nadia Burkart, and Marco F. Huber, "A Survey on the Explainability of Supervised Machine Learning," *Journal of Artificial Intelligence Research*, vol. 70, pp. 245-317, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[28] Zhe Li, Weiqing Sun, and Lingfeng Wang, "A Neural Network Based Distributed Intrusion Detection System on Cloud Platform," *2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems*, Hangzhou, China, pp. 75-79, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[29] Suaad S. Alarifi, and Stephen D. Wolthusen, "Detecting Anomalies in IaaS Environments through Virtual Machine Host System Call Analysis," *2012 International Conference for Internet Technology and Secured Transactions*, London, UK, pp. 211-218, 2012. [Google Scholar] [Publisher Link]

[30] Udaya Tupakula, Vijay Varadharajan, and Naveen Akku, "Intrusion Detection Techniques for Infrastructure as a Service Cloud," *2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, Sydney, NSW, Australia, pp. 744-751, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[31] Umer Ahmed Butt et al., "A Review of Machine Learning Algorithms for Cloud Computing Security," *Electronics*, vol. 9, no. 9, pp. 1-25, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[32] H. Hourani, and M. Abdallah, "Cloud Computing: Legal and Security Issues," *2018 8th International Conference on Computer Science and Information Technology (CSIT)*, Amman, Jordan, pp. 13-16, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[33] Nidal Hassan Hussein, and Ahmed Khalid, "A Survey of Cloud Computing Security Challenges and Solutions," *International Journal of Computer Science and Information Security*, vol. 14, no. 1, pp. 52-26, 2016. [Google Scholar] [Publisher Link]

[34] Javier Martínez Torres, Carla Iglesias Comesaña, and Paulino J. García-Nieto, "Machine Learning Techniques Applied to Cybersecurity," *International Journal of Machine Learning and Cybernetics*, vol. 10, pp. 2823-2836, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[35] Mouaad Mohy-eddine et al., "An Efficient Network Intrusion Detection Model for IoT Security using K-NN Classifier and Feature Selection," *Multimedia Tools and Applications*, vol. 82, pp. 23615-23633, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[36] Anar A. Hady et al., "Intrusion Detection System for Healthcare Systems using Medical and Network Data: A Comparison Study," *IEEE Access*, vol. 8, pp. 106576-106584, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[37] Maryam Douiba et al., "An Improved Anomaly Detection Model for IoT Security using Decision Tree and Gradient Boosting," *The Journal of Supercomputing*, vol. 79, pp. 3392-3411, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[38] Anna L. Buczak, and Erhan Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153-1176, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[39] Nadia Chaabouni et al., "A OneM2M Intrusion Detection and Prevention System based on Edge Machine Learning," *NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium*, Budapest, Hungary, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[40] Imtiaz Ullah, and Qusay H. Mahmoud, "Design and Development of a Deep Learning-based Model for Anomaly Detection in IoT Networks," *IEEE Access*, vol. 9, pp. 103906-103926, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[41] Azka Wani, S. Revathi, and Rubeena Khaliq, "SDN-based Intrusion Detection System for IoT using Deep Learning Classifier (IDSIoT-SDL)," *CAAI Transactions on Intelligence Technology*, vol. 6, no. 3, pp. 281-290, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[42] Mouaad Mohy-eddine et al., "An Effective Intrusion Detection Approach based on Ensemble Learning for IIoT Edge Computing," *Journal of Computer Virology and Hacking Techniques*, vol. 19, pp. 469-481, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[43] Hanaa Attou et al., "Towards an Intelligent Intrusion Detection System to Detect Malicious Activities in Cloud Computing," *Applied Sciences*, vol. 13, no. 17, pp. 1-19, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[44] Amirah Alshammari, and Abdulaziz Aldribi, "Apply Machine Learning Techniques to Detect Malicious Network Traffic in Cloud Computing," *Journal of Big Data*, vol. 8, no. 90, pp. 1-24, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[45] Feng Jiang et al., "Deep Learning Based Multi-Channel Intelligent Attack Detection for Data Security," *IEEE Transactions on Sustainable Computing*, vol. 5, no. 2, pp. 204-212, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[46] Zhiqiang Liu, and Yucheng Shi, "A Hybrid IDS using GA-based Feature Selection Method and Random Forest," *International Journal of Machine Learning Computing*, vol. 12, no. 2, pp. 43-50, 2022. [CrossRef] [Google Scholar] [Publisher Link]