Data and Metadata. 2025; 4:699
doi: 10.56294/dm2025699

ORIGINAL

Evidence Detection in Cloud Forensics: Classifying Cyber-Attacks in IaaS Environments using machine learning

Detección de evidencias en la informática forense en la nube: clasificación de ciberataques en entornos IaaS mediante aprendizaje automático

Suhaila Abuowaida[1], Hamza Abu Owida[2], Suleiman Ibrahim Shelash Mohammad[3,4], Nawaf Alshdaifat[5], Esraa Abu Elsoud[6], Raed Alazaidah[6], Asokan Vasudevan[7], Muhammad Turki Alshurideh[8]

[1]Department of Computer Science, Faculty of Prince Al-Hussein Bin Abdallah II for IT, Al al-Bayt University. Mafraq, Jordan.
[2]Department of Medical Engineering, Faculty of Engineering, Al-Ahliyya Amman University. Amman, 19328, Jordan.
[3]Electronic Marketing and Social Media, Economic and Administrative Sciences Zarqa University. Jordan.
[4]Research follower, INTI International University. 71800 Negeri Sembilan, Malaysia.
[5]Faculty of IT, Applied Science Private University. Amman, Jordan.
[6]Faculty of Information Technology, Zarqa University. Zarqa, Jordan.
[7]Faculty of Business and Communications, INTI International University. 71800 Negeri Sembilan, Malaysia.
[8]Department of Marketing, School of Business, The University of Jordan. Amman 11942, Jordan.

Cite as: Abuowaida S, Abu Owida H, Shelash Mohammad SI, Alshdaifat N, Abu Elsoud E, Alazaidah R, et al. Evidence Detection in Cloud Forensics: Classifying Cyber-Attacks in IaaS Environments using machine learning. Data and Metadata. 2025; 4:699. https://doi.org/10.56294/dm2025699

Submitted: 01-10-2024 **Revised:** 29-12-2024 **Accepted:** 17-02-2025 **Published:** 18-02-2025

Editor: Dr. Adrián Alejandro Vitón Castillo

Corresponding Author: Suleiman Ibrahim Shelash Mohammad

ABSTRACT

Introduction: cloud computing is considered a remarkable paradigm shift in Information Technology (IT), offering scalable and virtualized resources to end users at a low cost in terms of infrastructure and maintenance. These resources offer an exceptional degree of flexibility and adhere to established standards, formats, and networking protocols while being managed by several management entities. However, the existence of flaws and vulnerabilities in underlying technology and outdated protocols opens the door for malicious network attacks.
Method: this study addresses these vulnerabilities by introducing a method for classifying attacks in Infrastructure as a Service (IaaS) cloud environments, utilizing machine learning methodologies within a digital forensics framework. Various machine learning algorithms are employed to automatically identify and categorize cyber-attacks based on metrics related to process performance. The dataset is divided into three distinct categories—CPU usage, memory usage, and disk usage—to assess each category's impact on the detection of attacks within cloud computing systems.
Results: decision Tree and Neural Network models are recommended for analyzing disk-related features due to their superior performance in detecting attacks with an accuracy of 90 % and 87,9 %, respectively. Neural Network is deemed more suitable for identifying CPU behavior, achieving an accuracy of 86,2 %. For memory-related features, K-Nearest Neighbor (KNN) demonstrates the best False Negative Rate (FNR) value of 1,8 %.
Discussion: our study highlights the significance of customizing the selection of classifiers based on the specific system feature and the intended focus of detection. By tailoring machine learning models to particular system features, the detection of malicious activities in IaaS cloud environments can be enhanced, offering practical insights into effective attack classification.

Keywords: Cloud; Cloud Forensics; Machine Learning; Classifiers; CPU; Attack.

RESUMEN

Introducción: la computación en la nube se considera un cambio de paradigma notable en las tecnologías

de la información (TI), ya que ofrece recursos escalables y virtualizados a los usuarios finales a un bajo costo en términos de infraestructura y mantenimiento. Estos recursos ofrecen un grado excepcional de flexibilidad y se adhieren a estándares, formatos y protocolos de red establecidos, al tiempo que son administrados por varias entidades de gestión. Sin embargo, la existencia de fallas y vulnerabilidades en la tecnología subyacente y protocolos obsoletos abre la puerta a ataques maliciosos a la red.
Método: este estudio aborda estas vulnerabilidades mediante la introducción de un método para clasificar los ataques en entornos de nube de Infraestructura como Servicio (IaaS), utilizando metodologías de aprendizaje automático dentro de un marco de análisis forense digital. Se emplean varios algoritmos de aprendizaje automático para identificar y categorizar automáticamente los ataques cibernéticos en función de métricas relacionadas con el rendimiento del proceso. El conjunto de datos se divide en tres categorías distintas (uso de CPU, uso de memoria y uso de disco) para evaluar el impacto de cada categoría en la detección de ataques dentro de los sistemas de computación en la nube.
Resultados: los modelos de árbol de decisiones y red neuronal se recomiendan para analizar las características relacionadas con el disco debido a su rendimiento superior en la detección de ataques con una precisión del 90 % y el 87,9 %, respectivamente. La red neuronal se considera más adecuada para identificar el comportamiento de la CPU, logrando una precisión del 86,2 %. Para las características relacionadas con la memoria, K-Nearest Neighbor (KNN) demuestra el mejor valor de tasa de falsos negativos (FNR) del 1,8 %.
Discusión: nuestro estudio destaca la importancia de personalizar la selección de clasificadores en función de la característica específica del sistema y el enfoque de detección previsto. Al adaptar los modelos de aprendizaje automático a características particulares del sistema, se puede mejorar la detección de actividades maliciosas en entornos de nube IaaS, lo que ofrece información práctica para una clasificación eficaz de los ataques.

Palabras clave: Nube; Análisis forense de la nube; Aprendizaje automático; Clasificadores; CPU; Ataque.

INTRODUCTION

Cloud computing has become a global technology that has transformed the way organizations store, process, and manage their data. However, with the increased reliance on cloud services, the risk of cyber-attacks has also risen significantly. To address this challenge, researchers have explored the use of machine learning techniques to automatically detect and classify cyber-attacks in cloud environments.[1] Cloud computing reduces equipment purchase and maintenance expenses by leveraging cloud infrastructure. Cloud storage companies use crucial security features like authentication, access control, and encryption to protect their systems and the data they manage. The large amount of data that is sent between companies and cloud service providers might result in deliberate breaches of critical information. The same characteristics that facilitate employee and IT system employment of online services also make it difficult for organizations to restrict unauthorized access.[2,3] Businesses utilizing cloud services face additional security risks due to open APIs and authentication. Hackers with advanced skills can access cloud systems without permission by using these vulnerabilities, which poses serious risks to data security. It is crucial that cloud security is applied correctly and kept up to date to reduce disruptions and unauthorized access. Cloud service providers must implement stringent security measures to protect their resources and data, guaranteeing the provision of trustworthy and secure services. Cloud providers' incorporation of segmentation and isolation functions is essential for systems that support many users.

The creation of a robust cloud security framework requires that data be organized according to the risks associated with it. Considering the diverse array of applications for cloud services, such as enterprise operations, software solutions, social networking, and business utilities, it is paramount to recognize and mitigate the potential risks involved.[4,5] To address these challenges, machine learning techniques are used to strengthen data management and security protocols.[6] Machine learning, for example, can be used to create sophisticated intrusion detection systems that instantly recognize and react to questionable activity. Large datasets are analysed by these systems to find trends and anomalies that might point to a security breach.[7]

The present study examines the efficacy of using process-level performance metrics, such as CPU usage, memory usage, and disk usage, to detect and classify cyber-attacks in cloud computing. The paper commences by surveying the existing literature on the application of both machine learning and non-machine learning approaches to addressing security threats in the cloud domain. It then delves into the proposed methodology, which involves the utilization of various machine learning algorithms to analyze the performance metrics and identify patterns indicative of cyber-attacks.

The findings of this study demonstrate the potential of leveraging machine learning techniques to enhance the security of cloud computing environments. The analysis of the dataset, which is divided into three categories

(CPU usage, memory usage, and disk usage), reveals the distinct impact of each metric on the detection and classification of cyber-attacks. The stud structure is organized as follows. Section II is the summarization of the previous work related to this study. Section III describes the proposed methodology that implements the machine learning algorithms, section IV presents the results evaluations.

Related work

Recent efforts have focused on addressing the security problems of cloud computing using machine learning techniques. One notable study by D. C. Le et al.[8] discussed the challenges of insider threats which are the most expensive and difficult-to-detect forms of assault since insiders have access to a company's networked systems and are familiar with its structure and security processes. Detecting insider malware has a set of challenges such the unbalanced data and behavioural drifts and shifts. Machine learning is used to analyze data at several levels of detail under realistic situations to identify harmful behaviors, especially malicious insider attacks. Random Forest achieved good detection performance and F-score with low false positive rates. Their proposed work achieved an accuracy of 85 % and a false positive rate of only 0,78 %.

One of the most recent studies;[9] developed a Cyber-Attacks detection technique that combines the Principal Component Analysis, the Fuzzy C-Means technique for cluster creation, and the deep learning-based AutoEncoder method for attack detection in the cloud environment. They used the CE-CIC-2018 dataset with seven different attack types. The proposed technique achieved an accuracy equal to 97,7 %.

Sarosh[10] came up with an intrusion detection infrastructure that hybridizes K – means algorithm to flow the network with Support vector machine for training the model. They used UNSW-NB15 dataset for evaluating the proposed work performance based on some evaluation metrics like accuracy, and average detection time. The proposed model outperformed individual machine learning models such as random trees and decision trees with an accuracy of 89,7 %.

Bhatta[11] explored the ability of four main supervised machine learning techniques (Naive Bayes, AdaBoost, K-Nearest Neighbours, and Random Forest) for detecting abnormal network traffic using the UNSW-NB15 dataset. they used accuracy, Precision, recall, and F-score for evaluation. The results indicated that the Random Forest classifier outperformed the rest three classifiers that were examined in this paper with an accuracy of 95 %, precision of 93 %, recall of 95 %, and F-score of 93 %.

AlSaleh[12] is concerned on detecting the effectiveness of a Bayesian-based Convolutional Neural Network (BaysCNN) model in detecting Distributed Denial of Service (DDoS). attacks. For evaluation, they used the CICDDoS2019 dataset with accuracy, reliability, and efficiency as performance metrics. For feature selection, they used Principal Component Analysis (PCA). Results showed that the BaysCNN model obtains an average accuracy equal to 99,66 %.

Bakro et al.[13] proposed a hybrid features selection method based on combining an Information Gain (IG), Chi-Square (CS), and Particle Swarm Optimization (PSO). For handling imbalance data, they used the Synthetic Minority Over-Sampling Technique (SMOTE), and Random Forest (RF) classifier for detecting attacks. The multi-class classification scenarios outperformed the existing approaches with an accuracy exceeding 98 %.

Mayuranathan M et al.[14] developed an Intrusion Detection System (IDS) based on integrated Improved Heap Optimization (IHO) for data preprocessing, and Chaotic Red Deer Optimization (CRDO) for feature selection. They used a deep Kronecker neural network (DKNN) for detecting cloud attacks. The proposed model achievedaccuracy rates of 97,221 % for two Benchmark datasets called DARPA, and CSE-CICIDS2018.

METHOD

This section outlines the methodology used to develop an attack detection system in IaaS cloud environments. The proposed approach integrates cloud forensics and Machine Learning (ML) techniques within a digital forensic framework. The National Institute of Standards and Technology (NIST) defines Cloud computing forensic science as the utilization of scientific principles, technological methodologies, and established techniques to analyze previous cloud computing incidents by identifying, gathering, safeguarding, scrutinizing, interpreting, and documenting digital evidence.[15]

Cloud forensics plays a crucial role in cloud protection, representing a significant advancement in the field. The process of cloud forensics involves several stages, each with a clear and vital purpose. It helps identify the source of an attack and enhances overall security by focusing on effective attack detection. Two critical factors in cloud forensics are necessary: a high specific forecast rate, which means that a significant portion of the system's forecasts accurately identify real security concerns, indicating high prediction accuracy.[16] And a low false positive detection rate.Our model categorizes data fragments as CPU, memory, or disk usage before being fed into classifiers. This categorization helps improve the accuracy and efficiency of the detection process. The proposed model for our methodology is divided into 4 phases as shown in figure 1.

https://doi.org/10.56294/dm2025699

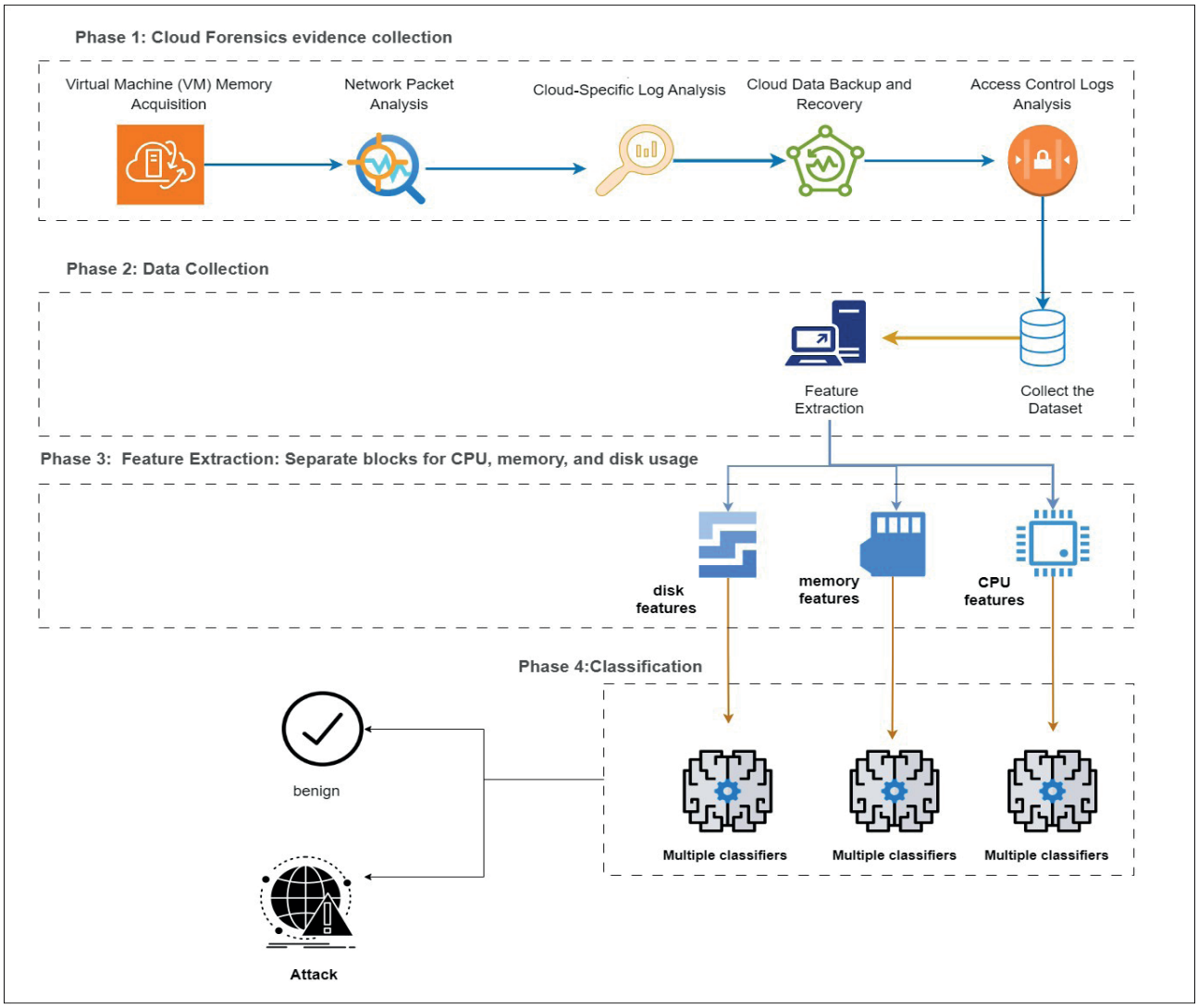

Figure 1. Proposed methodology

Virtual Machine (VM) Memory Acquisition

Forensic investigators obtain memory from virtual machines by creating snapshots by setting up a private cloud using an Intel® Core™ i5-4590 Processor, 12 GB of RAM, and 1 TB of HDD. The setup includes a KVM (Kernel-based Virtual Machine) type-1 hypervisor and OpenNebula as the cloud management platform. These snapshots capture the VM's memory, which can then be analyzed to identify processes, open files, and network connections, which would help in understanding the state of the VM during normal operation and under attack.

Data Collection and Features Extraction

A wide range of cyberattacks was modeled to provide an extensive dataset. These included malware injections, unauthorized access attempts, and DDoS attacks. Every kind of attack was chosen to symbolize typical security risks that cloud infrastructures must contend with. The collected dataset includes 19,190 instances with 44 attributes that were carefully categorized according to the known condition of the virtual machine (attack or normal). The features include various slopes of network, CPU, memory, and disk parameters, the features and their descriptions are listed in table 1. These attributes were categorized into three main categories: CPU, memory, and disk, as shown in table 2.

Table 1. Features and their descriptions

Feature	Description	Feature	Description
VMID	Virtual Machine Identifier	UUID	Universally Unique Identifier
dom	Domain	rxbytes slope	Received Bytes Slope
rxpackets slope	Received Packets Slope	rxerrors slope	Received Errors Slope
rxdrops slope	Received Drops Slope	txbytes slope	Transmitted Bytes Slope

https://doi.org/10.56294/dm2025699

txpackets slope	Transmitted Packets Slope	txerrors slope	Transmitted Errors Slope
txdrops slope	Transmitted Drops Slope	timecpu slope	CPU Time Slope
timesys slope	System Time Slope	timeusr slope	User Time Slope
state slope	State Slope	memmax slope	Maximum Memory Slope
mem slope	Memory Slope	cpus slope	CPUs Slope
cputime slope	CPU Time Slope	memactual slope	Actual Memory Slope
memswap in slope	Memory Swap In Slope	memswap out slope	Memory Swap Out Slope
memmajor fault slope	Major Memory Fault Slope	memminor fault slope	Minor Memory Fault Slope
memunused slope	Unused Memory Slope	memavailable slope	Available Memory Slope
memusable slope	Usable Memory Slope	memlast update slope	Last Update Memory Slope
memdisk cache slope	Disk CacheMemory Slope	memhugetlb pgalloc slo	peHugeTLB Page Allocation Slope
memhugetlb pgfail slope	HugeTLB Page Fail Slope	memrss slope	RSS Memory Slope
vdard req slope	VDA Read Requests Slope	vdard bytes slope	VDA Read Bytes Slope
vdawr reqs slope	VDA Write Requests Slope	vdawr bytes slope	VDA Write Bytes Slope
vdaerror slope	VDA Error Slope	hdard req slope	HDA ReadRequests Slope
hdard bytes slope	HAD Read Bytes Slope	hdawr reqs slope	HDA WriteRequests Slope
hdawr bytes slope	HAD Write Bytes Slope	hdaerror slope	HDA Error Slope
Label			

Table 2. Categorization of features into CPU, memory, and disk usage

CPU Usage	Memory Usage	Disk Usage
timecpu slope	memmax slope	vdard req slope
timesys slope	mem slope	vdard bytes slope
timeusr slope	memactual slope	vdawr reqs slope
cpus slope	memswap in slope	vdawr bytes slope
cputime slope	memswap out slope	vdaerror slope
	memmajor fault slope	hdard req slope
	memminor fault slope	hdard bytes slope
	memunused slope	hdawr reqs slope
	memavailable slope	hdawr bytes slope
	memusable slope	hdaerror slope
	memlast update slope	
	memdisk cache slope	
	memhugetlb pgalloc slope	
	memhugetlb pgfail slope	
	memrss slope	

RESULTS AND DISCUSSION

In this section of the study, we intend to outline and examine the results that have been obtained from the three different categories that have been defined and investigated. The thorough analysis carried out here provides insightful information about the overall effectiveness and performance displayed by the novel approaches that have been proposed.

Classification

CPU Category

In the CPU category, 6 features have been extracted, and the new dataset split in this category consists of 19190 instances, as shown in figure 2 and figure 3. Several classifiers were used in this study to evaluate their effectiveness. Decision Trees[17] were used with parameters like min samples leaf and min samples split, which

control the minimum sample size required for node splitting and leaf formation, respectively, and max depth, which limits the tree's depth to avoid overfitting. The Support Vector Machines (SVM) algorithm aims to locate the ideal hyperplane, or decision boundary, in an N-dimensional space to separate data points belonging to distinct classes. The hyperplane tries to maintain the largest possible buffer between the nearest points of various classes.[18] Logistic Regression (LR) is a technique that uses a set of independent variables to estimate the likelihood of an event occurring, such as voting or not. Using the sigmoid function, which accepts input as independent variables and outputs a probability value between 0 and 1, logistic regression is employed for binary classification. For instance, with two classes, Class 0 and Class 1, an input falls into Class 1 if the logistic function value is higher than the threshold of 0,5; otherwise, it falls into Class 0. Since it is a continuation of linear regression and is primarily applied to classification problems, it is known as regression.[19]

Na¨ıve Bayes, like other ML algorithms, simulates the distribution of inputs within a specific class or category, with the benefit of being very fast.[20] The K-Nearest Neighbors (KNN) algorithm is used in supervised machine learning for both classification and regression tasks. This method is highly applicable in practical situations due to its non-parametric nature, indicating the absence of assumptions about data distribution.[21] Finally, the configuration of Neural Network Classification involved specifying hidden layer sizes, which indicates the number of neurons in each hidden layer.[22,34,35,36,37,38,39,40] Each of these classifiers possesses unique parameters that impact their performance and suitability for various data types. The proposed model was exported to the workspace on MATLAB® to perform the mentioned classifiers. The results for detecting attacks depending on the CPU features are shown in figure 4.

Table 3. CPU category

Status	Instances
Attack	4612
Normal	14578
Total	19190

According to our unbalanced dataset and the results that we have. Precision and Recall are generally preferred over accuracy because they provide insights into the model's performance specifically for the minority class, and since the attack status is the minority class we have chosen the FNR to calculate the Precision and Recall values, and the TPR to the majority class, which is the normal status in our case, to understand where the model might be misclassifying. To calculate the Precision, Recall, and F-score values, the equations [1 2 3] were used.

Table 4. Comparison of classifier performance for CPU

Classifiers	Accuracy	Error Rate Validation	TPR (Normal)	FNR (Normal)	TPR (Attack)	FNR (Attack)	AUC
Decision Tree	88,20 %	11,80 %	96,30 %	3,70 %	62,70 %	37,30 %	91,95 %
Logistic Regression	85,90 %	14,10 %	99,70 %	0,30 %	42,20 %	57,80 %	85,19 %
Na¨ıve Bayes	85,70 %	14,30 %	99,50 %	0,50 %	42,10 %	57,90 %	80,50 %
SVM	86,20 %	13,80 %	99,70 %	0,30 %	43,60 %	56,40 %	87,16 %
KNN	85,80 %	14,20 %	98,70 %	1,30 %	45,00 %	55,00 %	71,80 %
Neural Network	86,20 %	13,80 %	98,70 %	1,30 %	46,80 %	53,20 %	88,40 %

$$\text{Precision} = \frac{TP}{TP+FP} \quad (1)$$

$$\text{Recall (Sensitivity)} = \frac{TP}{TP+FP} \quad (2)$$

$$\text{FI Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recal}} \quad (3)$$

The results are presented in table 5. The results show that the Decision Tree demonstrates notable efficacy in terms of precision when detecting attacks, indicating its capability to accurately recognize attacks with reduced False Positives. Conversely, the Neural Network displays a more equitable performance, showcasing increased recall for attacks and a slightly higher overall F-score. The Neural Network's capacity to uphold high precision for regular instances while enhancing recall for attacks significantly positions it as a more resilient option overall, particularly in situations where both precision and recall hold paramount importance. Figures 2 and 3 illustrate the results for all metrics for both attack and benign traffic.

https://doi.org/10.56294/dm2025699

Table 5. Comparison of classifier performance with precision, Recall, and F-score for CPU

Classifiers	Precision (Normal)	Recall (Normal)	F-score (Normal)	Precision (Attack)	Recall (Attack)	F-score (Attack)
Decision Tree	72,1 %	96,3 %	82.%3	94,5 %	62,7 %	75,1 %
Logistic Regression	99,7 %	42,2 %	59,2 %	42,2 %	95,8 %	58,6 %
Na¨ıve Bayes	99,5 %	42,1 %	59,1 %	42,1 %	95.%8	58,5 %
SVM	99,7 %	43,6 %	60,5 %	43,6 %	94,4 %	60,0 %
KNN	98,7 %	45,0 %	61,6 %	45,0 %	95,0 %	60,9 %
Neural Network	98,7 %	46,8 %	63,2 %	46,8 %	94,8 %	62,4 %

The Receiver Operating Characteristic (ROC) curve along with the corresponding Area Under the Curve (AUC) metric can offer valuable insights into classifier performance, particularly in elucidating the balance between sensitivity (true positive rate) and 1-specificity (false positive rate). ROC curves serve to visually assess the trade-offs between sensitivity and specificity for various classifiers, where a classifier positioned closer to the top-left corner (exhibiting higher TPR and lower FPR) signifies superior overall performance. Within the realm of imbalanced datasets, ROC curves, and AUC metrics play a pivotal role in providing a more comprehensive understanding of a classifier's ability to differentiate between classes, particularly the minority class (e.g., attacks) as shown in figure 4.

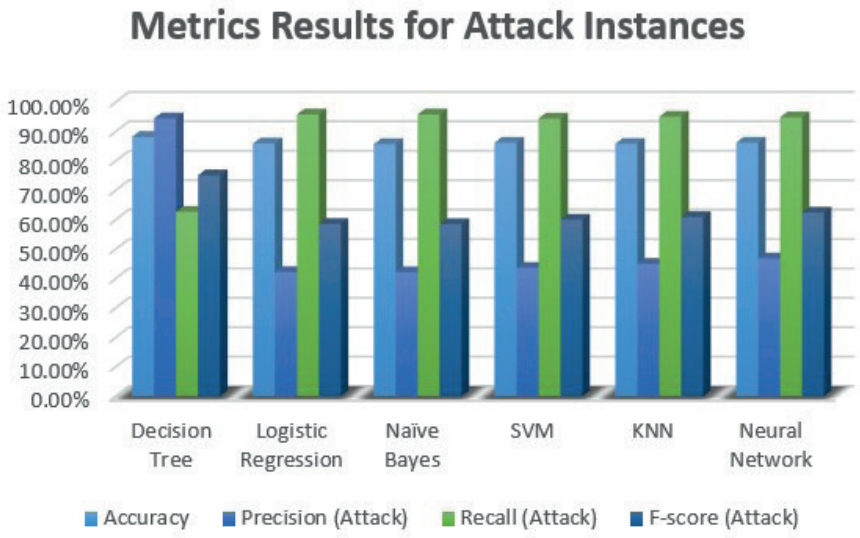

Figure 2. Metrics results for attack instances for CPU

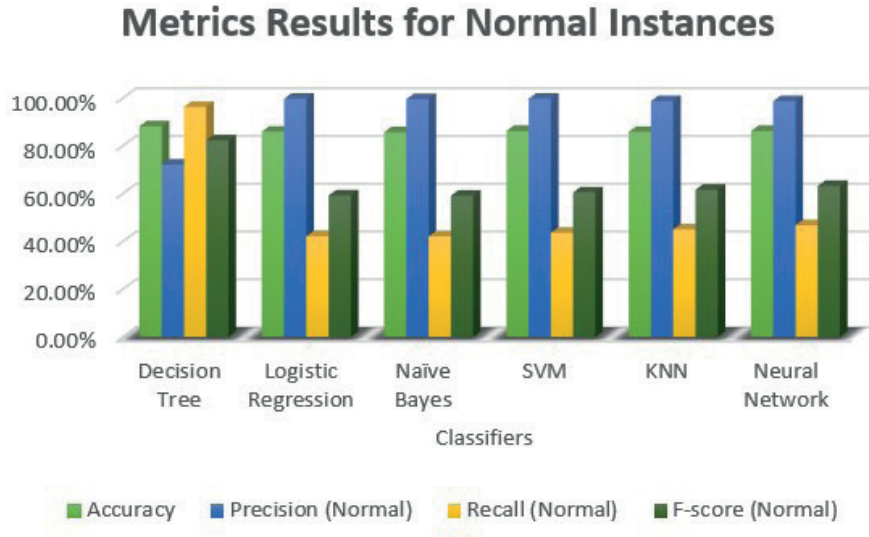

Figure 3. Metrics results for normal instances for CPU

https://doi.org/10.56294/dm2025699

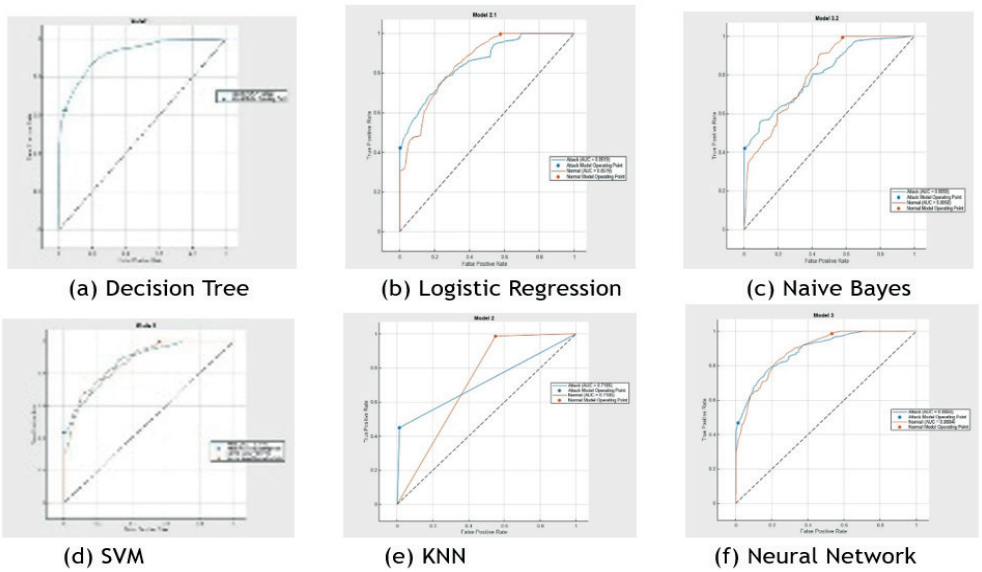

Figure 4. ROC Curves for different classifiers for CPU phase

Memory Category

The dataset in this category consists of 18 features (table 2) and 19190 instances. In the classification phase, a classification leaner application is used to train models to classify data. Supervised machine learning has been explored using various classifiers. The classifiers that have been used in this experiment are the same those used in the previous phase. The proposed model was exported to the workspace on MATLAB® to perform the mentioned classifiers. The results for the detecting attacks depending on the Memory features are shown in figure 6. The results for other evaluation metrics for all classifiers are shown in table 7, and figure 5 illustrates the ROC curves for different classifiers.

Table 6. Comparison of classifier performance for memory features

Classifiers	Accuracy	Error Rate Validation	TPR (Normal)	FNR (Normal)	TPR (Attack)	FNR (Attack)	AUC
Decision Tree	66,10 %	33,90 %	54,20 %	45,80 %	77,00 %	23,00 %	68,30 %
Logistic Regression	65,50 %	34,50 %	49,50 %	50,50 %	79,90 %	20,10 %	64,60 %
Na¨ıve Bayes	65,60 %	34,40 %	49,90 %	50,10 %	79,90 %	20,10 %	66,30 %
SVM	65,50 %	34,50 %	49,50 %	50,50 %	79,90 %	20,10 %	64,40 %
KNN	60,20 %	39,80 %	18,30 %	81,70 %	98,20 %	1,80 %	58,20 %
Neural Network	66,00 %	34,00 %	54,30 %	45,70 %	76,70 %	23,30 %	68,15 %

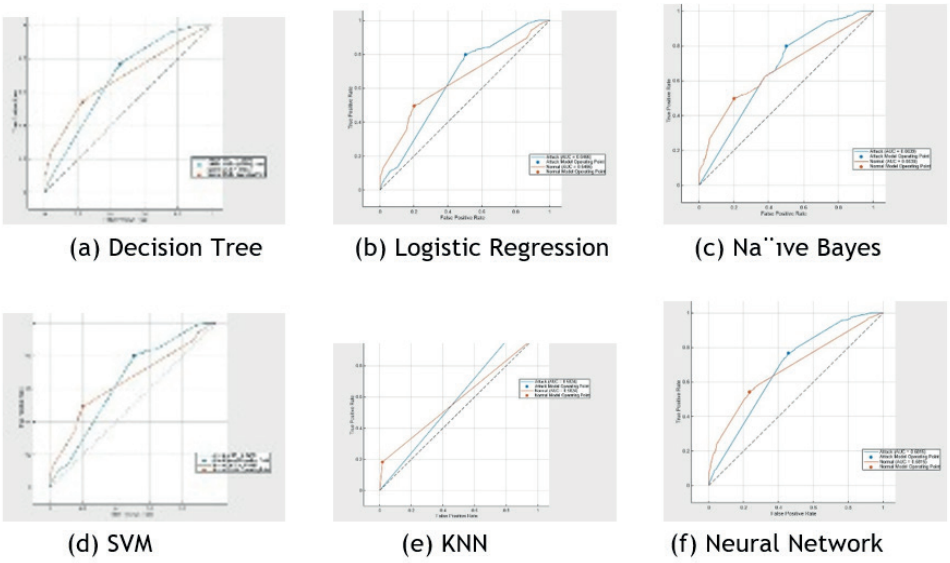

Figure 5. ROC curves for different classifiers for memory phase

https://doi.org/10.56294/dm2025699

Figures 6 and 7 illustrate that the most significant classifier in this phase is KNN due to its ability to correctly detect attack instances with TPR equal to 98,20 %. Also, it rarely misses an attack, which indicates the model is highly reliable in flagging true attack instances, minimizing the risk of undetected intrusions. This is very important in a cloud environment where security breaches can have significant consequences.

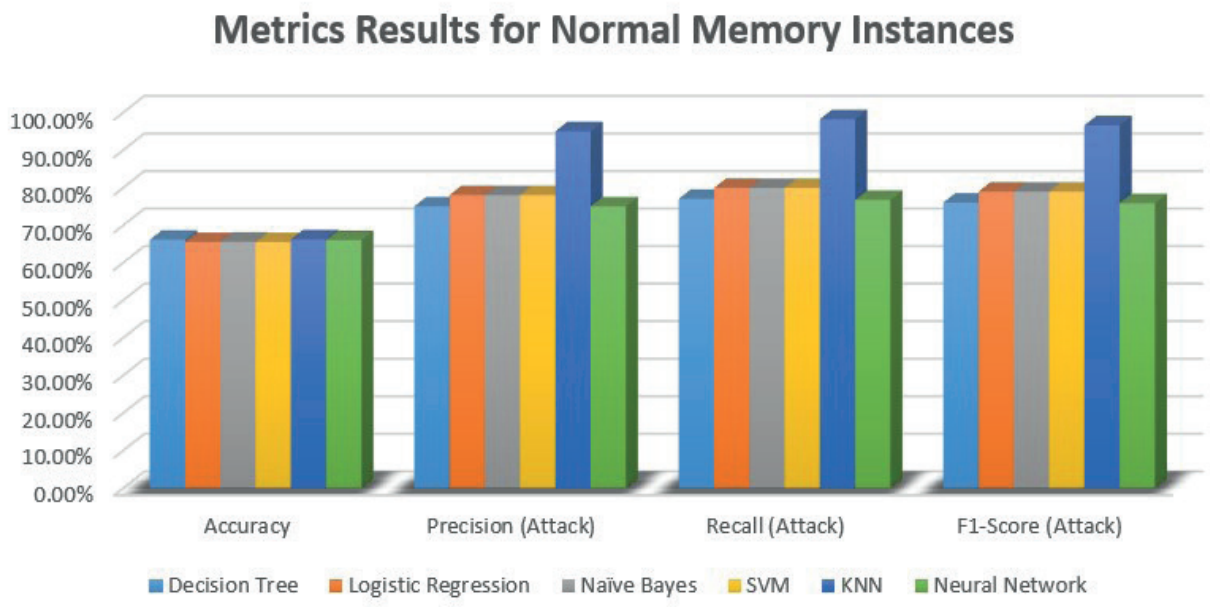

Figure 6. Metrics results for attack memory instances

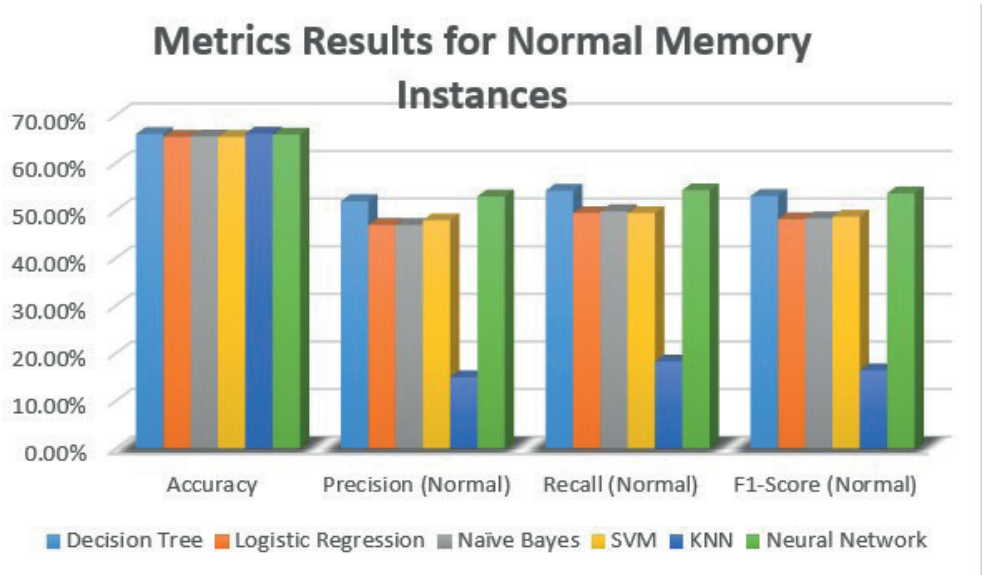

Figure 7. Metrics results for normal memory instances

Table 7. Comparison of classifier performance with precision, recall, and F-score for memory phase

Classifiers	Precision (Normal)	Recall (Normal)	F-Score (Normal)	Precision (Attack)	Recall (Attack)	F-Score (Attack)
Decision Tree	52,00 %	54,20 %	53,08 %	75,00 %	77,00 %	75,99 %
Logistic Regression	47,00 %	49,50 %	48,22 %	78,00 %	79,90 %	78,94 %
Na¨ıve Bayes	47,00 %	49,90 %	48,42 %	78,00 %	79,90 %	78,94 %
SVM	48,00 %	49,50 %	48,74 %	78,00 %	79,90 %	78,94 %
KNN	15,00 %	18,30 %	16,43 %	95,00 %	98,20 %	96,57 %
Neural Network	53,00 %	54,30 %	53,64 %	75,00 %	76,70 %	75,84 %

https://doi.org/10.56294/dm2025699

Disk Category

The dataset in this category consists of 11 features (as shown in table 8) and 19190 instances. The classifiers that have been used in this experiment are the same as those used in the previous phases. The proposed model was exported to the workspace on MATLAB® to perform the mentioned classifiers. The results for the detecting attacks depending on the Memory features are shown in figure 8, while the ROC curves are presented in figure 8. Based on the evaluation metrics outlined in equations 1 2 3 equations, the performance of various classifiers for disk features is summarized in table 9. According to these and as presented in figure The insights are further illustrated in figure 9. In the realm of identifying disk-based anomalies and intrusions in cloud environments, the selection of a suitable classifier plays a pivotal role in enhancing performance. The assessment of different classifiers such as Decision Tree, Logistic Regression, Na¨ıve Bayes, SVM, KNN, and Neural Network yields valuable insights into their efficacy across diverse metrics like Accuracy, Precision, Recall, F-Score, and AUC.

Table 8. Comparison of classifier performance for disk features

Classifiers	Accuracy	Error Rate Validation	TPR (Normal)	FNR (Normal)	TPR (Attack)	FNR (Attack)	AUC
Decision Tree	90 %	10 %	96,40 %	3,60 %	69,70 %	30,30 %	89,50 %
Logistic Regression	84,90 %	15,10 %	98,80 %	1,20 %	40,90 %	59,10 %	78,60 %
Na¨ıve Bayes	62,40 %	37,60 %	53,80 %	46,20 %	89,70 %	10,30 %	86,90 %
SVM	80,10 %	19,90 %	92,30 %	7,70 %	41,60 %	58,40 %	67,60 %
KNN	82,00 %	18 %	80,30 %	19,70 %	87,60 %	12,40 %	83,90 %
Neural Network	87,90 %	12,10 %	96,30 %	3,70 %	61,20 %	38,80 %	85,20 %

Decision Tree and Neural Network display the highest accuracy overall, with Decision Tree achieving 90,00 % and Neural Network achieving 87,90 %. These classifiers exhibit robust performance in detecting the normal class, characterized by high Precision, Recall, and F-Score. However, their performance in identifying attacks is moderate, as evidenced by lower Precision, Recall, and F-Score for the attack class.

Logistic Regression demonstrates outstanding performance in recognizing normal features with a Precision, Recall, and F-Score of 98,80 %.

When it comes to detecting attacks, Na¨ıve Bayes emerges as the top performer, achieving the highest Precision (89,70 %), Recall (89,70 %), and F-Score (89,70 %) for the attack class. Its robust performance in attack identification, coupled with a high AUC of 86,90 %, establishes it as a reliable choice in scenarios prioritizing attack detection. Nevertheless, its efficacy in normal class detection is comparatively lower, with Precision and Recall both at 53,80 %.

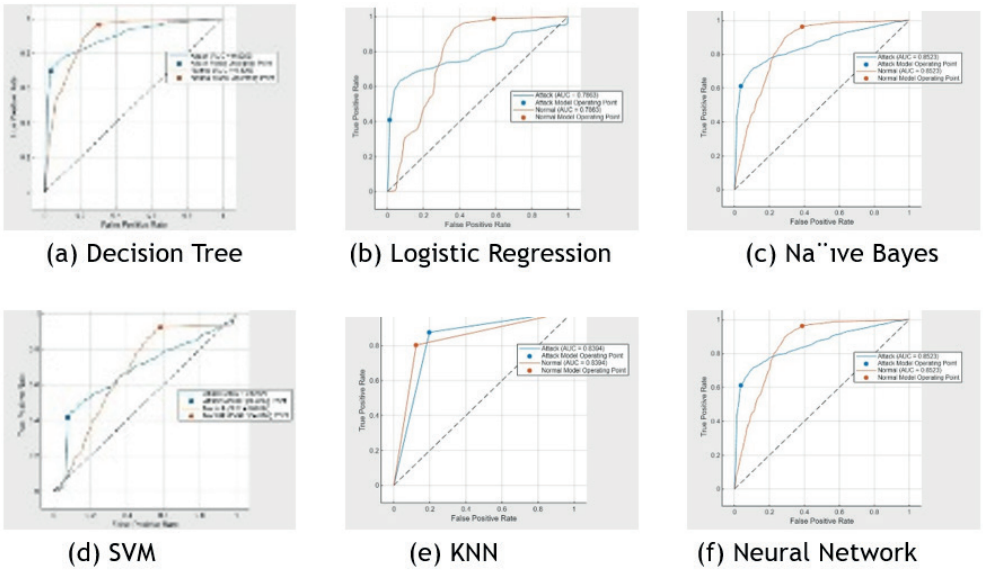

Figure 8. ROC Curves for Different Classifiers for Disk Phase

KNN also demonstrates proficiency in identifying attacks, with Precision (87,60 %), Recall (87,60 %), and F-Score (87,60 %) comparable to Na¨ıve Bayes. It delivers a balanced performance with a moderate AUC of 83,90 %. However, KNN's capability in detecting normal features is less effective when compared to Decision Tree and Neural Network.

https://doi.org/10.56294/dm2025699

For the detection of normal features, Logistic Regression offers the highest Precision, Recall, and F-Score. This classifier proves highly adept at accurately pinpointing normal class features but shows less effectiveness in detecting attacks.

Neural Network also exhibits strong performance in normal class detection, boasting high Precision, Recall, and F-Score, while maintaining a balanced performance in attack detection.

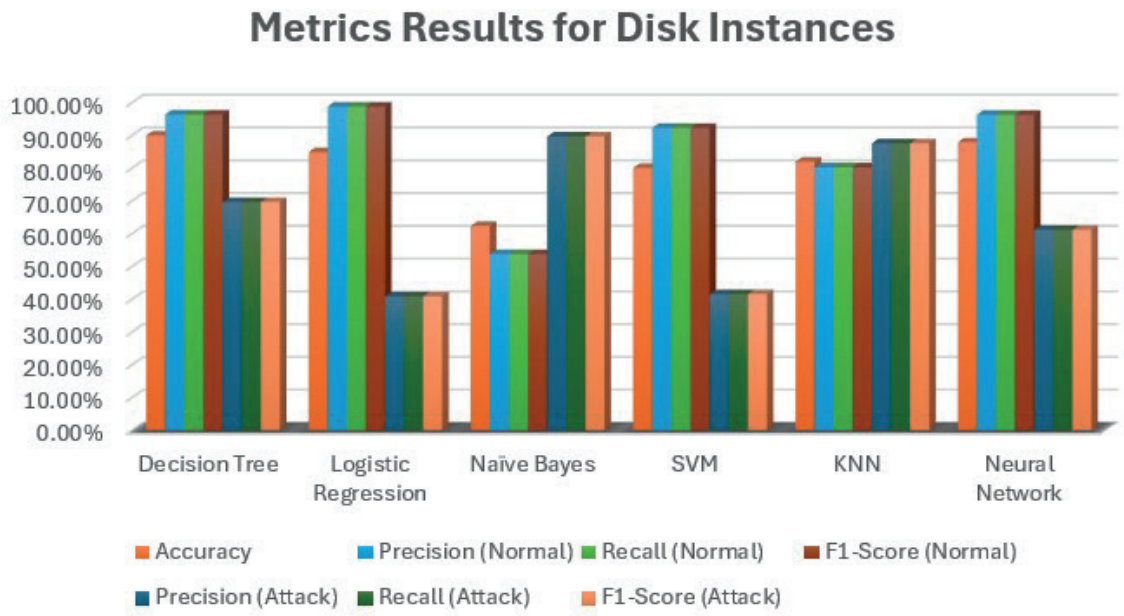

Figure 9. Metrics results for disk instances

Table 9. Classifier performance metrics for disk features

Classifiers	Precision (Normal)	Recall (Normal)	F-Score (Normal)	Precision (Attack)	Recall (Attack)	F-Score (Attack)
Decision Tree	96,40 %	96,40 %	96,40 %	69,70 %	69,70 %	69,70 %
Logistic Regression	98,80 %	98,80 %	98,80 %	40,90 %	40,90 %	40,90 %
Na¨ıve Bayes	53,80 %	53,80 %	53,80 %	89,70 %	89,70 %	89,70 %
SVM	92,30 %	92,30 %	92,30 %	41,60 %	41,60 %	41,60 %
KNN	80,30 %	80,30 %	80,30 %	87,60 %	87,60 %	87,60 %
Neural Network	96,30 %	96,30 %	96,30 %	61,20 %	61,20 %	61,20 %

Comparing our results with previous studies

ML is a key component that improves cloud security by increasing the accuracy of threat detection and incident response capabilities.[41,42,43,44,45,46,47,48,49,50,51] Neural Networks (NNs) have demonstrated considerable improvements in system security by accurately detecting vulnerabilities in cloud environments. Moreover, machine learning based tactics provide proactive threat detection, automated incident handling, and flexible security protocols in dynamic cloud environments, reducing the mean time to detect security breaches and enhancing overall security effectiveness.[52,53,54,55,56,57,58] Compared to other research on cloud security attack detection as mentioned in table 10, our study adopts a more detailed strategy by segmenting the dataset into three groups to do a thorough examination of the most significant characteristics. Weconducted another experiment using all features without any categorization for features and it shows great accuracy rates—typically 99,9 %. So, we look for the precise features that have a major influence on the research's results. The study shows that Decision Trees and Neural Networks are the most effective classifiers for analyzing disk-related attributes by classifying the dataset because of their remarkable attack detection capabilities. Moreover, studies show that Neural Networks are the best choice for standard CPU identification.

The findings emphasize how important it is to use classifiers strategically to increase intrusion detection systems' effectiveness in cloud computing environments. By emphasizing the effectiveness of feature-specific classifiers, this study provides a deeper understanding of how to optimise cloud security mechanisms against different types of attacks.

Table 10. Summary of research results and methods used in cloud security studies

Reference	Results	Methods Used
Mohamed et al.[23]	Machine learning improves cloud-based IoT security; for example, Convolutional Neural Networks achieve 98 % accuracy in threat detection, enhancing cloud system prevention, response, and recovery.	Convolutional Neural Networks, Random Forests, Decision Trees, and Support Vector Machines
Malaiyappan et al.[24]	According to the research study, machine learning improves cloud security by enabling automated incident response, adaptive security measures, and proactive threat detection, with accuracy of 95 %.	Random Forest, Neural Network
Senthilkumar et al.[25]	Using the NSL-KDD dataset and XGBoost feature selection, the research combines Deep Learning and Support Vector Machine for Cloud Computing security, improving threat detection and outperforming alternative techniques.	SVM
Mamidi[26]	By proactively identifying risks, analyzing patterns, forecasting future attacks, and bolstering authentication procedures, AI and machine learning improve cloud security and ultimately reinforce cloud infrastructure against changing cyber threats.	AI-driven techniques for threat detection
Vijayan et al.[27]	In comparison to conventional techniques, the model for cloud security proposed in this research achieves high accuracy (99 %) and F-scores (98 %) in detecting cyber risks.	ACC value of 99 %, F-scores of 98 %
Dhinakaran et al.[28]	High accuracy and threat detection rates can be attained by using advanced machine learning models, such as Random Forest, Deep Neural Networks, and reinforcement learning (Q-learning), to improve data security in cloud computing systems.	Random Forest: 95 %, Deep Neural Network: 97 % accuracy
Khatarkar et al.[29]	The study suggests an Integrated Augmented Intelligence strategy to improve Cloud Network Security by using machine learning for Explainable AI, Real-time Incident Response, and Threat Detection.	A comprehensive comparative analysis is performed against six traditional methods
Nasir et al.[30]	Cloud security is improved by machine learning, specifically the Random-Forest-Classifier model, which achieves 99,88 % accuracy in intrusion detection, outperforming conventional techniques, and effectively handles changing cybersecurity threats.	Random-Forest-Classifier achieved 99,88 % accuracy
Kavitha et al.[31]	In this research, an Artificial Neural Network with simulated bee colony optimization is used to construct a Machine Learning model with 93,8 % accuracy for improving data security in cloud computing.	ANN algorithm with simulated bee colony achieves 93,8 % accuracy
Alla et al.[32]	The study evaluated machine learning (KNN, SVM, DT, and LR) for cloud security anomaly detection, obtaining a high accuracy rate of 96,3 %.	Accuracy= 96,3 %, precision=93,8 %, recall=95,2 %, F-score=95,9 %
This study	The study takes a more granular approach by dividing the dataset into three categories to conduct an in-depth analysis of the most influential features.	DT, LR, Naive Bayas, SVM, KNN, NN

CONCLUSIONS

This study provides a comprehensive comparison of the performance of various machine learning classifiers in detecting attacks across distinct categories of system utilization: CPU, memory, and disk. The results reveal that classifier effectiveness varies significantly depending on the specific system feature under examination. This study underscores the importance of tailoring classifier selection to the specific system feature and detection objective. For instance, Decision Trees and Neural Networks demonstrate exceptional attack detection capabilities in disk feature analysis, while Neural Networks outperform others in typical CPU detection tasks. The findings highlight the critical role of strategically employing classifiers to enhance the efficiency of threat detection systems in cloud computing environments.

BIBLIOGRAPHIC REFERENCES

1. M. I. Ghafoor, M. S. Roomi, M. Aqeel, U. Sadiq, S. U. Bazai, “Multi-features classification of smd screen in smart cities using randomised machine learning algorithms,” 2nd International Informatics and Software Engineering Conference (IISEC), pp. 1–5, 2021. DOI: 10.1109/IISEC54230.2021.9672380

2. S. R. Seelam, M. Shobana, S. V. P. R. Pulagurla, N. A. Kundeti, "Securecloud guardian: Machine learning-driven privilege escalation detection and mitigation for cloud environments," Tech. rep., EasyChair (2024). https://easychair.org/publications/preprint/JHg8

3. G. Ziheng, G. Jiang, "A novel intrusion detection mechanism in cloud computing environments based on artificial neural network and genetic algorithm," Telecommunications and Radio Engineering, pp. 51-64, 2024. DOI: 10.1615/TelecomRadEng.2024048769

4. M. Dawood, S. Tu, C. Xiao, H. Alasmary, M. Waqas, S. U. Rehman, "Cyberattacks and security of cloud computing: a complete guideline," Symmetry, vol. 15, no. 11, 2023. https://doi.org/10.3390/sym15111981

5. S. Ayyub, P. Kaushik, "Secure searchable image encryption in cloud using hyper chaos," The International Arab Journal of Information Technology (IAJIT), vol.16, no. 2, 251-259, 2019.

6. L. Hasimi, D. Zavantis, E. Shakshuki, A. Yasar, "Cloud computing security and deep learning: An ANN approach," Procedia Computer Science, vol. 231, pp. 40-47, 2024. https://doi.org/10.1016/j.procs.2023.12.155

7. T. Beucler, I. Ebert-Uphoff, S. Rasp, M. Pritchard, P. Gentine, "Machine learning for clouds and climate," Clouds and their climatic impacts: Radiation, circulation, and precipitation, pp. 325–345, 2023. https://doi.org/10.1002/9781119700357.ch16

8. D. C. Le, N. Zincir-Heywood, M. I. Heywood, "Analyzing data granularity levels for insider threat detection using machine learning," IEEE Transactions on Network and Service Management, vol. 17, no.1, pp. 30–44, 2020. DOI: 10.1109/TNSM.2020.2967721

9. Q. K. Kadhim, O. F. Alwan, I. Y. Khudhair, "Deep learning methods to prevent various cyberattacks in cloud environment," Revue d'Intelligence Artificielle, vol. 38, no. 3, pp. 893-900, 2024. DOI: https://doi.org/10.18280/ria.380316

10. Sarosh, "Machine learning based hybrid intrusion detection forvirtualized infrastructures in cloud computing environments," Journal of Physics: Conference Series, vol. 2089, pp. 012072, 2021. DOI: 10.1088/1742-6596/2089/1/012072

11. U. Bhatta, "How to integrate cloud service, data analytic and machine learning technique to reduce cyber risks associated with the modern cloud based infrastructure," arXiv preprint arXiv:2405.11601, DOI:10.48550/arXiv.2405.11601

12. AlSaleh, A. Al-Samawi, L. Nissirat, "Novel machine learning approach for ddos cloud detection: Bayesian-based cnn and data fusion enhancements," Sensors, vol. 24, no. 5, 2024. https://doi.org/10.3390/s24051418

13. Bakro, R. R. Kumar, A. Alabrah, Z. Ashraf, M. N. Ahmed, M. Shameem, A. Abdelsalam, "An improved design for a cloud intrusion detection system using hybrid features selection approach with ml classifier," IEEE Access, vol. 11, pp. 64228–64247, 2023. DOI:10.1109/ACCESS.2023.3289405

14. Mayuranathan, S. Saravanan, B. Muthusenthil, A. Samydurai, "An efficient optimal security system for intrusion detection in cloud computing environment using hybrid deep learning technique," Advances in Engineering Software, vol. 173, no. 3, pp. 103236, 2022. DOI:10.1016/j.advengsoft.2022.103236

15. J.-H. Park, S.-H. Na, J.-Y. Park, E.-N. Huh, C.-W. Lee, H.-C. Kim, "A study on cloud forensics and challenges in saas application environment," IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), pp. 734–740, 2016. DOI:10.1109/HPCC-SmartCity-DSS.2016.0107

16. M. Brundage, S. Avin, J. Clark, H. Toner, P. Eckersley, B. Garfinkel, A. Dafoe, P. Scharre, T. Zeitzoff, B. Filar, et al., "The malicious use of artificial intelligence: Forecasting, prevention, and mitigation," arXiv preprint arXiv:1802.07228, 2018 DOI:10.48550/arXiv.1802.07228

17. S. B. Kotsiantis, "Decision trees: a recent overview," Artificial Intelligence Review, vol. 39, pp. 261–283, 2013. DOI:10.1007/s10462-011-9272-4

18. Steinwart, A. Christmann, “Support vector machines,” Springer Science & Business Media, 2008.

19. D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, M. Klein, Logistic regression, Springer, 2002.

20. P. Murphy, et al., “Naive bayes classifiers,” University of British Columbia, vol. 18, no. 60, pp. 1–8, 2006.

21. E. Peterson, “K-nearest neighbor,” Scholarpedia, vol. 4, no. 2, 2009.

22. S. Dreiseitl, L. Ohno-Machado, “Logistic regression and artificial neural network classification models: a methodology review,” Journal of Biomedical Informatics, vol. 35, no. 5-6, pp. 352–359, 2002. https://doi.org/10.1016/S1532-0464(03)00034-0

23. Mohamed, K. Alosman, “A comprehensive machine learning framework for robust security management in cloud-based internet of things systems,” Jurnal Kejuruteraan, vol. 36, no. 3, pp. 1055–1065, 2024. https://doi.org/10.17576/jkukm-2024-36(3)-18

24. N. A. Malaiyappan, S. Prakash, S. V. Bayani, M. Devan, “Enhancing cloud compliance: A machine learning approach,” AIJMR-Advanced International Journal of Multidisciplinary Research, vol. 2, no. 2, 2024. https://doi.org/10.62127/aijmr.2024.v02i02.1036

25. R. Senthilkumar, S. Yasotha, P. Manochithra, J. Senthil, G. Sivakumar, “An efficient investigation of cloud computing security with machine learning algorithm,” International Conference on Inventive Computation Technologies (ICICT), pp. 678–683, 2024. DOI:10.1109/ICICT60155.2024.10544578

26. S. R. Mamidi, The role of AI and machine learning in enhancing cloud security, Journal of Artificial Intelligence General science (JAIGS), vol. 3, no. 1, pp. 403–417, 2024. DOI: https://doi.org/10.60087/jaigs.v3i1.161

27. G. Vijayan, K. Dharun, M. Dhinesh, S. Mahalakshmi, “Establishing cloud security using modern learning approaches,” International Conference on Inventive Computation Technologies (ICICT), pp. 1283–1286, 2024. DOI:10.1109/ICICT60155.2024.10544686

28. Dhinakaran, M. Sundhari, S. Ambika, V. Balaji, R. T. Rajasekaran, “Advanced machine learning techniques for enhancing data security in cloud computing systems,” IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT), Vol. 5, 2024, pp. 1598–1602, 2024.

29. P. Khatarkar, D. P. Singh, A. Sharma, “Machine learning protocols for enhanced cloud network security,” IEEE International Conference on ICT in Business Industry & Government (ICTBIG), pp. 1–6, 2023. DOI:10.1109/ICTBIG59752.2023.10456016

30. H. Nasir, A. Ayaz, S. Nizamani, S. Siraj, S. Iqbal, M. K. Abid, “Cloud computing security via intelligent intrusion detection mechanisms,” International Journal of Information Systems and Computer Technologies, vol. 3, no. 1, pp. 84–92, 2024. DOI: https://doi.org/10.58325/ijisct.003.01.0082

31. S. Kavitha, S. Gadde, R. Thatikonda, S. A. Vaddadi, E. Naresh, P. K. Pareek, “Enhancing data security in cloud computing with optimized feature selection and machine learning for intrusion detection,” 2023. DOI:10.21203/rs.3.rs-3572347/v1

32. R. Alla, G. Thangarasu, “Performance evaluation of evolutionary under sampling and machine learning techniques for network security in cloud environment,” Asia Simulation Conference, pp. 270–278, 2023.

33. S. R. Mamidi, The role of AI and machine learning in enhancing cloud security, vol. 3, no. 1, 2024. (2024). doi:10.60087/jaigs.v3i1.161.

34. J. N. A. Malaiyappan, S. Prakash, S. V. Bayani, M. Devan, Enhancing cloud compliance: A machine learning approach, vol. 2, no. 2, 2024.doi:10.62127/aijmr.2024.v02i02.1036.

35. Al Sharah A, Owida HA, Alnaimat F, Abuowaida S. Application of machine learning in chemical engineering: outlook and perspectives. Int J Artif Intell. 2024 Mar;13(1):619-30. http://doi.org/10.11591/ijai.v13.i1.pp619-630

36. Owida HA, Moh'd BA, Turab N, Al-Nabulsi J, Abuowaida S. The Evolution and Reliability of Machine Learning Techniques for Oncology. International Journal of Online & Biomedical Engineering. 2023 Aug 1;19(8). doi: https://doi.org/10.3991/ijoe.v19i08.39433

37. Alshdaifat, N., Osman, M. A., & Talib, A. Z. (2022). An improved multi-object instance segmentation based on deep learning. Kuwait Journal of Science, 49(2). doi: https://doi.org/10.48129/kjs.10879

38. Abuowaida, S. F. A., Chan, H. Y., Alshdaifat, N. F. F., & Abualigah, L. (2021). A novel instance segmentation algorithm based on improved deep learning algorithm for multi-object images. Jordanian Journal of Computers and Information Technology (JJCIT), 7(01). doi: 10.5455/jjcit.71-1603701313

39. Abuowaida, S. F., & Chan, H. Y. (2020). Improved deep learning architecture for depth estimation from single image. Jordanian Journal of Computers and Information Technology, 6(4). doi: 10.5455/jjcit.71-1593368945

40. Alshdaifat, N., & Rahman, M. N. A. (2024). The effect of technological context on smart home adoption in Jordan. Indonesian Journal of Electrical Engineering and Computer Science, 33(2), 1186-1195., doi: 10.11591/ijeecs.v33.i2.pp1186-1195.

41. Alomoush, A., Alkhawaldeh, A., ALBashtawy, M., Hamaideh, S., Abdelkader, R., Mohammad, K., ... & Al-Qudah, M. (2024). Self-Medication and its Associated Factors among University Students: A Cross-Sectional Study. Iranian Journal of Nursing and Midwifery Research, 29(2), 268-271., doi: 10.4103/ijnmr.ijnmr_123_23

42. Alkhawaldeh, A., Alsaraireh, M., ALBashtawy, M., Rayan, A., Khatatbeh, M., Alshloul, M., ... & Alhroub, N. (2024). Assessment of cognitive impairment and related factors among elderly people in Jordan. Iranian Journal of Nursing and Midwifery Research, 29(1), 120-124., doi: 10.4103/ijnmr.ijnmr_456_22.

43. H. Khafajeh, "Cyberbullying Detection in Social Networks Using Deep Learning", The International Arab Journal of Information Technology (IAJIT) ,Volume 21, Number 06, pp. 1054 - 1063, November 2024, doi: 10.34028/iajit/21/6/9.

44. R. Alazaidah, (2023, December). A Comparative Analysis of Discretization Techniques in Machine Learning. In 2023 24th International Arab Conference on Information Technology (ACIT) (pp. 1-6). IEEE. doi: 10.1109/ACIT58888.2023.10453749.

45. H. A. Owida, J. I. Al-Nabulsi, N. M. Turab, M. Al-Ayyad, R. Alazaidah, & N. Alshdaifat (2025). Progression of polymeric nanostructured fibres for pharmaceutical applications. Bulletin of Electrical Engineering and Informatics, 14(1), 409-420. doi: 10.11591/eei.v14i1.7315.

46. H. M. Turki, E. Al Daoud, G. Samara, R. Alazaidah, M. H. Qasem, M. Aljaidi, ... & N. Alshdaifat. (2025). Arabic fake news detection using hybrid contextual features. International Journal of Electrical & Computer Engineering (2088-8708), 15(1). doi: http://doi.org/10.11591/ijece.v15i1.pp836-845

47. Mohammad, A.A.S., Shelash, S.I., Saber, I.T., Vasudevan, A., Darwazeh, R.N., Almajali, R., & Fei, A. (2025). Internal Audit Governance Factors and their effect on the Risk-Based Auditing Adoption of Commercial Banks in Jordan. Data and Metadata, 4, 464.

48. Mohammad, A.A.S., Al-Hawary, S.I.S., Hindieh, A., Vasudevan, A., Al-Shorman, H.M., Al-Adwan, A.S., Alshurideh, M.T., & Ali, I. (2025). Intelligent Data-Driven Task Offloading Framework for Internet of Vehicles Using Edge Computing and Reinforcement Learning. Data and Metadata, 4, 521.

49. Alhalalmeh, M.I., Al Sarayreh, A., Al-Ayed, S.I., Al-Tit, A.A., Alqahtani, M.M., Hunitie, M.F.A., & Mohammad, A.A.S. (2025). The Impact of Dynamic Capabilities on Entrepreneurial Performance: An Empirical Study of SMEs. In intelligence-driven circular economy: regeneration towards sustainability and social responsibility (pp. 465-479). Cham: Springer Nature Switzerland.

50. Salameh, W.E.M.K.B., Mohammad, A.A.S., Alshurideh, M.T., Alolayyan, M.N., Hunitie, M.F.A., Rababah, M.W., & Al-hawajreh, K.M. (2025). Evaluation of Applying the PDCA Cycle on Medication Administration in the Emergency Departments. In intelligence-driven circular economy: regeneration towards sustainability and social responsibility (pp. 319-330). Cham: Springer Nature Switzerland.

51. Mohammad, A.A.S., Alolayyan, M.N., Al-Daoud, K.I., Al Nammas, Y.M., Vasudevan, A., & Mohammad, S.I. (2024). Association between Social Demographic Factors and Health Literacy in Jordan. Journal of Ecohumanism, 3(7), 2351-2365.

52. Mohammad, A.A.S., Khanfar, I.A., Al-Daoud, K.I., Odeh, M., Mohammad, S.I., & Vasudevan, A. (2024). Impact of perceived brand dimensions on Consumers' Purchase Choices. Journal of Ecohumanism, 3(7), 2341-2350.

53. Mohammad, A.A.S., Khanfa, I.A., Al Oraini, B., Vasudevan, A., Mohammad, S.I., & Ala'a, M. (2024). User acceptance of health information technologies (HIT): an application of the theory of planned behavior. Data and Metadata, 3, 394-394.

54. Mohammad, A.A.S., Khanfar, I.A., Al Oraini, B., Vasudevan, A., Mohammad, S.I., & Fei, Z. (2024). Predictive analytics on artificial intelligence in supply chain optimization. Data and Metadata, 3, 395-395.

55. Mohammad, A.A.S., Alolayyan, M. N., Al-Daoud, K. I., Al Nammas, Y. M., Vasudevan, A., & Mohammad, S. I. (2024). Association between Social Demographic Factors and Health Literacy in Jordan. Journal of Ecohumanism, 3(7), 2351-2365.

56. Mohammad, A.A.S., Khanfar, I. A., Al Oraini, B., Vasudevan, A., Mohammad, S. I., & Fei, Z. (2024). Predictive analytics on artificial intelligence in supply chain optimization. Data and Metadata, 3, 395-395.

57. Mohammad, A.A.S., Alshebel, M., Al Oraini, B., Vasudevan, A., Mohammad, S.I.S., Jiang, H., & Al Sarayreh, A. (2024). Research on Multimodal College English Teaching Model Based on Genetic Algorithm. Data and Metadata, 3, 421.

58. Mohammad, A.A.S., Al-Daoud, K.I., Al Oraini, B., Mohammad, S.I.S., Vasudevan, A., Zhang, J., & Hunitie, M.F.A. (2024). Using Digital Twin Technology to Conduct Dynamic Simulation of Industry-Education Integration. Data and Metadata, 3, 422.

FINANCING

This research is funded by Zarqa University.

CONFLICT OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

AUTHOR CONTRIBUTIONS

Conceptualization: Suhaila Abuowaida, Hamza Abu Owida.
Investigation: Raed Alazaidah, Suleiman Ibrahim Shelash Mohammad.
Methodology: Suhaila Abuowaida and Esraa Abu Elsoud.
Writing - original draft: Asokan Vasudevan, Nawaf Alshdaifat.
Writing - review and editing: Shengqi Liu.