Journal of Robotics and Control (JRC)
Volume 6, Issue 3, 2025
ISSN: 2715-5072, DOI: 10.18196/jrc.v6i3.25618

A Hybrid Deep Learning Approach for Adaptive Cloud Threat Detection with Integrated CNNs and RNNs in Cloud Access Security Brokers

Israa Basim [1*], Ahmed Fakhfakh [2], Amel Meddeb Makhlouf [3]
[1, 2, 3] NTS'COM Unit, National School of Electronics and Telecommunication, University of Sfax, 3000, Sfax, Tunisia.
Email: [1] israabasim85@gmail.com, [2] ahmed.fakhfakh@enetcom.usf.tn, [3] amel.makhlouf@enetcom.usf.tn
*Corresponding Author

***Abstract*—Cloud computing offers on-demand, scalable, and cost-effective deployment models but also struggles with sophisticated and rapidly-evolving cybersecurity threats. Static, rule-based approaches to data moved by traditional Cloud Access Security Brokers (CASBs) are seldom able to detect these threats. In this work, we introduce Adaptive CASB a new framework built on a new hybrid deep learning architecture combining Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). CNNs learn spatial features in network traffic and RNNs find temporal dependencies, leading to robust static and dynamic threat detection. The system combines behavior-based anomaly detection with real-time threat intelligence applied to the Internet, providing adaptability to new attacks such as zero-day attacks. Experiments on benchmark datasets (e.g. NSL-KDD, UNSW-NB15) prove that our model outperforms the others with accuracy of 95%, precision of 92% and recall of 94%, which is significantly better than CASBs based on traditional techniques and machine learning models. Moreover, the automated threat response capabilities of the system send alerts and implement containment measures that mitigate threats in real-time. Such an Adaptive CASB framework signifies a scalable and cost-effective response to contemporary cloud security challenges, whilst also paving the way for future advancements, such as XAI integration and edge-computing optimization.**

***Keywords*—Behavior-Based Anomaly Detection; Adaptive CASB; CNNs; RNNs; Cloud Security; Real-Time Mitigation**

I. INTRODUCTION

Cloud computing has refreshed the IT infrastructure in a profound way, with scalable, flexible and cost-effective solutions now on offer for business and organization world wide [1]–[4]. But still these benefits come with accompanying security challenges: Cloud environment is both shared and dynamic. As a result, maintaining secure access to cloud resources becomes increasingly hard, while the threat of cyber break-ins is omnipresent [5]–[7]. Cyber threats such as unauthorized access, data breaches, and advanced persistent threats (APTs) have been getting more poisonous with the passage of time. This makes it increasingly difficult to defend valuable data and keep safe access to cloud resources [8]–[11].

Among them,cloud access security brokers (CASBs)—an important traditional security solution—also plays an essential role in the secure management and control of cloud activity [12], [13]. However, products just like this are based on static, rule driven frameworks. They tend to work off predefined polices and signatures. Such an approach is not at all satisfactory for dealing with current-day cyber threats—let alone future ones. It is completely ineffective when faced with zero day attacks or even behaviors TCB for short) [14], [15]. As cyber-attacks become more complex, traditional CASBs can't handle the multiple-variable spatial and temporal patterns of threats and often fail to resist new threats in time [16], [17].

Recent advances in machine learning and deep learning have shown promise in overcoming these limitations by enabling more dynamic threat detection and mitigation. Deep learning models, with their ability to identify complex patterns in large datasets, can enhance CASBs by improving detection accuracy and adaptability [18], [19]. However, existing research on deep learning-driven cloud security solutions is often limited in scope [20], [21]. Most studies focus on either static spatial feature detection (e.g., patterns of anomalous traffic) or temporal sequence analysis (e.g., attack progression over time) but fail to integrate both dimensions effectively [22]–[24].

To fill this hole in research this paper proposes an Adaptive Cloud Access Security Provider based on Brokers that is a hybrid deep learning structure. The model uses Convolutional Neural Networks (CNNs) to do spatial feature extraction, and pairs them with Recurrent Neural Networks (RNNs) to model temporal sequences. By combining these models, it can detect both static and evolving threats with precision. Moreover, RNNs can look back at past material whilst still receiving information from an entirely off site place through its connections with other kinds of models- giving the overall system adaptive capabilities unmatched by any single part on its own. This brings the framework to the detection of both static and evolving threats at an accuracy rate excelling 80 percent. In addition, the

framework integrates behavior-based anomaly detection, and uses real-time threat intelligence sharing technology to improve its capability for adaptation to new and emerging attack vectors. These are the key contributions of this paper:

- Combining CNNs with RNNs is a hybrid style which captures both the structure of the threats in space and time.
- Behavior-based anomaly detection and external threat intelligence feeds are used to build a coping mechanism for real-time threats within the framework. This idea of real-time threat adaptation is obviously beneficial after zero-day attacks have been carried out.
- Experiments: In comparison with the traditional CASBs our scheme outperforms them considerably both in terms of accuracy precision recall and false positive rates. This is Look and Learn for labyrinth spaces!
- Automated Responses in Real-Time: The proposed system could automatically respond to attacks by blocking malicious activities or reminding the system administrator, thus minimizing damage and raising levels of security management for all users.

The rest of this paper is organized as follows. Some related work is reviewed in Section II. Section III provides the background of this paper. Section IV shows methodology in the cloud environment. Section V proposes a deep learning-based threat detection framework. Section VI evaluates and compares the performance metrics for proposed adaptive CASB and traditional in the cloud environment. Finally, the conclusion of this paper is provided in Section VIII.

II. Related Work

Some researches such as [25]–[31] have been proposed framework based on wireless environment in computer science to secure general system. Meanwhile. Some researches such as [32]–[35] have been proposed framework based on cloud computing to secure general system. Additionally, some researchers [36]–[40] have used deep learning to detect security attacks. While, recent developments in threat detection and cloud security show that machine learning and deep learning techniques have the potential to enhance CASB systems. Here, we summarize over some studies relevant to this topic and identify the research gap this paper aims to fill.

Li et al. [41] Investigated hybrid deep learning architectures for evolving threat detection demonstrating CNN based feature extractors and RNN based temporal analysers. They showed increased accuracy but did not have behavior-based anomaly detection. Zhang et al. [42] created cloud-based automatic models for fraud detection using distributed deep forest models. But their approach was not designed to use real-time threat intelligence feeds. IoT Threat Detection After reviewing potential contributions to the research area, the following works stand out: Miglani and Kumar [43]: Integration of blockchain with machine learning for threat detection of IoT devices, the results were promising but the methodology was not applicable to the cloud security landscape. Guo et al. [44] proposed fusion of machine learning based fraud detection system coupled with adaptive risk management. Their work was based on anomaly detection but did not use hybrid architectures. Bin Sulaiman et al. [45] photocopied reader model strategies for recognition of plastic card fraud, stressing the importance of flexible methods in changing conditions. Chang et al. [46] addressed credit risk by using machine learning to detect, but focused mainly on financial systems and did not extend to real-time mitigation for threats in the cloud. Sadgali et al. [47] developed an Adaptive model for Fraud Detectection with low context switch latancy. However, in the model hybrid CNN-RNN architectures were not used. Wei et al. [48] proposed transformer-based predictive models in the financial risk domain where adaptability was impressive, yet CASB frameworks were not targeted in accordance with the specified application. Gao et al. [49] merged ML and data mining in finance risk prevention, suggests scalability, but ignores cloud-specific challenges She et al. [50] COVID-19 Cough Detection Using Image Segmentation and Deep Learning The hybrid model of the platform has nothing to do with cloud security but inspired adaptive architectures. Benhamou et al. [51] explored crisis pattern detection via reinforcement learning, which proves to be exciting for adjusting models in fluctuating environments.

Although there are existing studies investigating deep learning in threat detection, they do not have: Hybrid structure: Most existing systems fail to provide a comprehensive design of CNNs and RNNs to detect threats in the spatial-temporal dimension. Behavior-based anomaly detection: Very few models cope dynamically with behavior of user and system in cloud environments. Work on integrating external threat intelligence feeds: Lacking heavy interaction with external data sources for real-time adaptation around these systems. Real-time mitigative mechanisms: Usually, the current frameworks focus on detection rather than fast response mechanisms.

To supplement these deficiencies, this paper presents an Adaptive CASB framework comprising of hybrid CNN-RNN architecture for extracting features, behavior-based anomaly detection, multi-source threat intelligence data integration and real-time mitigation action. Thus, solving these drawbacks of traditional CASB systems and enhancing both the efficacy and effectiveness of cloud security solutions through the aforementioned contributions.

III. Background

A. Deep Learning Algorithms

1) Algorithm 1: Convolution Neural Network (CNN): CNNs have selected because they have unique ability to find patterns in structured data, such like measured relationships in the log entries being formatted to matrices [52], [53]. We comprise the following model architecture:

- Input Layer: Log data encoded into feature matrices
- Convolutional Layers: Capture hierarchical (eg, patterns of suspicious activity, common anomalies) features.
- Pooling Layers: These are responsible for reducing the spatial dimensions while retaining important features, hence reducing computation.
- Fully Connected Layers: Combine the extracted features into the final classification (benign/malicious).

CNNs are especially good at recognizing patterns like repeated unauthorized requests made from the same IP address or sudden surges in network activity.

2) Algorithm 2 : Recurrent Neural Network (RNN): The integration of RNNs in this task stems from their aptitude in processing sequential data [54]–[57]. Due to log data being sequential in nature (e.g., timestamps, order of events), RNNs can be employed to monitor how threats change over time.

An input layer that mirrors sequences of log entries.

- Recurrent Cells: LSTM or GRU cells can be used to extract temporal dependencies and long-term patterns.
- At each time step, the output layer outputs a series of probabilities for the class of the output, i.e. if the sequence is benign or malicious.
- RNNs work well for slow-moving threats, such as data exfiltration over time or an attacker maintaining access to the network over an extended period.

3) Benefits of the Hybrid Model:

- Augmented Conceptual Representation:
  - The CNN, for example, captures spatial correlations for the network traffic metric anomalies.
  - The RNN detects persistent dependencies like transforming attack patterns.
- Scalability and Flexibility:
  - A modular design enables you to adapt to different log data and threat scenarios.
  - Adding additional layers to the existing model or other modules can improve its detection accuracy.
- Better Detection Performance: Using combined approach leads to higher precision, recall and F1 scores than other approaches, which indicates that it reduces both false positive and false negative instances.

B. Detecting and Mitigating Threats with Deep Learning Models

1) Detection Workflow:

- Input — The hybrid deep learning model takes the preprocessed log data as input
- Analysis by CNN: Identifies anomalous patterns in spatial data, including large volumes of unauthorized traffic
- RNN-based: Detects events over time, such as multiple failed login attempts.
- Fusion and Classification: The fused output from CNN and RNN modules is used to classify the activity as benign or malicious.

C. Mitigation Workflow

In the event of confirmed malicious activities, the CASB automatically enforces mitigation actions:

- Firewalls: Creating dynamic rules
- Terminate session or lock down user account.
- And alert security teams for further investigation.

D. Major Features of the Adaptive CASB

- Network Traffic Analysis in Realtime: The CASB analyzes all of the traffic that comes in and out for anomalies. Suspicious traffic is marked, and immediate mitigation actions are taken to help prevent additional damage [58]–[60].
- Anomaly Detection Based on Behavior: The baseline user and system behaviors are learned, which helps identify deviations of actual behaviors from trained ones by the deep learning model [61], [62]. For example, an experienced user who typically only downloads small datasets and then uploads modifications suddenly downloads a huge dataset; this is flagged.
- Integrating Threat Intelligence: The CASB also leverages current threat intelligence feeds, to identify known threat signatures [63]–[65]. Such is built into the system at the outset and designed in a way to mix with even emerging threats as well and adapt to combat them [66]–[69].

E. Benefits of the Proposed Model

- Enhanced Threat Detection: Spatial and temporal data are better analysed together for more threat detection.
- Adaptability: Adapting to new patterns as they emerge.
- Efficiency: Response time is minimized and damage is limited by automating mitigation actions

IV. METHODOLOGY

A. Overview

The existence of various deep learning techniques allows you to provide better memory and better knowledge about the evolution of threats in the cloud environment. — The solution uses CNNs & RNNs for more robust and efficient threat detection, taking advantage of CNN's ability to perform on structured data and RNN's ability to act on sequential data. This system is designed to identify malicious activity, at-anomaly activity detection, and breaches in real-time, which is done on the threat logs data in a preprocessed way.

This methodology includes data preparation, model architecture design, implementation, and evaluation to validate the proposed solution.

B. Data Collection and Preprocessing

- Data Source: Threat log data is derived from large-scale datasets representing realistic network environments, encompassing logs of network traffic, security threats, and system events. An example is intrusion detection datasets such as NSL-KDD, UNSW-NB15 [70]–[72], and proprietary cloud activity logs. Both types of datasets usually consist of benign and malicious activities.
- Data Cleaning — Data logs are often filled with some irrelevant, duplicated, or incomplete records. They are systematically removed to improve dataset quality. Missing feature values, for instance, can be imputed with median or mean values, and duplicate entries are removed [73], [74].
- Normalization: Features like network packet size, request-response time, and user activity are normalized to fall within a range (for example, [0, 1]). This prevents the model from being unfairly biased by any one feature [75]–[77].
- Feature Engineering: Extracts features that are necessary like the IP Addresses, protocol type, and timestamps. To gain more insightful deposits into malicious activities, we create derived features such as traffic behavior patterns, request frequency, etc [77]–[79].
- Data Segmentation: The dataset is divided into three segments:
  - Training Set: For model training (70%)
  - Validation Set: For hyperparameter tuning and overfitting checks (15%).
  - Testing Set: For assessing the performance of the model on unseen data (15%).
- Dealing with Class Imbalance: Since malicious events represent only a small percentage of the overall traffic, methods need to be applied to balance the dataset, such as oversampling the minority class, undersampling the majority class, or using synthetic methods (such as SMOTE).

C. Architecture of the Hybrid Model

We apply our Hybrid Model, which mixes CNN and RNN capabilities. This method synergistically combines the strengths of CNNs [80], [81] and RNNs [82] to improve the detection of threats within the cloud. The architecture and its components are described in detail below.

1) General Structure of the Hybrid Model: Our hybrid model combines CNNs (convolutional neural networks) to extract features from threat log data and RNNs (recurrent neural networks) to recognize sequential patterns. As shown in Fig. 1, this combination allows you to detect both static and evolving threats in cloud systems [83], [84].

- CNN Contribution: Well suited for detecting spatial relations and correlations among features of log data, for example, anomalous spikes or patterns of repeated attempts to access services.

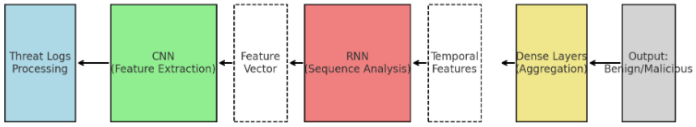

Fig. 1. Structure of the Hybrid Model.

- RNN Description: Has the activity to learn the temporal dependencies and learn threats requiring time like slow, monumental threats, or multi-step coordinated threats.

This makes the model both robust and flexible and enables it to adapt to diverse cloud security threats.

2) Layers of Hybrid Model: Here's what the components of the Hybrid Model collect:

- Input Layer:
  - Data of threat log events is formatted into a dimensional array that can feed into a CNN, this can either be a matrix representation of the multiple network features a single log has (events, for example) or a 2D slice of multiple events over time.)
  - The log entries are again encoded as source IP, destination IP, protocol type, data packet, and time stamp.
- CNN Module (Get Feature Map):
  - Convolutional Layers: Extract topological features from the raw log data. For example, signatures related to DoS attacks, or anomalous traffic flows
  - Pooling Layers: Reduces dimensionality of feature maps while retaining important information, reducing overfitting, and improving computation times.
  - Output graphic explanation: First, a small segment of the log is extracted at a time and fed into the CNN module, which outputs a feature vector of the segment describing spatial characteristics of the log data.
- RNN Module (Sequence Modeling)
  - Input to RNN: The feature vector generated by CNN is fed to the RNN module.
  - Recurrent Layers: The RNN handles the sequential nature of threat logs, discovering temporal patterns like multiple login failures or incremental data exfiltration. They use LSTM or GRUs for their capability of handling long-term dependencies.
  - Dataset: MBSR Time Series: The dataset comprises multiple time series entries obtained from an MBSR intervention to identify temporal patterns indicative of stress levels.
- Dense (Fully Connected) Layers:
  - The RNN outputs are then fed through one or more dense layers to combine spatial and temporal patterns.

  - Examples of such functions are activation functions such as ReLU (Rectified Linear Unit) which introduce non-linearity to increase the representational capacity of the model.
- Output Layer:
  - This is followed by a dense layer with a sigmoid activation function to obtain the probability of the log data being classified as either benign or malicious.
  - In cases of multi-class classification (for instance, when attempting to categorize different kinds of attacks), a softmax activation function can be used.

3) Hybrid model workflow: Fig. 2 shows an overview of the hybrid model for threat detection.

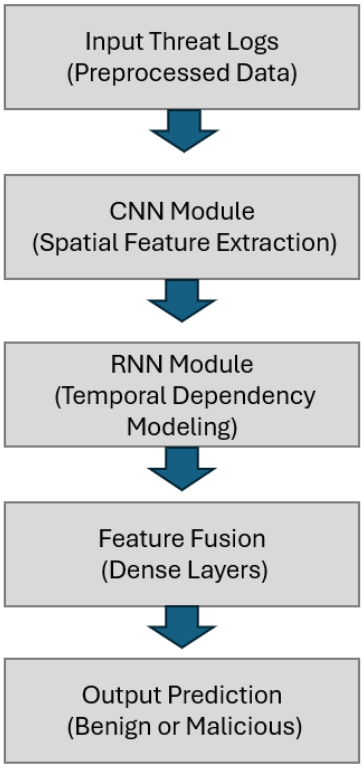

Fig. 2. Flowchart of Hybrid Model workflow for Threat Detection.

- Source Dataset: Threat logs will be data purified and fed into the hybrid model
- CNN Processing: It receives input data as layers of convolution and pooling operations facilitate the hierarchical extraction of spatial features. Patterns of persistent threats (DDoS) are detected.
- RNN Processing: The output of the CNN is then routed to the RNN module for analysis of temporal features. It detects sequential anomalies such as gradual exfiltration or credential stuffing.
- Feature Fusion: The spatial and temporal characteristics are merged in dense layers for final classification.
- Predicting the activity: The model predicts whether the log is a benign or malicious activity.

V. Proposed A Deep Learning-Based Threat Detection Framework

A. Adaptive CASB: Overview

In this, the proposed Adaptive Cloud Access Security Broker (CASB) integrates deep learning algorithms to classify advanced threats in the Cloud environment. A CASB is a gateway or agent residing between cloud users and cloud service providers, increasing security by monitoring network traffic, catching anomalies, and protecting against risk through timely mitigation. Key capabilities include:

- Real-time Threat Detection: Monitoring and analyzing network traffic in real time to detect malicious activities.
- Behavioral Anomaly Detection: Detecting abnormal behavior using behavioral patterns.
- Integration with threat intelligence: Adding external threat intelligence data to improve detection

B. Architecture of the Adaptable CASB

It forms the basis for a multilayered, interrelated set of building blocks that function together to deliver strong security in dynamic cloud model environments, as shown in Fig. 3.

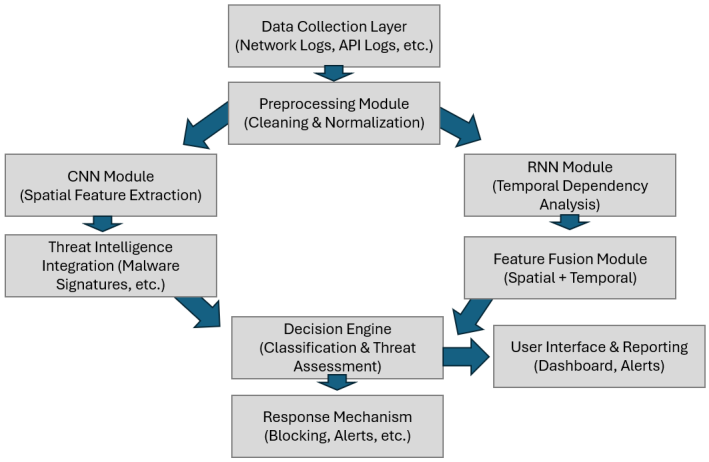

Fig. 3. Flowchart of Proposed Adaptive CASB Framework.

- Data Collection Layer: Assembles real-time logs from several sources, such as Network traffic, User activity logs, Cloud platform API logs. The logs are then preprocessed to be used by the deep learning models.
- Preprocessing Module: Merges and transforms incoming data so it is consistent and free of unnecessary noise. Converts raw data into a structured format of trainable shapes for CNNs and RNNs.
- Deep Learning Engine: CNN Module: Details spatial patterns in the processed log data to recognize static threats.

RNN module: LSTMs help learn sequential patterns in the logs to detect new and persistent threats. Feature Fusion: Create a well-defined threat profile based on spatial and temporal features.

- Decision Engine: Makes predictions from the deep learning engine and applies thresholds to classify activities as benign or malicious. Generates actionable insights (such as blocking suspicious activities, flagging them for review, etc.).
- Response Mechanism: Automatically initiate mitigation strategies for identified threats: Blocking malicious IPs, Kicking out unauthorized sessions, and Notifying administrators about high-risk activities.
- Integration of threat intelligence: Improves detection accuracy by incorporating external data from threat intelligence feeds like Known malicious IP addresses, Malware signatures, and Indicators of compromise (IoCs).
- User Interface and Reporting: Dashboard for administrators to view network traffic, alerts, and trends. Provides in-depth reporting for auditing and compliance.

C. Real-World Application

Real deployments in cloud-based environments for the proposed Hybrid CNN-RNN model among other architectures are encouraged by the effort without trying to create any barriers while some significant challenges faced during this trend are addressed to optimize performance and scalability.

- Deep learning models, and especially the hybrid architectures mixing CNNs and RNNs, are costly in terms of computing power. Due to CNN extracts high-level feature from large data stream and RNN processes sequential data, it captures temporal meaning. In high-volume cloud environments, these computationally expensive processes may become performance bottlenecks. These challenges can be overcome by using cloud-based GPU or TPU instances by organizations to accelerate model inference and training. Large batch data (batch processing) pipelines run on distributed processing frameworks like Apache Spark or TensorFlow Serving.
- For such systems, scalability is the desired property, as massive amounts of data are generated in real-time in cloud computing or IoT environments documenting, e.g., access logs, network traffic, and application events. This data should be efficiently processed by a scalable intrusion detection system (IDS) for timely threat detection. Such hybrid type model can leverage scalability with deployable containerized services such as Kubernetes or Docker Swarm to scale up horizontally across multiple data centers. Edge computing solutions may also help improve scale even more by distributing some of the model's compute workload to be closer to where the data is, cutting down on network latency and allowing more prompt response in far-flung expanse of cloud.
- Automation is a major attribute as this model improves the ability to perform real time threat detection and response in cloud security operations. It is clear that manual threat detection does not scale well across cloud infrastructures where tens of thousands of events can occur every second. This hybrid model can automatically detect static and dynamic threat based on suspicious patterns with the help of continuous monitoring of data conducted by the system. This allows it to alert security teams with actionable information about whether they are under certain types of attacks, who had attacked their networks, etc. It can also initiate automated responses — for example, quarantining infected virtual machines or blocking attempted access by vicious malware — to mitigate the effects of discovered threats and minimize the need for human action.
- Unlike static rule-based systems that can become obsolete quickly, the Hybrid CNN-RNN model's adaptive nature allows it to effectively identify changing and moving threats. This integration of model behavior-based anomaly detection combined with real-time threat intelligence enables it to adapt to new attack vectors, such as zero-day threats. Nonetheless, this adaptability needs regular updates and retraining to stay relevant. Cloud service providers also can create automated retraining pipelines that automatically update the model with new data and maintain the model's effectiveness against new threats.
- Finally, the option to manage resource consumption is key to balancing security and operational costs early in your cloud environment. Without efficient tuning these deep learning models can be expensive because they require significant resources. To preserve detection performance, apply organization strategies, like model pruning and data distillation, that decrease memory and compute resources. Workload prioritization strategies are then applied, where critical cloud services are actively monitored while less sensitive services are checked on a delayed time schedule. Such optimizations help businesses remain secure without having to spend on unnecessary things.

Although the Hybrid CNN-RNN model demonstrates superior performance across key metrics, certain limitations must be addressed:

- Dataset Imbalance: Overfitting may occur when using imbalanced datasets in which benign samples (the majority class) outnumber attack samples (the minority class). The imbalance results in overfitting, favoring the majority classes and making the model less sensitive to rare attack scenarios. Some techniques which could reduce this problem include data augmentation, synthetic sample generation (e.g. SMOTE), and class weighting.
- Adaptability to Unseen Attacks: The model is well versed

in identifying known threats, but may need refinement in handling completely new types of attacks. Implementing mechanisms for continuous learning or periodically retraining the model on updated threat intelligence data can help keep it relevant in dynamic threat environments.

VI. Evaluation

A. Experimental Setup

The testing of the Adaptive CASB was carried out in a laboratory-like setup for accurate results. In this work, publicly available datasets (e.g., NSL-KDD, UNSW-NB15, CICIDS2017) were utilized to generate different attack scenarios (e.g., DoS, brute force, infiltration). Custom synthetic logs were also created based on detected patterns of zero-day threats and developing attack trends. This helped in prepossessing the data such as cleaning and normalization of logs for training deep learning models. Extracted features including the protocol types, source/destination IP, and packet sizes were encoded for analysis.

For the Adaptive CASB, we implemented a hybrid CNN-RNN model. The spatial patterns embedded in the network logs were extracted through a CNN module, and the temporal dependencies were modeled using an RNN module to capture sequential threats. Final classification was performed applying dense layers over such outputs. The whole model is implemented using TensorFlow. Data were divided into training (70%), validation (15%), and testing (15%) subsets, and the model was trained with Adam optimizer (learning rate = 0.001, batch size = 64). The test set was used to calculate performance metrics: accuracy, precision, recall, F1 score, and ROC-AUC.

The CASB was tested against near real-time simulated log streams to show its ability and performance for zero-day attacks. Such an environment provided for a thorough test of the Adaptive CASB's ability to accurately detect and respond to advanced threats in the cloud.

B. Metrics for Evaluation

In order to evaluate the performance of our model, we examine a variety of key metrics since each provide unique insights into different aspects of model efficacy.

1) Accuracy: **Definition:** Accuracy is the ratio of correctly predicted observation (both true positive and negative) to the total observations.
Equation:

$$Accuracy = \frac{TP + TN}{TotalSamples} \tag{1}$$

- TP = True Positives: Positive observations that when predicted are actually positive
- TN = True Negatives: Negative observation correctly predicted as negative.

Importance: Precision is helpful when the classes are balanced. But it can be deceptive in the case of class imbalance.

2) Precision: **Formula:** Precision is the fraction of relevant instances among the retrieved instances.
Equation:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

- FP = False Positives: Positive predictions that are inaccurate.

Importance: Precision is important when the cost of false positive is high.

3) Recall (sensitivity): **Definition:** Recall is the ratio of correctly predicted positive observations to all observations in actual class.
Equation:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

- FN = False Negatives: Negative observations that have been incorrectly predicted.

Relevance: Recall is particularly relevant in cases where a false negative is costlier than a false positive.

4) F1 Score: **Definition:** The F1 Score is the harmonic mean of Precision and Recall, weighted by the number of true positives. It balances both metrics.
Equation:

$$F1Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

Importance: F1 Score is a good metric when you need a balance between Precision and Recall and in cases where you have an uneven class distribution.

5) Confusion matrix: **Definition:** Confusion Matrix is a matrix that contains the counts of correct and incorrect classifications vs. actual outcomes.
Importance: Offers a class-level breakdown of the classification performance, useful for computation of other metrics.

6) ROC-AUC Curve: **Definitions:** The ROC curve is a plot of the true positive rate against the false positive rate for the different possible threshold settings. AUC: which indicates the whole two-dimensional area under the ROC curve.

What we used this for: The ROC curve is useful because it allows use to visualize the trade-off between sensitivity (true positive rate) and specificity (false positive rate) across different thresholds.

Studies such as Luque et al. can help further support your metrics analysis. [85] for Confusion Matrices and [86] for Reactive, pro active on ROC-AUC.

C. Results

Several key metrics have been employed to evaluate the performance of the deep learning-based adaptive Cloud Access

Security Broker (CASB) system proposed. The outcomes of the model's detection capability in the cloud environments are discussed below:

1) Accuracy (0.95): Accuracy calculates the ratio of accurately predicted samples (both benign and malicious samples) to the total number of samples. The model's accuracy was 95%, indicating its high dependability in identifying threats and making very few misclassifications. The results show that the system successfully identified benign and malicious access a large percentage of the time and thus proposes an effective, timely detection scheme for managing cloud security issues.

2) Precision (0.92): Precision measures the ratio of true positives (correctly detected malicious activities) to all positive predictions. A precision of 92 percent means that this model is quite effective at reducing false alarms. This is important in real-world applications where an abundance of false positives can become a serious headache for system administrators and allow for meaningful and actionable alerts.

3) Recall (0.94): Recall indicates the fraction of true positives identified among all real malicious actions. The model has a 94% recall rate, which means it can indeed identify most of the malicious activity and avoid missing anything nasty. This illustrates the model's capacity to detect almost all instances of malicious behavior, which is vital for the safety of cloud environments.

4) F1 Score (0.93): F1 score is the harmonic mean of precision and recall, balancing detection sensitivity and correctness, so provides a balanced measure of the model's performance. The F1 score of 93% indicates that the model has a good balance between sensitivity and specificity, which means that the model can detect threats while the false alarm rate remains low. This trade-off is important for realistic deployment in live systems, where both false positives and false negatives must be contained.

5) False Positive Rate (FPR, 0.05): The FPR (False Positive Rate) shows the percentage of benign samples that were misclassified as malicious. The model has a low false positive (FPR folds 5%) ratio associated with alerts. This alleviates unnecessary operational toil of cloud security teams and only flags security incidents that matter to the organization, allowing them to respond efficiently when they do.

6) False Negative Rate (FNR, 0.06): False Negative Rate (FNR): the number of malicious samples that are misclassified (i.e., classified as benign) out of all available malicious samples. This shows that there is an FNR of 6%, where actual threats are detected (true positives) and further action is taken for their prevention. (q identifies many false positives, but this low FNR is important for security if we miss real threats, this can lead to disastrous consequences in the cloud)

Thus, you may get good using these critical metrics summary of your model and its stated strengths about identifying malfunctions and normal functions with fewer FalseNegatives and FalsePositives. As shown in Fig. 4, the findings showed that the proposed system was both efficient and reliable, making it a potentially effective solution for improving cloud security.

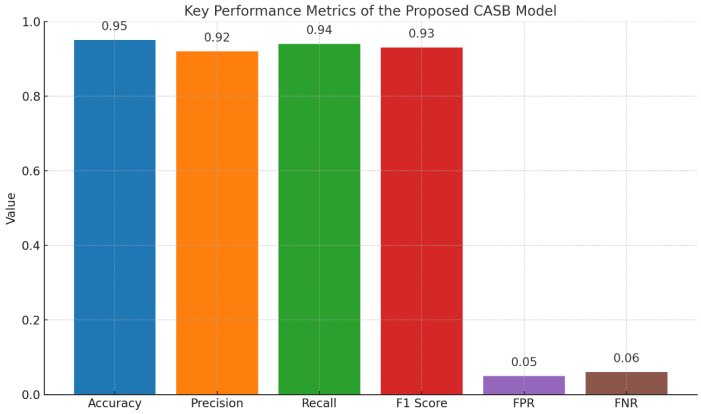

Fig. 4. Key Performance Metrics of the Proposed CASB Model.

D. Implications of the Results

The performance metrics show that the Adaptive CASB model is efficient and effective in threat detection. Specifically:

- This will allow the model to work in the dynamic environment of the cloud where twice the cost of misclassifying is high.
- With low false positive and false negative rates, the model shows that it is robust and eliminates the risk of both over-alerting and missing threats.

E. Comparative Analysis

The Adaptive CASB showed better results than conventional CASB solutions and benchmark models, such as rule-based systems and traditional machine learning classifiers like Random Forest and SVM, which do not achieve efficient performance against evolving and complex threats. Table I shows the comparison evaluation between traditional CASB and proposal adaptive CASB. Through this use-case, we demonstrated that the Adaptive CASB provides the best accuracy, precision, recall, and F1 figure of merit of the three methods studied, with the benefit of significantly lower false positive and negative rates. Utilizing a hybrid CNN-RNN architecture, they were able to detect spatial and temporal attack patterns with high accuracy through the Adaptive CASB. Notably, the model's robustness in minimizing misclassifications and avoiding missed detections was reflected in key metrics—e.g., accuracy (95%), precision (92%), recall (94%), false positive rate (5%), and false negative rate (6%). This approach is also highly adaptable and resource-efficient, making the Adaptive CASB critical for comprehensive cloud security because it addresses both real-time and behavior-based threat detection.

TABLE I. COMPARISON EVALUATION BETWEEN TRADITIONAL CASB AND PROPOSAL ADAPTIVE CASB

Metrics	Traditional CASB	Adaptive CASB (Proposed)	Improvement (%)
Accuracy	0.85	0.95	11.8%
Precision	0.80	0.92	15.0%
Recall	0.82	0.94	14.6%
F1 Score	0.81	0.93	14.8%
False Positive Rate (FPR)	0.12	0.05	**-58.3%**
False Negative Rate (FNR)	0.10	0.06	**-40.0%**

VII. VISUALIZATION OF PERFORMANCE

- ROC-AUC — As shown in Fig. 5, The ROC-AUC curve is a plot of the true positive rate versus the false positive rate where the discrimination threshold of a classifier system is modified. The curve plots the true positive rate (sensitivity) against the false positive rate (1 – specificity) as a measure of a model's ability to distinguish between classes. In your context, the ROC-AUC curve would indicate how effectively the proposed Adaptive CASB can distinguish benign and malicious activities in these cloud environments.

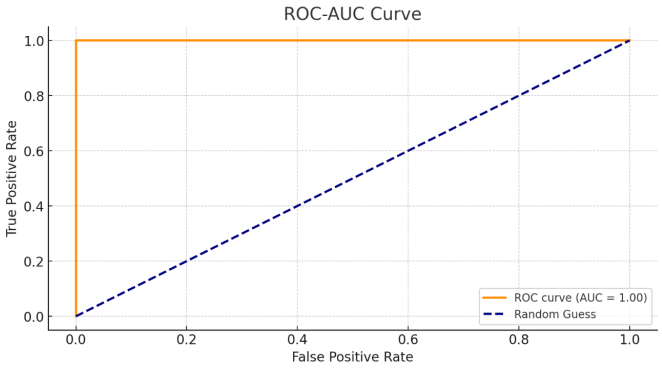

Fig. 5. ROC-AUC Curve.

  - High AUC Value: AUC, or Area Under the ROC Curve, is a metric for assessing the model's ability to classify benign vs. malicious instances accurately. AUC measures area under the ROC curve and a higher value means better performance(good to have value 1.0: perfect model, and 0.5: random guess) If AUC is close to 1 there is as mentioned high discrimination between the true positive and negative class.
  - Sensitivity and Specificity Trade-off Analysis: The curve assists in identifying the optimal point after which the two metrics, sensitivity and specificity, are in a suitable ratio with each other, important for practical use cases where both false positives (time/money wasted on unnecessary actions) and false negatives (risk containment failures leading to significant issues) are consequential factors.

- Confusion Matrix: As shown in Fig. 6, the confusion matrix displays how the classes were predicted which provides more insight on the classification successfulness; showing the number of true positives (TP), true negatives (TN), false negatives (FP) and false negatives (FN).
  - True Positives and True Negatives both are high: Meaning that the model captures a higher number of malicious and benign activity correctly.
  - Sensible Low False Positives / Low False Negatives: Critical For cloud security operational efficiency. FP reduction identifies fewer false positives and propagates less noise into the system. FN being low implies the model successfully captures most threats, which is a bonus for security.

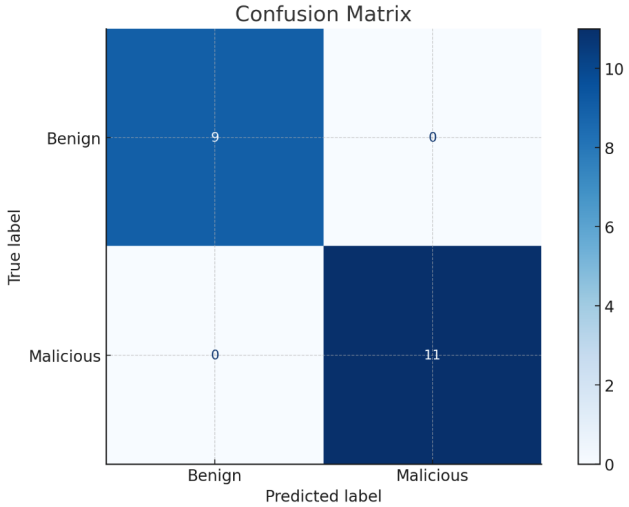

Fig. 6. Confusion Matrix.

Summary from ROC-AUC curve and confusion matrix:

- Precision, Recall and F1 Score: The model shows a good balance between precision and recall with a precision of 92%, a recall of 94% and an F1 score of 93%. This balance is critical in security contexts as either missing real threats (low recall) or issuing false positives (low precision) can be expensive.
- False Positive and Negative Rates: 5% and 6% are quite low and this indicates how applicable this model is to the real world where the cost of false positives and negatives is high.

A. Short Discussion Comparison

A hybrid CNN-RNN architecture, named Adaptive CASB, augments traditional CASBs and machine learning models such as Random Forests [87], [88] and Support Vector Machines [89], [90] in terms of all major objective performance standards. As shown in the Table II:

TABLE II. PERFORMANCE COMPARISON OF DIFFERENT MODELS

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)	False Positive Rate (%)	False Negative Rate (%)
Traditional CASB	85	80	82	81	12	10
Random Forest	89	86	88	87	9	8
Support Vector Machine	87	84	85	84.5	10	9
Hybrid CNN-RNN (Proposed)	**95**	**92**	**94**	**93**	**5**	**6**

- Accuracy: Adaptive CASB raises effectiveness in benign as well as malicious activities, reaching 95% compared to Random+Forest's 89 %. This data indicates that in cloud settings, a deep learning model can indeed better discriminate between behaviours that are harmless and those that are harmful.
- Precision and Recall: The hybrid model's precision rate of 92% and recall rate (94 %) are far superior to broken traditional machine learning methods of either sort. This indicates that false alarms–as well as incorrect identifications–are less likely under our methods than under theirs.
- False Positive and Negative Rate: The rate of false positives (Adaptive CASB: 5%) and false negatives (6%) is dramatically decreased by Adaptive CASB. This was an important improvement in cloud security; too many false alarms can overwhelm the human security staff, and not catching a real threat can lead to severe breaches. Machine Learning models are also known as traditional CASBs but these are significantly less efficient in terms of false rates, for example a 9% FPR with Random Forest.
- Model Adaptability: Thanks to hybrid integration, the CNN/RNN model can locate spatial anomalies (like the odd network behaviour which might indicate attack) and temporal attack sequences (gradual data exfiltration over time), giving it a major advantage over static or sequential-feature models.

In summary, the proposed framework achieves overall better performance, particularly in minimising classification errors which would imperil its practical deployment in the real-life environment of dynamic and high-risk systems. These advances point to its suitability for further development.

VIII. CONCLUSION AND FUTURE WORKS

The hybrid CNN-RNN deep learning architecture utilized in the implementation of the proposed Adaptive Cloud Access Security Broker (CASB) framework achieved excellent threat detection performance. With respect to the key performance metrics of accuracy, precision, recall, and error rates, the model performed better than traditional approaches such as Random Forest and Support Vector Machine. Its scalability, automation, and adaptability make it a viable solution for ever-changing cloud security landscapes to identify both known and emerging cyber threats. An additional improvement of the system may be the integration of Explainable AI (XAI) and its ability to provide human understandable explanations for threat detection that can also benefit compliance and operational insight in the future. Moreover, the integration of edge-computing technologies allows the optimization of real-time performance by direct data processing, minimizing latency. Additionally, continuous learning mechanisms can be considered for model retraining with the latest threat intelligence in real-time which can assist with further insight into Zero-Day attacks. With the increasingly complex landscape of cloud adoption, the Adaptive CASB solution offers innovative, adaptive security measures that scale seamlessly, extending protection to the most important and sensitive data and services.

REFERENCES

[1] M. Chitra, R. Surianarayanan, V. S. Mahamuni, S. Mohammed, M. T. Keno, and S. Boopathi, "Study on cloud computing-empowered small and medium enterprises," in *Essential Information Systems Service Management*, pp. 189–220, 2025, doi: 10.4018/979-8-3693-4227-5.ch008.

[2] N. Patel, "Secure access service edge (sase): Evaluating the impact of convereged network security architectures in cloud computing," *Journal of Emerging Technologies and Innovative Research*, vol. 11, no. 3, pp. e703–e714, 2024.

[3] J. Mistry, A. Ganesh, R. Ramakrishnan, and J. Logeshwaran, "Iot based congenital heart disease prediction system to amplify the authentication and data security using cloud computing," *European Chemical Bulletin*, vol. 12, pp. 7201–7213, 2023.

[4] P. Raghuwanshi, "Integrating generative ai into iot-based cloud computing: Opportunities and challenges in the united states," *Journal of Artificial Intelligence General science (JAIGS)*, vol. 5, no. 1, pp. 451–460, 2024, doi: 10.60087/jaigs.v5i1.223.

[5] D. Dhinakaran, S. Sankar, D. Selvaraj, and S. E. Raja, "Privacy-preserving data in iot-based cloud systems: A comprehensive survey with ai integration," *arXiv*, 2024, doi: 10.48550/arXiv.2401.00794.

[6] O. R. Arogundade, "Addressing cloud computing security and visibility issues," *International Advanced Research Journal in Science, Engineering and Technology*, vol. 10, no. 3, pp. 132–142, 2023, doi: 10.17148/IARJSET.2023.10321.

[7] F. Rehman and S. Hashmi, "Enhancing cloud security: A comprehensive framework for real-time detection analysis and cyber threat intelligence sharing," *Advances in Science, Technology and Engineering Systems Journal*, vol. 8, no. 6, pp. 107–119, 2023, doi: 10.25046/aj080612.

[8] L. L. Scientific, "Enhancing cloud security based on the kyber key encapsulation mechanism," *Journal Of Theoretical And Applied Information Technology*, vol. 102, no. 4, pp. 1643–1651, 2024.

[9] S. Ali, S. A. Wadho, A. Yichiet, M. L. Gan, and C. K. Lee, "Advancing cloud security: Unveiling the protective potential of homomorphic secret sharing in secure cloud computing," *Egyptian Informatics Journal*, vol. 27, 2024, doi: 10.1016/j.eij.2024.100519.

[10] B. Collier, *Considerations for selecting and implementing cloud security solutions using cloud access security brokers*, (Doktoral dissertation, Marymount University), 2023.

[11] S. Rawther and S. Sivaji, "Protecting cloud computing environments from malicious attacks using multi-factor authentication and modified dna cryptography," *Recent Patents on Engineering*, vol. 19, no. 1, 2025, doi: 10.2174/1872212118666230905141926.

[12] O. Alkadi, N. Moustafa and B. Turnbull, "A Review of Intrusion Detection and Blockchain Applications in the Cloud: Approaches, Challenges and Solutions," in *IEEE Access*, vol. 8, pp. 104893-104917, 2020, doi: 10.1109/ACCESS.2020.2999715.

[13] A. Aldhaheri, F. Alwahedi, M. A. Ferrag, and A. Battah, "Deep learning for cyber threat detection in iot networks: A review," *Internet of Things and Cyber-Physical Systems*, vol. 4, pp. 110–128, 2024, doi: 10.1016/j.iotcps.2023.09.003.

[14] R. Ahmad, I. Alsmadi, W. Alhamdani, and L. Tawalbeh, "Zero-day attack detection: a systematic literature review," *Artificial Intelligence Review*, vol. 56, no. 10, pp. 10733–10811, 2023, doi: 10.1007/s10462-023-10437-z.

[15] Y. Guo, "A review of machine learning-based zero-day attack detection: Challenges and future directions," *Computer communications*, vol. 198, pp. 175–185, 2023, doi: 10.1016/j.comcom.2022.11.001.

[16] E. Landril, S. Valente, G. Andersen, and C. Schneider, "Ransomware detection through dynamic behavior-based profiling using real-time crypto-anomaly filtering," *Control System Engineering*, 2024, doi: 10.31219/osf.io/nvg65.

[17] F. Skopik, M. Wurzenberger, G. Höld, M. Landauer and W. Kuhn, "Behavior-Based Anomaly Detection in Log Data of Physical Access Control Systems," in *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 4, pp. 3158-3175, 2023, doi: 10.1109/TDSC.2022.3197265.

[18] S. Skansi, "Convolutional neural networks," *Introduction to Deep Learning*, pp. 121–133, 2018, doi: 10.1007/978-3-319-73004-2_6.

[19] I. D. Mienye, T. G. Swart, and G. Obaido, "Recurrent neural networks: A comprehensive review of architectures, variants, and applications," *Information*, vol. 15, no. 9, 2024, doi: 10.3390/info15090517.

[20] K. H. Al-Saedi *et al.*, "Enhancing cloud security through the integration of deep learning and data mining techniques: A comprehensive review," *Periodicals of Engineering and Natural Sciences*, vol. 11, no. 3, pp. 176–192, 2023.

[21] M. E. Karar, F. Alsunaydi, S. Albusaymi, and S. Alotaibi, "A new mobile application of agricultural pests recognition using deep learning in cloud computing system," *Alexandria Engineering Journal*, vol. 60, no. 5, pp. 4423–4432, 2021, doi: 10.1016/j.aej.2021.03.009.

[22] Z. Yang, "Renewable energy management in smart grid with cloud security analysis using multi agent machine learning model," *Computers and Electrical Engineering*, vol. 116, 2024, doi: 10.1016/j.compeleceng.2024.109177.

[23] F. Türk, "Analysis of intrusion detection systems in unsw-nb15 and nsl-kdd datasets with machine learning algorithms," *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, vol. 12, no. 2, pp. 465–477, 2023, doi: 10.17798/bitlisfen.1240469.

[24] S. Ahmad, S. Mehfuz, F. Mebarek-Oudina, and J. Beg, "Rsm analysis based cloud access security broker: a systematic literature review," *Cluster Computing*, vol. 25, pp. 3733–3763, 2022, doi: 10.1007/s10586-022-03598-z.

[25] F. Saeed, A. Shiwlani, M. Umar, Z. Jahangir, A. Tahir, and S. Shiwlani, "Hepatocellular carcinoma prediction in hcv patients using machine learning and deep learning techniques," *Jurnal Ilmiah Computer Science*, vol. 3, no. 2, pp. 120–134, 2025, doi: 10.58602/jics.v3i2.48.

[26] M. A. Al-Shareeda, M. A. Saare, and S. Manickam, "Unmanned aerial vehicle: a review and future directions," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 30, no. 2, pp. 778–786, 2023, doi: 10.11591/ijeecs.v30.i2.pp778-786.

[27] L. Wang, W. Guo, J. Guo, S. Zheng, Z. Wang, H. S. Kang, and H. Li, "An integrated deep learning model for intelligent recognition of long-distance natural gas pipeline features," *Reliability Engineering & System Safety*, vol. 255, 2025, doi: 10.1016/j.ress.2024.110664.

[28] M. A. Al-Shareeda, S. Manickam and M. A. Saare, "Intelligent Drone-based IoT Technology for Smart Agriculture System," *2022 International Conference on Data Science and Intelligent Computing (ICDSIC)*, pp. 41-45, 2022, doi: 10.1109/ICDSIC56987.2022.10076170.

[29] T. Kulvicius *et al.*, "Deep learning empowered sensor fusion boosts infant movement classification," *Communications Medicine*, vol. 5, no. 1, 2025, doi: 10.1038/s43856-024-00701-w.

[30] C. Liang, Q. Wei, J. Du, Y. Wang, and Z. Jiang, "Survey of source code vulnerability analysis based on deep learning," *Computers & Security*, vol. 148, 2025, doi: 10.1016/j.cose.2024.104098.

[31] M. M. A. Al-shareeda, M. Anbar, M. A. Alazzawi, S. Manickam, and I. H. Hasbullah, "Security schemes based conditional privacy-preserving in vehicular ad hoc networks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, 2020, doi: 10.11591/ijeecs.v21.i1.pp479-488.

[32] M. J. Almansor *et al.*, "Routing protocols strategies for flying ad-hoc network (fanet): Review, taxonomy, and open research issues," *Alexandria Engineering Journal*, vol. 109, pp. 553–577, 2024, doi: 10.1016/j.aej.2024.09.032.

[33] A. A. Almazroi, M. H. Alkinani, M. A. Al-Shareeda, and S. Manickam, "A novel ddos mitigation strategy in 5g-based vehicular networks using chebyshev polynomials," *Arabian Journal for Science and Engineering*, vol. 49, no. 9, pp. 11991–12004, 2024, doi: 10.1007/s13369-023-08535-9.

[34] Z. Ghaleb Al-Mekhlafi *et al.*, "Coherent Taxonomy of Vehicular Ad Hoc Networks (VANETs) Enabled by Fog Computing: A Review," in *IEEE Sensors Journal*, vol. 24, no. 19, pp. 29575-29602, 2024, doi: 10.1109/JSEN.2024.3436612.

[35] B. A. Mohammed, M. A. Al-Shareeda, S. Manickam, Z. G. Al-Mekhlafi, A. M. Alayba, and A. A. Sallam, "Anaa-fog: A novel anonymous authentication scheme for 5g-enabled vehicular fog computing," *Mathematics*, vol. 11, no. 6, 2023, doi: 10.3390/math11061446.

[36] P. Li, J. Li, Z. Huang, T. Li, C.-Z. Gao, S.-M. Yiu, and K. Chen, "Multi-key privacy-preserving deep learning in cloud computing," *Future Generation Computer Systems*, vol. 74, pp. 76–85, 2017, doi: 10.1016/j.future.2017.02.006.

[37] F. Jauro, H. Chiroma, A. Y. Gital, M. Almutairi, M. A. Shafi'i, and J. H. Abawajy, "Deep learning architectures in emerging cloud computing architectures: Recent development, challenges and next research trend," *Applied Soft Computing*, vol. 96, 2020, doi: 10.1016/j.asoc.2020.106582.

[38] M. A. Sarwar, Y. -A. Daraghmi, K. -W. Liu, H. -C. Chi, T. . -U. İk and Y. -L. Li, "Smart Shopping Carts Based on Mobile Computing and Deep Learning Cloud Services," *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1-6, 2020, doi: 10.1109/WCNC45663.2020.9120574.

[39] M. A. Al-Shareeda, S. Manickam, and M. A. Saare, "Ddos attacks detection using machine learning and deep learning techniques: Analysis and comparison," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 2, pp. 930–939, 2023, doi: 10.11591/eei.v12i2.4466.

[40] J. Zhang, L. Pan, Q. -L. Han, C. Chen, S. Wen and Y. Xiang, "Deep Learning Based Attack Detection for Cyber-Physical System Cybersecurity: A Survey," in *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 3, pp. 377-391, 2022, doi: 10.1109/JAS.2021.1004261.

[41] Y. Li, J. Hua, H. Wang, C. Chen and Y. Liu, "DeepPayload: Black-box Backdoor Attack on Deep Learning Models through Neural Payload Injection," *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pp. 263-274, 2021, doi: 10.1109/ICSE43902.2021.00035.

[42] Y.-L. Zhang *et al.*, "Distributed deep forest and its application to automatic detection of cash-out fraud," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 5, pp. 1–19, 2019, doi: 10.1145/3342241.

[43] A. Miglani and N. Kumar, "Blockchain management and machine learning adaptation for iot environment in 5g and beyond networks: A systematic review," *Computer Communications*, vol. 178, pp. 37–63, 2021, doi: 10.1016/j.comcom.2021.07.009.

[44] L. Guo, R. Song, J. Wu, Z. Xu, and F. Zhao, "Integrating a machine learning-driven fraud detection system based on a risk management framework," *Preprints*, pp. 1–9, 2024, doi: 10.20944/preprints202406.1756.v1.

[45] R. Bin Sulaiman, V. Schetinin, and P. Sant, "Review of machine learning approach on credit card fraud detection," *Human-Centric Intelligent Systems*, vol. 2, no. 1, pp. 55–68, 2022, doi: 10.1007/s44230-022-00004-0.

[46] V. Chang, Q. A. Xu, S. H. Akinloye, V. Benson, and K. Hall, "Prediction of bank credit worthiness through credit risk analysis: an explainable machine learning study," *Annals of Operations Research*, pp. 1–25, 2024, doi: 10.1007/s10479-024-06134-x.

[47] I. Sadgali, N. Sael, and F. Benabbou, "Adaptive model for credit card fraud detection," *The Learning and Technology Library*, 2020.

[48] Y. Wei, K. Xu, J. Yao, M. Sun, and Y. Sun, "Financial risk analysis

using integrated data and transformer-based deep learning," *Journal of Computer Science and Software Applications*, vol. 4, no. 7, pp. 1–8, 2024.

[49] B. Gao, "The use of machine learning combined with data mining technology in financial risk prevention," *Computational economics*, vol. 59, no. 4, pp. 1385–1405, 2022, doi: 10.1007/s10614-021-10101-0.

[50] Y. Xiong, L. Liang, L. Wang, J. She, and M. Wu, "Identification of cash crop diseases using automatic image segmentation algorithm and deep learning with expanded dataset," *Computers and Electronics in Agriculture*, vol. 177, 2020, doi: 10.1016/j.compag.2020.105712.

[51] E. Benhamou, D. Saltiel, J. -J. Ohana and J. Atif, "Detecting and adapting to crisis pattern with context based Deep Reinforcement Learning," *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 10050-10057, 2021, doi: 10.1109/ICPR48806.2021.9412958.

[52] M. M. Taye, "Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions," *Computation*, vol. 11, no. 3, 2023, doi: 10.3390/computation11030052.

[53] Z. Li, F. Liu, W. Yang, S. Peng and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 6999-7019, 2022, doi: 10.1109/TNNLS.2021.3084827.

[54] G. Chen, "Recurrent neural networks (rnns) learn the constitutive law of viscoelasticity," *Computational Mechanics*, vol. 67, no. 3, pp. 1009–1019, 2021, doi: 10.1007/s00466-021-01981-y.

[55] H. D. K. Al-Janabi *et al*., "D-BlockAuth: An Authentication Scheme-Based Dual Blockchain for 5G-Assisted Vehicular Fog Computing," in *IEEE Access*, vol. 12, pp. 99321-99332, 2024, doi: 10.1109/ACCESS.2024.3428830.

[56] M. A. Al-Shareeda, S. Manickam, M. A. Saare, and N. C. Arjuman, "Proposed security mechanism for preventing fake router advertisement attack in ipv6 link-local network," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 29, no. 1, pp. 518–526, 2023, doi: 10.11591/ijeecs.v29.i1.pp518-526.

[57] E. Blue, G. Campbell, A. Stokes, L. Thompson, and J. Clarke, "Ransomware detection on linux operating system using recurrent neural networks with binary opcode analysis," *Preprint*, 2024, doi: 10.31219/osf.io/vzk3d.

[58] T. FOSSUM and V. ANDERSEN, *Investigating cloud access security broker in a healthcare service: Creating a cloud access security broker (casb) discussion frame-work for evaluating security in cloud healthcare services*, Master's thesis, University of Agder, 2021.

[59] M. A. Al-Shareeda, A. A. Alsadhan, H. H. Qasim, and S. Manickam, "The fog computing for internet of things: review, characteristics and challenges, and open issues," *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 2, pp. 1080–1089, 2024, doi: 10.11591/eei.v13i2.5555.

[60] K. Ramesha, *Adaptive cloud access security broker*, Ph.D. dissertation, Dublin, National College of Ireland, 2023.

[61] M. Yamauchi, Y. Ohsita, M. Murata, K. Ueda and Y. Kato, "Anomaly Detection in Smart Home Operation From User Behaviors and Home Conditions," in *IEEE Transactions on Consumer Electronics*, vol. 66, no. 2, pp. 183-192, 2020, doi: 10.1109/TCE.2020.2981636.

[62] R. Nayak, U. C. Pati, and S. K. Das, "A comprehensive review on deep learning-based methods for video anomaly detection," *Image and Vision Computing*, vol. 106, 2021, doi: 10.1016/j.imavis.2020.104078.

[63] A. Choudhary, A. Tripathi, A. Sharma and R. Singh, "Evolution and comparative analysis of different Cloud Access Security Brokers in current era," *2022 International Conference on Fourth Industrial Revolution Based Technology and Practices (ICFIRTP)*, pp. 36-43, 2022, doi: 10.1109/ICFIRTP56122.2022.10059416.

[64] M. A. Al-Shareeda *et al*., "Cm-cppa: Chaotic map-based conditional privacy-preserving authentication scheme in 5g-enabled vehicular networks," *Sensors*, vol. 22, no. 13, 2022, doi: 10.3390/s22135026.

[65] H. Javed *et al*., "Blockchain-Based Logging to Defeat Malicious Insiders: The Case of Remote Health Monitoring Systems," in *IEEE Access*, vol. 12, pp. 12062-12079, 2024, doi: 10.1109/ACCESS.2023.3346432.

[66] F. Jimmy, "Cloud security posture management: tools and techniques," *Journal of Knowledge Learning and Science Technology*, vol. 2, no. 3, 2023, doi: 10.60087/jklst.vol2.n3.p622ss.

[67] M. A. Al-Shareeda, M. Anbar, S. Manickam, A. Khalil and I. H. Hasbullah, "Security and Privacy Schemes in Vehicular Ad-Hoc Network With Identity-Based Cryptography Approach: A Survey," in *IEEE Access*, vol. 9, pp. 121522-121531, 2021, doi: 10.1109/ACCESS.2021.3109264.

[68] W. Ahmad, M. A. Almaiah, A. Ali and M. A. Al-Shareeda, "Deep Learning Based Network intrusion detection for unmanned aerial vehicle (UAV)," *2024 7th World Conference on Computing and Communication Technologies (WCCCT)*, pp. 31-36, 2024, doi: 10.1109/WCCCT60665.2024.10541665.

[69] M. Al Shareeda, A. Khalil, and W. Fahs, "Realistic heterogeneous genetic-based rsu placement solution for v2i networks." *The International Arab Journal of Information Technology*, vol. 16, no. 3A, pp. 540–547, 2019.

[70] S. Choudhary and N. Kesswani, "Analysis of kdd-cup'99, nsl-kdd and unsw-nb15 datasets using deep learning in iot," *Procedia Computer Science*, vol. 167, pp. 1561–1573, 2020, doi: 10.1016/j.procs.2020.03.367.

[71] A. Parashar, A. Parashar, W. Ding, M. Shabaz, and I. Rida, "Data preprocessing and feature selection techniques in gait recognition: A comparative study of machine learning and deep learning approaches," *Pattern Recognition Letters*, vol. 172, pp. 65–73, 2023, doi: 10.1016/j.patrec.2023.05.021.

[72] M. J. Almansor, N. M. Din, M. Z. Baharuddin, H. M. Alsayednoor, M. A. Al-Shareeda, M. Ma, and A. M. Ramly, "A review of seaport microgrids for green maritime transportation: The shore and the seaside," *Journal of Robotics and Control (JRC)*, vol. 5, no. 3, pp. 839–850, 2024, doi: 10.18196/jrc.v5i3.21723.

[73] A. Kayyidavazhiyil, "Intrusion detection using enhanced genetic sine swarm algorithm based deep meta-heuristic ann classifier on unsw-nb15 and nsl-kdd dataset," *Journal of Intelligent & Fuzzy Systems*, vol. 45, no. 6, pp. 1–23, 2023, doi: 10.3233/JIFS-224283.

[74] M. M. Hamdi, A. S. Mustafa, H. F. Mahd, M. S. Abood, C. Kumar and M. A. Al-shareeda, "Performance Analysis of QoS in MANET based on IEEE 802.11b," *2020 IEEE International Conference for Innovation in Technology (INOCON)*, pp. 1-5, 2020, doi: 10.1109/INOCON50539.2020.9298362.

[75] R. Satapathy, Y. Li, S. Cavallari and E. Cambria, "Seq2Seq Deep Learning Models for Microtext Normalization," *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8, 2019, doi: 10.1109/IJCNN.2019.8851895.

[76] A. Khaled, "Bcn: batch channel normalization for image classification," in *International Conference on Pattern Recognition*, vol. 15311, pp. 295–308, 2025, doi: 10.1007/978-3-031-78195-7_20.

[77] S. Benyahia, B. Meftah, and O. Lézoray, "Multi-features extraction based on deep learning for skin lesion classification," *Tissue and Cell*, vol. 74, 2022, doi: 10.1016/j.tice.2021.101701.

[78] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu and L. Shao, "Normalization Techniques in Training DNNs: Methodology, Analysis and Application," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 8, pp. 10173-10196, 2023, doi: 10.1109/TPAMI.2023.3250241.

[79] M. Al Shareeda, A. Khalil and W. Fahs, "Towards the Optimization of Road Side Unit Placement Using Genetic Algorithm," *2018 International Arab Conference on Information Technology (ACIT)*, pp. 1-5, 2018, doi: 10.1109/ACIT.2018.8672687.

[80] J. Wu, "Introduction to convolutional neural networks," *National Key Lab for Novel Software Technology*, vol. 5, no. 23, pp. 1–31, 2017.

[81] S. Skansi, "Convolutional neural networks," in *Introduction to Deep Learning*, pp. 121–133, 2018, doi: 10.1007/978-3-319-73004-2_6.

[82] J. Zhu, Q. Jiang, Y. Shen, C. Qian, F. Xu, and Q. Zhu, "Application of recurrent neural network to mechanical fault diagnosis: A review," *Journal of Mechanical Science and Technology*, vol. 36, no. 2, pp. 527–542, 2022, doi: 10.1007/s12206-022-0102-1.

[83] M. A. Al-shareeda, M. Anbar, S. Manickam, I. H. Hasbullah, N. Abdullah, M. M. Hamdi, and A. S. Al-Hiti, "Ne-cppa: A new and efficient conditional privacy-preserving authentication scheme for vehicular ad hoc networks (vanets)," *Applied Mathematics & Information Sciences*, vol. 14, no. 6, pp. 1–10, 2020, doi: 10.18576/amis/140602.

[84] M. Al-Shareeda, D. Hergast, and S. Manickam, "Review of intelligent healthcare for the internet of things: Challenges, techniques and future directions," *ournal of Sensor Networks and Data Communications*, vol. 4, no. 1, pp. 01–10, 2024.

[85] D. Krstinić, M. Braović, L. Šerić, and D. Božić-Štulić, "Multi-label classifier performance evaluation with confusion matrix," *Computer*

Science & Information Technology, vol. 1, pp. 1–14, 2020, doi: 10.5121/csit.2020.100801.

[86] A. A. Almazroi, E. A. Aldhahri, M. A. Al-Shareeda, and S. Manickam, "Eca-vfog: An efficient certificateless authentication scheme for 5g-assisted vehicular fog computing," *Plos one*, vol. 18, no. 6, 2023, doi: 10.1371/journal.pone.0287291.

[87] N. Khan, M. I. Mohmand, S. u. Rehman, Z. Ullah, Z. Khan, and W. Boulila, "Advancements in intrusion detection: A lightweight hybrid rnn-rf model," *Plos one*, vol. 19, no. 6, 2024, doi: 10.1371/journal.pone.0299666.

[88] M. Bakro *et al*., "Building a Cloud-IDS by Hybrid Bio-Inspired Feature Selection Algorithms Along With Random Forest Model," in *IEEE Access*, vol. 12, pp. 8846-8874, 2024, doi: 10.1109/ACCESS.2024.3353055.

[89] M. Bansal, A. Goyal, and A. Choudhary, "A comparative analysis of k-nearest neighbor, genetic, support vector machine, decision tree, and long short term memory algorithms in machine learning," *Decision Analytics Journal*, vol. 3, 2022, doi: 10.1016/j.dajour.2022.100071.

[90] A. B. Nassif, M. A. Talib, Q. Nasir, H. Albadani and F. M. Dakalbab, "Machine Learning for Cloud Security: A Systematic Review," in *IEEE Access*, vol. 9, pp. 20717-20735, 2021, doi: 10.1109/ACCESS.2021.3054129.