

Artificial Intelligence-Driven Hybrid Ensemble Model for Cyber Attack Detection in Internet of Things-Enabled Cloud Systems

Anwer Mustafa Hilal^{1*}, Salahaldeen Duraibi²

Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, AlKharj, Saudi Arabia¹

Department of Electrical and Electronics Engineering, College of Engineering and Computer Science, Jazan University, Jazan, Saudi Arabia²

Abstract—The Internet of Things (IoT) is an expanding network of interconnected devices, sensors, and applications that share and communicate data via the internet. Sustainable Cities and Communities is a one main goal of Sustainable Development Goal (SDG) which aims to make urban areas inclusive, safe, resilient, and sustainable by 2030. It consists connection of numerous devices and links to human beings. Industry, Innovation and Infrastructure (SDG 9) focuses on the development of robust infrastructure, promoting sustainable industrialization and develop innovations. Since the IoT devices are susceptible to attacks, SDG 9 intends to develop an effective foundation for a sustainable future. IoT devices are often constrained by limited computational resources, memory, and energy capacity, making them more susceptible to security breaches related to traditional computing systems. Simultaneously, it needs artificial intelligence (AI) to study the data, which is kept in cloud architecture, and make faster and more trustworthy intellectual decisions. Cloud computing ensures controlling the cutting-edge technological setting so that organizations face increasing difficulty in avoiding data breaches and advanced cyber-risks. The rising sophistication and measure of the cloud computing environment need progressive security mechanisms to deal with developing attacks. The cloud integration of IoT systems further increases the attack surface, as vast amounts of sensitive data are transmitted, processed, and stored across multiple cloud platforms. AI has converted the cybersecurity setting, providing state-of-the-art features that considerably improve threat response and detection. By utilizing deep learning and predictive model, AI methods can analyze huge quantities of data to perceive patterns and abnormalities indicative of possible cyber threats. This paper develops an Artificial Intelligence Driven Cyber Attack Detection Using Advanced Ensemble Models and Dimensionality Reduction Technique (AICAD-AEMDRT). The main of the AICAD-AEMDRT algorithm is to develop an AI-driven approach for detecting cyber-attacks in IoT and cloud platforms to enhance security and system resilience. Primarily, the min-max standardization system has been deployed in the data pre-processing to convert input data into a valuable format. Followed by, the Fisher score and principal component analysis method for the chosen of attributes process to choose the most appropriate features. Eventually, the proposed AICAD-AEMDRT technique designs ensemble models namely the variational autoencoder-generative adversarial network, gated recurrent unit, and Elman neural for the attack classification model. The experimental assessment of the AICAD-AEMDRT arises utilizing a benchmark

database. The performance results specified the greater result of the AICAD-AEMDRT model compared to recent systems.

Keywords—Artificial intelligence; cyber attack detection; ensemble model; dimensionality reduction; Internet of Things; cloud environment

I. INTRODUCTION

Internet of Things (IoT) is regarded as the next promising advancement in technology. Along with cloud, artificial intelligence (AI) and IoT cover the data communication system [1]. Furthermore, the greater the amount of IoT devices and cloud-based big data, and the significance of data and network security. However, the progress and gains of IoT device utilization have made a higher Return on Investment (ROI) value; the IoT device security and the continuous cyberattacks on the cloud network infrastructure are considered crucial security concerns [2]. The propagation of cyberattacks targeting organizations across several businesses repeatedly states the vital necessity for strong incident response abilities. It is more crucial than ever to reinforce cyber defenses against advanced attacks in the current digital environment, where businesses of all kinds are using cloud computing. Current cybersecurity tactics are based on Artificial Intelligence (AI) for proactive cyber threat detection, as attackers are continuously improving its techniques and tactics to take advantage of susceptibilities in cloud systems [3]. This outline summarizes the importance of AI-empowered proactive threat identification within cloud platforms and explains how significant it is to safeguard sensitive data, key infrastructure, and digital assets from various cyberattacks [4]. In this framework, AI appears as a transformative power, allowing organizations to adopt proactive approaches to detect and mitigate cyberattacks in cloud environments. By employing deep learning (DL) and machine learning (ML), AI approaches can examine complex patterns, recognize abnormalities, and predict potential security violations with unparalleled accuracy and efficiency. AI-driven cyber threat recognition's proactive method is vital to reduce threats and stop potential security issues before they turn into serious violations [5]. AI and DL allow the system to learn from past experiences, forecast possible threats, and respond to novel attacks [6]. The aim is to offer organizations a strong, practical security solution, which safeguards vital data and cloud operations. The major intention of the attempt is to raise threat

*Corresponding author.

intelligence in cloud computing by the employment of an AI-driven model.

This paper presents an Artificial Intelligence Driven Cyber Attack Detection Using Advanced Ensemble Models and Dimensionality Reduction Technique (AICAD-AEMDRT). At first, the min-max standardization model has been used. Followed by, the Fisher score and principal component analysis (PCA) method for feature reduction process to choose the most appropriate attributes from a data. Eventually, the proposed AICAD-AEMDRT technique designs ensemble models namely variational autoencoder-generative adversarial network (VAE-GAN), gated recurrent unit (GRU) method, and Elman neural network (ENN) system for the attack classification procedure. The experimental assessment of the AICAD-AEMDRT arises utilizing a benchmark database.

II. PRIOR STUDIES ON CYBERATTACK DETECTION

Kshetri [7] investigated the transformative ability of agentic AI within cybersecurity. It focuses on exploring in what way agentic AI transforms cybersecurity practices, specifically in tackling novel and emerging threats. Kasula et al. [8] introduced "SecureCloudAI," an innovative AI-driven security system created to strengthen delicate data in a cloud environment. SecureCloudAI employs a hybrid method, which merges ML paradigms, such as Random Forest (RF), and DL methods, such as LSTM, for detecting, classifying, and responding to possible security violations. This framework provides strong web intrusion detection, network traffic analysis, and malware detection, although upholding efficacy and scalability in big cloud infrastructures. Vadisetty and Polamarasetti [9] proposed a generative AI-driven distributed cybersecurity structure to enhance threat recognition. Those generative methods, namely VAEs, and GANs, are believed to develop the structure of cyberattack simulation, identify anomalies, and making real-time solutions. Depending on a decentralized design, this presented framework guarantees adaptive threat monitoring and reduction across distributed nodes.

Saxena et al. [10] proposed a unique multiple risks analysis-enabled VM attack prediction method (MR-TPM) for securing computation data and minimizing security breaches through actively predicting VM threats. It contains several cybersecurity risk factors connected to the VM's configuration and management, with an examination of the behavior of the user. All those risk factors are measured for generating risk scores and passed as input into an ML-based classifier for estimating the likelihood of threat for every VM. Hassan et al. [11] examined the present environment of cyber-physical security in industrial settings, emphasizing the constraints of conventional models and the developing function of AI in solving those problems. A suggested AI-driven cybersecurity technique has been presented, highlighting its modular design, practical threat detection abilities, and flawless amalgamation with present protocols. Rahman et al. [12] recommended "DistB-SDCloud", a framework for improved cloud security for smart IoT applications. This framework utilizes a distributed BC methodology for providing integrity, privacy, and security, although remaining scalable and flexible. Clients in the industrial sector take advantage of the decentralized or dispersed, and effective BC infrastructure. Furthermore, defined

an SDN approach for improving the resilience, steadiness, and load balance of the cloud environment.

III. PROPOSED SYSTEM ARCHITECTURE

This paper develops an AICAD-AEMDRT system. The aim of AICAD-AEMDRT system is to develop an AI-driven approach for detecting and mitigating cyber-attacks in IoT and cloud environments to enhance security and system resilience. To achieve that, the proposed AICAD-AEMDRT approach involves various level of data pre-processing, FS process, and ensemble classification model. Fig. 1 represents the complete flow process of AICAD-AEMDRT approach.

A. Data Pre-Processing

Primarily, the min-max standardization system is used in the data preprocessing stage for transforming input data into a beneficial format. Min-Max Normalization normalizes transaction features between zero and one, enabling uniform distribution. This evades bias towards greater values and enhances the performance of model to improve gradient descent convergence in training. Min-Max Normalization guarantees data falling between zero and one.

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

B. Dimensionality Reduction Procedure

Owing to the higher feature quantity gained in the stage of extraction of the features, numerous features are equivalent, leading to feature redundancies, particular unwanted features need to be eliminated. Fisher score has imitate the significance of dissimilar attributes, creating the distance among instances of the similar class as few as promising and the distance among instances of dissimilar class as great as potential.

Describe $x^{(k)}$ to represent sample value x on k th feature, $m_i^{(k)}$ represents the mean of samples values of i on feature k , $d_i^{(k)}$ to designate the variance values of category i on the feature k , and $m^{(k)}$ to represent the sample mean of k . Based on this, the score of Fisher $F(k)$ represents the ratio of class variance of between- $S_B^{(k)}$ to from the- $S_W^{(k)}$.

$$S_B^{(k)} = \sum_{i=1}^c \frac{n_i}{n} (m_i^{(k)} - m^{(k)})^2 \quad (2)$$

$$S_W^{(k)} = \sum_{i=1}^c \frac{n_i}{n} (d_i^{(k)})^2 \quad (3)$$

After computing the Score of Fisher for the attributes, the features with high and low scores are maintained and removed according to the computation outcomes. The redundant features effects is decreased and the recognition precision was enhanced.

At present, PCA is the generally applied unsupervised linear feature selection technique. The basic concept is to remove the key characteristics from the data whereas maintaining additional features of the new data, such that the data of dimensionality reduction have the major variances in the predictable sizes to reach the aim of decreasing unwanted data. Several researches

have demonstrated that the usage of PCA technology can successfully increase the detection outcomes.

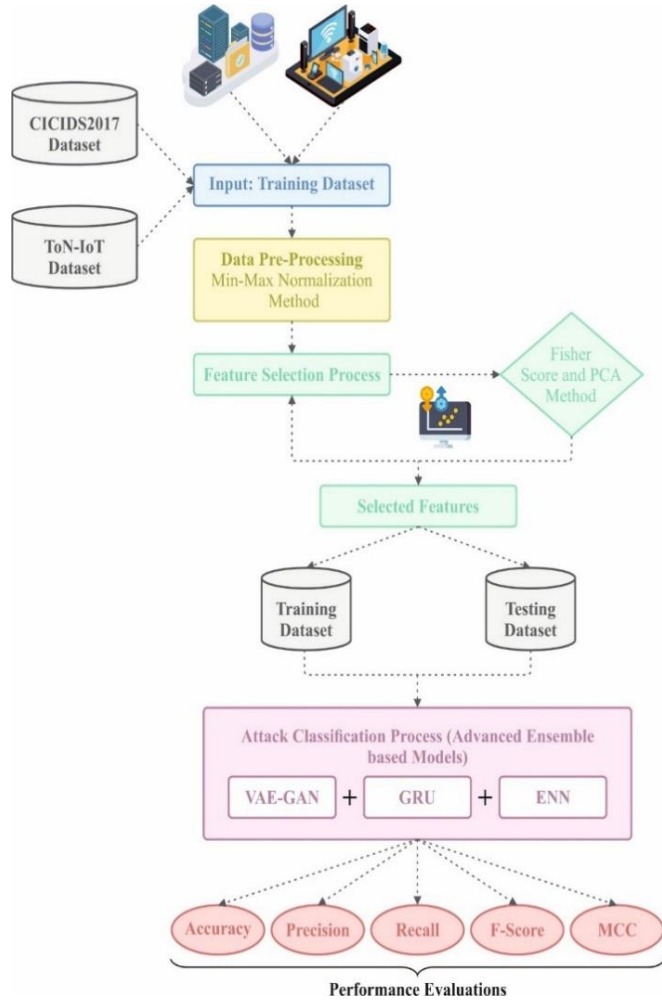


Fig. 1. Overall work flow process of AICAD-AEMDRT model.

The sample data were normalized by finishing the process of demeaning the data. Next, the matrix of covariance was measured. The eigenvectors and eigenvalues are measured for the matrix of covariance. The eigenvalues are organized depending on their dimensions and the novel principal component dataset was gained by maintaining the highly organized eigenvectors. Utilized PCA are sorted by the Score of Fisher that minimizes the similarities among features and further considerably decreases the feature dimensionality. It enhances the recognition precision in addition to the speed of the recognition.

C. Ensemble Classification Model

Eventually, the proposed AICAD-AEMDRT technique designs ensemble models namely the VAE-GAN method, GRU method, and ENN system for the attack classification process.

1) *VAE-GAN method*: VAE and GAN are 2 traditional generating methods. However, GAN creates higher-quality data, attaining the Nash equilibrium is inspiring, resulting in collapsed mode, while the generator provides difficult outputs. On the other hand, variance measures the VAE training loss

function that may lead to produced data that absences accuracy to the new features of the dataset. Still, the latent vectors learned in decoding are applied to create the new data. To completely use the beneficial characteristics of the two main generative methods, considered a highly reliable VAE-GAN method able to learn features. In the training model, the encoded E removes and compacts the instance attributes in an entire database and encodes into the hidden area z over linear networks, whereas z refers to significant data feature data be seized inactively. The created novel samples were given by generator G according to the latent variable description z , and $p(z)$ represents the normal standard distribution. By using variational inference, G retains posterior distribution $q(z|x)$ nearer to the predictable dispersion and selects KL as a function of loss to compute the distance among the dual distributions.

$$L = -E_{q(z|x)}[\log p(x|z)] + D_{KL}(q(z|x)||p(z)) \quad (4)$$

Here, discriminator D and encoder E make GAN, and encoding is reflected as generator G . Unlike conventional GAN samples, this generator G creates instances related to the training instances. However, to determine the precision of the created data, the discriminator further categorizes the data.

$$L_{GAN} = \log(D(x)) + \log(1 - D(G(z))) \quad (5)$$

An individual networking architecture of VAE-GAN allows this generator to remove deeper data features gained by the encoding. This encoding bonds z and actual instance x , thus providing the discriminator the capability to identify dissimilar data attributes. Incorporating the benefits of dual conventional networks not only guarantees the model stability, but also guarantees the created data qualities. Thereby, able to remove deeper data features.

2) *GRU model*: GRU is an improved substitute to LSTM network. GRU [13] aims to tackle the vanishing gradient problem in RNN that degrades the tasks connected to long-term dependency. The GRU comprises 3 major elements, comprising update, reset, and input gates. The input gate assesses the innovative input combines with the existing hidden unit. The reset gate defines the level of a previous hidden unit is rejected.

a) *Input gate*: It exhibits how much novel input data must be combined to present hidden units. Either previously hidden unit or existing input. Additionally, it generates a value between zero and one that depicts the level of information to be incorporated with current hidden unit. The value 0 shows that innovative data is incorporated with present hidden unit. Conversely, the value 1 represents novel information comprised of this hidden unit.

$$i_t = \text{sigmoid}(W_i[h_{t-1}, x_t] + b_i) \quad (6)$$

Now the previous hidden unit is depicted as h_{t-1} . The variable x_t indicates existing input, and the sigmoid function are demonstrated by sigmoid. W_i and b_i refers to weigh and bias. The output gate is i_t , and this output is employed to compute hidden unit ($\tilde{h}_t$) which comprises an innovative input. This process has been measured to employ the succeeding equation.

$$\tilde{h}_t = \tanh(W[h_{t-1}, i_t \times x_t] + b) \quad (7)$$

Here the hyperbolic tangent activation function has been depicted *tanh*.

b) *Update gate*: This gate measures how many previous hidden units must be kept and how many novel exterior inputs must be combined to existing hidden units. Either the previous hidden unit or current input is taken inputs. Additionally, this gate produces a value among zero and one that denotes the proportion of preceding hidden unit. The value of zero indicates no data from the earlier hidden unit has been retain, whereas a value of one signifies the information from the prior hidden unit are completely recollected. Generally, the current state is measured to employ the succeeding equation:

$$z_t = \text{sigmoid}(W_z[h_{t-1}, x_t] + b_z) \quad (8)$$

Now the prior hidden unit is h_{t-1} . The activation function of sigmoid refers to *sigmoid*. b_z and W_z refers the biased and weighted. The variable x_t indicates existing input. Besides, the result of the present gate was utilized to assess the existing hidden unit (h) instead of a present hidden unit that is represented by h_{t-1} . A wide range of hidden units are allocated weight h_t . The current hidden unit is calculated:

$$h_t = z_t \times h_{t-1} + (1 - z_t) \times \tilde{h}_t \quad (9)$$

The present gate allows the model to enhance the hidden unit selectively based on input through diverse time periods.

c) *Reset gate*: The GRU controls the degree of previously hidden unit must be extracted. This gate assesses either existing input or the resultant hidden unit. It produces a value within the interval of zero to one and specifies how many can be removed. The value of 0 exhibits that subsequent hidden unit is discarded, and 1 depicts previous hidden unit is retained. The existing gate is computed to employ the succeeding equation:

$$r_t = \text{sigmoid}(W_r[h_{t-1}, x_t] + b_r) \quad (10)$$

Here the output of this gate is r_t utilized to calculate the enhanced previous hidden unit ($\tilde{h}_t$) that comprises novel input.

$$\tilde{h}_{t-1} = r_t \times h_{t-1} \quad (11)$$

Now, the reset gate has been intended to combine the existing hidden unit ($\tilde{h}_t$) with previous hidden unit ($\tilde{h}_{t-1}$). It enables the network to disregard segments of preceding hidden units which is not related to existing input selectively; additionally, it aids to reduce issues connected to overfitting inside the training data. The gating system aids in evading either vanishing or exploding gradient issues in longer sequence that make it simpler to train the network.

3) *ENN technique*: ENN [14] is a kind of RNN that is employed in various applications namely time-series prediction, attack detection, and more. Unlike other feedforward NN that just associate the movement in a single direction (input to output layers), RNNs have response links that enable them to employ preceding results as input to existing iterations. Generally, this method comprises three layers such as output and input layers, and a hidden layer (HL). The input layer attains an input to the network and the output layers offer

solutions. HL is a major distinctive factor of ENN. It comprises feedback links from output to input, allowing it to keep the data from the preceding iteration. Within training procedure, the output and input pairs were projected, and the output is in comparison with the preferred output.

$$I(t) = \{f_1, f_2, f_3, \dots, f_n\} \quad (12)$$

Now $I(t)$ indicates the set of input features at time, signifies the removed features, and denotes overall feature counts of a set. The network handles every vector feature and upgrades the state of HL depending on the preceding state of HL and present input. The HL of ENN has been depicted in the succeeding equation.

$$H_i(t) = A_n(U * I(t) + W * H(t - 1)) \quad (13)$$

Now A_n indicates activation function, $H_i(t)$ signifies HL at a time, W represents the weight matrix among HL and U depicts the weighted matrix among input state and HL. The output layer of ENN has been specified.

$$O_p(t) = R_n(V * H(t)) \quad (14)$$

Here R_n represents activation function, V indicates the weighted matrix among output layer and HL and O_p denote output at time. The training of ENN modifies the weighted matrices, W and V to decreases the error among predicted and actual outputs. The error was computed to employ cost function and back propagated over the system to upgrade weight.

$$A_{Cl} = \begin{cases} \text{if } (I_{ap} = T_{ap}); \text{Attack} \\ \text{else}; \text{Benign} \end{cases} \quad (15)$$

Here I_{ap} represents incoming network traffic data, T_{ap} indicates the trained attacking pattern and A_{Cl} signifies classification function of the attack.

IV. PERFORMANCE ANALYSIS

The experimental analysis of AICAD-AEMDRT model is inspected under two databases, namely, CICIDS2017 [15] and ToN-IoT [16]. This database contains total of 17500 samples under seven classes as revealed in Table I. The no. of attributes is 78 but only 30 attributes are chosen.

TABLE I. DETAILS OF CICIDS2017 DATABASE

CICIDS2017 Database	
Class	No. of Samples
"BruteForce"	"2500"
"DoS"	"2500"
"WebAttacks"	"2500"
"Infiltration"	"2500"
"Bot"	"2500"
"DDoS"	"2500"
"PortScan"	"2500"
Total Samples	17500

Fig. 2 depicts the classifier outcomes of AICAD-AEMDRT approach on CICIDS2017 database. Fig. 2a-2b demonstrations the confusion matrix with classification of seven classes at

70%:30%. Fig. 2c displays the PR analysis, signifying maximal performance across every class. Ultimately, Fig. 2d demonstrates the ROC outcome, showing efficacious results with superior rates of ROC for 7 classes.

Table II defines the attack recognition of AICAD-AEMDRT algorithm under CICIDS2017 database. Based on 70%TRPHE, the AICAD-AEMDRT gains an average $accu_y$ of 99.41%, $prec_n$ of 97.95%, $recal_l$ of 97.94%, F_{score} of 97.94%, and MCC of 97.60%. Similarly, on 30%TSPHE, the AICAD-AEMDRT accomplishes an average $accu_y$ of 99.40%, $prec_n$ of 97.90%, $recal_l$ of 97.92%, F_{score} of 97.90%, and MCC of 97.55%.

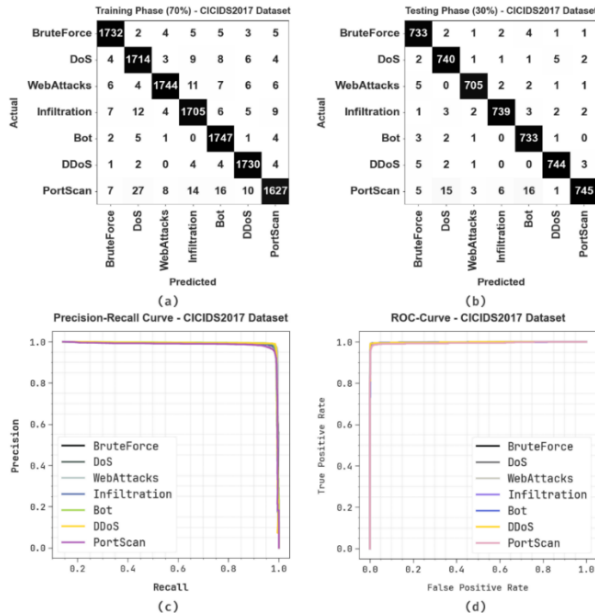


Fig. 2. CICIDS2017 database (a-b) Confusion matrices and (c-d) PR and ROC curves.

TABLE II. ATTACK DETECTION OF THE AICAD-AEMDRT ON CICIDS2017 DATABASE

Class Labels	$Accu_y$	$Prec_n$	$Recal_l$	F_{score}	MCC
TRPHE (70%)	99.41	97.95	97.94	97.94	97.60
TSPHE (30%)	99.40	97.90	97.92	97.90	97.55

Table III and Fig. 3 exemplify the comparative study of AICAD-AEMDRT on the CICIDS2017 database with current techniques [17, 18]. The outcomes underlined that the proposed AICAD-AEMDRT model got the highest $accu_y$, $prec_n$, $recal_l$, and F_{score} of 99.41%, 97.95%, 97.94%, and 97.94%, respectively. While the present approaches, namely LSTM, GRU, Modified Bi-LSTM, Random Forest, SMOTE-AdaBoost, DNN, and HFS+Light GBM, have shown worse outcomes.

TABLE III. COMPARATIVE ANALYSIS OF THE AICAD-AEMDRT MODEL ON CICIDS2017 DATABASE

CICIDS 2017 Database				
Technique	$Accu_y$	$Prec_n$	$Recal_l$	F_{score}
LSTM	99.11	97.62	95.71	95.11

GRU	95.35	89.96	94.12	93.25
Modified Bi-LSTM	98.47	95.92	89.88	97.56
Random Forest	95.56	92.90	96.03	95.03
SMOTE-AdaBoost	99.39	96.89	89.68	95.43
DNN	92.85	93.56	95.29	96.33
HFS+Light GBM	98.57	92.23	91.86	94.83
AICAD-AEMDRT	99.41	97.95	97.94	97.94

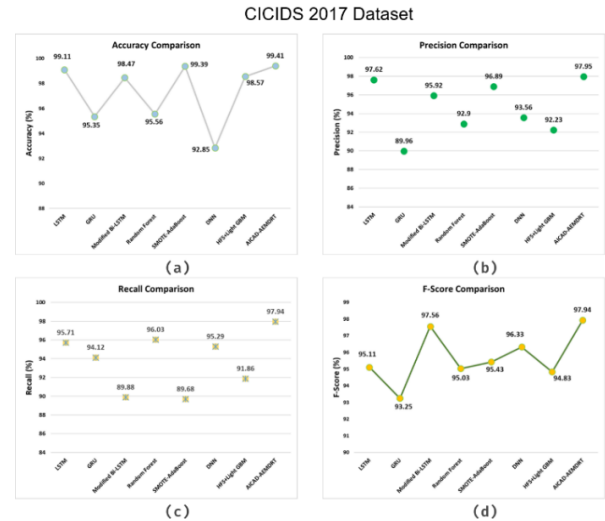


Fig. 3. Comparison analysis of AICAD-AEMDRT model on CICIDS2017 database.

TABLE IV. DETAILS OF ToN-IoT DATABASE

ToN-IoT Database	
Classes	No. of Samples
“Normal”	“78369”
“MiTM”	“336”
“DoS”	“5440”
“DDoS”	“5987”
“Password”	“6016”
“Injection”	“5867”
“XSS”	“5951”
“Ransomware”	“5976”
“Backdoor”	“6015”
Total Instances	119957

The ToN-IoT database comprises 119957 instances in total under nine classes as presented below in Table IV. The number of attributes is 42 but only 25 attributes are chosen.

Fig. 4 represents the classifier outcomes of AICAD-AEMDRT under ToN-IoT database. Fig. 4a-4b illustrates the confusion matrices with accurate recognition of nine class labels under 70%:30%. Fig. 4c exhibits the PR examination, specifying greater outcome 9 each class. However, Fig. 4d explains the ROC inspection, exhibiting skillful performance with increased outcomes of ROC for nine classes.

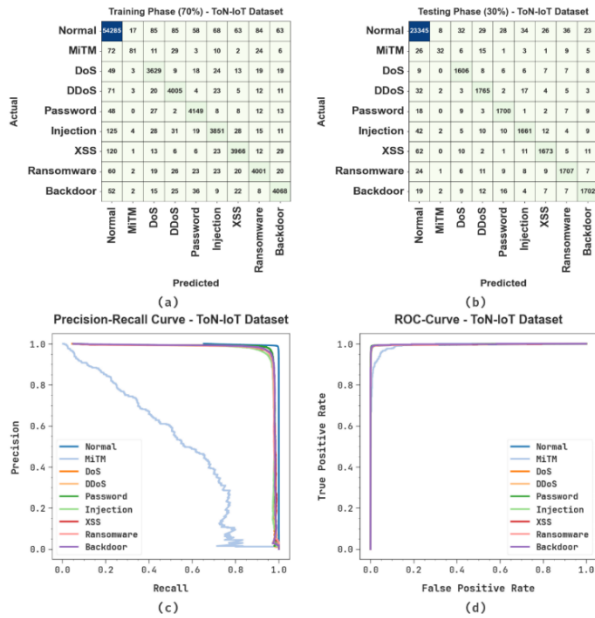


Fig. 4. ToN-IoT database (a-b) Confusion matrices and (c-d) PR and ROC curves.

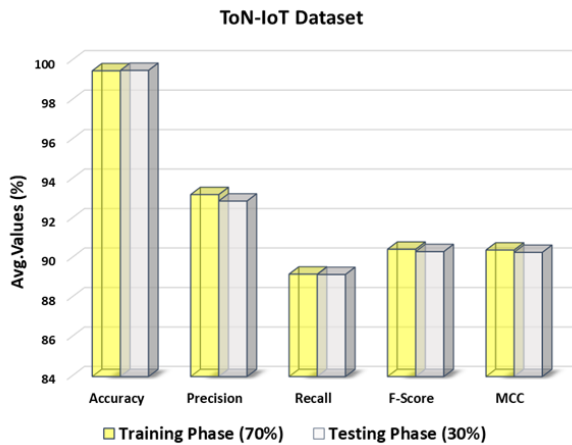


Fig. 5. Average of AICAD-AEMDRT on ToN-IoT database.

Fig. 5 portrays the attack recognition of AICAD-AEMDRT approach on ToN-IoT database. With 70%TRPHE, the proposed AICAD-AEMDRT technique obtains average $accuracy$ of 99.49%, $prec_i$ of 93.22%, $recal_i$ of 89.19%, F_{Score} of 90.45%, and MCC of 90.41%. Likewise, at 30%TSPHE, the AICAD-AEMDRT algorithm obtains an average $accuracy$ of 99.51%, $prec_i$ of 92.89%, $recal_i$ of 89.17%, F_{Score} of 90.33%, and MCC of 90.29%.

Table V and Fig. 6 depict the comparative investigation of AICAD-AEMDRT system under ToN-IoT database with current methods. The table values underlined that the present systems, namely, NB, Health Guard, CNN and GRU, CuDNNLSTM, GRU, LSTM, and RNN, got worse performance. Whereas, the proposed AICAD-AEMDRT model attained the highest $accuracy$, $prec_i$, $recal_i$, and F_{Score} of 99.51%, 92.89%, 89.17%, and 90.33%, appropriately.

TABLE V. COMPARATIVE STUDY OF AICAD-AEMDRT SYSTEM ON ToN-IoT DATABASE

ToN-IoT Database				
Algorithm	$Accur_y$	$Preci_n$	$Recal_l$	F_{Score}
NB Method	90.62	90.66	81.74	83.71
Health-Guard	88.68	92.06	83.94	84.32
CNN and GRU	98.90	88.47	85.68	89.73
CuDNNLSTM	98.81	80.99	80.37	82.60
GRU	98.96	80.38	82.33	85.22
LSTM	98.94	87.82	81.48	89.19
RNN	98.91	87.35	85.97	83.31
AICAD-AEMDRT	99.51	92.89	89.17	90.33

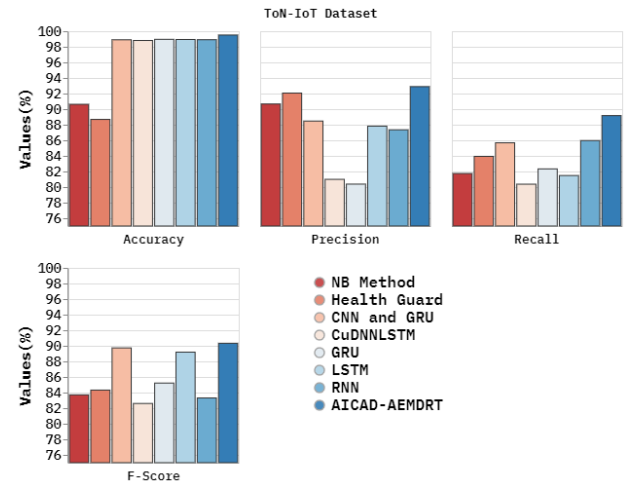


Fig. 6. Comparison analysis of AICAD-AEMDRT model on ToN-IoT database.

V. CONCLUSION

In this work, an effective cyberattack detection AICAD-AEMDRT model has been presented for detecting and mitigating cyber-attacks in IoT and cloud platforms to enhance security and system resilience. At the primary stage, the min-max standardization technique is applied in the data preprocessing step to convert input data into a valuable format. Followed by, the Fisher score and PCA method for the feature reduction to elect the most appropriate attributes. Eventually, the proposed AICAD-AEMDRT technique designs ensemble models, namely, the VAE-GAN method, GRU method, and ENN system for the attack detection model. The experimental assessment of the AICAD-AEMDRT arises utilizing a benchmark database. The performance results specified the greater result of the AICAD-AEMDRT model compared to recent systems.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in Kaggle repository at <https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset>, <https://www.kaggle.com/datasets/dhoogla/cictoniotreference> number, referenced number [15, 16].

ACKNOWLEDGMENT

The authors extend their appreciation to Prince Sattam bin Abdulaziz University for funding this research work through the project number (PSAU/ 2025/01/37848).

REFERENCES

- [1] S. Jayadatta, "A study on latest developments in artificial intelligence (AI) and internet of things (IoT) in current context," *J. Appl. Inf. Sci.*, vol. 11, no. 2, pp. 21–28, 2023.
- [2] H. A. Alterazi, "An Intelligent Cybersecurity Framework for IoT Environments Using a Multi-Stage Heuristic Algorithm Fused with a Spatiotemporal Attention Mechanism," *Eng. Technol. Appl. Sci. Res.*, vol. 16, no. 1, pp. 31970–31975, Feb. 2026.
- [3] A. R. P. Reddy, "The role of artificial intelligence in proactive cyber threat detection in cloud environments," *NeuroQuantology*, vol. 19, no. 12, pp. 764–773, 2021.
- [4] S. A. Alshaya, "IoT Device Identification and Cybersecurity: Advancements, Challenges, and an LSTM-MLP Solution," *Eng. Technol. Appl. Sci. Res.*, vol. 13, no. 6, pp. 11992–12000, Dec. 2023.
- [5] N. U. Prince, M. A. Faheem, O. U. Khan, K. Hossain, A. Alkhayyat, A. Hamdache, and I. Elmouki, "AI-powered data-driven cybersecurity techniques: Boosting threat identification and reaction," *Nanotechnol. Percept.*, vol. 20, S10, 2024.
- [6] A. A. A. H. Alzubaidi, "Cyber Attack Classification on IOT Devices Using Federated Machine Learning Infrastructure and AI," *Eng. Technol. Appl. Sci. Res.*, vol. 15, no. 6, pp. 30470–30476, Dec. 2025.
- [7] N. Kshetri, "Transforming cybersecurity with agentic AI to combat emerging cyber threats," *Telecommun. Policy*, p. 102976, 2025.
- [8] V. K. Kasula, A. R. Yadulla, B. Konda, and M. Yenugula, "Fortifying cloud environments against data breaches: A novel AI-driven security framework," *World J. Adv. Res. Rev.*, vol. 24, pp. 1613–1626, 2024.
- [9] R. Vadisetty and A. Polamarasetti, "Generative AI-Driven Distributed Cybersecurity Frameworks for AI-Integrated Global Big Data Systems," in *2024 Int. Conf. Emerg. Technol. Innov. Sustain. (EmergIN)*, pp. 595–600, Dec. 2024.
- [10] D. Saxena, I. Gupta, R. Gupta, A. K. Singh, and X. Wen, "An AI-driven VM threat prediction model for multi-risks analysis-based cloud cybersecurity," *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol. 53, no. 11, pp. 6815–6827, 2023.
- [11] Y. G. Hassan, A. Collins, G. O. Babatunde, A. A. Alabi, and S. D. Mustapha, "AI-powered cyber-physical security framework for critical industrial IoT systems," *Machine Learning*, p. 27, 2023.
- [12] A. Rahman, M. J. Islam, S. S. Band, G. Muhammad, K. Hasan, and P. Tiwari, "Towards a blockchain-SDN-based secure architecture for cloud computing in smart industrial IoT," *Digit. Commun. Netw.*, vol. 9, no. 2, pp. 411–421, 2023.
- [13] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the Properties of Neural Machine Translation: Encoder–Decoder Approaches," *arXiv:1409.1259*, 2014.
- [14] J. L. Elman, "Finding structure in time," *Cogn. Sci.*, vol. 14, no. 2, pp. 179–211, 1990, doi: 10.1016/0364-0213(90)90002-E.
- [15] <https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset>
- [16] <https://www.kaggle.com/datasets/dhoogla/cictoniot>
- [17] X. Hu, X. Meng, S. Liu, and L. Liang, "An improved algorithm for network intrusion detection based on deep residual networks," *IEEE Access*, 2024.
- [18] M. Maaz, G. Ahmed, A. S. Al-Shamayleh, A. Akhunzada, S. Siddiqui, and A. H. Al-Ghushami, "Empowering IoT resilience: hybrid deep learning techniques for enhanced security," *IEEE Access*, 2024.