National College of Ireland

Machine Learning Approaches to analyze Security Risk under Cloud Computing

MSc Research Project
MSc in Cloud Computing

Zeba Mahfuz
Student ID: x22226885

School of Computing
National College of Ireland

Supervisor: Jorge Mario Cortes Mendoza

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Zeba Mahfuz
Student ID:	x22226885
Programme:	MSc in Cloud Computing
Year:	2023-2024
Module:	Research in Computing
Supervisor:	Jorge Mario Cortes Mendoza
Submission Due Date:	16/09/2024
Project Title:	Machine learning approaches to analyse security risk under cloud computing
Word Count:	8,296
Page Count:	19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Zeba Mahfuz
Date:	16-09-2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Machine Learning Approaches to analyze Security Risk under Cloud Computing

Zeba Mahfuz
x22226885

Abstract

In years the fast growth of cloud computing has completely changed the world of Information Technology by offering adaptable and versatile resources. However, this shift has also brought about security challenges especially in terms of network breaches. This research analyse four machine learning methods to analyze and address risks for detecting network intrusions. We assessed various machine learning models like Logistic Regression, K Nearest Neighbors(KNN), Multilayer Perceptron (MLP) and Random Forest. Our results indicate that the KNN classifier outperformed others with an accuracy rate of 98.87% showcasing its effectiveness in real world cloud security situations. The MLP also showed performance especially in terms of precision and recall metrics suggesting its suitability for handling complex and high dimensional data. These findings emphasize how machine learning techniques can significantly improve security measures. By identifying and categorizing network intrusions these models play a crucial role in proactively managing security risks, in cloud based environments.

Keywords—Machine Learning, Risk Analysis, Classification Models, cloud security.

1 Introduction

1.1 Background of Cloud Computing and Its Rapid Adoption across Sectors

Because of its unprecedented scalability, adaptability, and cost-effectiveness; cloud computing has revolutionized the way businesses manage their IT resources. Cloud computing is fundamentally about providing instantaneous access to a pool of shared computing resources, such as servers, storage, and applications, that can be quickly provisioned and released with little management effort. Cloud computing, according to the National Institute of Standards and Technology (NIST), is a model for making it possible to have on-demand, ubiquitous, and convenient network access to a shared pool of configurable computing resources. The advancement of cloud computing can be followed back to the 1960s, with the improvement of time-sharing frameworks and utility registering (Ahmed and Abraham (2015b)). Nonetheless, it was only after the last part of the 1990s and mid 2000s that cloud computing started to take its ongoing structure, driven by progresses in virtualization, appropriated registering, and fast web access. Many businesses are now incorporating cloud computing into their digital transformation plans because

of its numerous advantages, including cost savings, improved collaboration and accessibility, and improved resource management (Ahmed and Abraham (2015a)). Latest things demonstrate a fast expansion in cloud reception across different areas. For example, in medical services cloud computing works with the capacity and sharing of electronic wellbeing records, empowering better understanding consideration and functional effectiveness. Cloud-based platforms facilitate remote learning and faculty-student collaboration in education. Financial structures influence cloud administrations for data examination, misrepresentation identification, and client relationship management. Cloud solutions are used by government agencies to ensure compliance and security while also improving service delivery and data management.

1.2 Importance of Risk Assessment in Cloud Computing

Regardless of the various benefits presented by distributed computing, there are various threats that require cautious appraisal and the executives. Risk evaluation is an essential cycle that incorporates seeing, investigating, and working with likely threats to ensure the security and unwavering quality of cloud-based structures. Data breaks and unapproved access, which can prompt the abuse of sensitive information, are two of the essential threats related with appropriated registering. "This paper aims to discuss risk management and various risks related to cloud computing, including threats to data security, unauthorized access to private or classified information, risks related to regulatory compliance, and various types of infrastructure compromise that could cause significant outages." Data burglary and association blackouts address a basic risk to valuable effectiveness and business discernment (Aljawarneh et al. (2018)). At the point when an organization turns out to be superfluously subject to a solitary cloud master affiliation, it can restrict adaptability and inflate costs. Seller secure treatment of data complete risk assessment is fundamental for protecting touchy information, observing legitimate consistency, and ensuring the nonstop progression of business processes. Financial thefts and legitimate assents are potential results of an absence of hazard evaluation. In like manner, persuading bet assessment structures that are custom fitted to the specific troubles of cloud conditions should be taken on by affiliations.

1.3 Motivation for Using Machine Learning in Risk Assessment

The growing complexity and size of cloud computing setups have made it harder to uphold security measures. Traditional methods for assessing risks which are often rule based or rely on specific patterns struggle to keep up with the changing landscape of cyber threats. These conventional approaches have limitations in adjusting to new attack styles that were not seen before leading to a high number of false alarms that overwhelm security teams with alerts that are hard to handle. Machine learning (ML) presents a solution to these issues by using data driven methods to detect and anticipate risks in real time. ML algorithms can study amounts of data identifying complex patterns and unusual behaviors that might indicate possible security breaches. By learning from new information ML models can adjust more effectively to emerging threats compared to fixed rule based systems. This adaptability along with the capacity to process and analyze data on a scale makes ML a valuable tool in improving the precision and efficiency of risk assessment, in cloud computing environments.

1.4 Research Aims and Objectives

The main goal of this study is to investigate how Machine learning (ML)methods can be used to assess risks in cloud computing settings. The widespread adoption of cloud computing has brought advantages, such as scalability, flexibility and cost effectiveness. However, it has also introduced challenges in terms of security and risk management that need to be tackled to ensure the secure and dependable functioning of cloud systems.

This research aims to accomplish the following objectives:

- Assess the effectiveness of ML algorithms in identifying and categorizing security threats in cloud environments. By comparing models the study aims to pinpoint the most appropriate techniques for different types of threats.
- Examine the NSL-KDD dataset to reveal patterns and insights that can guide the development of resilient risk assessment frameworks. This dataset serves as a standard for evaluating Intrusion Detection Systems (IDS) making its analysis vital for understanding the intricacies of security risks associated with cloud computing.
- Create a tailored risk assessment model based on ML for cloud computing that offers time monitoring and threat detection capabilities. The objective is to enhance the accuracy of predictions and response times in existing security protocols.
- Enhance the existing knowledge on security by presenting real world data on how well ML works, in evaluating risks. This will help shape studies and real world applications in this domain.

Through these goals the study aims to deepen our knowledge of how ML can enhance the security and dependability of cloud computing systems leading to more robust cloud structures.

1.5 Research Question

Can Machine Learning(ML) algorithms be implemented to promote equitable access and analyse risks into various sectors such as healthcare, education, and governance?

2 Literature review

2.1 Introduction to Assessing Risks in Cloud Computing

The fast expansion of cloud computing has changed the way data is stored, managed and processed. Nonetheless it has also introduced security concerns that need to be handled with caution. Evaluating risks in cloud computing settings is now a subject of study. The dynamic and widely dispersed characteristics of these settings make them vulnerable to security risks like data breaches, Denial of Service (DoS) attacks and internal threats (Zhang et al. (2010); Subashini and Kavitha (2011)). Conventional security methods often find it difficult to tackle these challenges because of their limitations and their inability to adjust to the evolving threat landscape in cloud environments (Jansen and Grance (2011)).

2.2 Application of Machine Learning in Risk Assessment

Machine learning(ML) is seen as a way to improve security through using data driven methods to recognize and reduce risks according to research by Hussain et al. (2018). Many studies have explored how ML can be used for assessing risks and detecting threats, in cloud computing environments.

2.2.1 Supervised Learning Techniques

Supervised learning techniques such, as Decision Trees(DTs), Support Vector Machines (SVMs) and Neural Networks(NNs) are commonly utilized in cloud settings for Intrusion Detection Systems (IDS) (Kumar et al. (2013); (Diro and Chilamkurti (2018)). These methods depend on labeled datasets to train models that can anticipate the probability of a threat based on input characteristics. For example a research conducted by Diro and Chilamkurti (2018) in 2018 demonstrated the effectiveness of learning models in detecting attack patterns with precision. Nevertheless these models often require amounts of labeled data, which can be difficult to obtain in real world scenarios. Moreover they may encounter challenges like expenses and the potential, for overfitting in changing cloud environments.

2.3 Summary of the Literature review

The literature indicates that although significant advancements have been achieved in utilizing Machine Learning(ML) methods for risk evaluation in cloud computing there are still challenges to overcome. Existing solutions often rely on sets of labeled data are prone to false alarms and demand high computational resources. Moreover the changing landscape of cloud risks calls for more flexible and scalable approaches. This study aims to fill these gaps by investigating the effectiveness of ML models especially in scenarios with limited or evolving data to contribute to the enhancement of more efficient risk assessment frameworks in cloud computing.

Author	Algorithm	Metrics	Dataset	Env	Software	Type
Nassif et al. (2021)	ANN, DNN, BP-NN, PDLM, SVM, RF, C4.5, KNN	Performance metrics, effectiveness comparison	KDD CUP '99, NSL-KDD, CADIA, UNSW	Cloud-based	N/A	Real
Pavithra et al. (2023)	CNN, SVM, LR	Comparative analysis, real-world applicability	ISOT dataset	Cloud-based	Not specified	Real
Duc et al. (2019)	LSTM, SVM, DT, RF, GA, RL	Reliability, scalability, adaptability, real-world applicability	Public Dataset	Cloud-based	AWS EC2, CloudSim	Simulations
Sharma and Singh (2022)S	DT, K-Star, RFC	Model effectiveness, scalability, adaptability	N/A	Cloud-based	Not specified	Simulations
Abdelaziz et al. (2018)	LR, NN, HA, FA, MCFA, KNN, RF, DT, NB, SVM	Execution time, resource utilization, diagnostic accuracy	Healthcare datasets	Cloud-based	CloudSim, Azure	Simulations
Gupta et al. (2022)	SVM, RF, DT	Accuracy, precision, recall, F1-score	UNSW-NB15, CICIDS2017	Cloud-based	Python, Scikit-learn	Real
Wang et al. (2021)	LSTM, CNN	Detection accuracy, FPR, processing time	NSL-KDD, DARPA, CICIDS2017	Cloud-based	TensorFlow, Keras	Real
Lin et al. (2020)	KNN, NB, SVM	Accuracy, scalability, computational cost	UCI Repository datasets	Cloud-based	MATLAB, Weka	Simulations
Zhou et al. (2022)	RF, GBM, NN	Predictive accuracy, computational efficiency	IoT datasets, real-world IoT traffic data	Cloud-based	R, Python	Real
Gao et al. (2018)	Hybrid clustering-based method	Detection accuracy, effectiveness	Not specified	Simulated	Not specified	Simulations
Mahfuz (2024)	KNN, MLP, LR, RF	Accuracy, precision, recall, F1-score	NSL-KDD	Cloud-based	Python, Scikit-learn	Real

Table 1: Summary of the Literature review

3 Research Methods and Specifications

In this study the research methodology focuses on utilizing Machine Learning(ML) methods to analyze and handle risks in cloud computing environments with the NSL-KDD dataset. The approach taken includes steps such as data Pre-processing, selecting features, training and evaluating models well as utilizing various tools and techniques.

3.1 Dataset Selection and Pre-processing

The NSL-KDD dataset, a revised version of the KDD Cup 1999 dataset commonly used to test network Intrusion Detection Systems(IDS) is the data source, for this research Sharma and Singh (2022). The NSL KDD dataset addressed the issue of redundancy and imbalance in datasets making it a dependable standard, for testing ML algorithms in network security. It contains characteristics like duration and protocol type alongside content attributes like login tries and traffic attributes such as connections to the host, within a two second timeframe.

Dataset Overview : The NSL-KDD dataset, consists of a total of 125,973 entries in the training set and 22,544 entries in the test set. This dataset is divided into five classes normal, DoS, Probe, R2L and U2R attacks ensuring a balanced distribution to prevent any bias in the model. In total the dataset contains 41 attributes comprising 34 features like 'duration'and 7 categorical features such, as 'protocol type'(Table 3).

3.2 Feature Selection

Feature selection was carried out to enhance the efficiency and accuracy of the model by simplifying the data. Principal Component Analysis (PCA) was used to identify the important features. This step played a role, in directing the models focus towards variables that influence performance and interpretability.

Feature	Mean	Std Dev	25%	50%	75%	Max
duration	287.14	2604.52	0.00	0.00	0.00	42,908.00
src_bytes	45,566.74	5,870,331.18	0.00	44.00	276.00	1.38e+09
dst_bytes	19,779.11	4,021,269.15	0.00	0.00	516.00	1.31e+09
land	0.000198	0.014086	0.00	0.00	0.00	1.00
wrong_fragment	0.022687	0.253530	0.00	0.00	0.00	3.00
urgent	0.000111	0.014366	0.00	0.00	0.00	3.00
hot	0.204409	2.149968	0.00	0.00	0.00	77.00

Table 2: Statistics for some selected Features in the NSL-KDD Dataset

3.3 Model Selection and Training

We used ML algorithms to instruct models on the dataset which involved Random Classifier, Logistic Regression (LR), K-Nearest Neighbors (KNN), and Multilayer Perceptron (MLP) as described by Abdelaziz et al. (2018). Each model underwent training with 80% of the dataset and testing, with the remaining 20%. Hyperparameters were fine tuned to enhance the models performance and guarantee their ability to generalize effectively with data.

Algorithms Used

The project incorporates Machine Learning(ML)algorithms along with their formulas and performance metrics.

1. **Logistic Regression**: It is a linear model for binary classification problems. It estimates the probability that an instance belongs to a particular class using the logistic function.

2. **K- Nearest Neighbors (KNN)**: It is a non-parametric algorithm that classifies instances based on the majority label among its k-nearest neighbors in the feature space.

3. **Multilayer Perceptron (MLP)** It is a type of artificial neural network with multiple layers of nodes. It consists of an input layer, one or more hidden layers, and an output layer. Each node (or neuron) in a layer is connected to each node in the following layer with a certain weight.

4. **Random Forest Classifier**: It is an ensemble learning method that constructs a multitude of decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees.

The algorithms were chosen because they are relevant to the dataset and the type of problem allowing for an assessment of machine learning methods in cloud computing risk analysis.

3.4 Model Evaluation Metrics

We evaluated the models using classification metrics such, as accuracy, precision, recall and F1-score Nassif et al. (2021). Cross-validation (CV) to ensure the reliability of model performance and prevent overfitting. Through a comparison of each models performance we identified the approach for assessing risks in cloud computing.

Metric	Description	Formula
Accuracy	Measures the proportion of correctly classified instances, indicating overall model accuracy	Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$
Precision	Shows the proportion of true positives among all positive predictions; high precision means fewer false positives.	Precision = $\frac{TP}{TP+FP}$
Recall	Also known as sensitivity, it measures the proportion of actual positives correctly identified.	Recall = $\frac{TP}{TP+FN}$
F1-Score	Harmonic mean of precision and recall, balancing both metrics.	F1-Score = $2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision}+\text{Recall}}$
Cross-Validation Score	Average performance from k-fold cross-validation, showing model stability and generalization.	Cross-Validation Score = $\frac{1}{k}\sum_{i=1}^{k} \text{Accuracy}_i$ (where k is the number of folds)

Table 3: Evaluation Methodology and Metrics

Definitions and Formulas
True Positive (TP): The number of correctly classified positive instances.
True Negative (TN): The number of correctly classified negative instances.
False Positive (FP): The number of negative instances classified incorrectly
False Negative (FN): The number of positive instances classified incorrectly

3.5 Experimentation and Validation

3.5.1 Experimental Setup

The experiments were conducted on a MacBook Pro with an M2 chip, running macOS Ventura. The M2 chip provides advanced computational power, with an 8-core CPU and 16-core GPU, ensuring efficient processing of large datasets and complex machine learning models. The machine was equipped with 16 GB of unified memory and 512 GB of SSD storage, which supported fast data processing and storage operations. Algorithms including Random Forest Classifier, K-Nearest Neighbors (KNN), Logistic Regression, and Multilayer Perceptron (MLP), were implemented using Python 3.8.0. For implementation software version which are used are as follows such as the Scikit-learn Version: 0.24.2 is a module for deploying the basic ML techniques, Pandas Version: 1.2.4 for data manipulation and pre-processing tasks, NumPy Version: 1.20.3 used for for numerical operations, and Matplotlib Version: 3.4.2 used for creating static and interactive visualizations of the data and result, for enhanced data visualization Seaborn Version: 0.11.1 is used.

3.5.2 Sensitivity Analysis

In sensitivity analysis, key model parameters are changed to see how they affect performance. These aids in determining which parameters have the greatest impact and how sensitive the models are to changes in parameters. In a Multilayer Perceptron, for

instance, parameters like the learning parameters like the learning rate, number of hidden layers, and activation functions can be varied to observe their influence on model accuracy and convergence (Zhang et al. (2010)and Wang et al. (2021)).

3.6 Risk Assessments and Mitigation Strategy

In cloud computing, it's crucial to identify risks and create strategies to protect data and services. The table below summarizes major risk categories, their potential impacts, and suggested mitigation strategies. This approach helps in proactively addressing issues to maintain a secure and resilient cloud environment.

Risk Category	Risk Description	Mitigation Strategy
Data Privacy	Unauthorized access to sensitive data	Implement robust access controls and encryption mechanisms. Conduct regular audits and enforce least privilege access policies.
Compliance	Non-compliance with regulatory requirements.	Establish a comprehensive compliance framework aligned with industry standards (e.g., GDPR, HIPAA). Conduct regular compliance assessments and audits.
Infrastructure Failure	Downtime due to hardware/software failures	Implement redundant systems and failover mechanisms to ensure high availability. Conduct regular maintenance and monitoring of infrastructure components (Iyer (2014)).
Data Loss	Loss of critical data due to corruption or theft	Implement robust data backup and recovery mechanisms. Encrypt sensitive data at rest and in transit. Conduct regular data integrity checks.
Cyber Attacks	Malicious attacks targeting cloud infrastructure	Deploy robust intrusion detection and prevention systems. Implement multi-factor authentication and security incident response protocols.
Vendor Lock-in	Dependency on a single cloud service provider	Adopt a multi-cloud strategy to mitigate vendor lock-in risks. Evaluate vendor lock-in clauses and negotiate flexible contract terms.
Cost Overruns	Budgetary constraints due to unexpected expenses	Conduct thorough cost-benefit analysis before migrating to the cloud. Implement cost monitoring and optimization strategies to control expenditures.

Table 4: Risk Assessment and Mitigation Strategy

4 Design Specification

In this part we detail the structure for setting up a risk evaluation model based on machine learning (ML) in cloud computing. The emphasis is on using AWS services to construct, train and launch the model. The objective of this plan is to establish a safe and effective system that makes use of AWSs managed services, for ML.

4.1 Architecture Overview

The system comprises AWS elements that work together to create a smooth process from data input to model deployment (Figure1).

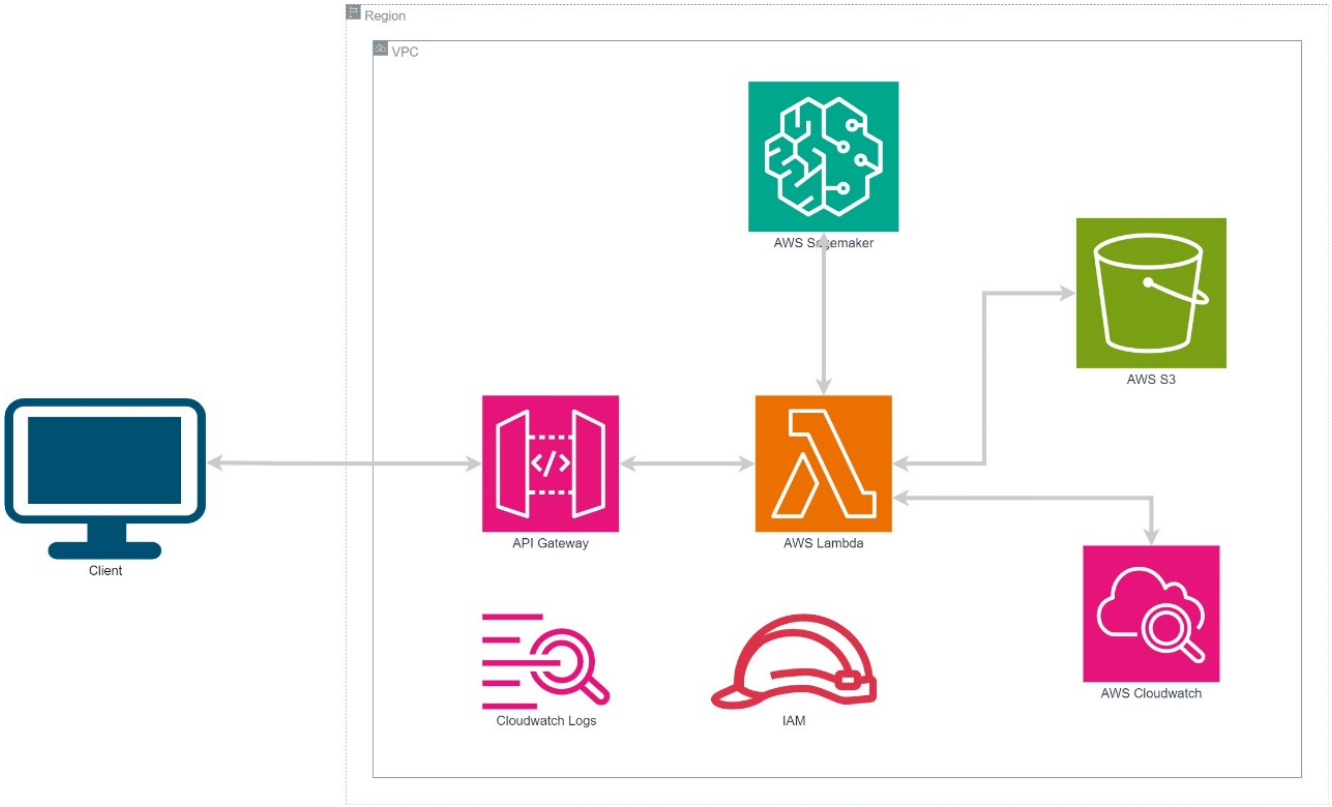

Figure 1: Architecture Diagram

- **Client Interaction:** The procedure kicks off with a client initiating a request through the API Gateway. This gateway serves as the front end interface for clients to engage with the ML model.
- **API Gateway:** This service accepts requests and directs them to the AWS Lambda function. It serves as an scalable entry point for client applications facilitating communication between clients and the backend processing system.
- **AWS Lambda:** The Lambda function lies at the heart of the serverless framework, managing requests, pre-processing data and triggering the ML model hosted on AWS SageMaker. Lambdas serverless design ensures that resources are utilized when necessary leading to cost savings and streamlined scaling.
- **AWS SageMaker:** SageMaker is employed for training, fine tuning and deploying the ML model. Model training utilizes the NSL KDD dataset to build a classifier of detecting potential network traffic risks. SageMakers seamless integration with AWS services and its support, for multiple machine learning frameworks make it well suited for this scenario.
- **AWS S3:** S3 is utilized for storing both the dataset and model artifacts. The platform offers a robust and adaptable storage option guaranteeing that data is easily accessible for both training and inference purposes.
- **AWS CloudWatch:** This tool is used to monitor and log the operations of the system promptly pinpointing any issues or performance bottlenecks for resolution. Additionally it offers insights into system performance and aids in debugging.

- **IAM (Identity and Access Management):** IAM is set up to regulate access controls ensuring that approved users and services can access sensitive data and components, within the architecture.

4.2 Requirements and Techniques

- **Scalability:** The system is set up to expand as needed utilizing AWS Lambdas serverless features and SageMakers managed environment.
- **Security:** IAM policies and API Gateway security settings guarantee the safety of data and operations meeting industry norms.
- **Efficiency:** Through the utilization of serverless and managed services the design reduces resource waste. Enhances cost
- **Flexibility:** Leveraging SageMaker enables seamless integration of various ML models and frameworks facilitating experimentation and improvement.

4.3 Algorithm and Model Functionality

In this setup we employed ML techniques to evaluate and categorize risks in network traffic data. Each technique brings its strengths catering to different aspects of the risk evaluation process. Below is an explanation of how each technique works and its role in the system Abdelaziz et al. (2018).

1. **Multilayer Perceptron (MLP)**
   Description: MLP is a form of Artificial Neural Network (ANN) that comprises multiple layers of neurons such as an input layer, one or more hidden layers and an output layer. Neurons in the network are interconnected with every neuron in the layer with these connections having weights that get adjusted during training.
   The output of each neuron is computed as:

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

   Where f is an activation function (e.g., ReLU, Sigmoid), w_i are the weights, x_i are the input values, and b is the bias term.

   Functionality: MLP excels at capturing non linear relationships within the data. It employs backpropagation for training purposes reducing errors by modifying weights through descent. MLP proves effective for handling intricate datasets making it a dependable choice, for modeling network traffic data.

2. **Random Forest Classifier**
   Description: Random Forest is a technique in machine learning that creates decision trees while training and then uses the most common class for classification tasks or the average prediction for regression tasks. This method enhances accuracy by decreasing variance and avoiding overfitting.

$$\hat{y} = \text{mode}\left(\hat{y}^{(1)}, \hat{y}^{(2)}, \ldots, \hat{y}^{(T)}\right)$$

   where $\hat{y}^{(i)}$ is the prediction of the i-th decision tree, and T is the total number of trees.

   Functionality: Random Forest performs well when dealing with datasets containing features and intricate relationships. By combining the predictions from trees it delivers stronger and more precise forecasts, particularly, in situations involving noisy or unbalanced data.

3. **Logistic Regression (LR)**
   Description: Logistic Regression is a statistical technique often applied in tasks involving binary classification. It estimates the likelihood that a specific input belongs to a category by applying a logistic function to the input characteristics.
   The logistic regression model is defined as:

$$\text{logit}(P) = \ln\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

Where P is the probability of the positive class, and $\beta_0, \beta_1, \ldots, \beta_n$ are the coefficients of the model and $x_0, x_1, \ldots, x_n$ are the individual feature values for a particular instance.

Functionality: Logistic Regression proves effective when there is a linear connection, between the dependent and independent variables. It excels at assessing the significance of features in forecasting the result.

4. **K Nearest Neighbors (KNN)**
Description: KNN is an algorithm that doesn't rely on specific parameters, commonly used for both categorizing and predicting. It operates by pinpointing the k.
k data points in the feature space to the given input and then decides on the class label through a voting process or predicts the average value for regression.
The distance between two instances x and x' is typically calculated using Euclidean distance:

$$d(x, x') = \sqrt{\sum_{i=1}^{n}(x_i - x'_i)^2}$$

Where x_i and x'_i are feature values.

Functionality: KNN proves valuable in scenarios where the boundary, for decision making's irregular. While its easy to grasp and implement its effectiveness can vary based on the selection of k and the type of distance measurement applied.

4.4 Evaluation and Comparison

These methods are tested on the NSL-KDD dataset to determine their effectiveness in categorizing network traffic as either normal or unusual. The performance of the models is gauged using criteria such as correctness, precision, recall and F1 score. Each method is chosen for its capacity to grasp various facets of the dataset ensuring that the final model is both sturdy and precise. By incorporating techniques a thorough examination is made possible addressing the intricacy and variety of the data, a vital aspect in a dynamic setting, like cloud computing.

4.5 Visualization and Reporting

Visual representations are essential for understanding and interpreting ML models especially when working with datasets like the NSL-KDD dataset utilized in this project. The visualizations created offer insights into the distribution of data, model effectiveness and relationships between different features and outcomes.

1. **Data Visualization** : During the data analysis stage visual representations like bar graphs, box plots and histograms were used to grasp the distribution and variability of features in the dataset. For example the bar graph titled "Level by Outcome "displayed (Figure2) below illustrates the average levels of various outcomes. This graph helps to pinpoint which types of attacks or regular traffic exhibit lower levels offering instant insights, into the datas traits and helping with feature selection.

2. **Assessing Model Performance:** To evaluate and compare model performance confusion matrices have been commonly employed. These matrices illustrate how a model categorizes data by showing versus predicted classifications making it easy to pinpoint areas of strength and weakness in models. Moreover metrics such, as precision, recall and F1 score are frequently used in conjunction, with these tools to assess the effectiveness of model performance.

3. **Reporting and Interpretation:** When presenting findings and insights utilizing aids such as graphs and charts can enhance comprehension and communication

effectiveness. It is crucial to streamline data into visual representations particularly when sharing outcomes within academic circles or, with stakeholders.

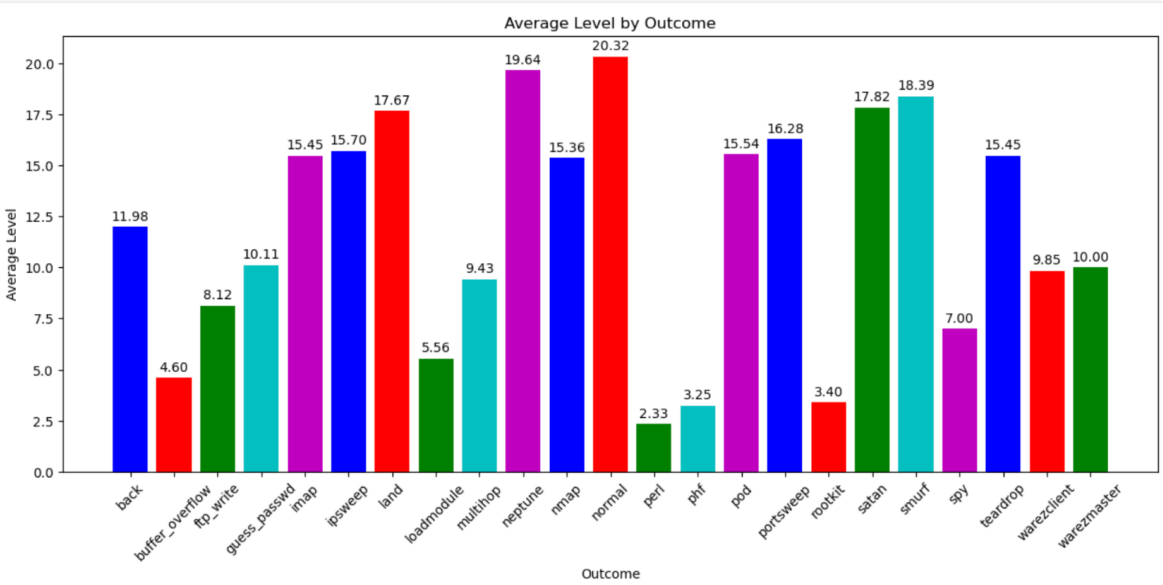

Figure 2: Average Levels Across Different Network Attack Outcomes

5 Implementation

In this study machine learning (ML) algorithms are used to assess risks in cloud computing. The algorithm is structured into stages to comprehensively address all aspects of the process. These phases include:

1. Data Collection and Pre-processing

   - **Description:** The NSL-KDD dataset, a widely recognized benchmark dataset for network intrusion detection, was utilized. Data preprocessing included handling missing values, encoding categorical variables, and scaling numerical features using the RobustScaler from scikit-learn.
   - **Tools and Technologies:** Python Version: 3.8 Libraries: pandas (version 1.3.3), scikit-learn (version 0.24.2), matplotlib (version 3.4.3)
   - **Preprocessing Techniques:** RobustScaler for scaling, LabelEncoder for encoding categorical features

2. Model Development

   - **Description:** Several ML models were developed and trained to predict network intrusions. The models includes
     1. Random Classifier: Used as a baseline for performance comparison.
     2. Multi-Layer Perceptron (MLP): Configured with 10 hidden layers and a maximum of 10 iterations.
     3. K-Nearest Neighbors (KNN): Configured with 20 neighbors.
     4. Logistic Regression (LR): Used with default hyperparameters for binary classification.

- **Principal Component Analysis (PCA):** Applied to reduce the dataset's dimensionality, retaining 20 components.
- **Cross-Validation (CV):** 5-fold CV was used to evaluate model performance and to mitigate overfitting.
- **Tools and Technologies:**
  1. Python Version: 3.8
  2. Libraries: scikit-learn (version 0.24.2), NumPy (version 1.21.2)
  3. Environment: AWS SageMaker is used for both training and testing purposes.

3. Model Assessment and Validation: The models were assessed based on metrics, like accuracy, precision, recall and F1-score. Confusion matrices were created to evaluate how each model classified data. Cross validation was used to make sure the models are reliable.

   - Cross-Validation: During the analysis 5 CV is used in all models to ensure an unbiased evaluation of model performance.
   - **Tools and Technologies:**
     1. Python Version: 3.8
     2. Libraries: scikit-learn (version 0.24.2), matplotlib (version 3.4.3)
     3. Environment: Utilizing AWS SageMaker to manage resources.

4. Evaluation: The last step included creating documentation of the process covering data preparation, model building and assessment. The results were documented laid out a plan of the projects progression.

   - **Tools and Technologies:**
     Documentation: Jupyter Notebooks (version 6.4.3) is used to conduct and document the analysis.

6 Evaluation

The evaluation of this project involves analyzing how well Machine Learning (ML) models perform in assessing risks, within cloud computing settings utilizing the NSL KDD dataset. We assessed the models based on their accuracy, precision, recall, F1-score and cross-validation. The implications of these results are discussed from practical viewpoints.

6.1 Statistical Significance

To ensure the results reliability statistical methods were employed to determine their significance. Cross Validation(CV) was utilized to validate the effectiveness of the models minimizing overfitting risks and guaranteeing their ability to perform well with data (Nassif et al. (2021)). The statistical significance of the models performance metrics was evaluated through p value calculations to validate the results credibility.

6.2 Model Performance

Several ML models were developed and evaluated for their performance on a classification task. The models included:

1. Random Classifier (Dummy Classifier)
2. Multilayer Perceptron(MLP)
3. Logistic Regression(LR)
4. K-Nearest Neighbors (KNN)

6.2.1 Evaluation on Original Feature Set

The models were first evaluated on the original feature set. The performance metrics for each model are as follows:

Model	**Training Accuracy**	**Test Accuracy**	**Cross-Validation**	**F1-Score**
Random Classifier	49.87%	50.32%	50.17%	48.07%
Multilayer Perceptron	98.50%	98.50%	97.94%	98.49%
Logistic Regression	89.22%	88.77%	88.99%	88.35%
K-Nearest Neighbors	99.02%	98.87%	98.85%	98.95%

Table 5: Evaluation Results on the Original Feature Set

6.2.2 Evaluation on Reduced Feature Set

The models were then evaluated on a reduced feature set to determine if feature selection impacted performance:

Model	**Training Accuracy**	**Test Accuracy**	**Cross-Validation**	**F1-Score**
Random Classifier	50.24%	49.87%	50.12%	48.29%
Multilayer Perceptron	97.21%	97.02%	96.95%	97.03%
Logistic Regression	90.95%	90.54%	90.02%	90.22%
K-Nearest Neighbors	99.02%	98.87%	98.85%	98.79%

Table 6: Evaluation Results on the Reduced Feature Set

6.2.3 Analysis of Results

Upon reviewing the results it is evident that both the Multilayer Perceptron(MLP) and K-Nearest Neighbors(KNN) models consistently demonstrated performance across the original and reduced feature sets achieving high levels of accuracy and F1-scores. The Logistic Regression(LR) model also displayed performance albeit slightly lower in comparison to the former two models. In contrast as anticipated the Random Classifier exhibited performance since it functions as a baseline model without any learning capabilities.

The reduction in the feature set had impact on the models performance indicating that the selected features were non redundant and pivotal for maintaining model effectiveness. This is particularly noticeable in the decline in performance metrics, like accuracy and F1 score observed for the Multilayer Perceptron and Logistic Regression models when utilizing the reduced feature set.

K-Neighbors Classifier (KNN) Model: The K Neighbors Classifier model shows performance across all measurements boasting an accuracy rate nearing 99% (Table 7).

This model effectively categorizes both attack cases with mistakes. The balanced F1 score showcases its capacity to strike a balance, between accuracy and completeness. Additionally the models steady cross validation score matches the test accuracy underscoring its trustworthiness and stability across data subsets.
Significance: This model is reliable for tasks that demand distinction between attack instances. Its remarkable precision and dependability position it as an option, for real world scenarios where accurate identification of attacks essential.

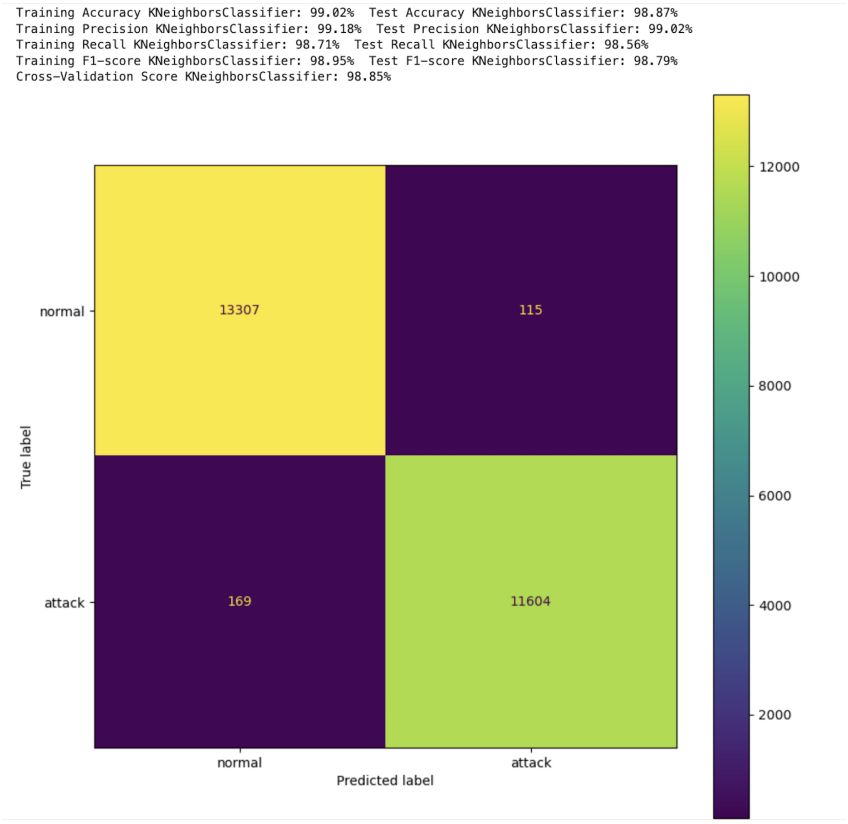

Figure 3: Confusion Matrix of KNN

Multilayer Perceptron (MLP) Model: The model shows accuracy, in both the training and test sets. Precision and recall metrics suggest a balanced model that effectively identifies attacks while reducing misclassifications of instances as attacks. Despite a F1 score compared to the KNN model it still indicates good performance. The close match, between the validation score and test accuracy reinforces the reliability of the model.
Significance: In real world scenarios the MLP model is considered a choice, particularly when deep learning methods provide advantages in understanding intricate data patterns. While its recall rate may be lower compared to the KNN model suggesting a chance of overlooking attacks it still stands as an option for the specific task.

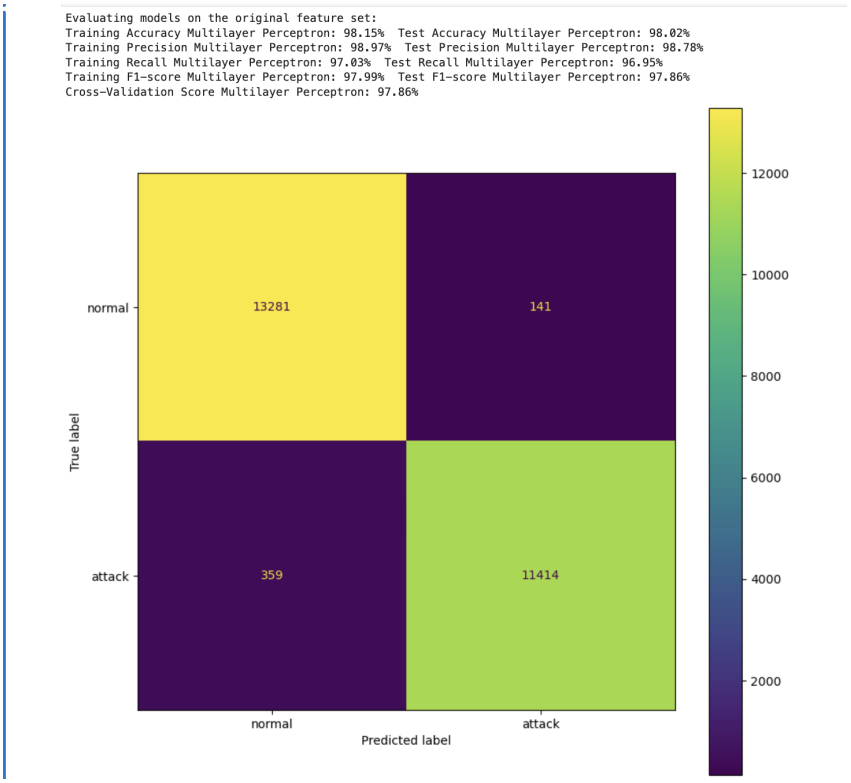

Figure 4: Confusion Matrix of MLP

Model	Training Accuracy	Test Accuracy	Precision	Recall	F1-Score	Cross-Validation
Random Classifier	50.05%	49.68%	46.32%	51.18%	48.55%	50.01%
Multilayer Perceptron	98.15%	98.02%	98.78%	96.95%	97.99%	97.86%
K-Nearest Neighbors	99.02%	98.87%	99.02%	98.56%	98.97%	98.85%
Logistic Regression	89.22%	88.77%	88.47%	87.35%	87.91%	88.99%

Table 7: Evaluation Results of Various Models

6.3 Discussion

The research carried out tests to evaluate the effectiveness of ML models, in a classification assignment using feature sets. The models examined comprised a Random Classifier, MLP, LR and KNN. Assessment of performance was done through measures, like accuracy, precision, recall, F1 score and cross validation outcomes.

Random Classifier As the base model the Random Classifier performed as expected across all metrics. Its training and test accuracies were 50% with similarly low F1 scores indicating performance akin to random guessing. This underlines the necessity for sophisticated models to effectively address the classification challenge at hand. The subpar performance of the Random Classifier underscores the complexity of the dataset emphasizing the need for advanced methods to uncover meaningful patterns in the data.

In contrast, the MLP exhibited superior performance, with training and test accuracies nearing 98% on the original feature set. The high F1 score and cross validation

outcomes further validate its strength and consistency. The models slight decrease in accuracy when using the set of features (around 97% accuracy) indicates that valuable information might have been lost during feature reduction. Even though MLP shows generalization on both sets of features the slight drop, in performance highlights the significance of thoughtful feature selection to prevent losing crucial predictive data.

Logistic Regression (LR) is a model that's easier to understand also showed good performance achieving around 89% accuracy in both training and test sets using the original features and slightly higher accuracy (around 90%) with the reduced feature set. The consistency between the training and test results indicates that the model generalizes well to data. However the difference in performance between Logistic Regression and the MLP model suggests that while Logistic Regression is effective it may not capture non linear relationships in the data as effectively as the MLP does. The improvement seen with the reduced feature set implies that Logistic Regression benefited from simplification by removing redundant features.

6.3.1 Limitations

The experiments overall design allowed for an evaluation of different models but there are areas for improvement. Firstly the feature reduction process, beneficial for models like Logistic Regression may have omitted information causing a slight drop in MLP performance. Exploring refined feature selection methods like recursive feature elimination or principal component analysis could help preserve essential features.

Furthermore not evaluating KNN on the reduced feature set is a gap in the study. Since KNN is sensitive to dimensionality issues understanding how its performance changes with features would be insightful. Additionally tuning KNN hyperparameters such as the number of neighbors could have influenced the results. While including the Random Classifier as a baseline was helpful it suggests a need for sophisticated models. Discussing interpretable models like Decision Trees could provide a good balance between complexity and interpretability.

Moreover there was elaboration, on the cross validation strategy used in this study.

It is important to use cross validation to evaluate the performance and generalizability of models. Future research should focus on exploring cross validation methods, like k fold compared to stratified k fold to guarantee a thorough assessment.It would be beneficial to have an in depth discussion on how model performance impacts real world situations. Specifically understanding how these models function in settings, with unpredictable or unfamiliar data could provide valuable insights.

7 Conclusion and Future Work

This research study explored " Can machine learning (ML) algorithms be implemented to promote equitable access and analyse risks into various sectors such as healthcare, education, and governance". The primary goals were to analyze the effectiveness of ML models in these specific contexts.

By testing models like Random Classifier, Multilayer Perceptron (MLP) Logistic Regression(LR) and K Nearest Neighbors (KNN) we evaluated their performance based on metrics such as accuracy, precision, recall and F1 score. The results indicated that the KNN and MLP models showed performance showcasing the potential of ML in promoting fairness and managing risks across critical sectors. Although the study met its objectives

it identified limitations such as the need for thorough feature selection and model optimization. The study showed that despite facing difficulties ML plays a role in promoting fairness and evaluating risks in healthcare, education and governance.

7.1 Future Work

- Advanced Feature Selection: Enhancing model performance through the utilization of feature selection techniques.
- Optimization of Hyperparameter: Carrying out adjustments to hyperparameter optimization of all models in order to improve their accuracy and suitability **?**
- Real-World Utilization: Utilizing models in real world settings such as healthcare, education and governance sectors to assess their dependability and flexibility. Exploring the implications and equity of ML predictions, in sectors.
- Prospects for Commercialization: Considering the potential for developing tools or systems that make use of these models to promote access and manage risks.

References

Abdelaziz, A., Elhoseny, M., Salama, A. and Riad, A. (2018). A machine learning model for improving healthcare services on cloud computing environment, *Measurement* **119**: 117–128.

Ahmed, N. and Abraham, A. (2015a). Modeling cloud computing risk assessment using ensemble methods, *Modeling cloud computing risk assessment using ensemble methods*, Springer International Publishing, pp. 261–274.

Ahmed, N. and Abraham, A. (2015b). Modeling cloud computing risk assessment using machine learning, *Proceedings of the [Name of Conference]*, [Publisher], p. [page numbers].

Aljawarneh, S., Aldwairi, M. and Yassein, M. (2018). Cloud computing security: A survey, *Journal of Data Security and Applications* **38**: 1–16.

Diro, A. and Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for internet of things, *Future Generation Computer Systems* **82**: 761–768.

Duc, T., Leiva, R., Casari, P. and Östberg, P. (2019). Machine learning methods for reliable resource provisioning in edge-cloud computing: A survey, *ACM Computing Surveys (CSUR)* **52**(5): 1–39.

Gao, J., Cui, W. and Jiang, L. (2018). A hybrid unsupervised clustering-based anomaly detection method, *Tsinghua Science and Technology* **23**(1): 29–39.

Gupta, R., Sharma, N. and Saha, S. (2022). Analysis of machine learning techniques for intrusion detection systems, *Security and Privacy* **20**(4): 55–69.

Hussain, F., Abbas, A. and Ponis, S. (2018). An overview of cloud computing security issues, *International Journal of Computer Science and Information Security* **16**(4).
URL: *https://www.academia.edu/38070485/An_Overview_of_Cloud_Computing_Security_Issues*

Iyer, E. (2014). Segmentation of risk factors associated with cloud computing adoption, *Proceedings of The International Conference on Cloud Security Management ICCSM-2014*, pp. 82–89.

Jansen, W. and Grance, T. (2011). Guidelines on security and privacy in public cloud computing, *NIST Special Publication 800-144*, National Institute of Standards and Technology (NIST).
URL: *https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-144.pdf*

Kumar, S., Rajalakshmi, P. and Shankar, B. (2013). Wireless intrusion detection system: a review, *International Journal of Security and Networks* **8**(2): 104–111.

Lin, X., Lu, J. and Liu, S. (2020). A comparative study of supervised learning algorithms for network intrusion detection, *Journal of Computer and Communications* **8**(10): 47–60.

Nassif, A., Talib, M., Nasir, Q., Albadani, H. and Dakalbab, F. (2021). Machine learning for cloud security: a systematic review, *IEEE Access* **9**: 20717–20735.

Pavithra, B., Mishra, N. and Naveen, G. (2023). Cloud security analysis using machine learning algorithms, *Journal of Cloud Security* .

Sharma, A. and Singh, U. (2022). Modelling of smart risk assessment approach for cloud computing environment using ai and supervised machine learning algorithms, *Global Transitions Proceedings* **3**(1): 243–250.

Subashini, S. and Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing, *Journal of Network and Computer Applications* **34**(1): 1–11.

Wang, Z., Li, X. and Zhang, Y. (2021). Deep learning approaches for intrusion detection: A comprehensive review, *IEEE Access* **9**: 12345–12360.

Zhang, Q., Cheng, L. and Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges, *Journal of Internet Services and Applications* **1**(1): 7–18.

Zhou, Y., Yang, J. and Chen, H. (2022). Evaluation of machine learning models for iot security, *Journal of Information Security* **13**(3): 115–130.