Research Article

Journal of Contemporary Education Theory & Artificial Intelligence

Surveying Security Threat Landscape and Defense Frameworks for Cloud-Based Big Data Systems

Shravan Kumar Rajaram[1*], Avinash Attipalli[2], Siddharth Konkimalla[3], Chandrababu Kuraku[4], Sunil Jacob Enokkaren[5]

[1]Microsoft Inc , Sr Technical Support Engineer,
srajaram@microsoft.com
[2]University of Bridgeport, Department of Computer Science,
Attipalli.avinash@gmail.com
[3]Adobe Inc, Sr Network Development engineer,
Siddharth.konkimalla@gmail.com
[4]Campbellsville University , Chandrababu.kuraku@gmail.com
[5]Sikkim Manipal University, Department of Information Technology,
sunil.jacob.enokkaren@gmail.com

IJCACI and Washington University of Science and Technology (WUST) Conference Proceedings 2025
https://scrs.in/conference/ijcaci2025/page/Best%20Paper%20Award

***Corresponding author:** Shravan Kumar Rajaram

Citation: Shravan Kumar R, Avinash A, Siddharth K, Chandrababu K, Sunil Jacob E (2025) Surveying Security Threat Landscape and Defense Frameworks for Cloud-Based Big Data Systems. J Contemp Edu Theo Artific Intel: JCETAI-118.

Received Date: 05 Sep, 2025; **Accepted Date:** 17 Sep, 2025; **Published Date:** 25 Sep, 2025

Abstract

Technology has made life much easier, and also posed numerous security threats following the rapid changes that have occurred. Due to growing dependence on the Internet, cyberattacks are gaining enormous momentum, and it is inevitable to have highly effective countermeasures. An Intrusion Detection System (IDS) is an important part of information security, a safeguard to suspicious traffic on the network. Most recently, IDS has become one of the applications where Machine Learning (ML) techniques are incorporated to do better at detecting and classifying threats. The current research paper introduces an effective security threat detection framework based on Multi-Layer Perceptron (MLP) model, which is tested and analyzed using the dataset called CSE-CIC-IDS2018, which considers all possible operations. The methodology is to collect the data systematically including preprocessing, normalization of the features, selection of the features, and training of the models. A balanced dataset is created by dividing into training and testing data in a ratio of 70:30 and the model is trained with MLP in a bid to classify the different types of network intrusion. Experimental findings confirm the high accuracy ratio of 98.97% in the proposed model and precision, recall, and F1-score of 99 having much better results than the standard models like KNN and AdaBoost. The model is a well generalized with the characteristics of stability in training and validation stages. This study demonstrates the viability of deep learning-based approaches in increasing cloud security and creates a basis of integrating smart, scalable and flexible intrusion detection in cloud-based big data environments.

Keywords: *Intrusion Detection System (IDS), Cybersecurity, Machine Learning (ML), Multi-Layer Perceptron (MLP), CSE-CIC-IDS2018 Dataset, Feature Selection, Cloud Security, Network Threat Detection.*

I. Introduction

As cloud computing becomes an integral part of modern IT infrastructures, providing flexible and scalable solutions for data storage and processing, it simultaneously presents a growing challenge for cybersecurity. The widespread adoption of cloud services by businesses, governments, and individuals has led to the proliferation of devices connecting to cloud environments, such as smartphones, IoT devices, and workstations. These devices often act as access points to sensitive data, making them prime targets for cyberattacks.

A security assessment is a crucial, on-site analysis to evaluate the current state of security, spot excesses or shortcomings, define the necessary level of protection, and offer suggestions to raise the operation's overall security. The development of harmful software, or malware, presents a significant obstacle to IDS design. It has become more difficult to recognise unfamiliar and obfuscated malware in the increasingly complicated world of hostile attacks because malware writers utilise numerous evasion strategies to conceal information in order to prevent detection by an IDS. In addition, there are now more security threats than ever before, such as zero-day assaults that specifically target internet users. The pervasiveness of information technology has made computer safety an absolute necessity [1].

Hadoop and other big-data technologies are necessary to effectively handle threat intelligence and predict assaults. This method of analyzing real-time security data to provide insights on the latest global threats has the potential to change cloud security from a reactive to a proactive and predictive approach [2].

The biggest security dangers are internal rather than external. Insider threat identification and prediction are critical mitigating approaches [3][4][5]. However, identifying and detecting breaches against intelligent network attacks is difficult due to the vast amount of security data as well as the high probability of false alarms [6].

The most current threat landscape highlights how difficult it is to prevent an assault & breach of security since attackers may exploit flaws in people, procedures, and technology. Cyber criminals have evolved their strategies, techniques, and processes to the point that they are highly difficult to detect and challenging to investigate and fix [7] [8].

The advanced technologies used during these solutions include Artificial Intelligence (AI), machine learning, and encryption methods which are at the center stage of enhancing digital defense. Such application as AI-driven security tools allow spotting regularities and anomalies in incredibly large amounts of information and prevent the worsening of the situation to the scale of a serious breach [9][10]. Meanwhile, the encryption technologies that become available make the data secure when transferred or being in the resting state in the clouds. In discussing these developments, it is quite clear that these defense measures must be quite potent to protect the digital infrastructures, and not forgetting the need to make a conscious effort that will help to adopt strategies that are adaptive regarding the dynamic threat environment.

ML An area of study within computer science. Knowledge in learning ML, a branch of artificial intelligence concerned with pattern recognition as well as computational learning theory, has grown and matured. Investigate Computer Science for Humans. It addresses issues surrounding the building and research of algorithms that can learn and predict from data [11]. In contrast to purely static program instructions, such algorithms are constructed to be able to predict or make a decision based on data. The process involves building a model with feedback from experts. Identifying network events as either normal or attack events which might include U2R, R2L attacks, Denial of Service (DOS), as well as Probe is a multinomial type classification issue for intrusion detection models.

A. Motivation and Contribution of Paper

Cloud computing and big data's increased use and its usage in almost all industries created a paramount focus on the security of cloud-based infrastructures. Potent cyber threats, like DDoS, brute-force, and input attacks, often disrupt the work of such environments extensively and threaten the security of sensitive information. The problem is that traditional ML models are not scalable enough and they may have limitations when it comes to learning components, so they do not effectively detect and respond to such emerging threats. This calls for better intelligent, adaptive, and precise IDS to be developed. It is driven by this necessity that the current research paper examines the implementation of a DL-based solution based on an MLP model to improve on the detection of threats within a cloud-based big data system to achieve a more consistent and expansive defensive strategy. The study contributes in the following way:

- The study was initiated by collecting the comprehensive CSE-CIC-IDS2018 dataset from Kaggle, providing a robust foundation of the security threat landscape in cloud systems.
- Implementing robust data preparation techniques, including duplicate removal, normalization, and parallel processing to enhance data quality and model efficiency. A proper feature extraction combined with proper feature selection to narrow down features.
- Developing and evaluating an MLP (Multilayer Perceptron) model for accurate classification of network intrusions.
- Accuracy, precision, recall, F1-score, as well as loss are just a few of the measures used to provide a thorough performance review.

B. Novelty & Justification of the Study

This study is justified by the growing number and sophistication of security threats that are targeted at cloud big data systems and applications, necessitating smarter and scalable methods of detecting it beyond present traditional approaches. Although the current models like K-Nearest Neighbors (KNN) and AdaBoost may lack the flexibility to adapt to changing trends of attacks and high dimensions, this study proposes a DL-based model that uses an MLP model in disaster clouds with high-dimensionality. The main distinctive feature of the work is the end-to-end methodological protocol, with comprehensive preparation of the data, selection of normative solutions and model training aimed at effective threat detection. The proposed model adopts the state-of-the-art learning capabilities in a nonlinear relationship of the MLP in order to affect a high detection rate with generous generalization and provides a scalable resilient infrastructure in real time intrusion detection within dynamic cloud infrastructures.

C. Structure of paper

The paper's outline is as follows: Section II examines relevant research on cloud-based systems' danger environment. Section III explains the suggested methodology, including the deployment of the model and dataset. Section IV provides crucial insights and experimental outcomes. Section V finishes with the study's shortcomings and future research goals.

II. Literature Review

This literature analysis provides a comprehensive evaluation of current improvements in the security threat environment in cloud systems. Table I summarizes the examined studies, detailing the methodology used, performance outcomes, important findings, identified limitations, and proposed paths for future study.

Saad et al. (2019) work helps keep cloud networks safe and guards against harmful intrusions. Bidirectional long short-term memory is used in this work to detect incidents across cloud-based unified threat management systems. The results are contrasted with a baseline method called K-nearest neighbor. The UTM platform is used to acquire time series input samples for both testing as well as training. With an MSE of 0.0186, KNN achieves an accuracy score of 98.47%, but BLSTM outperforms KNN with an accuracy score of 98.6% with a loss of 0.002 [12].

Torkura et al. (2018) use the principles of chaos engineering by incorporating Broker Monkey, a component that continuously causes failures in their reference CSB system, Cloud RAID. Thus, CSB Auditor's effectiveness that is, its capacity to identify the modifications made by Broker Monkey is constantly evaluated. By using the Common Configuration Scoring System, vulnerability scores for severity are determined, CSB Auditor uses security metrics for risk analysis, eliminating the drawback of inadequate security metrics in current cloud auditing systems. Numerous techniques, including failure injection strategies based on chaotic engineering, have been used to test CSB Auditor. Their experimental assessment It demonstrates a 96% detection as well as recovery rate, confirming the effectiveness of their strategy against the previously identified security concerns [13].

Salman et al. (2017) Instead of only finding anomalies, as is often the case in modern research, look into both detecting and classifying abnormalities. They have developed and tested learning models for the detection and classification of various assaults using a widely used publicly accessible dataset. Specifically, they have employed two supervised ML methods: random forest (RF) and linear regression (LR). They demonstrate how similarities across assaults might lead to less accurate categorization even in cases where detection is flawless. Although they are unable to classify certain assaults, their results show above 99% detection accuracy and 93.6% categorization accuracy. Additionally, they contend that the same ML methods may be used to apply this classification to multi-cloud setups [14].

Jiao et al. (2017) also offer a real-time TCP-based DDoS detection technique that extracts valuable information from TCP traffic and then using two decision- tree classifiers to differentiate malicious data from valid traffic. They employ a variety of actual and simulated datasets, including the CAIDA DDoS Attack 2007 dataset, the ISCX IDS dataset, as well as a Baidu Cloud Computing Platform dataset, to evaluate the proposed technique. Experiment results show that the proposed technique has a low false alarm rate (less than 1%), and a high assault detection rate (more than 99%). The victim-end DDoS defines system at Baidu's cloud computing data centre will employ this tactic [15].

Nugroho, Wahyudin and Cahyana (2017) suggested utilizing a cloud-based PHP learning system to assist students in the eleventh grade with dynamic web development courses. With the help of their system, teachers may control learning resources including exercises and materials, users, and source code. The learning component that offers PHP course materials, exercises, and code practical is available to the students. With an average score of 81.53%, 82.59%, as well as 78.13%, respectively, from 21 students, three content reviewers, and four system reviewers, they might use the LORI 1.5 instrument as well as their proposed PHP learning methods to conduct an experiment and determine that their approach is suitable. As a result, the suggested method is regarded as good and acceptable in reviews conducted by students, content reviewers, or system reviewers [16].

Nikolai and Wang (2016) In an IaaS cloud computing surroundings, a defines-in-depth approach is provided by several distributed intrusion detection and prevention system sensors. To assist cloud providers with various safety measures, they propose and introduce a streaming cloud intrusion monitoring and classification system (SCIMCS), which categorizes known attacks and filters out noise signals. They use a three-step process: classify attacks, detect anomalies, and then summarize and score. In an IaaS cloud environment running Eucalyptus, they perform genuine assaults to show off the efficacy of their system, achieving a 95.9% overall alert reduction and a zero-miss rate for problematic alerts. Furthermore, for trained assaults, they exhibit a 100% classification rate [17].

Tong et al. (2016) After analyzing and describing the features of bug-induced failures logs from bug repository as well as Q&A websites, a general automated technique for recognizing bug-induced failure logs from logfiles on open cloud platforms is presented. MPIN and SPIN are the two log classification techniques presented in the approach.They test their methodology using logs gathered from OpenStack and Hadoop bug repositories as well as five Q&A websites. The results of the tests show that the recommended approach has an accuracy of 83.9% in identifying bug-induced failure reports in OpenStack logs and in Hadoop logs with an accuracy of 82.52% [18].

Table I provides a summary of the literature on threats to cloud security, reviewing many different detection models and frameworks, noting high detection rates but limitation in the scope of attacks, models and scalability of deployment to an enterprise environment.

Table 1: Summary of literature Overview and Review on Security Threat Landscape in Cloud Systems.

Study	Methodology	Dataset	Key Findings	Limitations	Future Direction
Saad et al. (2019)	Comparing BLSTM and KNN for cloud UTM platform incident detection	Time series input from UTM	BLSTM achieved 0.002 loss and 98.6% accuracy; KNN achieved 98.47% accuracy and 0.0186 MSE	Limited model comparison (only KNN baseline); lacks deployment details	Integrate other DL models and real-time deployment scenarios
Torkura et al. (2018)	Chaos engineering with BrokerMonkey and CSBAuditor for cloud risk analysis	CloudRAID reference system	Achieved >96% detection and recovery using severity scoring via Common Configuration Scoring System	Limited to specific cloud architecture and simulated faults	Apply method to real-world enterprise-scale cloud setups
Salman et al. (2017)	LR and RF anomaly detection and classification	Public attack dataset	>99% detection, 93.6% categorization accuracy; highlights challenge in differentiating similar attacks	Struggles with categorization due to attack similarities	Explore ensemble learning and feature engineering for improved categorization
Jiao et al. (2017)	TCP-based DDoS detection using two decision trees	ISCX IDS, CAIDA 2007, Baidu datasets	>99% detection rate with <1% false alarm	Model limited to TCP-based attacks	Extend to UDP/ICMP and hybrid DDoS detection with DL
Nugroho et al. (2017)	Cloud-based PHP learning system for school-level programming	Feedback via LORI 1.5 instrument	Average score: 81.53% (students), 82.59% (content), 78.13% (system) – overall acceptable	Specific to one programming course and limited school-level evaluation	Broaden system to support more programming languages and grade levels

Nikolai and Wang (2016)	Streaming intrusion monitoring (SCIMCS) in IaaS with alert filtering and classification	Eucalyptus-based cloud environment	95.9% alert reduction, 100% classification, 0% miss rate	Real attacks but limited cloud infrastructure scale	Extend to hybrid/multi-cloud environments and real-time response
Tong et al. (2016)	Log failure classification using MPIN and SPIN algorithms	Logs from OpenStack, Hadoop, Q&A sites	Precision: 83.9% (OpenStack), 82.52% (Hadoop)	Limited precision; lacks recall analysis	Improve algorithm accuracy and apply to microservices-based cloud platforms

III. Methodology

The approach used for risk and security threat evaluation and defense frameworks for cloud-based big data environments followed a similar methodology involving data collection, feature selection, pre-processing as well as model evaluation. First, the CSE-CIC-IDS2018 dataset was collected via Kaggle to provide a good amount of data for realistic cloud network traffic. Cleaning procedures include eliminating duplicate and superfluous items and using feature normalization to guarantee consistent scaling are part of the data preparation step. The most pertinent attributes are then extracted for threat categorization using feature selection approaches. After the dataset has been refined, it is divided into training (70%) as well as testing (30%) sets. To detect possible security risks, an MLP model is used for categorization. Standard criteria, F1-score, precision, accuracy, and recall are among the metrics used to methodically assess the model's performance and determine its efficacy. The entire workflow is illustrated in Figure. 1, which presents the flowchart diagram of the security threat identification process in cloud systems.

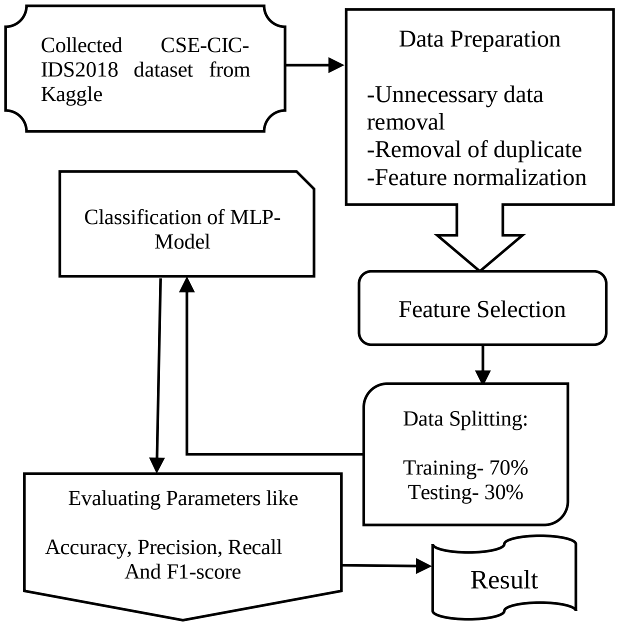
Collected CSE-CIC-IDS2018 dataset from Kaggle
Data Preparation
-Unnecessary data removal
-Removal of duplicate
-Feature normalization
Classification of MLP-Model
Feature Selection
Data Splitting:
Training- 70%
Testing- 30%
Evaluating Parameters like
Accuracy, Precision, Recall And F1-score
Result

Fig. 1. Flowchart Diagram of the Security Threat in Cloud Systems.

This is a thorough, sequential explanation of the procedures shown in the flowchart.

D. Data Collection

The C dataset was used in this investigation. In 2018, CSE and CIC proposed the CSE-CIC-IDS2018 dataset as a collaborative endeavour. The dataset may be found on Kaggle. 10 CSV files, each representing 10 days of the network flow that was recorded, and over 16.2 million samples make up the dataset. Moreover, the CIC Flow Meter program retrieved over 80 characteristics. Six main forms of intrusion assaults are included in this dataset: brute force, bot, infiltration, DoS, distributed denial of service (DDoS), as well as online attacks.

E. Data Preparation

The datasets must be thoroughly pre-processed to eliminate inconsistencies, irrelevancies, and missing values that can negatively impact the performance of security threat detection in cloud environments. Several crucial activities are usually included in this pre-processing phase, such as feature normalization, addressing missing values, and eliminating redundant and superfluous data, as well as implementation of parallel processing techniques to enhance computational efficiency.

- **Unnecessary data removal:** This method strips the dataset of security event data that should be examined by a system's data analysis module to identify cyberattacks of any extraneous information. We refer to the portion of security event data that is not useful for detection as superfluous data. When such material is eliminated from the dataset, its size is decreased, which shortens processing time.
- **Removal of duplicates:** This strategy eliminates information that doesn't aid in the attack detection process. Duplicate records, or entries referring to the same action within a specific period, are likewise abundant in the gathered data.
- **Feature Normalization:** Normalization is necessary to eliminate the impact of the original feature value scales on the numerical characteristics [19]. Every feature is normalized according to Equation (1):

$$z_i = \frac{x_i - min(x)}{max(x) - min(x')} \quad (1)$$

F. Feature Selection

The process of choosing a subset of significant attributes from the original dataset is known as feature selection, eliminating irrelevant or redundant attributes to enhance classification performance and reduce memory storage. It mitigates the curse of dimensionality, decreases computational complexity, and improves learning accuracy. Supervised feature selection methods are broadly classified into wrapper, filter, and embedded models. A widely used filter method is Information Gain, which evaluates attribute significance based on entropy concerning the target class.

G. Data Splitting

Data splitting is essential in ML to evaluate model performance. 30% of the data is used for testing, while 70% is used for training., the 70:30 ratio in this study guarantees that the model learns efficiently while permitting objective evaluation of unseen data.

H. Classification of MLP Model

The suggested MLP is a feed-forward ANN extension technique that maps the input pictures to a classification. The back-propagation technique is used to map the characteristics of the datasets used for testing and training. Nodes are constructed as directed graphs by the MLP and subsequently linked. The graph's nodes each have a quasi-activation function of their own. Furthermore, the MLP datasets were trained using supervised learning techniques, which are also used for the classification of non-linear data. A stochastic fitness function is used to overcome the challenges.

An artificial neural network classifier has been utilized to perform a binary classification to distinguish between pixels with cloudy and clear skies. To categorize regions of interest, artificial neural network (ANN) algorithms employ a technique that mimics the human brain's comprehension, learning, problem-solving, and decision-making processes. In its broadest sense, a neural network is a device created to simulate how the brain carries out a certain activity or function of interest [20]. There are three components to the ANN structure. The number of nodes that comprise the input layer the first layer is determined by the input parameters.

The last layer is called the output layer, as well as the intended output determines how many nodes it has. "Hidden layer" describes the layer or layers that are located between the input & output layers. Most ANNs use hidden layers that perform certain non-linear data processing tasks [21]. The bias b_k produces either a positive or negative net effect with regards to the activation function input [22]. According to Equations (2) and (3), a single neuron k may be described by expressing the following two formulas:

$$u_k = \sum_{i=1} w_{ki} x_i \qquad (2)$$

$$y_k = \varphi(u_k + b_k) \qquad (3)$$

In this context $x_1, \dots, x_n$ denote the input signals, $w_{k1}, \dots, w_{kn}$ denote the synaptic weights of neuron k, u_k denotes the output of the linear combiner as a result of the input signals, b_k denotes the bias, $\varphi(\cdot)$ denotes the activation function, as well as y_k is the neuron's output signal.

I. *Evaluation Parameters*

The efficiency of the MLP-based security threat detection model was assessed using different performance measures like precision, recall, F1-score, as well as accuracy. These indications are crucial for confirming that the model can accurately detect linkages and for spotting security risks in cloud computing. To achieve consistent and stable output, the model was tested by the CSE-CIC-IDS2018dataset. The next part comments on the parameters of the confusion matrix:

- **True Positives (TP):** The number of legitimate security risks detected. These are the cases when the model can sense an actual intrusion or malicious activity in the cloud environment correctly.
- **True Negatives (TN):** The number of benign activities that are correctly classified. The benign activities in the cloud system are normal, non-malicious cloud operations that the model categorizes as benign.
- **False Positives (FP):** The number of benign activities that the model incorrectly classified as a threat. The false positives could cause unnecessary alarms or alerts, as well as direct resources to the false positives which may disturb normal cloud operations.
- **False Negatives (FN):** The number of actual risks categorized as benign. These errors are the most serious. They represent a security issue that cannot be seen, affecting security and integrity of the cloud system.

The metrics of evaluation given to single classes were calculated with the help of conventional formulas that are usually used in classification procedures:

1) Accuracy

The total percentage of correctly classifying cases as compared to the total cases. It also implies overall performance of model, but could be less indicative in case of unbalanced data. It can be more formally defined as in Equation. (4):

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \qquad (4)$$

2) Precision

The proportion between the number of security risks successfully anticipated and the total number of forecasted threats. High precision indicates that the model produces fewer false alarms, which is critical in avoiding unnecessary response actions in cloud environments. Precision is determined using the following formula (5):

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

3) Recall

The proportion of real security risks that the model accurately detected. Maintaining cloud security requires the detection of the majority of genuine threats, which is ensured by high recall. Mathematically, we define it as in Equation. (6):

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

4) F1-Score

The harmonic mean of recall & accuracy is used to gauge how well the model performs in threat detection. When there is an uneven distribution of benign as well as serious instances, it is very beneficial, helping to assess the trade-off between missing threats and raising false alerts and recall, and it is calculated as demonstrated below in Equation. (7):

$$\text{F1} - \text{Score} = 2 \times \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \qquad (7)$$

5) Loss

Loss in the cloud security threat landscape is the measurable or perceived damage financial, reputational, operational, or data-related that results from unauthorized access, data breaches, service disruptions, or other security incidents affecting cloud systems.

IV. Results Analysis and Discussions

Windows 10, a graphics processing unit (GPU), a DL framework (PyTorch), and the programming language (python 3.10) make up the experimental environment described in this article. This configuration allowed the suggested model to run in an efficient and stable environment. The suggested MLP model demonstrated its excellent capabilities in handling the growing security threat environment in cloud systems by achieving an exceptional classification accuracy of 98.97% using the well-recognised CSE-CIC-IDS2018 dataset. The MLP model's performance measures, as shown in Table II, demonstrate its high classification capability. A measure of the accuracy of this model stood at 98.97 percent, implying high chances that the predictions were correct. In addition, it achieved a precision, recall, as well as F 1 -F1-score of 99 percent, which represents a well-balanced performance, in terms of all the evaluation parameters. These findings indicate that the MLP model can be very effective in terms of cutting down FP incidences as well as FN incidences, and at the same time, able to identify positive incidences accurately, making it a very efficient option to use in the classification exercise in the considered context.

Table 2: Results of MLP Model for Security Threat Landscape in Cloud Systems.

Metrics	MLP
Accuracy	98.97
Precision	99
Recall	99
F1-Score	99

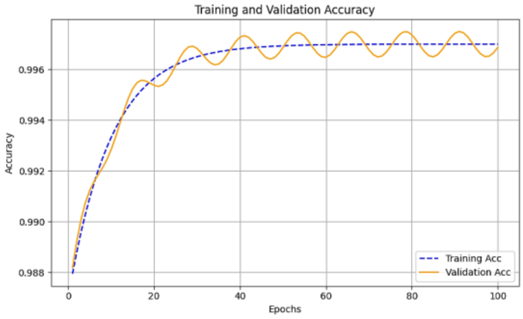

Fig. 2. Training and Validation Accuracy Analysis of MLP Model.

Figure 2 shows the validation and training accuracy of MLP model according to 100 epochs. The training precision, shown by a blue dashed line, increases steadily and monotonic, reaching its plateau value of slightly over 99.7, which is a sign that the model has learned well the training data. The validation accuracy, is also shown in orange colored solid line and has a similar increasing trend with a high level being achieved but with small variation over the epoch periods. These oscillations in the validation accuracy suggest minor variations in generalization performance across epochs, but overall, it remains closely aligned with the training accuracy.

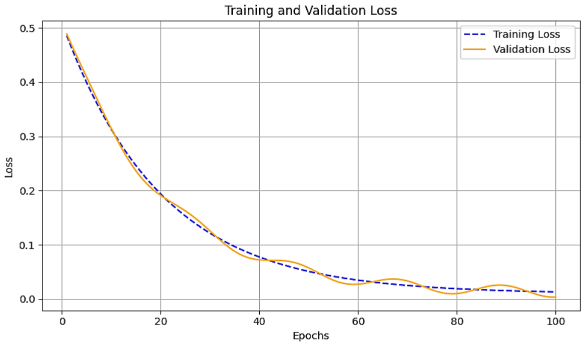

Fig. 3. Training and Validation Loss Analysis of MLP Approach Under CSE-CIC-IDS2018 Dataset.

Figure 3 illustrates the validation and training loss of the MLP model over 100 epochs. Both the training loss (blue dashed line) and validation loss (orange solid line) depict a continuous and stable decreasing trend ultimately suggesting evidence of learning and improved model performance. In the beginning, both losses drop significantly, which shows how fast learning is in the early epochs. The model demonstrated robust and efficient training based on the loss curves, and it achieved low error rates on both the training as well as validation datasets.

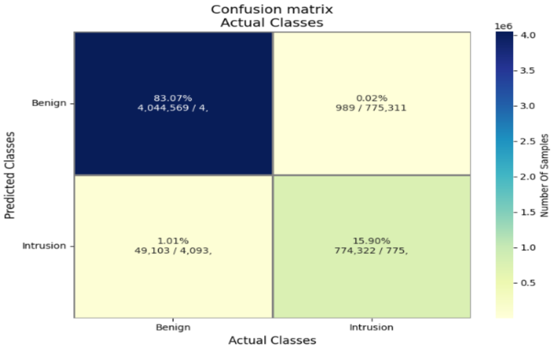

Fig. 4. Confusion Matrix of MLP Model.

A binary classification model's capacity to differentiate between "Benign" as well as "intrusion" classes is evaluated in Figure. 4. The model correctly identified 83.07% of benign cases but misclassified a tiny 0.02% as intrusions. For intrusion cases, it correctly identified 15.90%, while incorrectly labelling 1.01% as benign. The color intensity reflects the number of samples in each category. Overall, the model exhibits high accuracy (98.97%) and a low error rate (1.03%), as indicated by various performance metrics.

J. Comparative Analysis

Table III highlights the performance comparison of existing models used for detecting security threats in cloud systems. The KNN methods attain a performance of 86 per cent, which implies that it has limited efficacy in detecting threats on a measure of proximity. AdaBoost used as mentioned in performs much better with 95.6% accuracy as its metric because it uses the ensemble learning technique to amplify the weak learners. Among them, the MLP model is the most capable of learning complex nonlinear classes and successfully identifying advanced cloud security threats since it shows the highest accuracy of 98.97%.

Table 3: Existing Models Performance for Security Threat in Cloud Systems

Models	Accuracy
KNN [23]	86
AdaBoost [24]	95.6
MLP	98.97

The model of MLP proposed gives a higher accuracy of 98.97 percent in security threat detection across cloud-based systems and is much more superior than conventional models like that of KNN and AdaBoost. This great performance can be explained by the fact that there is a possibility of the MLP being able to capture non-linear and complex relationships in high dimensional feature spaces, leading to better classification of threats. In addition, the model shows strong generalization and training convergence speed, which makes it best suited to dynamic and real-time application in cloud environments in terms of intrusion detection. The benefit of applying MLP to the system is its flexibility, capability of being scaled to deal with emerging patterns of attacks and its possible incorporation into automated cloud security systems.

V. Conclusion and Future Work

Cloud systems operate with an ever-changing environment of security threats, include data breaches, advanced cyber-attacks, as well as exposure to vulnerabilities in common infrastructure, which requires constant improvements to the defence techniques. It is a good investigation on how to recognize and assess security risks of big data stored in clouds based on the CSE-CIC-IDS2018 dataset. The recommended MLP model proved to be more efficient than the compared ones as it attained an excellent figure of 98.97% in the threat classification task and 99%. The good classification is explained by good pre-processing methods, feature selection methods, and generalization of the nonlinear complicated relationships present in the data by the MLP. The MLP model demonstrated greater generalization and faster convergence than the traditional models like KNN and AdaBoost, and had lower misclassification rates as well, this makes it a promising candidate for use in real-time threat detection in ever-changing cloud environments. These findings validate the idea that the proposed model presents an opportunity to make the cloud security models stronger by anticipating several cyberattacks, including DDoS, brute, and injection, among others.

Future plans will include an addition of more advanced DL models like Recurrent Neural Network, to enhance the accuracy of detection further and accommodate lesser computational burden. Additionally, hybrid ensemble models combining MLP with attention mechanisms or Transformer-based approaches may enhance the detection of sophisticated attack vectors. Real-time deployment and validation of the model on live cloud infrastructure will be explored to evaluate its practical applicability. Furthermore, incorporating adaptive learning techniques to address concept drift and evolving threat patterns will be a critical area of research.

References

1. A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, “Survey of intrusion detection systems: techniques, datasets and challenges,” *Cybersecurity*, 2019, doi: 10.1186/s42400-019-0038-7.
2. T. Tene and D. P. C. S. R, “Real-Time Threat Prediction for Cloud Based Assets Using Big-Data Analytics,” *Res. gate*, vol. 4, no. 7, pp. 70–76, 2015.
3. I. A. Gheyas and A. E. Abdallah, “Detection and prediction of insider threats to cyber security: a systematic literature review and meta-analysis,” *Big Data Anal.*, 2016, doi: 10.1186/s41044-016-0006-0.
4. V. Kolluri, “A Pioneering Approach To Forensic Insights: Utilization Ai for Cybersecurity Incident Investigations,” *Int. J. Res. Anal. Rev. (IJRAR*, vol. 3, no. 3, 2016.
5. N. Hubballi and V. Suryanarayanan, “False alarm minimization techniques in signature-based intrusion detection systems: A survey,” *Comput. Commun.*, vol. 49, pp. 1–17, Aug. 2014, doi: 10.1016/j.comcom.2014.04.012.
6. M. Sahrom Abu, S. Rahayu Selamat, A. Ariffin, and R. Yusof, “Cyber Threat Intelligence – Issue and Challenges,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 10, no. 1, p. 371, Apr. 2018, doi: 10.11591/ijeecs.v10.i1.pp371-379.
7. S. Singamsetty, “Fuzzy-Optimized Lightweight Cyber-Attack Detection For Secure Edge-Based Iot,” vol. 6, no. 07, pp. 1028–1033, 2019, doi: 10.53555/jcr.v6.
8. S. Garg, “Predictive Analytics and Auto Remediation using Artificial Inteligence and Machine learning in Cloud Computing Operations,” *Int. J. Innov. Res. Eng. Multidiscip. Phys. Sci.*, vol. 7, no. 2, 2019.
9. Y. Hamid, M. Sugumaran, and V. Balasaraswathi, “IDS Using Machine Learning - Current State of Art and Future Directions,” *Br. J. Appl. Sci. Technol.*, vol. 15, no. 3, pp. 1–22, Jan. 2016, doi: 10.9734/BJAST/2016/23668.
10. M. M. Saad, T. Iqbal, H. Ali, M. F. Bulbul, S. Khan, and C. Tanougast, “Incident Detection over Unified Threat Management Platform on a Cloud Network,” in *Proceedings of the 2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019*, 2019. doi: 10.1109/IDAACS.2019.8924299.
11. K. A. Torkura, M. I. H. Sukmana, T. Strauss, H. Graupner, F. Cheng, and C. Meinel, “CSBAuditor: Proactive security risk analysis for cloud storage broker systems,” in *NCA 2018 - 2018 IEEE 17th International Symposium on Network Computing and Applications*, 2018. doi: 10.1109/NCA.2018.8548329.
12. T. Salman, D. Bhamare, A. Erbad, R. Jain, and M. Samaka, “Machine Learning for Anomaly Detection and Categorization in Multi-Cloud Environments,” in *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*, IEEE, Jun. 2017, pp. 97–103. doi: 10.1109/CSCloud.2017.15.
13. J. Jiao *et al.*, “Detecting TCP-based DDoS attacks in Baidu cloud computing data centers,” in *Proceedings of the IEEE Symposium on Reliable Distributed Systems*, 2017. doi: 10.1109/SRDS.2017.37.
14. E. P. Nugroho, Wahyudin, and R. Cahyana, “A development of cloud-based PHP learning system,” in *2017 3rd International Conference on Science in Information Technology (ICSITech)*, 2017, pp. 674–680. doi: 10.1109/ICSITech.2017.8257198.
15. J. Nikolai and Y. Wang, “A streaming intrusion monitoring and classification system for iaas cloud,” in *IEEE International Conference on Cloud Computing, CLOUD*, 2016. doi: 10.1109/CLOUD.2016.87.
16. J. Tong, Y. Li, H. Tang, and Z. Wu, “An approach to pinpointing bug-induced failure in logs for open cloud platforms,” in *IEEE International Conference on Cloud Computing, CLOUD*, 2016. doi: 10.1109/CLOUD.2016.45.
17. F. Farahnakian and J. Heikkonen, “A deep auto-encoder based approach for intrusion detection system,” in *International Conference on Advanced Communication Technology, ICACT*, 2018. doi: 10.23919/ICACT.2018.8323688.
18. CM Bishop, *Neural networks for pattern recognition*. 1995.
19. D. Lafont, O. Jourdan, and B. Guillemet, “Mesoscale cloud pattern classification over ocean with a neural network using a new index of cloud variability,” *Int. J. Remote Sens.*, 2006, doi: 10.1080/01431160500192512.
20. A. Taravat, S. Proud, S. Peronaci, F. Del Frate, and N. Oppelt, “Multilayer perceptron neural networks model for meteosat second generation SEVIRI daytime cloud masking,” *Remote Sens.*, 2015, doi: 10.3390/rs70201529.
21. F. Ullah *et al.*, “Cyber security threats detection in internet of things using deep learning approach,” *IEEE Access*, 2019, doi: 10.1109/ACCESS.2019.2937347.
22. F. S. De Lima Filho, F. A. F. Silveira, A. De Medeiros Brito Junior, G. Vargas-Solar, and L. F. Silveira, “Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning,” *Secur. Commun. Networks*, vol. 2019, 2019, doi: 10.1155/2019/1574749.

23. *Chundru, S. K., Vangala, S. R., Polam, R. M., Kamarthapu, B., Kakani, A. B., & Nandiraju, S. K. K. (2024). A Machine Learning-Based Framework for Predicting and Improving Student Outcomes Using Big Educational Data (Approved by ICITET 2024 Conference Proceedings). Available at SSRN 5315635.*
24. *Nandiraju, S. K. K., Chundru, S. K., Vangala, S. R., Polam, R. M., Kamarthapu, B., & Kakani, A. B. (2025). Towards Early Forecast of Diabetes Mellitus via Machine Learning Systems in Healthcare. European Journal of Technology, 9(1), 35-50.*
25. Krutthika H. K. & Rajashekhara R. (2019). Network-on-chip: A survey on router design and algorithms. *International Journal of Recent Technology and Engineering, 7*(6), 1687–1691. https://doi.org/10.35940/ijrte.F2131.037619
26. *Chalasani, R., Gangineni, V. N., Pabbineedi, S., Penmetsa, M., Bhumireddy, J. R., & Tyagadurgam, M. S. V. (2025). Big Data-Driven Approach for Lung Cancer Identification via Advanced Deep Transfer Learning Models. European Journal of Technology, 9(1), 51-67.*
27. *Vattikonda, N., Gupta, A. K., Polu, A. R., Narra, B., Buddula, D. V. K. R., & Patchipulusu, H. H. S. (2024). Machine Learning-Based Approaches for Detecting and Mitigating Distributed Denial of Service (DDoS) Attacks to Improved Cloud Security. European Journal of Technology, 8(6), 28-48.*
28. Krutthika H. K. & A.R. Aswatha. (2020). FPGA-based design and architecture of network-on-chip router for efficient data propagation. *IIOAB Journal, 11*(S2), 7–25.
29. *Polu, A. R., Narra, B., Buddula, D. V. K. R., Hara, H., Patchipulusu, S., Vattikonda, N., & Gupta, A. K. Analyzing the Role of Analytics in Insurance Risk Management: A Systematic Review of Process Improvement and Business Agility.*
30. *Madhura, R., Varshitha, P., Nikitha, S., Niveditha, K. M., & Bhat, M. (2024, December). RTL design of 16-bit RISC Processor Using Vedic Mathematics. In 2024 IEEE 33rd Asian Test Symposium (ATS) (pp. 1-4). IEEE.*
31. Krutthika H. K. & A.R. Aswatha (2020). Design of efficient FSM-based 3D network-on-chip architecture. *International Journal of Engineering Trends and Technology, 68*(10), 67–73. https://doi.org/10.14445/22315381/IJETT-V68I10P212
32. *Harinandan, R., Kumar, M., Vamshi, P., Padma, C. R., Krishnappa, K. H., & Raghunandan, J. R. (2024, August). Design and Development of a Real-time Monitoring System for ACL Injury Prevention. In 2024 2nd International Conference on Networking, Embedded and Wireless Systems (ICNEWS) (pp. 1-6). IEEE.*
33. *Krishnappa, K. H. (2024). Traffic pattern analysis for malicious node detection in NoC design. Journal of Communications, 9, 12.*
34. *Mukund Sai Vikram Tyagadurgam, Venkataswamy Naidu Gangineni, Sriram Pabbineedi, Mitra Penmetsa, Jayakeshav Reddy Bhumireddy, et al. (2024) AI-Powered Cybersecurity Risk Scoring for Financial Institutions Using Machine Learning Techniques. Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-482. DOI: doi.org/10.47363/JAICC/2024(3)452.*
35. HK, K. (2020). Design of Efficient FSM Based 3D Network on Chip Architecture. *INTERNATIONAL JOURNAL OF ENGINEERING, 68*(10), 67-73.
36. Krutthika, H. K. (2019, October). Modeling of Data Delivery Modes of Next Generation SOC-NOC Router. In *2019 Global Conference for Advancement in Technology (GCAT)* (pp. 1-6). IEEE.
37. Ajay, S., Satya Sai Krishna Mohan G, Rao, S. S., Shaunak, S. B., Krutthika, H. K., Ananda, Y. R., & Jose, J. (2018). Source Hotspot Management in a Mesh Network on Chip. In *VDAT* (pp. 619-630).
38. Nair, T. R., & Krutthika, H. K. (2010). An Architectural Approach for Decoding and Distributing Functions in FPUs in a Functional Processor System. *arXiv preprint arXiv:1001.3781*.
39. Gopalakrishnan Nair, T. R., & Krutthika, H. K. (2010). An Architectural Approach for Decoding and Distributing Functions in FPUs in a Functional Processor System. *arXiv e-prints*, arXiv-1001.
40. Krutthika H. K. & A.R. Aswatha. (2021). Implementation and analysis of congestion prevention and fault tolerance in network on chip. *Journal of Tianjin University Science and Technology, 54*(11), 213–231. https://doi.org/10.5281/zenodo.5746712.
41. *Pabbineedi, S., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., Tyagadurgam, M. S. V., & Gangineni, V. N. (2023). Scalable Deep Learning Algorithms with Big Data for Predictive Maintenance in Industrial IoT. International Journal of AI, BigData, Computational and Management Studies, 4(1), 88-97.*
42. *Chalasani, R., Vangala, S. R., Polam, R. M., Kamarthapu, B., Penmetsa, M., & Bhumireddy, J. R. (2023). Detecting Network Intrusions Using Big Data-Driven Artificial Intelligence Techniques in Cybersecurity. International Journal of AI, BigData, Computational and Management Studies, 4(3), 50-60.*
43. *Vangala, S. R., Polam, R. M., Kamarthapu, B., Penmetsa, M., Bhumireddy, J. R., & Chalasani, R. (2023). A Review of Machine Learning Techniques for Financial Stress Testing: Emerging Trends, Tools, and Challenges. International Journal of Artificial Intelligence, Data Science, and Machine Learning, 4(1), 40-50.*
44. *Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., Tyagadurgam, M. S. V., Gangineni, V. N., & Pabbineedi, S. (2023). A Survey on Regulatory Compliance and AI-Based Risk Management in Financial Services. International Journal of Artificial Intelligence, Data Science, and Machine Learning, 4(4), 46-53.*
45. *Bhumireddy, J. R., Chalasani, R., Vangala, S. R., Kamarthapu, B., Polam, R. M., & Penmetsa, M. (2023). Predictive Machine Learning Models for Financial Fraud Detection Leveraging Big Data Analysis. International Journal of Emerging Trends in Computer Science and Information Technology, 4(1), 34-43.*
46. *Gangineni, V. N., Pabbineedi, S., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., & Tyagadurgam, M. S. V. (2023). AI-Enabled Big Data Analytics for Climate Change Prediction and Environmental Monitoring. International Journal of Emerging Trends in Computer Science and Information Technology, 4(3), 71-79.*
47. *Polam, R. M. (2023). Predictive Machine Learning Strategies and Clinical Diagnosis for Prognosis in Healthcare: Insights from MIMIC-III Dataset. Available at SSRN 5495028.*
48. *Narra, B., Gupta, A., Polu, A. R., Vattikonda, N., Buddula, D. V. K. R., & Patchipulusu, H. (2023). Predictive Analytics in E-Commerce: Effective Business Analysis through Machine Learning. Available at SSRN 5315532.*

49. *Narra, B., Buddula, D. V. K. R., Patchipulusu, H. H. S., Polu, A. R., Vattikonda, N., & Gupta, A. K. (2023). Advanced Edge Computing Frameworks for Optimizing Data Processing and Latency in IoT Networks. JOETSR-Journal of Emerging Trends in Scientific Research, 1(1).*
50. *Patchipulusu, H. H. S., Vattikonda, N., Gupta, A. K., Polu, A. R., Narra, B., & Buddula, D. V. K. R. (2023). Opportunities and Limitations of Using Artificial Intelligence to Personalize E-Learning Platforms. International Journal of AI, BigData, Computational and Management Studies, 4(1), 128-136.*
51. *Madhura, R., Krishnappa, K. H., Shashidhar, R., Shwetha, G., Yashaswini, K. P., & Sandya, G. R. (2023, December). UVM Methodology for ARINC 429 Transceiver in Loop Back Mode. In 2023 3rd International Conference on Mobile Networks and Wireless Communications (ICMNWC) (pp. 1-7). IEEE.*
52. *Shashidhar, R., Kadakol, P., Sreeniketh, D., Patil, P., Krishnappa, K. H., & Madhura, R. (2023, November). EEG data analysis for stress detection using k-nearest neighbor. In 2023 International Conference on Integrated Intelligence and Communication Systems (ICIICS) (pp. 1-7). IEEE.*
53. *KRISHNAPPA, K. H., & Trivedi, S. K. (2023). Efficient and Accurate Estimation of Pharmacokinetic Maps from DCE-MRI using Extended Tofts Model in Frequency Domain.*
54. *Krishnappa, K. H., Shashidhar, R., Shashank, M. P., & Roopa, M. (2023, November). Detecting Parkinson's disease with prediction: A novel SVM approach. In 2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIE) (pp. 1-7). IEEE.*
55. *Shashidhar, R., Balivada, D., Shalini, D. N., Krishnappa, K. H., & Roopa, M. (2023, November). Music Emotion Recognition using Convolutional Neural Networks for Regional Languages. In 2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIE) (pp. 1-7). IEEE.*
56. *Madhura, R., Krishnappa, K. H., Manasa, R., & Yashaswini, K. P. (2023, August). Slack Time Analysis for APB Timer Using Genus Synthesis Tool. In International Conference on ICT for Sustainable Development (pp. 207-217). Singapore: Springer Nature Singapore.*
57. *Krishnappa, K. H., & Gowda, N. V. N. (2023, August). Dictionary-Based PLS Approach to Pharmacokinetic Mapping in DCE-MRI Using Tofts Model. In International Conference on ICT for Sustainable Development (pp. 219-226). Singapore: Springer Nature Singapore.*
58. *Krishnappa, K. H., & Gowda, N. V. N. (2023, August). Dictionary-Based PLS Approach to Pharmacokinetic Mapping in DCE-MRI Using Tofts Model. In International Conference on ICT for Sustainable Development (pp. 219-226). Singapore: Springer Nature Singapore.*
59. *Madhura, R., Krutthika Hirebasur Krishnappa. et al., (2023). Slack time analysis for APB timer using Genus synthesis tool. 8th Edition ICT4SD International ICT Summit & Awards, Vol.3, 207–217. https://doi.org/10.1007/978-981-99-4932-8_20.*
60. *Shashidhar, R., Aditya, V., Srihari, S., Subhash, M. H., & Krishnappa, K. H. (2023). Empowering investors: Insights from sentiment analysis, FFT, and regression in Indian stock markets. 2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIE), 01–06. https://doi.org/10.1109/AIKIIE60097.2023.10390502.*
61. *Jayakeshav Reddy Bhumireddy, Rajiv Chalasani, Mukund Sai Vikram Tyagadurgam, Venkataswamy Naidu Gangineni, Sriram Pabbineedi, Mitra Penmetsa. Predictive models for early detection of chronic diseases in elderly populations: A machine learning perspective. Int J Comput Artif Intell 2023;4(1):71-79. DOI: 10.33545/27076571. 2023.v4.i1a.169.*