RESEARCH

Enhancing IoT Network Security by Anomaly Detection and Intrusion Prevention Using Gannet Optimization-Based Adaptive Deep Capsule Network

WeiWei Hu[1] · Jafar A. Alzubi[2] · J. Shreyas[3] · Muna Al-Razgan[4] · Yasser A. Ali[5] · A. Karthikayan[6]

Received: 11 August 2024 / Revised: 14 July 2025 / Accepted: 23 July 2025

Abstract

The grouping of various interconnected devices along distinct applications leads the Internet of Things (IoT) to more vulnerability to security threats that affect the security of the data. Thus, Intrusion Detection Systems (IDS) are needed for IoT to mitigate cyber threats. A proven performance is offered by the deep learning models in detecting network traffic, minimizing the effects of cyberattacks, and providing enhanced security to IoT devices. Thus, this paper aims the implementation of anomaly identification and network intrusion prevention in IoT systems. Three main steps are involved in the developed framework. The first step is to get the necessary data from the publicly accessible data source. Once the necessary data is collected, the best and most appropriate weighted features are obtained from the input data. The optimal weighted features are obtained with the aid of a newly introduced Improved Gannet Optimization Algorithm (IGOA), which is responsible for optimizing the weight necessary to fuse the features to form the weighted fused features to assist in the upcoming intrusion detection procedure. To find the anomaly in the network, the weighted fused features are given as input to run via the Adaptive Deep Capsule Network (ADCapsNet). The generated IGOA is used to tune the hyper-parameter in the ADCapsNet framework to increase the detection performance. Then, the necessary actions are taken to prevent these intrusions from the network. In the end, the implemented model is evaluated by contrasting it with various traditional anomaly detection models.

Keywords Intrusion detection · Internet of things · Anomaly detection · Adaptive deep capsule network · Intrusion prevention · Improved gannet optimization algorithm

Abbreviations

1DCNN	One-dimensional convolutional neural network
ADCapsNet	Adaptive deep capsule network
AUC	Area under curve
CEA	Camellia encryption algorithm
CNN	Convolutional neural network
DNN	Deep neural network
DT	Decision tree
DTCN	Deep temporal convolutional network

https://doi.org/10.1007/s44196-025-00940-2

Published online: 01 October 2025

 https://doi.org/10.1007/s44196-025-00940-2

ESOA	Egret swarm optimization algorithm
FDR	False discovery rate
FFNN	Feed forward neural networks
FL	Federated learning
FNR	False negative rate
FPR	False positive rate
GAN	Generative adversarial networks
GOA	Gannet optimization algorithm
IDS	Intrusion detection system
IGOA	Improved gannet optimization algorithm
IIF	Improved isolation forest
IoT	Internet of things
KNN	K-Nearest neighbors
LCNN	Lightweight convolutional neural network
LSTM	Long short-term memory
MCC	Matthews correlation coefficient
MBO	Mine blast optimization
MLP	Multi-layer perceptron
NB	Naïve bayes
NN	Neural network
NPV	Negative predictive value
OSMA	Opposition slime mould algorithm
RandNN	Random neural networks
SOM	Self-organizing map
SVM	Support vector machine
TPR	True positive rate
VPN	Virtual private network

1 Introduction

IoT has been broadly used in different applications ranging from transportation to healthcare and smart city applications. Thus, the popularity of IoT has risen in the past few years. These interconnected physical devices are termed as "things" in the IoT setting. Anomalies are widely present in the IoT system, which has to be mitigated to prevent the data from possible security threats. The anomalies in the IoT system are detected by means of an Intrusion Detection System (IDS). Usually, this task of anomaly detection or IDS is done in the IoT environment with the aid of deep learning techniques [1]. An illegal or harmful activity that is performed on the IoT system to cause issues for data availability, confidentiality, and integrity is called an IoT intrusion. The integrity as well as the confidentiality of the data that is being transmitted over the IoT system is maintained with the inclusion of encrypted and secure communication channels by incorporating a Virtual Private Network (VPN) system [2]. As a result of the occurrence of several intrusions, the security of the IoT system is prone to privacy risks. To prevent the occurrence of such activities, both cybersecurity as well as physical security have to be monitored continuously throughout the IoT environment [3]. Besides, an additional scheme to protect the privacy of the entire IoT system by means of thorough evaluation is required. This is because the number of cyberattacks in the IoT environment has grown significantly, and the motivation for these attacks is also higher nowadays [4].

However, conventional Information Technology (IT)-based privacy prevention schemes like signature-based anomaly identification are not suitable for IoT environments because of their massive structure, incorporation of various software as well as hardware, and their implementation in uncontrolled environments [5]. Thus,

the prevention and detection of intrusion both play a key role in the IoT system. Most of the IDS focus on detecting anomalies; however, none of them focus on preventing such intrusions, thus leading to less security and more vulnerability to the data [6]. As there are no defense schemes like Honeypot or Blockchain, intruders can easily attack the target nodes, thus resulting in reduced security in the IoT system [7]. Enhancing cybersecurity in such essential networks is an important task to achieve all over the world. Thus, the initial stage in developing a cybersecurity system will be the identification of possible intrusions [8]. The resource-constrained nature, like the limited storage capacity, minimum battery life, and less computation of the IoT makes it difficult to execute an effective cybersecurity mechanism into its structure [9]. It is also noteworthy to mention the higher false positives and false alarms that are caused by the conventional IDS. These false rates make the existing IDS a complicated system to incorporate with the IoT system as the management of these IDS becomes a difficult task [10]. On a normal operation, IDS can generate hundreds of false alarms despite the fact that they are developed as a signature-based or anomaly-based system. This issue of higher false rates is seen in many existing network IDS [11].

Generally, resource-constrained devices use minimal processing efficiency with storage to execute specific tasks. Highly efficient and throughput-based Stream Processing Engine (SPE) techniques are commonly employed for the IoT–based devices for processing enormous and high-speed data in the network. Issues that arise due to spatial big data in resource-constrained devices are resolved by introducing cloud-based geospatial SPE techniques [38]. In resource-constrained devices, user-defined spatial observations are performed for transferring a certain set of rules to other cloud nodes. Throughput is termed as a major issue in the resource-constrained devices, which include the lightweight scheduling procedures with runtime measures. The cloud-based geospatial SPE techniques offer better support to large-scale devices by improving the throughput for accomplishing edge intelligence. Moreover, the multi-access edge computing techniques have the efficiency to enhance the performance of offloading tasks in the edge servers. The edge computing model provides huge support to different applications like unmanned aerial vehicles, healthcare, autonomous vehicles, and also for security. In the resource-constrained device, an offloading technique named Scheduled Multi-agent Deep Reinforcement Learning (SMDRL) [39] is employed to offer better suggestions and discussions in the resource-constrained regions. Designing a virtual energy queue technique for edge computing models helps to increase the quality of experience in the network. The major goal of SMDRL is to resolve the service delay and also the consumption of energy. In edge intelligence technology, deep learning and edge computing models are employed to advance the performance of the network edge. TreeNet [40] is a deep learning-based resource-constrained device employed to minimize the validation cost for edge devices. Moreover, the complexities of the network are resolved by including the disjoint sub-tasks, which aid in reducing network complexity. The deep learning-based TreeNet technique accomplishes higher processing speed with minimal error than other resource-constrained devices.

Nowadays, deep learning and machine learning are used in various applications including network intrusion identification, anomaly identification, and intrusion prevention [12]. With the aid of certain machine learning models like K-Nearest Neighbors (KNN), Self-Organizing Map (SOM), Naïve Bayes (NB), Support Vector Machine (SVM), Neural Network (NN), and Decision Tree (DT), the features of the packets that are continuously evaluated [13]. This enhanced development in anomaly detection and pattern recognition tasks is possible as a result of enhancements in artificial intelligence technology [14]. With the help of this artificial intelligence technology, security against threats and cybersecurity are carried out in an enhanced manner. Deep Neural Network (DNN) is widely used in the task of identification and classification of data to perform anomaly based intrusion detection tasks [15]. Thus, deep learning-based IDS have to be developed for the IoT system to carry out real-time anomaly detection tasks. As a result of the complicated and resource-constrained nature of the IoT, conventional IDS are not enough to offer security to these IoT devices [16].

Cloud-based structure makes these machine-learning-based IDS a complicated thing because of their heterogeneous and widespread nature [17]. The network's training is highly affected by issues like load balancing, higher consumption of bandwidth, and network resource congestion, which may lead to high latency, packet loss, traffic peaks, and transmission delays. All these problems make the machine learning-based IDS

a complicated task to achieve in the cloud-based environment, making the overall training infeasible. This is solved using deep learning technology. Thus, a deep learning-oriented IDS and intrusion prevention model for cloud-based IoT systems is developed in this paper.

The main objectives of this developed intrusion detection model are provided as follows.

- To generate an enhanced model for anomaly detection on IoT platforms using an advanced deep learning approach supported by heuristic techniques to provide enhanced protection on data against various malicious intruders in IoT systems.
- To implement a heuristic approach called the IGOA to tune the parameters in the CapsNet framework and to tune the weights required to execute the weighted feature fusion process to assist in the accurate detection of the intrusion in the IoT system.
- To generate a deep learning approach called ADCapsNet to perform the binary classification task of anomaly detection to take preventive measures to mitigate the intrusions and anomalies effectively from the IoT network.
- To validate the performance offered by the executed anomaly detection and mitigation framework by comparing its performance with various existing approaches.

This paper is arranged as follows. In Sect. 1, a detailed introduction to the topic of IDS and anomaly detection is provided. In Sect. 2, a concise review of various traditional anomaly detection frameworks and IDS is provided. In Sect. 3, the automatic anomaly detection and prevention framework using an optimized deep learning network for enhancing IoT network security is provided. The utilization of the weighted fused features for anomaly detection with the support of an improved optimization mechanism by analyzing the anomaly dataset is given in Sect. 4. The formulation of the deep learning-based anomaly detection and prevention for security enhancement in IoT networks is specified in Sect. 5. The results and discussion regarding the implemented technique are provided in Sect. 6. This work is concluded in Sect. 7.

2 Literature Survey

2.1 Related Works

In 2020, Ali et al. [18] have explored the presence of an attacker by validating the flow, user, and packet in a three-tier Intrusion Detection and Prevention System (IDPS). The work aimed to detect the compromised devices by keenly evaluating the packets in the system. The validation of the IoT user in the first tier was done using routers by implementing the technique of encrypted signatures and RFID tags. Then, a type-II fuzzy filter was used in the second tier along with switches to validate the authenticity of the data packets. The crucial attributes from these packets were then obtained to carry out the classification task. The packets that did not match with one another were evaluated with the aid of controllers. The packets in the suspicious queue from these controllers were then classified. The implemented model was simulated on OMNeT++ while considering factors such as delay, traffic load, rate of intrusion detection, throughput, and failure rate.

In 2023, Sharma et al. [19] have utilized DNN to act as anomaly-based IDS in the IoT system. The DNN model makes use of the features obtained from the filters. The highly correlated features were provided to this DNN model. Then, the tuning of the parameters in this DNN model was done to increase its overall detection performance. The implemented model was analyzed by the UNSW-NB15 dataset. With the utilization of this dataset, the executed model has attained around 84% detection accuracy. While on synthetic data generated by means of Generative Adversarial Networks (GANs), an accuracy of around 91% was attained.

In 2022, Saba et al. [20] have implemented anomaly-based IDS using a Convolutional Neural Network (CNN). These IDS utilized the power of IoT to perform the intrusion detection task by evaluating the traffic pattern in the

entire network. This model was capable of detecting any abnormality and intrusions by monitoring the network's traffic. While carrying out experimentation of BoT-IoT and NID Datasets, the implemented IDS has achieved around 93% and 99.5% accurate outcomes, accordingly.

In 2023, Ntizikira et al. [21] have implemented IDS using Honeypot and Blockchain technologies. The latency in communication was reduced using an edge computing system. At first, the Camellia Encryption Algorithm (CEA) was used to perform a three-tier authentication process. The gathered data were pre-processed using min–max normalization at the gateway to minimize the complexity and redundancy of data. The pre-processed data were then protected using a signature-based encryption method. Then the pre-processed data were classified using the Improved Isolation Forest (IIF) approach to detect the presence of an anomaly. On the edge level, the Honeypot was used to attract suspicious data to determine the patterns of the attackers. The behavior of the suspicious packets was classified using GAN, Lightweight Convolutional Neural Network (LCNN), and Multi-Layer Perceptron (MLP). After the detection of the anomalies, the prevention of these anomalies from attacking the IoT system was done by creating reports that were encrypted using the CEA technique. The HB-IDP was simulated on 3.26 (NS-3.26) and was evaluated to confirm its enhanced performance.

In 2023, Allah et al. [22] have implemented an IDS using Random Neural Networks (RandNN), Feed Forward Neural Networks (FFNN), and Long Short-Term Memory (LSTM). The complicated traffic patterns were handled using FFNN. The long-term dependencies in the traffic were captured using LSTM. The network data were learned using the RandNN framework. These frameworks provided essential security against several cyberattacks. The executed model was tested using the CIC-IoT22 dataset. The accuracy rate of 96.42%, 99.85%, and 99.93% was attained in detecting anomalies using RandFFNN, LSTM, and FFNN models, respectively.

In 2023, Sáez et al. [23] have implemented a framework to train various unsupervised structures to perform intrusion detection tasks on massive and distributed IoT and Industrial IoT (IIoT) systems. The overheads on the network and the isolation issues were tackled by implementing federated learning. This federated learning collaboratively trained the peers. The heterogeneity in federated learning was tackled by integrating an unsupervised device clustering approach with the federated learning pipeline. The executed model was executed and tested on a complicated network topology. This model was tested on real attacks that targeted the emulated devices.

In 2022, Pal et al. [24] have suggested a model to prevent and detect intrusions on the IoT ecosystem. The implemented scheme was an anomaly based IDS. The features that have a positive effect on the detection accuracy were initially identified by making use of a data filtering technique. This filtering technique has utilized the correlation coefficient to attain the best features. Then, by utilizing these features, the trust factor of the user was identified and classified using a classification approach. The implemented scheme on the NSL-KDD dataset has attained enhanced accuracy and reduced errors.

In 2022, Bacha et al. [25] have suggested a framework to thwart cyber attacks that arise in IoT environments. The suggested approach has utilized a dimensionality reduction approach to the features obtained from the data and has enhanced the performance of identifying the anomaly using the kernel principal component analysis mechanism. To perform the binary classification task, a kernel-based Extreme Learning Machine (ELM) was employed. Two benchmark datasets were utilized. Experimental outcomes proved the effectiveness of the executed scheme with respect to Area under Curve (AUC), specificity, F1-score, accuracy, and sensitivity.

2.1.1 Anomaly Detection Using Federated Learning

In 2024, Ibrahim et al. [49] have recommended a novel technique named Hybrid Differential Privacy with Federated Learning (HDP-FL) to enhance the privacy of the data. The developed technique helped to overcome privacy issues by adjusting certain parameters. Later, various performance computations were performed in the recommended framework over classical techniques in terms of security and privacy. In 2024, Adi et al. [50] have designed a novel federated learning technique for the IoT environment. The developed federated learning technique helped to advance the privacy, trust, and security of IoT. Moreover, a lightweight smart contract was used with Proof-of-Authority (PoA)-aided blockchain with a Gaussian differential privacy approach. Multiple performance

measures were suggested to observe the network efficiency. In 2024, Wang et al. [51] have implemented novel technique named Reliable Federated Learning With Privacy-Preserving (RFLPP) for IoT. Here, a Lightweight network was employed to protect the private data, and also an optimization technique was employed to enhance the network accuracy. Further, the privacy preservation models helped to fulfill privacy-related needs. The efficiency of the developed model was verified using privacy-based techniques. In 2024, Zhang et al. [52] have recommended Deep Federated Scattering Fingerprinting Aided by Differential Privacy (DFSF-DP) to offer better data privacy in the distributed training phase. Initially, a wavelet scattering network was used to extract the required features, and also the federated learning model eliminated the demands from the validation resources. In the experimental outcomes, the developed framework accomplished superior privacy and security than the traditional techniques.

2.1.2 Anomaly Detection Using Transformers

In 2024, Yao et al. [53] have implemented a new hybridized technique Partial Semantic Aggregation Vision Transformer (PSA-VT) for anomaly detection models. Initially, a pre-trained CNN was employed to acquire the local representation. Then, the feature reconstruction procedure was carried out through PSA-VT and then identified the anomaly was identified without any errors. In experiments developed model accomplished superior outcomes than others. In 2024, Kwon and Yu [54] have proposed Region-attentive Vision Transformer-based Autoencoder (RaViT-AE) to execute anomaly detection in images. In this phase, path projections were enhanced by applying the region attentive, which helped to learn the complicated patterns. In addition, a new loss function was considered for finding the higher-level semantic differences over the reconstructed and original images. The developed RaViT-AE gained superior results than conventional schemes in detecting the anomaly and offered robust outcomes. In 2024, Yao et al. [55] have designed a Dual-attention Transformer and Discriminative Flow (DADF) for the identification of anomalies. The DADF used a pre-trained network for collecting the multi-scale prior embeddings. Here, the ViT was employed with the dual attention procedures to accomplish local–global reconstruction. Moreover, the discriminative likelihood was maintained among the joint distribution in various scales. In experiments, superior outcomes were accomplished over prior schemes.

2.1.3 Anomaly Detection Using GAN

In 2023, Li et al. [56] have implemented Dilated Convolutional Transformer-based GAN (DCT-GAN) for improving anomaly detection. The developed DCT-GAN was suitable for tracking the time series information. DCT-GAN was efficient in improving the generalization efficiency. In addition, a generator was employed to attain the coarse and fine generated information through a dilated CNN. Experimental findings showcased superior outcomes in DCT-GAN than other schemes. In 2023, Ibrahim et al. [57] have designed an efficient scheme, Ensemble Active Learning Generative Adversarial Network (EAL-GAN) for detecting anomalies. EAL-GAN used a single generator with different discriminators, and also the anomalies were identified through an auxiliary classifier. Here, the conditional GAN was employed to produce the balanced training data. In experiments, EAL-GAN gained superior results with a high performance margin. In 2024, Cheng et al. [58] have suggested Attention Anomaly Detection Generative Adversarial Networks (Att-ADGANs). Here, the GAN was considered with an encoder as well as decoder structure that supports to acquire the feature of normal distribution. Progressive distillation procedure was considered through Series-Knowledge Distillation GAN (S-KDGAN) for eliminating the performance degradation. Analysis displayed that the suggested technique reduced the degradation issues and provided superior outcomes.

2.2 Problem Statement

IoT is a creative technology based on computer applications that is used in multiple sectors like transportation, smart homes, industries, and healthcare with various sensors. IoT has grown visibly and become an essential part of life. More devices are connected to the device through the internet, which makes the internet more susceptible

to many cyber attacks. Hence, distribution is the main task to prevent user privacy. In the early research, multiple techniques were used to detect anomalies and prevent intrusion. Their importance and issues are determined in Table 1. Type-II fuzzy [18] has the capacity to process linguistic uncertainties and helps to create the membership functions. However, it forms complications in time variability in the system. GAN [19] helps to solve the problems related to class imbalance in the dataset, but it makes the training slow and unstable, which leads to poor performance of the model. CNN [20] helps to enhance the IoT network security and performance. However, it takes much time to train the sequential data and requires a huge dataset to analyze the structure. The CEA algorithm [21] minimizes the difficulty present in the classification and feature extraction phase but takes more time for the encryption process. FFNN [22] has the capacity to handle difficult IoT networks, boost performance, and analyze the difficult non-linear correlation among the features. However, it has low performance in handling sequential data. FL [23] helps to train the large network of heterogeneous IoT devices, and it is not necessary to have human intervention. However, the process is more costly and difficult in a practical setting. Feature Engineering [24] helps to find out the attributes to predict the intrusion but it is very slow and labour-intensive and needs much training. Kernel principal component analysis [25] reduces the dimension of the feature data and

Table 1 Importance and issues in classical anomaly detection and intrusion prevention in the IoT networks

Author [citation]	Techniques	Advantages	Complications	Dataset	Performance Measures
Ali et al. [18]	Type-II fuzzy	It has the capacity to process linguistic uncertainties It helps to create the membership functions	It forms complications in time variability in the system	Publicly available benchmark dataset	Delay, failure rate, traffic load, throughput and detection rate
Sharma et al. [19]	GAN	It aids in overcoming class imbalance issues in the dataset	It makes the training slow and unstable, which leads to poor performance at the execution	UNSW-NB15	Accuracy 84%
Saba et al. [20]	CNN	It helps to enhance the IoT network security and performance	It takes a lot of time to train sequential data It requires a huge dataset to analyze the structure	Publicly available benchmark dataset	Accuracy
Ntizikira*et al.* [21]	CEA algorithm	It minimizes the difficulty present in the classification and feature extraction phase	It takes an enormous time for the encryption procedure	UNSW-NB15 and BoT-IoT	Security analysis
Allah et al. [22]	FFNN	It has the capacity to handle difficult IoT networks It boosts the performance and analyzes the difficult non-linear correlation among the features	It has low performance in handling sequential data	CIC-IoT22	Accuracy 96.42%
Sáez*et al.* [23]	FL	It helps to train the large network of heterogeneous IoT devices It is not necessary to have human intervention	The process is more costly and difficult in a practical setting	Traffic data	MSE and RMSE
Pal et al. [24]	Feature Engineering	It helps to find out the attributes to predict the intrusion	It is very slow and labour-intensive and needs much training	NSL-KDD	Accuracy 98.4%
Bacha*et al.* [25]	kernel principal component analysis	It decreases the dimension of the feature data It improves the performance of anomaly detection	It is more costly for large datasets It requires more computation	Benchmark dataset	Sensitivity, specificity, accuracy and Area under the curve

 https://doi.org/10.1007/s44196-025-00940-2

enhances the performance of anomaly detection. However, it is more costly for large datasets and requires more computation. Therefore, to sort out the above-mentioned issues, the paper proposed an anomaly detection and intrusion prevention model by a deep learning approach in the IoT sector.

3 Automatic Anomaly Detection and Prevention Framework Using an Optimized Deep Learning Network for Enhancing IoT Network Security

3.1 IoT Network Model

A collection of interconnected devices that use internet technology to facilitate communication between these interconnected devices is called the IoT. IoT is a collection of distinct sensing devices, software, and other technologies that are used to establish communication with other devices within the network to obtain an efficient transmission of data. This IoT technology can be found its application in multiple fields such as the military, the healthcare sector, smart city applications, smart home applications, and so on. IoT devices consist of a vast number of sensors that are deployed in various places throughout the network to gather information regarding the surrounding environment or to gather information from the task they are assigned. Then these sensors will communicate the collected information to the Base stations (BS). From the BS, the information is spread by means of the internet to distinct users, servers, and other required parties. The information that is gathered from these sensors is also sometimes stored in the database for future use. The typical IoT network is shown in Fig. 1

3.2 Security Challenges in IoT Network

Finding instances of data, groups, or incidents that move away from the usual pattern is the method of anomaly detection. Anything that differs from what is normal or what is anticipated is considered an anomaly. If the anomaly is not detected, it will result in the generation of faulty outcomes. Hence, anomaly detection is essential in various applications, which also suits well with IoT technology. In the past few decades, scientists have been

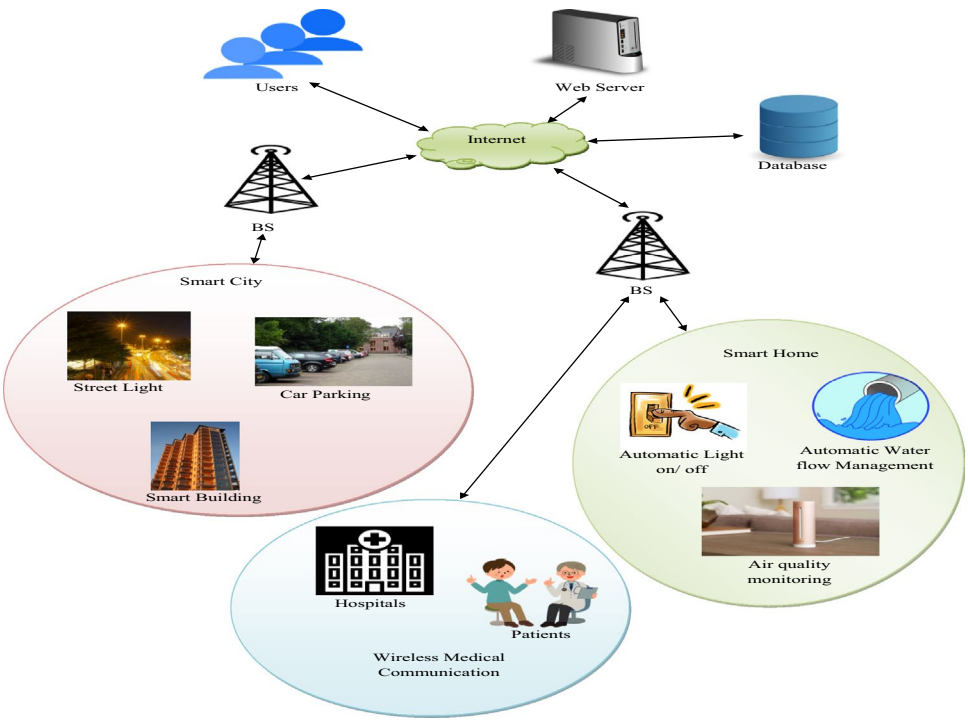

Fig. 1 IoT network model

automating this procedure with machine learning methods that aim to discover more effective ways to identify various kinds of anomalies. Anomaly detection is commonly employed in practice to identify suspicious activities, unanticipated prospects, or inaccurate data concealed inside the time series data. An unusual occurrence could point to fraud, criminal activity, network breach, or defective hardware in a system. There are numerous approaches to finding anomalies. However, machine learning techniques provide a solution for detecting them in an automated manner. Machine learning algorithms can be trained in a variety of strategies to find anomalies. When working with a labelled data set that distinguishes between normal and abnormal conditions, supervised machine learning approaches are most useful. Only recognized anomalies can be found using the supervised and semi-supervised approaches. However, almost all of the data lacks labels. Unsupervised anomaly detection algorithms, which can recognize unusual or rare events automatically, may be used in such instances. It is possible to train a wide variety of machine learning methods to recognize anomalies. Density-based techniques, such as K-Nearest Neighbour (KNN) and IF, identify an anomaly by comparing its density as the outlier has a larger and denser structure than the typical dataset. Using methods such as K-Means Cluster (KMC), which is a cluster-based anomaly detection method assesses the differences between every given point and clusters of related data to determine the anomaly. Based on relevant data, Bayesian network algorithms generate models to estimate the likelihood that an event will occur and then detect the major deviations from these prediction outcomes. A Neural Network (NN) is trained to anticipate an expected time series and then identify deviations from it. Each of these models can be applied to different scenarios and helps with anomaly identification. Models for automated anomaly detection are effective. Still, there are some difficulties. Scaling data infrastructure is necessary for handling anomalies with these machine learning approaches. Problems with the quality of the data can make anomaly detection less effective. Users may receive an abundance of misleading notifications from inadequate anomaly detection systems. Creating a reasonable baseline to account for common patterns that happen less frequently could take a while, which also has an effect on these machine learning models. Therefore, deep learning-based anomaly detection models are developed. One of the major problems in the development of the deep learning-based anomaly detection model is the quality of the dataset that has been utilized to train the model. Normally, the dataset may have redundant data, data of distinct formats, an incomplete dataset, the presence of null values, distinct measurements and scales, and a dataset made with human errors. Thus, an efficient model for detecting the presence of an anomaly in the IoT environment has to be developed using deep learning approaches.

3.3 Proposed Anomaly Detection and Prevention Model for Securing IoT Network

An innovative and simple network anomaly detection and intrusion prevention approach is implemented in the IoT-based environment with the utilization of deep learning techniques. Here, the CapsNet model is used as the basic building block of the executed anomaly detection model. In the primary phase of development, the essential data for training the suggested anomaly detection and intrusion prevention model is collected from various standard websites. These websites provide the dataset regarding various online anomaly events. The acquired data are then given for the feature extraction phase in which the most prominent and ideal set of features required for performing the network anomaly detection and intrusion prevention task is obtained. The feature extraction process assisted in the minimization of overall process steps and thus reduced the burden on the final detection model. The feature extraction also saves computation time by eliminating irrelevant and redundant data and facilitates the implemented model with only the required and significant attributes. The extracted features from the dataset are further used for the weighted feature fusion procedure. Following the extraction of features, the optimization of the weights of the features is carried out. The major goal behind the optimization of weights in the features is the reduction in the difficulty of the task that is being performed. This also eliminates the possible occurrence of losses. The optimization of weights is carried out using the suggested IGOA. The optimized weights received after being processed by the suggested IGOA are then multiplied by the extracted features to generate weighted features. The weighted features are finally provided as a source for the executed ADCapsNet-based anomaly

 https://doi.org/10.1007/s44196-025-00940-2

detection and intrusion prevention model. The enhanced ability of CapsNet to provide a detailed representation of the vectors makes it an ideal option for anomaly detection and intrusion prevention tasks in the IoT platform. The ADCapsNet utilizing the weighted features generates the final detected anomaly outputs. To boost the efficiency of the anomaly detection framework, the parameters such as the activation function, steps per epoch, and hidden neuron count are also optimized using the IGOA. The detected output is then utilized to take suitable preventive measures to prevent the occurrence of intrusion on the network. The architectural view of the executed anomaly detection and intrusion prevention model is depicted in Fig. 2.

In the developed framework, data regarding the IoT network components in the standard dataset are acquired for the analysis. Next, the collected IoT network data is passed to the feature extraction phase. Here, the significant features presented in the collected samples are extracted. Then, the weights presented in the extracted features are selected optimally through IGOA, where the weights are tuned optimally, which helps to enhance the relief score. Once the weights in the feature are selected optimally, and then the weighted features are offered to the anomaly detection phase. The developed framework used the ADCapsNet to classify the anomalies from the optimally weighted features. In ADCapsNet, different parameters such as activation function, steps per epoch and hidden neuron counts are tuned through IGOA. Here, the parameter tuning in ADCapsNet supports to maximize the MCC and accuracy and also effectively minimizing the FNR for providing more precise anomaly detection outcomes. Later, from the anomaly-detected outcomes, intrusion prevention procedures are carried out to mitigate

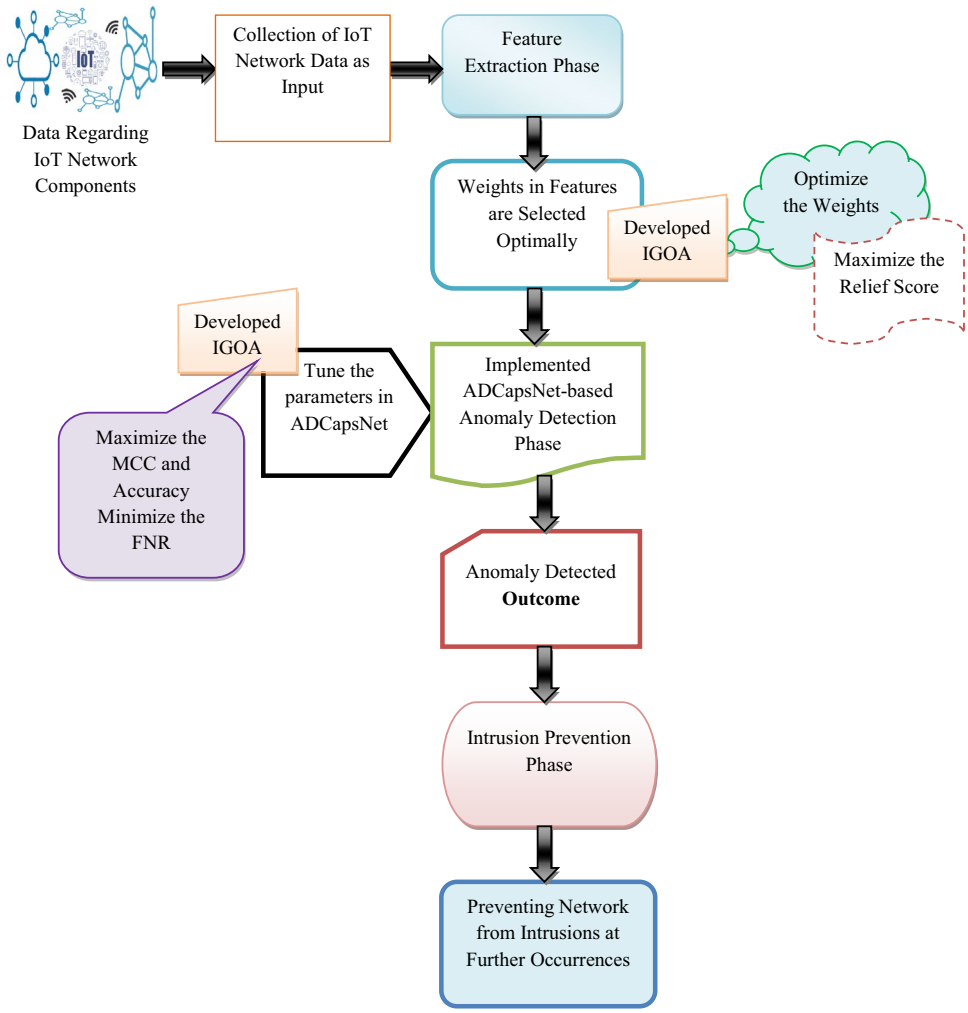

Fig. 2 Architectural view of the executed deep learning-based anomaly detection and intrusion prevention model

the intrusion from the IoT network. Finally, the intrusion prevention outcome is obtained from the developed framework.

4 Weighted Fused Features for Anomaly Detection with the Support of an Improved Gannet Optimization Algorithm by Analyzing Anomaly Dataset

4.1 Anomaly Detection Dataset: Description

The necessary data are gathered from distinct web pages. The collected data are represented by CD_{cd}^{col}, in which cd signifies the entire data count. The description of these datasets is given in Table 2.

4.2 Existing Optimization Algorithm: GOA

The algorithms that are deployed by considering the hunting activities undertaken by the gannet while searching and attacking the fish serve as the inspiration behind the development of the GOA [26]. Like most of the optimization algorithms, the GOA also has both exploitation and exploration phases. These two phases are further classified into four distinct modes, two for each phase. The diving mode in the U-shape and the diving mode in the V-shape come under exploration, whereas the arbitrary search and unexpected rotation come under the exploitation phase. Gannets are water birds that have a high liking for fish. Gannets usually search for food by flying at higher altitudes. When a fish is spotted by the gannet, it informs the entire flock about the presence of prey. Then, the gannet flock will create an array in a semi-circular or straight form to capture the prey. Then the gannets will dash into the water at high speed and try to capture the fish. The enhanced swimming ability of the gannet will assist it in capturing the fish inside the water body. As the gannets are capable of diving for longer distances even after the disappearance of their dive power, the gannets have a higher rate of success in hunting. The initial process in the GOA is the arbitrary initialization of the gannet population. At this stage, the feasible solution is considered to be the global optimal solution. The population matrix of the gannets is provided in Eq. (1).

$$A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,c} & \cdots & a_{1,C-1} & a_{1,C} \\ a_{2,1} & \cdots & a_{2,c} & \cdots & a_{2,C-1} & a_{2,C} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & a_{b,c} & \cdots & \cdots & \cdots \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ a_{B-1,1} & \cdots & a_{B-1,c} & \cdots & a_{B-1,C-1} & a_{B-1,C} \\ a_{B,1} & \cdots & a_{B,c} & \cdots & a_{B,C-1} & a_{B,C} \end{bmatrix} \tag{1}$$

Table 2 Dataset description on the gathered anomaly detection dataset

Sl. No	Dataset name	Availability	Description
1)	"CICIDS 2017 KNN"	"https://www.kaggle.com/code/saqlainhussainshah/cicids-2017-knn/data access date: 2023–12-20"	The benign and most recent common intrusions are included in this dataset, which closely reflects actual data
2)	"NSL-KDD dataset"	"https://www.unb.ca/cic/datasets/nsl.html access date: 2023–12-20"	One data set that is proposed to address some of the KDD'99 data set's inherent issues is NSL-KDD. There are eight data files in this dataset
3)	"Kddcup99"	"https://datahub.io/machine-learning/kddcup99 access date: 2023–12-20"	A total of 24 training intrusion types are present in the datasets, with an additional 14 types of stand-alone test data present in this dataset

In Eq. (1), the location at which the b^{th} gannet will be available is represented by the term a_b. The location at which a gannet is present in the search space can be computed with the aid of Eq. (2).

$$a_{b,c} = d_1 \times \left(D_c - E_c\right) + E; b = 1, 2, ..., B; c = 1, 2, ..., C \tag{2}$$

In Eq. (2), the search space's highest limit set in the dimension c is represented by the term $a_{b,c}$, the search space's minimum limit set in the dimension c is indicated by the term E_c, the population of gannets present in the population is denoted as B, size of the problem's dimension is indicated as C, and an arbitrary within [0, 1] is represented by the term d_1. An addition matrix called the memory matrix denoted by the term F is generated in the GOA after the initialization of the population. At first, the value of the population matrix A is assigned to this memory matrix. Then, this memory matrix will keep a record of the location of the gannets in every iteration. The fitness of the gannet's position in both the population matrix and the memory matrix is determined. If the fitness of the memory matrix F_b is more than the population matrix that is generated at present A_b, then the value of A_b is replaced by F_b. Or else, the values in the A will be utilized.

Exploration: In this phase, the behaviours utilized by the gannet while searching for fish in the water bodies are modelled mathematically. At first, the gannets will have a flight of around 10 m in height in the air to spot fish in the water bodies. After detecting a prey, the gannets will dive into the prey's position at high speed by utilizing either of the two dive modes as mentioned earlier. In U-shaped dive mode, the gannet will have a longer and deeper dive into the water, whereas in V-shaped dive mode, the dive will be shallower and of shorter length. The selection process between the two diving modes is given by Eq. (3).

$$H(a) = \begin{cases} \frac{1}{\pi} * a - 1 & a \in (\pi, 2\pi) \\ -\frac{1}{\pi} * a + 1 & a \in (0, \pi) \end{cases} \tag{3}$$

This is computed with the aid of Eq. (4).

$$e = 1 - \frac{g}{EG} \tag{4}$$

In Eq. (4), the term g indicates the ongoing iteration of the GOA, and the maximum limit of iteration count in the GOA is represented by the term EG. The parameter f is determined using Eq. (5), which demonstrates the U-shaped diving mode.

$$f = 2 * \cos\left(2 * \pi * d_2\right) * e \tag{5}$$

In Eq. (5),d_2 specifies the variable that is generated randomly within the limit [0, 1]. Parameter h values are computed as given in Eq. (6), which demonstrates the V-shaped diving mode.

$$h = 2 * H\left(2 * \pi * d_3\right) * e \tag{6}$$

In Eq. (6), d_3 represents the variable that is generated randomly within the limit [0, 1]. The same weight is provided to the chance in which the gannet will select the diving mode to capture the prey under the water. Since equal probability is assigned to these two diving modes, a new parameter i is generated at random to choose between these two diving modes arbitrarily. Once the diving method is determined, the process of updating the location of the gannets based on these diving patterns will take place. The process of updating the location of the gannets with respect to the type of diving mode being utilized is formulated mathematically with the help of Eq. (7).

$$F_b(e+1) = \begin{cases} A_b(e) + k_1 + k_2 & i < \frac{1}{2} \\ A_b(e) + j_1 + j_2 & i \geq \frac{1}{2} \end{cases} \tag{7}$$

In Eq. (7), the location of the b^{th} gannet is denoted as $A_b(e)$. The term j_1 indicates an arbitrary parameter within $[-f, f]$. The term k_1 indicates an arbitrary parameter within $[-h, h]$. The value of j_2 in Eq. (7) is computed and validated by Eq. (8).

$$j_2 = I * \left(A_b(e) - A_d(e)\right) \tag{8}$$

In Eq. (8), the location of the gannet, which is picked arbitrarily in the present iteration for the current population is denoted as $A_d(e)$. The value of I in Eq. (8) is computed using Eq. (9).

$$I = \left(2 * d_4 - 1\right) * f \tag{9}$$

In Eq. (9), d_4 indicates the arbitrary variable within [0, 1]. k_2 Value in Eq. (7) is computed by Eq. (10).

$$k_2 = J * \left(A_b(e) - A_l(e)\right) \tag{10}$$

In Eq. (10), the mean location of the gannet is denoted by $A_l(e)$. The value of J in Eq. (10) is considered by Eq. (11).

$$J = \left(2 * d_5 - 1\right) * h \tag{11}$$

In Eq. (11), d_5 indicates a random variable among the limits [0, 1]. The mean location of the gannet in the current population is calculated on the basis of Eq. (12).

$$A_l(e) = \frac{1}{B} \sum_{b=1}^{B} A_b(e) \tag{12}$$

Exploitation: Once the gannet dives into the water, it has to capture the fish spotted by it when it is in flight. However, the fish uses various turning movements to flee from the gannet. Then the gannet will chase the fish in the water at high speed. These processes are provided in the exploitation phase of the GOA. A huge amount of energy in the gannet is spent on the exploration phase in search of the fish. Thus, the gannet is left with less energy. The energy capacity of the gannet thus has to be determined. The energy capacity of the gannet, which helps to determine the ability of the gannet to capture a fish, is given by Eq. (13).

$$K = \frac{1}{L * e_2} \tag{13}$$

In Eq. (13), the value of L is computed by Eq. (14).

$$L = \frac{O * m^2}{N} \tag{14}$$

In Eq. (14), the term m denotes the speed at which the gannet enters the water while ignoring the water resistance. The value of m is taken as $1.5\,{}^{m}/_{s}$. The term O in Eq. (14) indicates the gannet's weight, which on average is considered to be as $2.5\,kg$. The term N in Eq. (14) is computed through Eq. (15).

 https://doi.org/10.1007/s44196-025-00940-2

$$N = \frac{2}{10} + \left(2 - \frac{2}{10}\right) * d_6 \tag{15}$$

In Eq. (15), the term d_6 denotes randomly generated parameters within [0, 1]. The value of e_2 in Eq. (13) is computed with the formula given in Eq. (16).

$$e_2 = 1 + \frac{g}{EG} \tag{16}$$

If the value of Eq. (16) is high, then the possibility of the gannet capturing the fish will also be high. But, as time increases, the energy in the gannet will gradually deplete. This leads to a reduction in energy capacity, thus, more chances are there for the fish to flee from the gannet. When the gannet is at a location where the prey is within the catchable proximity, then the gannet will utilize the immediate turn to obtain the fish. If not, then the Levy movement is utilized to find other fish. This process is mathematically formulated with the aid of Eq. (17).

$$F_b(e+1) = \begin{cases} A_M(e) + \left(A_b(e) - A_M(e)\right) * Q * e & K < n \\ e * \Delta + \left(A_b(e) - A_M(e)\right) + A_b(e) & K \geq n \end{cases} \tag{17}$$

The term n in Eq. (17) denotes a constant whose value is taken as $\frac{1}{2}$, and the ideal candidate (gannet) is represented by the term $A_M(e)$. The value of Δ in Eq. (17) is computed using Eq. (18).

$$\Delta = K * \left|A_b(e) - A_M(e)\right| \tag{18}$$

The value of the term Q in Eq. (18) is computed using Eq. (19).

$$Q = P(C) \tag{19}$$

In Eq. (19), the term P indicates the Levy flight function, which is calculated as given in Eq. (20).

$$P(C) = \frac{1}{100} \times \frac{\alpha \times \beta}{|\chi|^{\frac{1}{\delta}}} \tag{20}$$

The terms α and χ in Eq. (20) represent two arbitrarily generated variables within [0, 1], and δ indicate a constant on value 1.5. The value of β in Eq. (20) is computed using Eq. (21).

$$\beta = \left(\frac{\Gamma(1+\delta) \times \sin\left(\frac{\pi\delta}{2}\right)}{\Gamma\left(\frac{1+\delta}{2}\right) \times \delta \times 2^{\left(\frac{\delta-1}{2}\right)}} \right)^{\frac{1}{\delta}} \tag{21}$$

The GOA is an iteration-based procedure. At first, the solutions are selected randomly. Then, on every iterations the location of the gannet will be updated. Two methods of position updates are available in both the exploitation and exploration phases of the GOA. Equal weightage is provided to the position update in the exploration phase by utilizing any one of the diving modes. In the exploitation phase, the location update is dependent on the energy capacity of the gannet. However, equal weightage is given to the exploitation and exploration phases, and either of the two phases will be executed in all iterations. This process keeps on iteration till the end condition is met. The pseudocode of the GOA is provided in Algorithm 1.

Input the population B, problem dimension C, and maximum number of iterations G
The gannet's initial position is determined using Eq. (2)
The memory matrix F is computed
Evaluate the fitness of the population
While (is the stopping condition met?)
  For $(B = 1 to B_{\max})$
    For $(g = 1 to G)$
      If $\left(\frac{1}{2} < rand\right)$
        For F_b
          If $\left(\frac{1}{2} \le i\right)$
            The gannet's position is amended using the second part of Eq. (7)
          Else
            The gannet's position is amended using the first part of Eq. (7)
          End
        End
      Else
        For F_b
          If $\left(\frac{2}{10} \le n\right)$
            The gannet's position is amended using the first part of Eq. (17)
          Else
            The gannet's position is amended using the second part of Eq. (17)
          End
        End
      End
      For F_b
        The fitness of the memory matrix F_b is re-evaluated
        Replace A_b with F_b if the fitness of F_b is greater than A_b
      End
    End
  End
End

 https://doi.org/10.1007/s44196-025-00940-2

4.3 Proposed Optimization Algorithm: IGOA

The process of detecting anomaly detection is a tedious and time-consuming process as the quantity of data that needs to be handled by these detection models is high. But, for real-time implementation of these anomaly detection frameworks, it is essential to make the detections in a quick manner. The slower processing of these anomaly detection frameworks is due to the presence of various hyperparameters within these deep learning models. Therefore, there is a need to optimize these hyperparameters within the anomaly detection model. Thus, heuristic optimization schemes are introduced into the system. In this work, the GOA algorithm is selected and updated to optimize the hyperparameters such as hidden neuron count, activation function, and steps per epoch in the CapsNet model used for anomaly detection. The GOA is selected in this work because of its enhanced feasibility and enriched performance on several complicated optimization problems. However, the conventional GOA takes more time to converge due to the presence of more random variables. Thus, in the IGOA, these variables are amended. The IGOA algorithm is executed by amending the value of random parameters d_1 in the conventional GOA using an adaptive concept. The iteration count has a major impact on the process of upgrading this random value. The adaptive concept that is used for upgrading this random number d_1 is given by Eq. (22).

$$d_1 = -g * \left(\frac{(-1)}{G} \right) \tag{22}$$

In Eq. (22), the term g denotes the current iteration and G represents the highest iteration. The value of d_1 in the conventional GOA is in the range $[0,\ 1]$. The value of d_1 in Eq. (22) is used to amend the value of d_1 in Eq. (2).

4.3.1 Superiority of the IGOA with Other Optimization Algorithms

Generally, various optimization techniques are employed widely for tuning the parameters in the deep learning model and offer better outcomes in different classes. In the developed anomaly detection and intrusion prevention model, IGOA is employed to enhance the overall network efficiency by improving the security. When comparing with different optimization techniques such as Mine Blast Optimization (MBO) [31], Egret Swarm Optimization Algorithm (ESOA) [33], Opposition Slime Mould Algorithm (OSMA) [32], and GOA, the proposed IGOA has attained better performance that has been theoretically explained as follows. MBO [31] is efficient in offering better outcomes by analyzing the performance of the overall network. MBO is capable of offering quick convergence and also minimize the risks by improving the random search ability. Yet, the MBO requires tackling the multi-dimensionality issues that arise in the network. Moreover, its interaction process is complicated among the parameters. These kinds of issues that arise in the MBO affect the overall network efficiency in the complex classes. OSMA [32] is a nature-based technique efficient in tackling real-world complications, and also it includes simple implementation procedures. OSMA is efficient in tackling the vanishing gradient issues and also provides good advancements in the exploitation and exploration phase. However, it finds more issues in finding the global optimal outcomes in the search spaces due to local optimal trapping issues. Premature convergence issues generate more trouble while finding the optimal solutions in the search space. Moreover, they face more complications while performing the higher-dimensional optimization tasks. In addition, the developed framework used the ESOA [33], which is efficient in providing better balance in the search spaces and also offers good adaptability. Accomplishing higher adaptability helps to suitable in wide range of applications. Furthermore, its adaptability is higher in the complex search spaces, which helps to eliminate the local optimal issues. But, diversity issues arise

in the ESOA minimized the exploration and also it can easily get trapped while finding the local optima solutions. Classical optimization techniques are prone to slow convergence and also take more time to identify the global optimal solutions. GOA [26] has a simple structure to implement, and it is also efficient to use with a wide range of applications. Its global search efficiency is good, along with higher adaptability in various domains. Yet, GOA is required to eliminate the convergence issues that arise while handling the binary optimization issues. To rectify several issues that take place in the network, random numbers in the network are improved by a novel concept. The newly designed concept is named as IGOA, which is capable of tackling the premature convergence issues that support exploring a huge solution space under multiple populations. Moreover, it improves the robustness as well as adaptability in the modifying environments. In addition, the problem-solving tendency of the developed IGOA is improved for handling a wide range of constraints. The developed IGOA provides reliable outcomes by eliminating the premature convergence under multiple runs with various population counts. Attaining faster convergence supports eliminating the validation overhead issues that arise while finding the optimal solutions by considering the complex issues. IGOA is capable of resolving the risks and also supports for better decision making by adapting the dynamic environmental conditions. Thus, employing IGOA in the developed anomaly detection and intrusion prevention scheme helps accomplish better outcomes in the changing environment and also protects sensitive information from attackers.

 https://doi.org/10.1007/s44196-025-00940-2

The pseudocode of the IGOA is provided in Algorithm 2.

Input the population B, problem dimension C, and maximum number of iterations G
The gannet's initial position is determined using Eq. (2)
The memory matrix F is computed
Evaluate the fitness of the population
While (is the stopping condition met?)
Amend the value of d_1 using the adaptive concept as provided in Eq. (22)
For $(B = 1 to B_{\max})$
For $(g = 1 to G)$
If $\left(\frac{1}{2} < rand\right)$
For F_b
If $\left(\frac{1}{2} \le i\right)$
The gannet's position is amended using the second part of Eq. (7)
Else
The gannet's position is amended using the first part of Eq. (7)
End
End
Else
For F_b
If $\left(\frac{2}{10} \le n\right)$
The gannet's position is amended using the first part of Eq. (17)
Else
The gannet's position is amended using the second part of Eq. (17)
End
End
End
For F_b
The fitness of the memory matrix F_b is re-evaluated
Replace A_b with F_b if the fitness of F_b is greater than A_b
End
End
End
End

In the initial step iterations, populations and problem dimensions of IGOA are allocated. Next, the initial positions of the gannets are provided; along with their memory matrixes are validated. Later, the fitness for the population and stopping criteria are assigned. In IGOA, the random numbers d_1 presented in the bound [0, 1] are improvised by the novel concept offered in Eq. (22). Here, the random numbers are updated by considering the current iteration and the highest iteration, which helps to enhance the random numbers d_1. Next, the condition $\frac{1}{2} < rand$ is used to verify the performance. In case the condition $\frac{1}{2} < rand$ is not fulfilled then the condition $\left(\frac{1}{2} \leq i\right)$ is used to fulfill. In case, the condition $\left(\frac{1}{2} \leq i\right)$ fulfilled then the positions are improved through V diving modes else U diving mode is takes place according to Eq. (7). In case, the above mentioned condition is fulfilled then it moves towards the condition $\frac{2}{10} \leq n$, which utilize the immediate turning process for the updating of gannet position through Eq. (17) higher order else lower order is used to carry out the position updating. Later, the fitness functions of the memory matrixes are revaluated and also the overall fitness of the network is validated. Finally, optimal solutions are taken as the outcome. Here, the developed IGOA helps to obtain the optimal outcomes such as optimized weights, optimized hidden neurons, activation function and steps per epoch. Here, the validations are carried out for multiple run times to accomplish the optimal outcomes. Once the optimal outcomes are obtained from IGOA, then, the termination process is carried out, which stops the entire process.

The flowchart of the IGOA is provided in Fig. 3.

4.4 Weighted Feature Fusion with Proposed IGOA

The collected raw data CD_{cd}^{col} are considered for the feature extraction process. The extraction of the most viable features assists in the minimization of the overall detection process. Thus, it is essential to obtain the crucial features from the collected data. Feature extraction procedures are performed to analyze the information presented in each pixel for identifying the color distribution, corners, and edges. In the feature extraction phase, redundant information, incomplete data, and noise presented in the input samples are eliminated successfully. In this phase, input samples are converted into a minimal dimensional representation, which provides a high focus on the essential information and then filters out the noisy components by preserving the significant features presented in the input samples. Here, the random noises are filtered out by preserving significant information like texture and edges. Eliminating the noises from the input samples helps to minimize data complexity by preserving the edges. Reducing the noise from the input samples helps to enhance the system's robustness and also tackles the dimensionality issues that aid in avoiding the overfitting issues as well as reducing the validation cost. Once the required features are extracted then they are presented as EF_{ef}^{fea}. The weights from the extracted features are optimally selected by IGOA. The optimally selected weights are indicated as OW_{ef}^{GOA}. Then, these weights are multiplied by the extracted features to obtain the weighted features. The process of obtaining the weighted features is given by Eq. (23).

$$WF_{ef}^{Fuse} = EF_{ef}^{fea} * OW_{ef}^{GOA} \tag{23}$$

In Eq. (23), WF_{ef}^{Fuse} the obtained weighted features. The main goal behind the optimization of the weights is given by Eq. (24).

$$oe1 = \arg\min_{\left\{OW_{ef}^{GOA}\right\}} \left(\frac{1}{RF}\right) \tag{24}$$

The term $oe1$ in Eq. (24) indicates the objective function of the optimization problem. As provided in Eq. (24), the main goal behind the optimization of the weights by the IGOA is the enhancement of the relief score. In Eq. (24), the term RF indicates the maximized relief score. The weights for weighted feature formation are tuned using the IGOA within the limit [0.01, 0.99]. The relief score RF is given by the process of ranking features in

https://doi.org/10.1007/s44196-025-00940-2

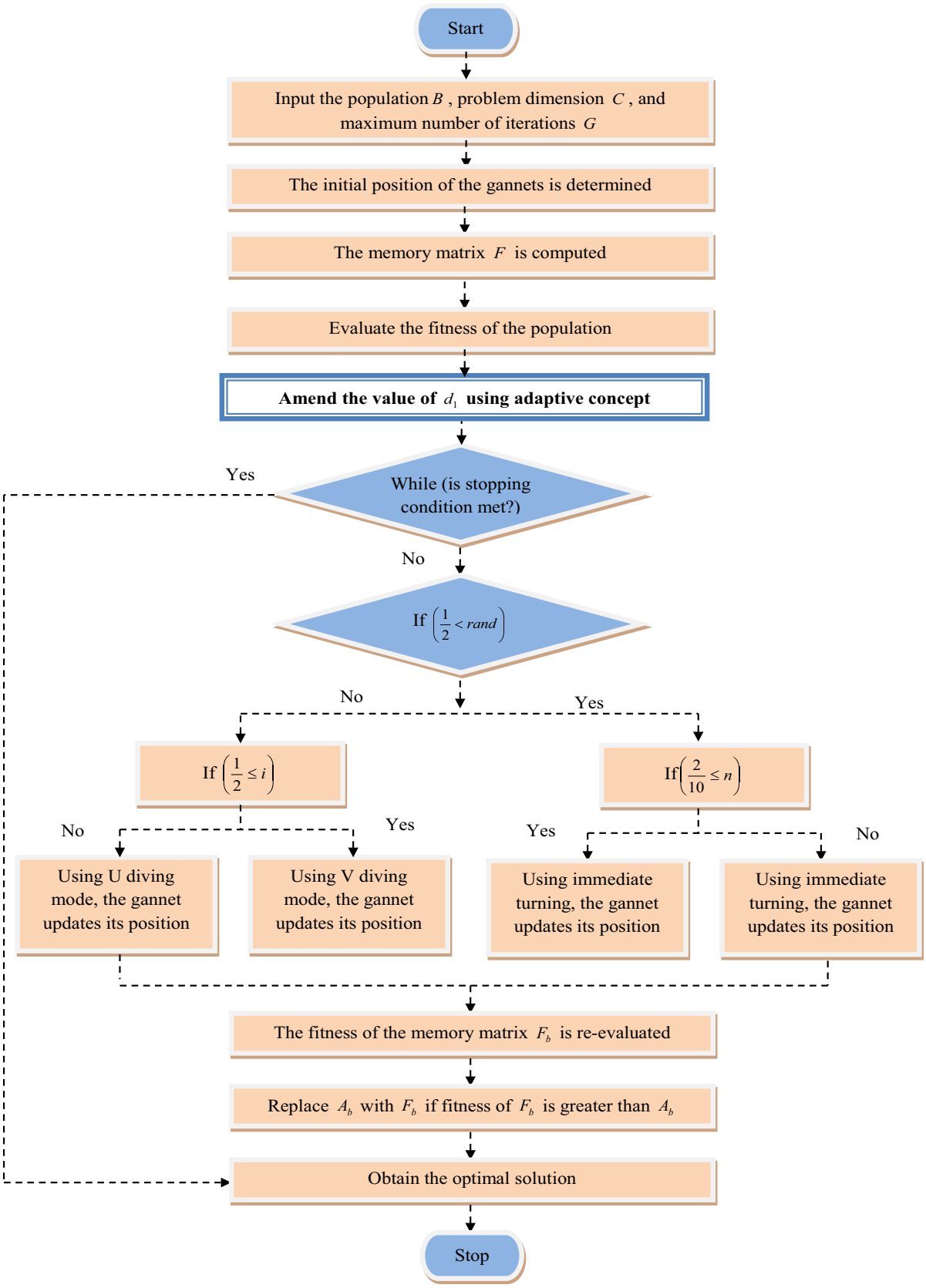

Fig. 3 Flowchart of the implemented IGOA

a given set of features based on the scores provided to them and selecting the top most scored features for the feature selection process. The process of weighted feature fusing using the proposed IGOA is depicted in Fig. 4.

5 ADCAPSNET: Adaptive Deep Capsule Network-Based Anomaly Detection and Prevention for Security Enhancement in IoT Network

5.1 Deep Capsule Network

A set of organised neurons that are arranged as capsules form the building block of the CapsNet [27]. The invariance obtaining characteristics irrespective of the orientation and transformation of the object is provided by the capsule's length. Whereas, the data is reconstructed by making use of the features in the capsules that are capable of capturing the equi-variance by considering distinct variations. The vectors generated by the capsules of the CapsNet are of distinct angles but have the same magnitude. The features that are extracted from these inputs are represented by the vector's alignment in the CapsNet. Thus, the spatial hierarchical relationships are effectively extracted by means of the CapsNet's vector representations. The main advantage of utilizing the CapsNet for the detection task is its minimized requirement of the number

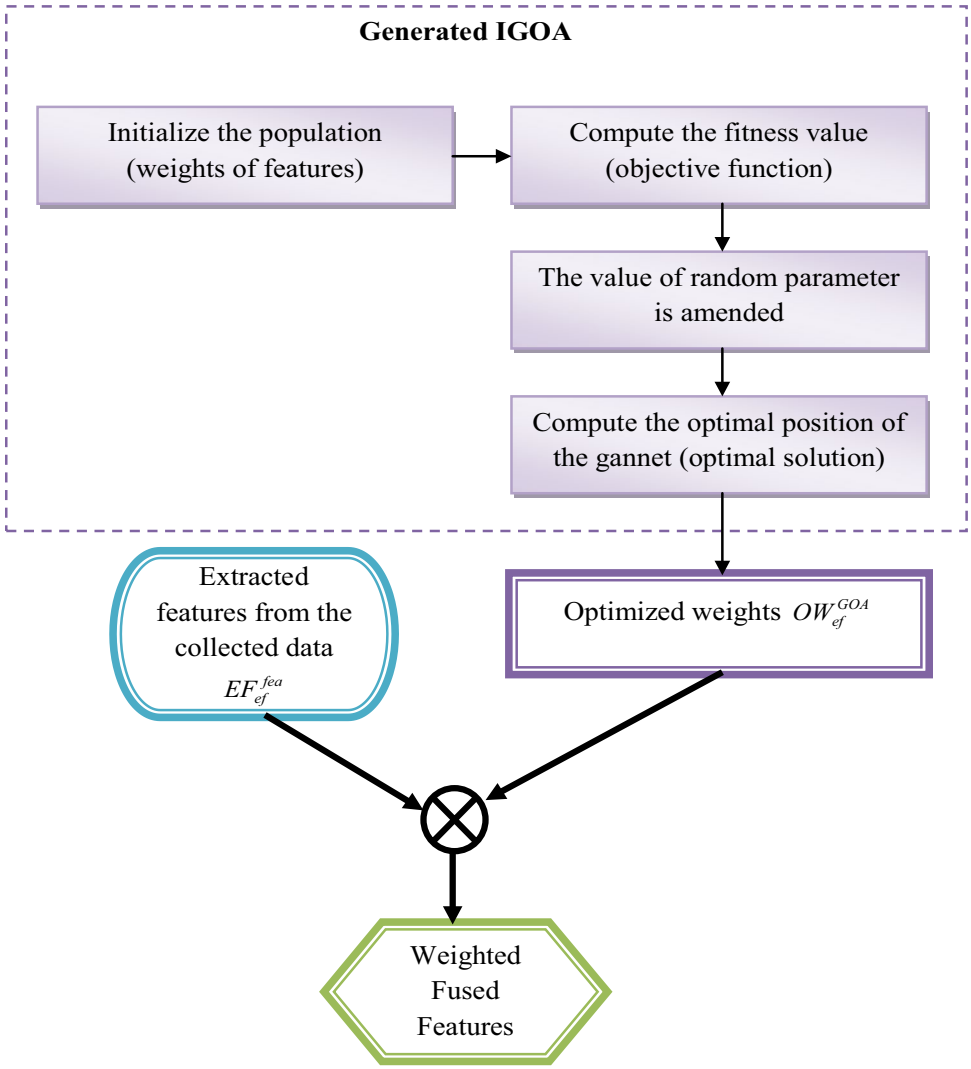

Fig. 4 Graphical representation of the weighted feature fusion by the implemented IGOA

of layers for performing the allocated task. The conventional neural network utilizes additional layers when enhanced accuracy is required. However, in CapsNet, the accuracy is attained by nesting the layers inside individual layers. The capsule in the CapsNet is the representation of the input information. The resultant vector from a capsule of the CapsNet is provided to the next layer to make the connection with a suitable parent capsule. Let us consider that the o^{th} capsule will generate an output as represented by the term q_o. The term q_o indicates the features that are extracted by the o^{th} capsule in the CapsNet. A transformation matrix S_{op} is applied to the input q_o to transform it to determine the parent capsule p 's prediction vector $\hat{T}p|o$. The transformation process is given by Eq. (25).

$$\hat{T}p|o = S_{op}q_o \tag{25}$$

In Eq. (25), the term o indicates the lowest layer in the CapsNet, and the forecast vector of the output of the capsule at this level p is represented by the term $\hat{T}p|o$. The transformation matrix (weight matrix) S_{op} is obtained in the network's back-propagation process. The weighted total of the entire prediction vector $q_{p|o}$ is represented by the term R_p. With the aid of the dynamic routing approach, the coupling coefficient r_{op} is determined. The level up to which the o^{th} capsules agree with the neighbouring capsule p is provided by this coupling coefficient term. The Softmax function, when performed on the comparison score, u_{op} is used to compute the value of the coupling coefficient r_{op}. The comparison score u_{op} is obtained by combining the characteristics of the features with the probability rather than combining the likelihood from every neuron in the CapsNet. The process of obtaining the coupling coefficient r_{op} is given by Eq. (26).

$$r_{op} = \frac{\exp^{(u_{op})}}{\sum_t v(u_{ot})} \tag{26}$$

In Eq. (26), the term u_{op} indicates the comparison score. This comparison score u_{op} is obtained by combining the likelihood and the feature properties of the capsule o^{th} with its parent capsule p. The value of the comparison score is amended by means of a repetition process on the execution of the dynamic routing approach. The comparison score u_{op} is estimated by utilizing Eq. (27).

$$u_{op} = u_{op} + \hat{T}_{p|o}\, s_p \tag{27}$$

The comparison score is determined by considering both the activity vector s_p and the prediction vector $\hat{T}_{p|o}$. The squashing activation function is used by CapsNet. The activation function will be the maximum value when the resultant vector is a single value. When the resultant vector has a minimum value, then the activation function becomes equal to zero. A non-linear squashing is utilized to determine the value of the activity vector s_p as given in Eq. (28).

$$s_p = \frac{\left\|R_p\right\|^2}{\left|1 + R_p\right|^2} \cdot \frac{R_p}{\left\|R_p\right\|} \tag{28}$$

In Eq. (28), the term $\frac{R_p}{\|R_p\|}$ indicates the prediction vector's overall normalized weighted sum value and the vector lengths are normalized by the factor given by $\frac{\|R_p\|^2}{|1+R_p|^2}$. The hierarchical as well as spatial relationships in the provided input are obtained by providing the information in one capsule to another by iteratively updating the values of the prediction vector and the coupling coefficient. Thus, efficient outputs are generated by the CapsNet with higher accuracy. The structural representation of Deep CapsNet is shown in Fig. 5.

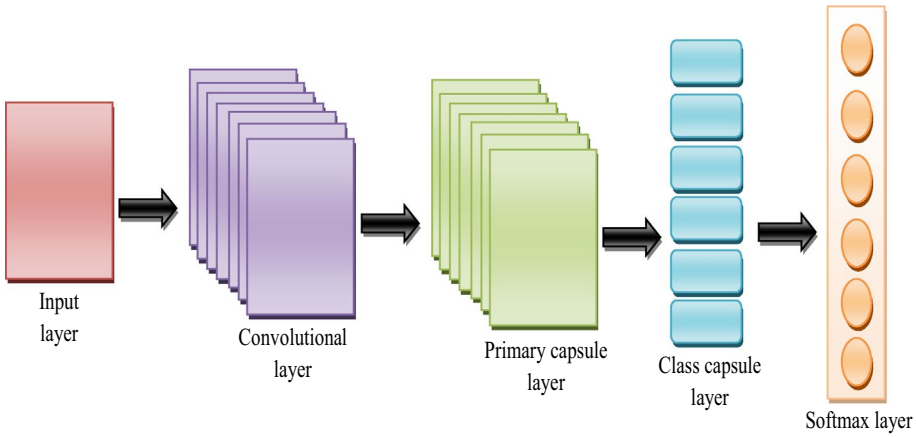

Fig. 5 Structural representation of deep capsnet model

5.2 Developed ADCapsNet-Aided Anomaly Detection and Intrusion Prevention

5.2.1 Rationale on Choosing ADCapsNet

In recent years, various mechanisms have been employed to execute anomaly detection and intrusion prevention. Among all the techniques, federated learning models are efficient in detecting anomalies and preventing intrusions that arise in the network. Yet, these techniques are prone to multiple issues that badly affect the overall efficiency of the network. Federated learning models need frequent communication over the participation devices and central server, which generates more complications in the resource-constrained scenario, such as IoT, edge computing models and so on. Moreover, managing the heterogeneity among the distributed network is challenging, and they are also prone to inaccurate outcomes and bias issues. In addition, data corruption issues arise in the network, which affects the global efficiency. Maintaining the security of sensitive information from attackers is complicated, and it leads to data breach issues. Moreover, it uses the limited validation power and reduced memory in the implementation phase, which leads to overfitting in the training phase. Henceforth, using the transformer-based techniques in anomaly detection as well as intrusion prevention schemes leads to an increase in false positive results, and they also require enormous data to execute the training process. In some cases, handling the attack patterns is complicated, and it also affects the decision-making efficiency of the network. Generating false positive outcomes leads to more wastage of resources and also security-related issues. Furthermore, the training process uses enormous time, which makes the implementation cost high. Misclassification issues lead to more losses in the network, and also understanding the complicated patterns takes more time and also affects decision making. To tackle all these issues that arise in the prior framework, the developed framework aims to design a novel anomaly detection and intrusion prevention model using deep learning schemes. So, a novel framework, ADCapsNet is suggested in this research work, which is designed by tuning the parameters in CapsNet to accomplish more precise anomaly detection and intrusion prevention results without any biases or misclassifications.

The obtained weighted features WF_{ef}^{Fuse} are given as input to the ADCapsNet model. As the CapsNet provides a detailed representation of the vector, it is much more suitable for anomaly detection tasks. In the realm of anomaly detection, ADCapsNet emerges as a transformative solution, addressing and surpassing the limitations inherent in existing models. Traditional approaches to anomaly detection often struggle with issues such as feature extraction inefficiencies, sensitivity to data variations, and limited generalization capabilities [34] [35]. To mitigate these limitations, DCapsNet offers a paradigm shift. This harnessing dynamic routing mechanisms and advanced feature representation techniques, DCapsNet not only overcomes these shortcomings but also achieves unprecedented levels of accuracy and robustness. This paradigm shift in anomaly detection is driven by the innovative capabilities of CapsNet, which enable more effective modelling of

complex data patterns and anomalies. As a result, ADCapsNet emerges as a powerful tool for anomaly detection tasks, capable of delivering superior performance and reliability compared to traditional methods.

However, the number of routing that takes place within the CapsNet due to the presence large number of inner loops makes the overall computation process a bit complicated. Also, the involvement of the hyper-parameters within the CapsNet makes the computation process much slower. Thus, the parameters in the Deep CapsNet are tuned using the implemented IGOA. This optimized IGOA-ADCapsNet model will then generate the anomaly-detected outcome. The detected anomaly from the ADCapsNet-based anomaly detection model is considered for taking measures to prevent intrusions in the system. The practice of safeguarding the software and the hardware present within a network is known as intrusion prevention. It involves the process of continuously tracking the malicious activity on the network and taking appropriate action, such as notifying, blocking, or removing the offending equipment, when any intrusion is detected. Unlike conventional IDS, the Intrusion Prevention System (IPS) is more sophisticated, with the ability to inform administrators about the existence of harmful activity in the network. The IPS needs to be strong enough to scan the traffic without causing any interference to the normal working of the network. This IPS has to run quickly, just like many other network security solutions. By removing harmful traffic from the path before it reaches other security controls, an IPS not only enhances the efficiency of the network but also helps defend the network against various attacks. An anomaly-based IPS is used in this work. Here, the anomaly detected from the ADCapsNet-based anomaly detection model is utilized to determine the occurrence of intrusions on the system and take necessary actions to prevent their occurrence in the future.

5.2.2 Discussion on Developed ADCapsNet Over Other Standard Models

This research work employs ADCapsNet for anomaly detection and intrusion prevention in the IoT network for the advancement of overall network security. Here, the developed ADCapsNet includes the potential of CapsNet as well as CNN, which helps to enhance the anomaly detection and intrusion prevention efficiency. In ADCapsNet, the capsule layer helps to collect the hierarchical relationship to enhance the robustness over a changing environment. CapsNet [27] is better at handling the hierarchical relation among the samples for providing a better understanding of anomaly detection. Its generalization is higher, and it also quickly encodes the required information by maintaining robustness. Yet, these techniques lead to computational complications, which slow down the training phase and also demand more parameters to carry out the executions. Initial settings of the CapsNet lead to sensitive issues in the hyperparameter settings and also call for precise tuning in the training phase. In some cases, their scalability is Limited, and also decision-making is complicated. The CNN-based technique, such as 1-Dimensional Convolutional Neural Network-Knowledge Distillation (1DCNN-KD) [44], is good in collecting the required features without any human supervision and also their implementation procedures are simple. It eliminates the memory overhead issues and also has a quick access time. It faces more issues while handling the hierarchical structures as well as local patterns. Its implementation is expensive and also leads to overfitting, which leads to poor generalization. Thus, maintaining the explicitness in ADCapsNet is crucial for maintaining the multi-level relationship among the nodes, and also it identifies the suitable deviations. The ADCapsNet helps to tackle the invariance issues while predicting the anomalies. The ADCapsNet identifies the anomalies by considering the reconstruction errors and also easily handles the class imbalance issues. ADCapsNet offers higher-dimensional outcomes in the complex regions. ADCapsNet is capable of providing superior robustness and also maintains robustness in the complex classes. ADCapsNet is efficient in handling the variations and offering better outcomes. Here, the adaptive concept is used in the ADCapsNet, which supports to rectify the gradient vanishing issues and also minimize the validation expense. Robustness of ADCapsNet is higher over various attacks by offering higher reliability. In addition, the false positive rates in the network are reduced in ADCapsNet to maintain a better relationship among the spatial information. Reducing the false positive rates helps to minimize the fluctuations over different attacks.

The illustration of the ADCapsNet-based anomaly detection and intrusion prevention approach is given in Fig. 6.

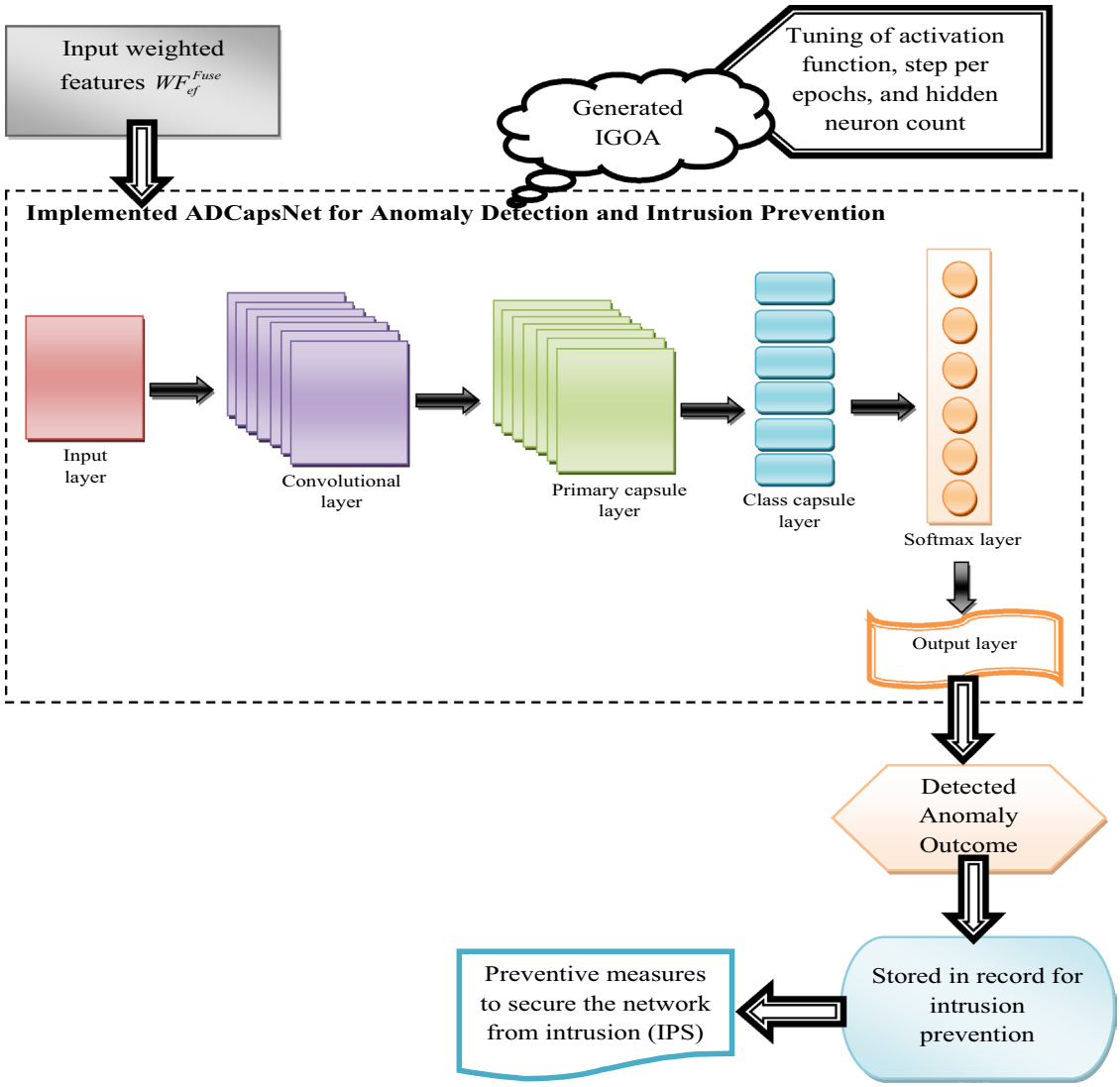

Fig. 6 Architecture of the implemented IGOA-ADCapsNet-based anomaly detection and intrusion prevention model

In the developed ADCapsNet-based anomaly detection and intrusion prevention model, optimally weighted features WF_{ef}^{Fuse} are offered as the input. Once the inputs are given to the input layer of CapsNet, they are passed to the convolution layer, which collects the spatial and temporal features from the input samples. Moreover, the hierarchical features are learned to reduce the complication. Next, the samples are forwarded to the primary capsule layer, which is efficient in handling the spatial hierarchies and also enhances the robustness over attacks. Then, the features are subjected to a class capsule layer, which is efficient in handling the hierarchical relationship and also collects the spatial relationship among the objects as well as predicted classes. At last, these features are given to the softmax layer, which helps maintain the interpretability along with confidence for enhancing the prediction. This layer offers numerical stability during the training phase and ensures the dynamic routing. Further, at the outcome layer collected hierarchical relationship and spatial information are used to provide the final anomaly detection outcomes. From the outcome layer, anomaly-detected outcomes are attained and then by considering these data, the intrusion prevention process is carried out to obtain better results. Here, the developed ADCapsNet helps to accomplish better outcomes by tuning the parameters of ADCapsNet such as steps per epoch, activation function, and hidden neuron counts through IGOA, which

supports to maximzie the MCC and accuracy and also reducing the FNR. Finally, more precise anomaly detection and intrusion prevention outcomes are attained from ADCapsNet.

5.3 Objective of Developed ADCapsNet Model

There is a need to optimize the hyper-parameters in the CapsNet model to make the overall process much more efficient and accurate. Here, the activation function, steps per epoch as well and hidden neurons in CapsNet are tuned by IGOA. The process of tuning these hyper-parameters helps to boost the overall accuracy and Matthews Correlation Coefficient (MCC) in anomaly detection tasks and decreases the False Negative Rate (FNR) that arises in this anomaly detection process. This objective function is given by Eq. (29).

$$oe2 = \underset{\left\{Af_{af}^{GOA}, Hr_{hr}^{GOA}, Sp_{sp}^{GOA}\right\}}{\arg\min} \left(\frac{1}{rf} + mv + pl\right) \tag{29}$$

In Eq. (29),Af_{af}^{GOA} denotes the tuned activation function, which is within the limit $[1, 5]$, $oe2$ specifies the objective function of the optimization problem, Sp_{sp}^{GOA} denotes the tuned steps per epoch, which is within the limit $[10, 50]$, mv denotes the maximized MCC value, rf represents the maximized accuracy rate, and pl denotes the minimized FNR, and Hr_{hr}^{GOA} indicates the optimized hidden neuron count, which is within limit $[5, 255]$. The enhanced accuracy rate rf is determined using Eq. (30).

$$\mathrm{rf} = \frac{gh + ih}{gh + ib + ih + gb} \tag{30}$$

The term ih denotes the true positives that are evaluated between the target and the detected outcome from the ADCapsNet, ib indicates the false positives, gb denotes the false negatives, and gh indicates the true negatives, respectively. The FNR pl is estimated with the support of Eq. (31).

$$pl = \frac{\mathrm{gb}}{ih + gb} \tag{31}$$

The computation of MCC mv is determined using Eq. (32).

$$mv = \frac{gh * ih - ib * gb}{\sqrt{(ih + ib)(ih + gb)(gb + ib)(gb + gb)}} \tag{32}$$

The term ih denotes the true positives that are evaluated between the target and the detected outcome from the ADCapsNet, ib indicates the false positives, gb denotes the false negatives, and gh indicates the true negatives, respectively.

5.4 Detailed Description of Anomaly Detection and Intrusion Prevention

Deep learning-based anomaly detection and intrusion prevention techniques help to identify threats in the initial stages and also resolve data breaching issues in the network. Moreover, this section addresses the transparency of executing anomaly detection and intrusion prevention in the early stages with deep learning techniques. These kinds of techniques easily detect unknown threads and identify various attacks, and also their patterns in the initial stage help the entire system from data breaching problems.

5.4.1 Step 1: Data Collection

In the initial phase, various data used for the validation are obtained from benchmark online resources.

5.4.2 Step 2: Optimal and Weighted Features Extraction

From the collected data, essential features are extracted, and also their weights are tuned optimally. Here, the weights are tuned within the bounds [0.01, 0.99] by the developed IGOA to obtain the optimal outcomes that help to reduce the errors by improving the relief score. In the developed framework, enhancing the relief score helps to allocate higher weights to the respective features, and then it is differentiated among various classes. Using optimal features for the validation helps reduce overfitting and also improves the training by selecting the most significant features.

5.4.3 Step 3: Parameter Tuning in Anomaly Detection and Intrusion Prevention Model

The main objective of the developed ADCapsNet-based anomaly detection and intrusion detection model is to enhance MCC and accuracy by reducing FPR. In the developed ADCapsNet model, reducing FPR helps to reduce the error by enhancing decision-making about the anomalies. Several objectives of the developed technique are fulfilled by tuning the parameters of ADCapsNet, like hidden neuron count in the range [5, 255], activation function in the bound [1, 5], and steps per epoch in limit [10, 50] by IGOA.

5.4.4 Step 4: Anomaly Detection and Intrusion Prevention Phase

Here, the ADCapsNet technique is used for anomaly detection and intrusion prevention, which is the enhanced version of the capsule network with parameter tuning that helps to enhance security by identifying the vulnerabilities and threats in the network. Moreover, these techniques identify the suspicious activities that take place in the network and warn the user through notifications. Here, the developed ADCapsNet detects the threads and malicious activities in the initial stages and also helps the user to protect sensitive information from attackers.

5.4.5 Step 5: Performance Analysis on Developed Model

At last, various experimental computations are executed in the developed ADCapsNet-based anomaly detection and intrusion prevention models over multiple performance measures.

6 Results and Discussion

6.1 Experimental Setup

The implemented anomaly detection and intrusion prevention framework was designed using a heuristic-aided deep learning approach and was implemented with the help of the Python platform. The model has utilized a total population count of 10, a total iteration count of 50, and a chromosome length equivalent to the number of features. The deployed anomaly detection and intrusion prevention framework was compared with other detection models such as Deep Temporal Convolutional Network (DTCN) [28], One Dimensional Convolutional Neural Network (1DCNN) [29], MobileNet [30], CapsNet [27], and conventional optimization techniques like MBO [31], OSMA [32], ESOA [33], and GOA [26], respectively, to prove its enhanced efficacy than other existing methods. Various lightweight IoT-based compression techniques utilized for the validation were Sequence Lossy Compression Algorithm for IoT (SZ4IoT) [41], Optimized Common Features Selection And Deep-Autoencoder (OCFSDA) [42], Granger Causality Inspired Graph Neural Network integrated with Efficient coviNet (GCIGNN-ENet) [43] and 1-Dimensional Convolutional Neural Network-Knowledge Distillation (1DCNN-KD) [44]. Different intrusion prevention mechanisms employed for the

observation were Binarized Spiking Neural Network with Blockchain (BSNN-BC) [45], hybrid Honeynet deployed in Docker with Tuning Of fiRewall (H-DOCTOR) [46], Honeypot and Blockchain-based Intrusion Detection and Prevention (HB-IDP) [47] and Combine Counter Mode Algorithm on Blockchain (CCMA-BC) [48]. Different resource requirements for the developed anomaly detection and intrusion prevention model are given as follows in Table 3.

6.2 Validation Indices

Various validation metrics used to examine the generated anomaly detection and intrusion prevention framework are listed in the following section.

a) The computation of False Positive Rate (FPR)*ek* follows Eq. (33).

$$ek = \frac{\text{ib}}{gh + ib} \tag{33}$$

b) Negative Predictive Value (NPV)*wx* is validated by Eq. (34).

$$wx = \frac{\text{gh}}{gh + gb} \tag{34}$$

c) The F1-score yc is determined with the help of Eq. (35).

$$\text{yc} = \frac{2*\text{ih}}{2 * (ih + ib + gb)} \tag{35}$$

d) The precision *on* is calculated using Eq. (36).

$$on = \frac{ih}{ih + ib} \tag{36}$$

e) The specificity *xu* is validated by Eq. (37).

$$xu = \frac{gb}{gb + ib} \tag{37}$$

Table 3 Resource requirement on developed framework

Software requirements	anaconda	Version 3
	pycharam	Version 3.11
Hardware Requirements	Random Access Memory (RAM)	8 GB
	Version	11
	Machine	Windows
	Read Only Memory (ROM)	500 GB
	Processor	i3
Libraries	Operncv-Python	
	prettytable	
	matplotlib	
	tensorflow	
	keras	
	tflearn	
	numpy	

f) The sensitivity *wk* is evaluated by utilizing the formula provided in Eq. (38).

$$wk = \frac{\text{ih}}{ih + gb} \tag{38}$$

g) The False Discovery Rate (FDR)*sy* is calculated with the aid of Eq. (39).

$$sy = \frac{\text{ib}}{ih + ib} \tag{39}$$

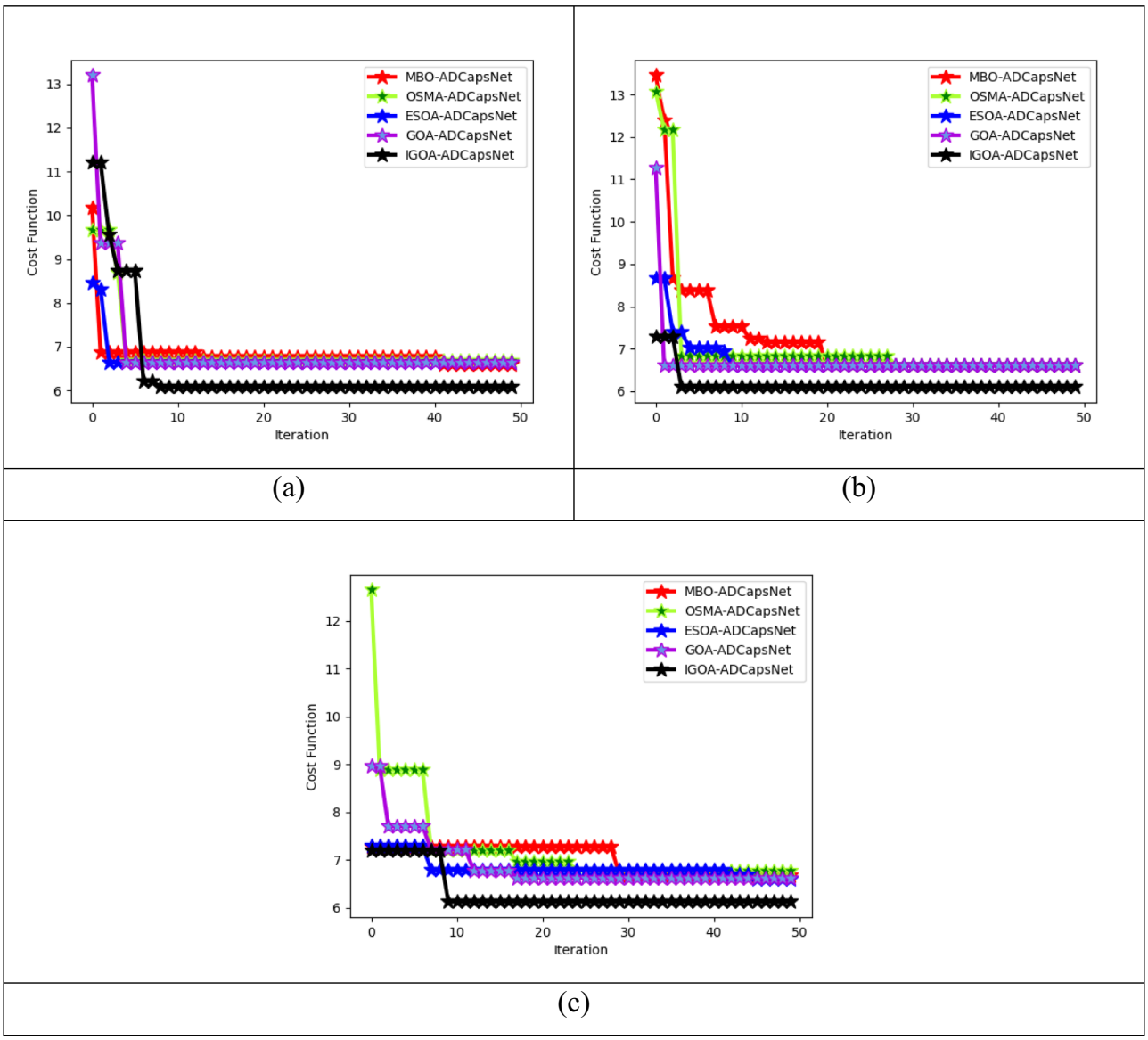

Fig. 7 Validation of convergence in designed anomaly detection and privacy prevention scheme regarding **a** Dataset 1, **b** Dataset 2, and **c** Dataset 3

6.3 Convergence Validation of the Executed Anomaly Detection and Intrusion Prevention Scheme

The convergence validation is conducted on the executed IGOA-ADCapsNet-based anomaly detection and privacy preservation scheme for validating the performance of the executed model while utilizing distinct existing optimization algorithms. The results regarding the convergence validation are illustrated in Fig. 7, corresponding to all three datasets that have been utilized for testing and training the model. On analyzing Fig. 7a, it is seen that the cost function of the executed IGOA-ADCapsNet-based anomaly detection and privacy preservation scheme is 8.96%, 8.96%, 10.29%, and 11.59% higher than the GOA-ADCapsNet, ESOA-ADCapsNet, OSMA-ADCapsNet, and MBO-ADCapsNet algorithms, respectively, while considering the 30th iteration. When Fig. 7b is evaluated, it is observed that the cost function of the suggested IGOA-ADCapsNet-based anomaly detection and privacy preservation scheme is 4.62%, 4.62%, 8.82%, and 15.07% enhanced than the GOA-ADCapsNet, ESOA-ADCapsNet, OSMA-ADCapsNet, and MBO-ADCapsNet algorithms, respectively, while considering the 15th iteration. While considering Fig. 7c, the executed IGOA-ADCapsNet-based anomaly detection and privacy preservation scheme shows 6.06%, 8.82%, 12.68%, and 16.23% improved cost function than the GOA-ADCapsNet, ESOA-ADCapsNet, OSMA-ADCapsNet, and MBO-ADCapsNet algorithms, respectively, at the 20th iteration. This analysis proves the enhanced convergence offered by the executed IGOA-ADCapsNet-based anomaly detection and privacy preservation scheme for all the datasets when compared with the other existing algorithms, making it an efficient model for detecting anomalies in the IoT network in a much faster manner.

6.4 Heuristic-Based Performance Analysis of the Generated Anomaly Detection and Intrusion Prevention Framework

The performance of the generated IGOA-ADCapsNet-based anomaly detection and privacy preservation framework is analyzed by comparing it with certain traditional algorithms, and the resultant outcomes are depicted in Fig. 8. The performance of the generated IGOA-ADCapsNet-based anomaly detection and privacy preservation framework is evaluated for all three datasets. Performance measures, including the Type 1 and Type 2 measures, are considered for validation purposes. In this section, we are considering the MCC analysis results. For Dataset 1, the MCC value of the generated IGOA-ADCapsNet-based anomaly detection and privacy preservation framework is 4.01%, 13.6%, 14.47%, and 17.48% more than the GOA-ADCapsNet, ESOA-ADCapsNet, OMSA-ADCapsNet, and MBO-ADCapsNet algorithms, respectively, for a k-fold value of 300. Similarly, the MCC of the generated IGOA-ADCapsNet-based anomaly detection and privacy preservation framework is 4.09%, 9.42%, 14.48%, and 20.03% better than the GOA-ADCapsNet, ESOA-ADCapsNet, OMSA-ADCapsNet, and MBO-ADCapsNet algorithms, respectively, for a k-fold value of 100 for Dataset 2. The generated IGOA-ADCapsNet-based anomaly detection and privacy preservation framework has an enriched MCC value of 3.125%, 7.84%, and 12%, than the ESOA-ADCapsNet, OMSA-ADCapsNet, and MBO-ADCapsNet algorithms, for a k-fold value of 500 for Dataset 3. However, a slight improvement in the GOA-ADCapsNet algorithm in the 500-fold is seen on dataset 3 validation than the suggested IGOA-ADCapsNet. This proves the second-best results produced by the GOA-ADCapsNet algorithm. However, while taking the validation on the whole, it is observed that the MCC provided by means of the generated IGOA-ADCapsNet-based anomaly detection and privacy preservation framework is higher, thus proving the ability to correctly predict both the classes (anomalous and non-anomalous) by the generated model than existing algorithms.

6.5 Techniques-Based Performance Examination of the Deployed Anomaly Detection and Intrusion Prevention Framework

The examination of the deployed IGOA-ADCapsNet-based anomaly detection and privacy preservation model when contrasted against conventional techniques generates results that are illustrated in Fig. 9. For this section,

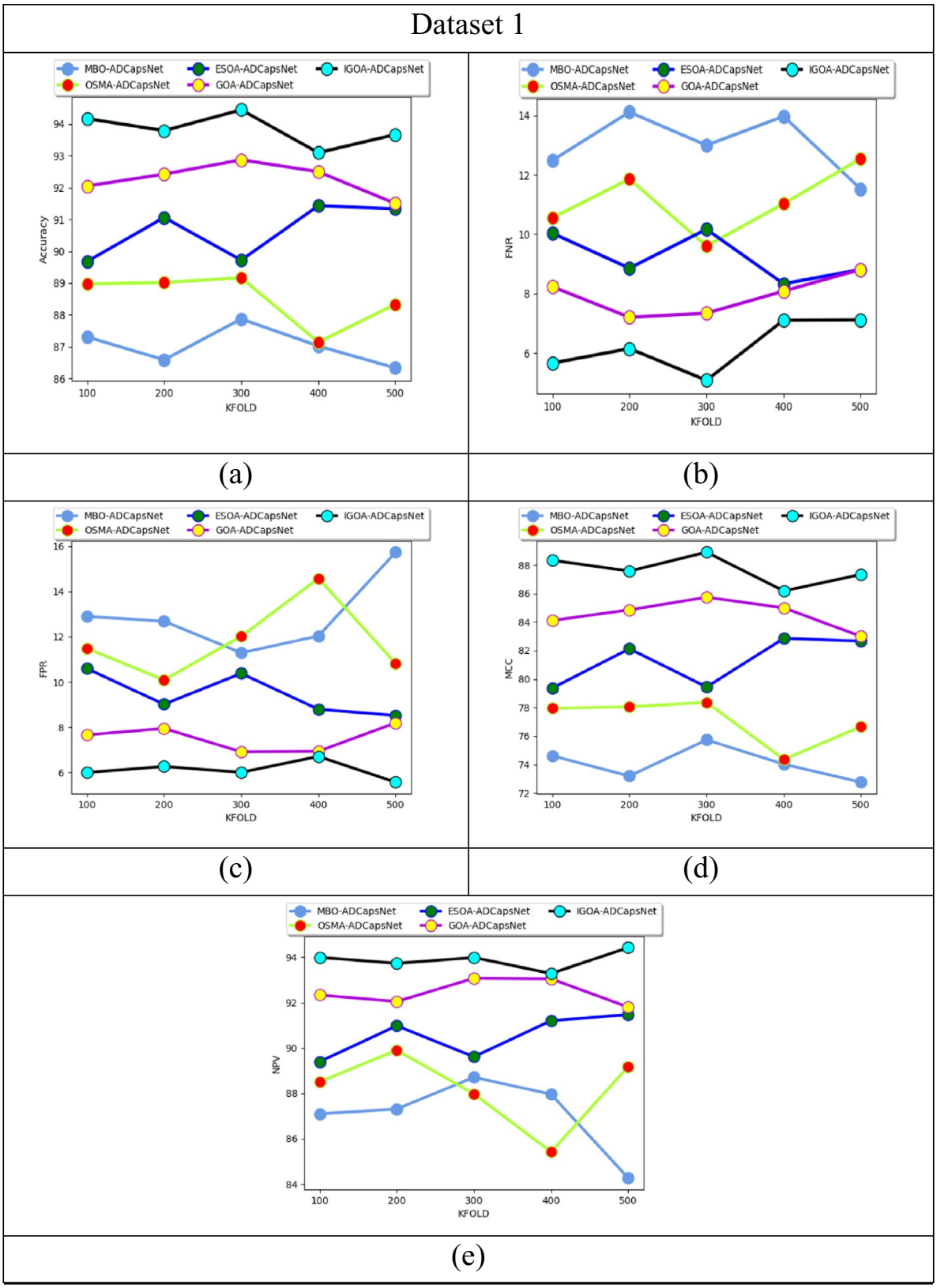

Fig. 8 K-fold-based performance analysis of the generated anomaly detection and intrusion prevention framework when contrasted against existing algorithms in terms of **a** Accuracy, **b** FNR, **c** FPR, **d** MCC, and **e** NPV

https://doi.org/10.1007/s44196-025-00940-2

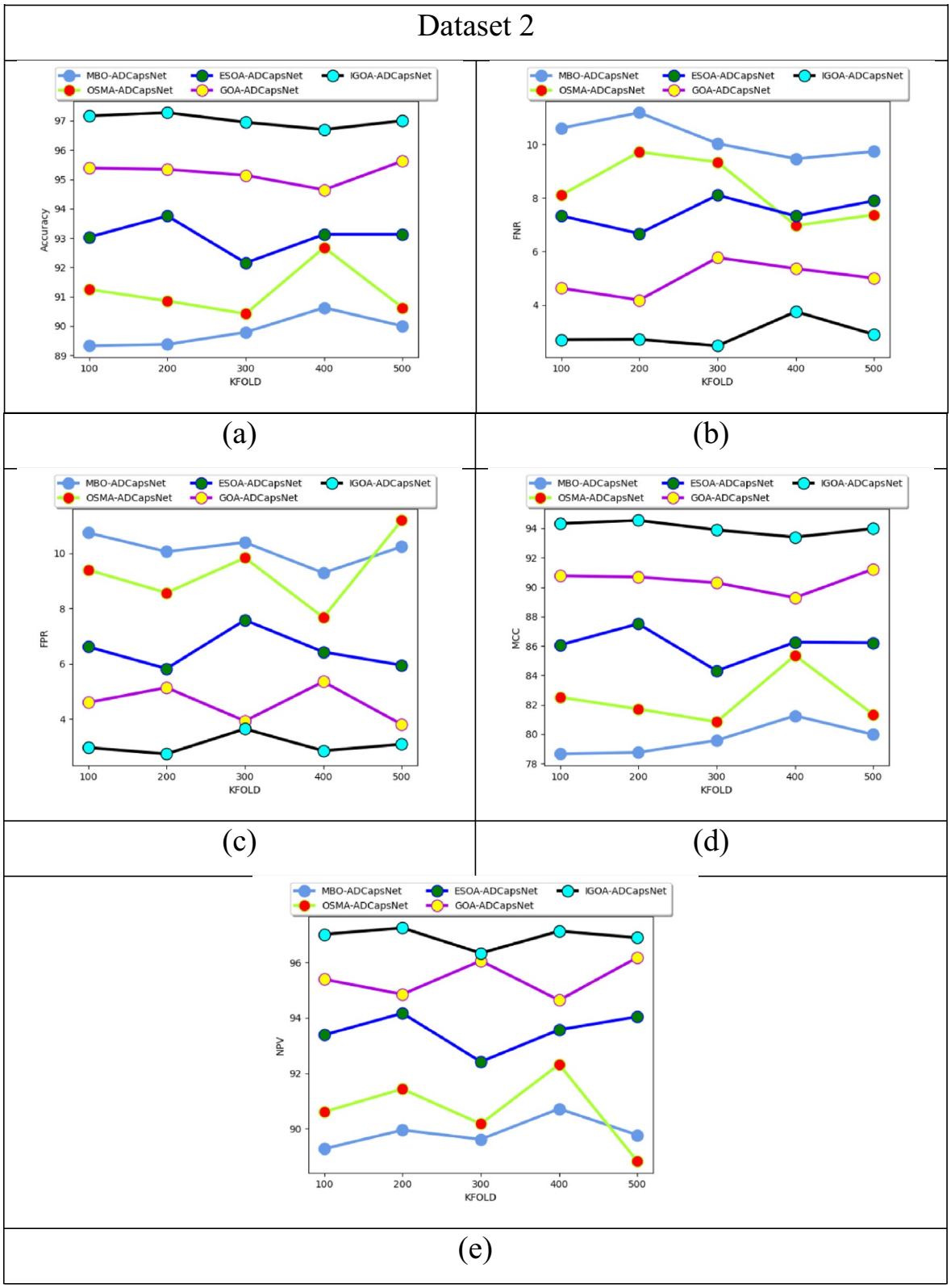

Fig. 8 (continued)

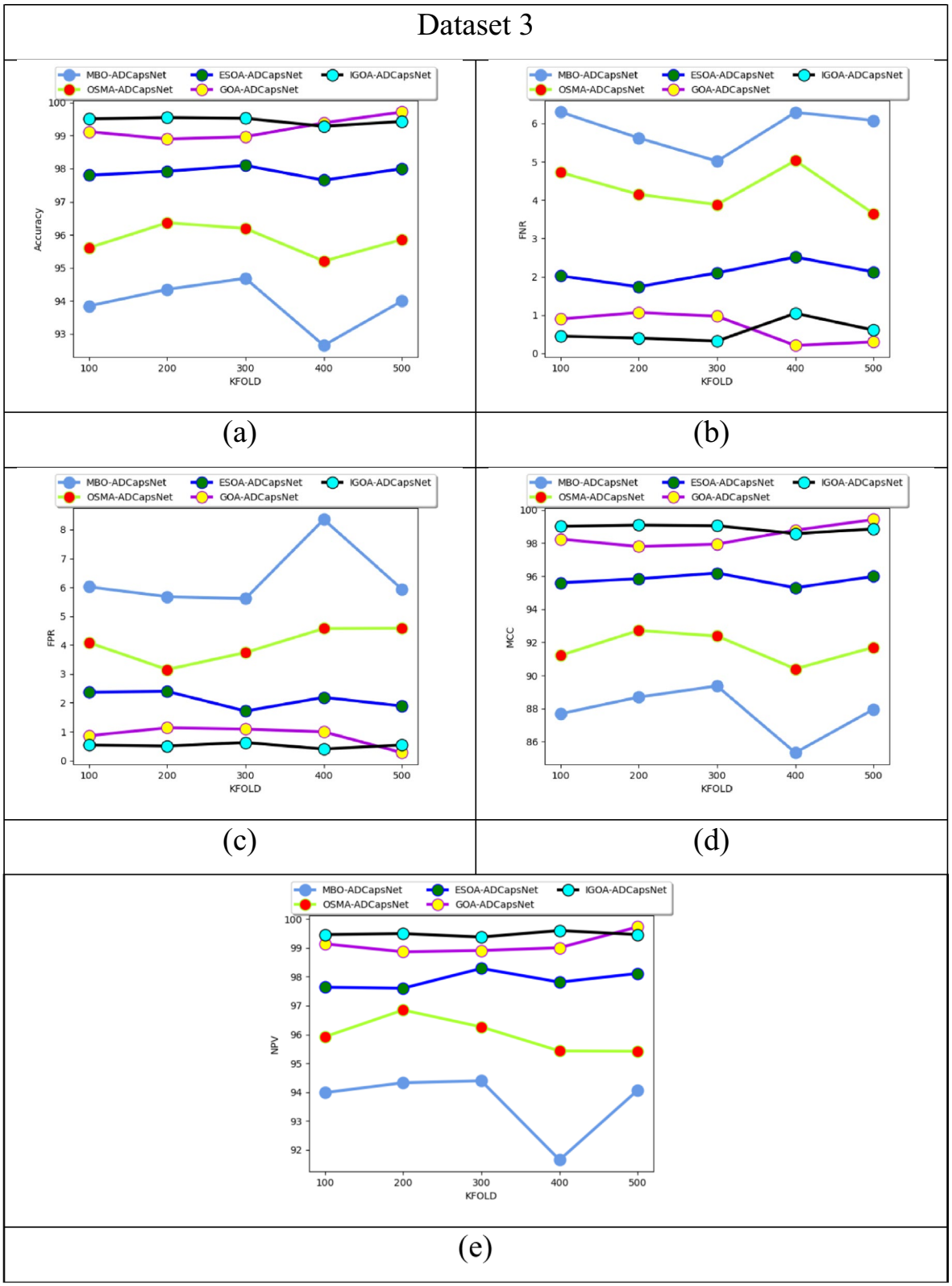

Fig. 8 (continued)

https://doi.org/10.1007/s44196-025-00940-2

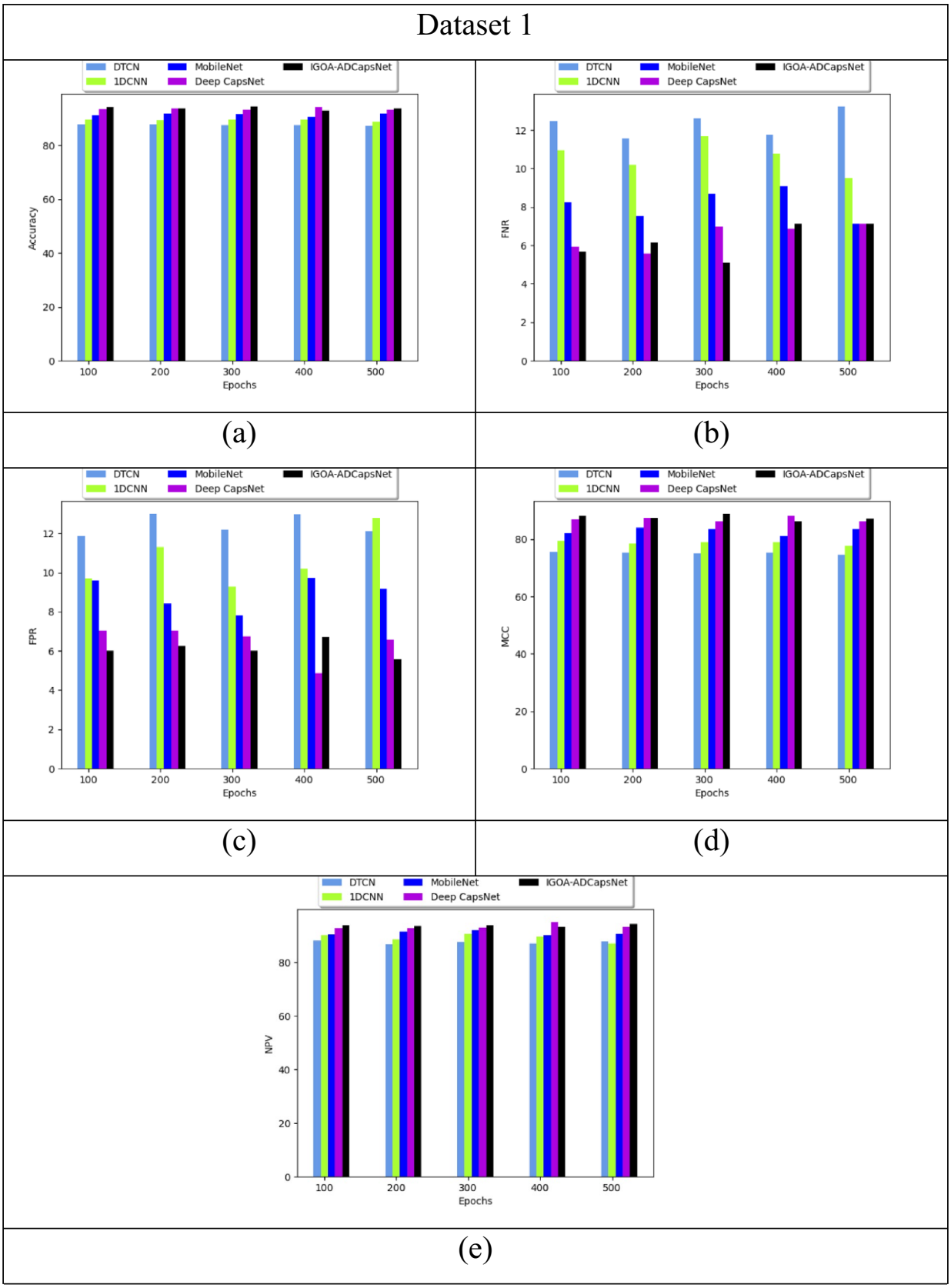

Fig. 9 K-fold-based performance examination of the deployed anomaly detection and intrusion prevention model when compared with conventional techniques in terms of **a** Accuracy, **b** FNR, **c** FPR, **d** MCC, and **e** NPV

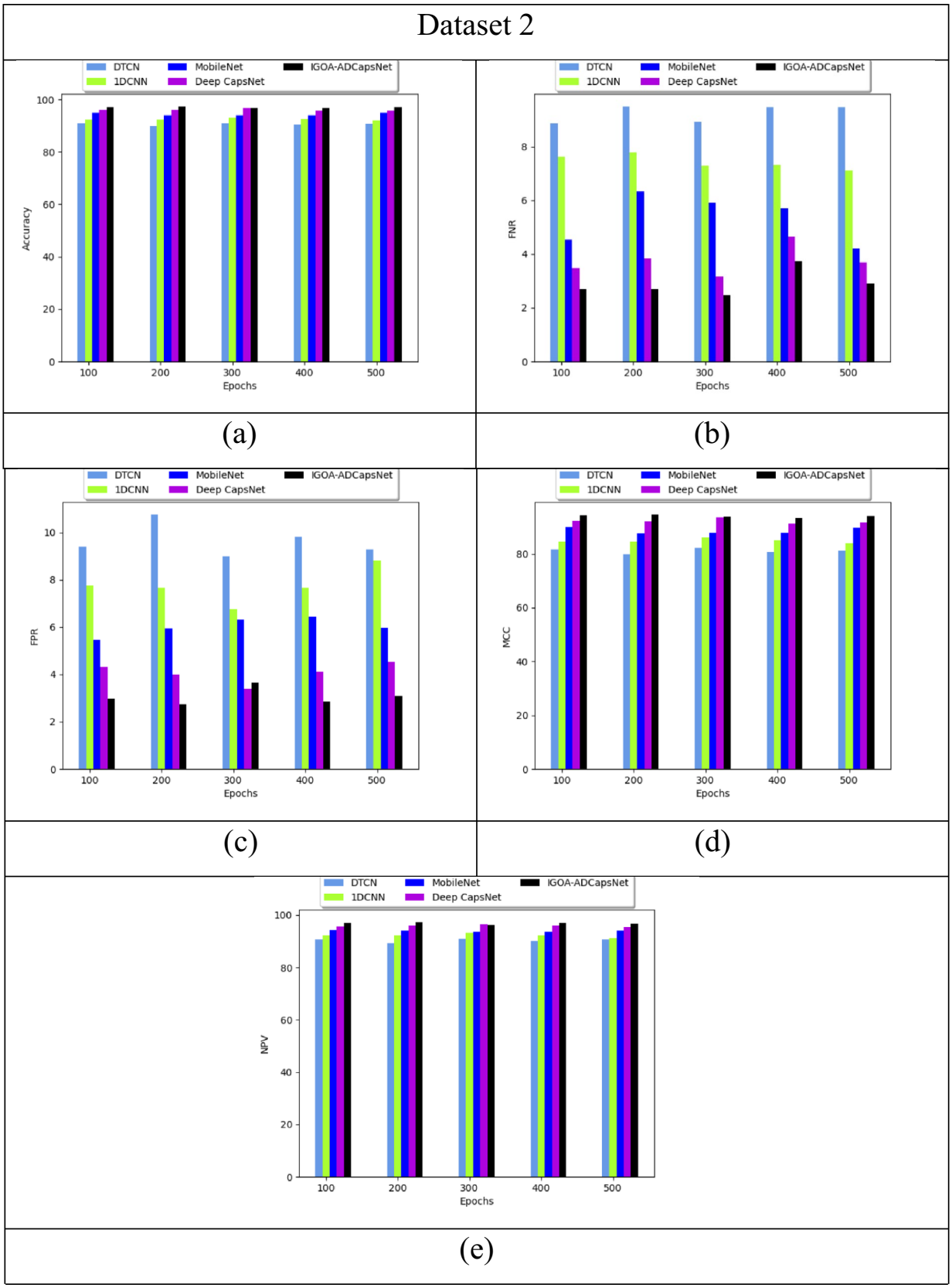

Fig. 9 (continued)

https://doi.org/10.1007/s44196-025-00940-2

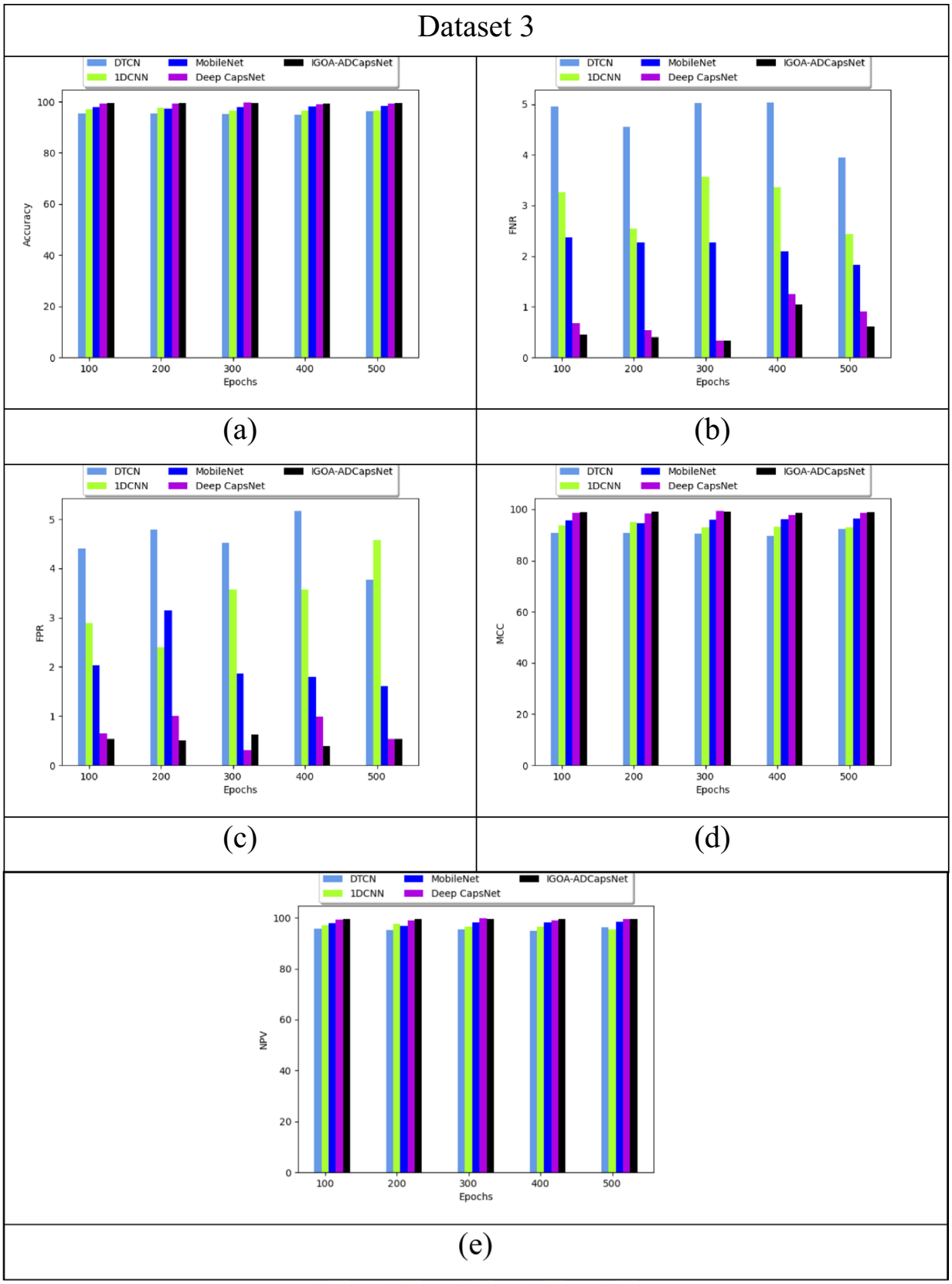

Fig. 9 (continued)

we are considering the FNR as the validation metric, and the corresponding discussion on the performance provided by the deployed IGOA-ADCapsNet-based anomaly detection and privacy preservation model for Datasets 1, 2, and 3 is presented. The FNR of the deployed IGOA-ADCapsNet-based anomaly detection and privacy preservation model is 4.78%, 22.22%, 36.36%, and 40.17% improved for Dataset 1; 25%, 36.84%, 48.57%, and 62.11% enhanced for Dataset 2; and 38.46%, 61.9%, 75.76% and 84% better for Dataset 3 than the conventional models Like Deep CapsNet, MobileNet, 1DCNN, and DTCM models, respectively, while considering the k-fold value as 400. Throughout the analysis, the deployed IGOA-ADCapsNet-based anomaly detection and privacy preservation model has offered better performance. The lower FNR values than the other techniques given out by the deployed IGOA-ADCapsNet-based model show that the number of negative predictions made by this technique is less, thus making it suitable for real-time anomaly detection tasks in the IoT ecosystem.

Table 4 Algorithm-based performance assessment of the implemented anomaly detection and intrusion prevention model

Algorithms/terms	MBO-ADCapsNet [31]	OSMA-ADCapsNet [32]	ESOA-ADCapsNet [33]	GOA-ADCapsNet [26]	IGOA-ADCapsNet
Dataset 1					
Accuracy	86.300	88.300	91.300	91.500	93.700
Sensitivity	88.500	87.500	91.200	91.200	92.900
Specificity	84.300	89.200	91.500	91.800	94.400
Precision	84.500	88.700	91.200	91.500	94.200
FPR	15.700	10.800	8.500	8.200	5.600
FNR	11.500	12.500	8.800	8.800	7.100
NPV	84.300	89.200	91.500	91.800	94.400
FDR	15.500	11.300	8.800	8.500	5.800
F1-Score	86.400	88.100	91.200	91.300	93.500
MCC	72.800	76.700	82.700	83.000	87.300
Dataset 2					
Accuracy	90.000	90.600	93.100	95.600	97.000
Sensitivity	90.300	92.600	92.100	95.000	97.100
Specificity	89.800	88.800	94.000	96.200	96.900
Precision	88.900	88.200	93.300	95.800	96.600
FPR	10.200	11.200	6.000	3.800	3.100
FNR	9.700	7.400	7.900	5.000	2.900
F1-Score	89.600	90.400	92.700	95.400	96.900
NPV	89.800	88.800	94.000	96.200	96.900
FDR	11.100	11.800	6.700	4.200	3.400
MCC	80.000	81.300	86.200	91.200	94.000
Dataset 3					
Accuracy	94.000	95.900	98.000	99.700	99.400
Sensitivity	93.900	96.400	97.900	99.700	99.400
Specificity	94.100	95.400	98.100	99.700	99.500
Precision	93.400	94.900	97.900	99.700	99.400
FPR	5.900	4.600	1.900	0.300	0.500
FNR	6.100	3.600	2.100	0.300	0.600
NPV	94.100	95.400	98.100	99.700	99.500
FDR	6.600	5.100	2.100	0.300	0.600
F1-Score	93.600	95.600	97.900	99.700	99.400
MCC	88.000	91.700	96.000	99.400	98.900

 https://doi.org/10.1007/s44196-025-00940-2

Table 5 Detection models-based performance assessment of the implemented anomaly detection and intrusion prevention model

Algorithms/terms	DTCN [28]	1DCNN [29]	MobilNet [30]	Deep CapsNet [27]	IGOA-ADCapsNet
Dataset 1					
Accuracy	87.300	88.800	91.800	93.200	93.700
Sensitivity	86.800	90.500	92.900	92.900	92.900
Specificity	87.900	87.200	90.800	93.400	94.400
Precision	87.400	87.300	90.700	93.200	94.200
FPR	12.100	12.800	9.200	6.600	5.600
FNR	13.200	9.500	7.100	7.100	7.100
NPV	87.900	87.200	90.800	93.400	94.400
FDR	12.600	12.700	9.300	6.800	5.800
F1-Score	87.100	88.900	91.800	93.000	93.500
MCC	74.700	77.700	83.700	86.300	87.300
Dataset 2					
Accuracy	90.600	92.000	94.900	95.900	97.000
Sensitivity	90.500	92.900	95.800	96.300	97.100
Specificity	90.700	91.200	94.000	95.500	96.900
Precision	89.800	90.500	93.600	95.100	96.600
FPR	9.300	8.800	6.000	4.500	3.100
FNR	9.500	7.100	4.200	3.700	2.900
NPV	90.700	91.200	94.000	95.500	96.900
FDR	10.200	9.500	6.400	4.900	3.400
F1-Score	90.200	91.700	94.700	95.700	96.900
MCC	81.200	84.000	89.800	91.700	94.000
Dataset 3					
Accuracy	96.100	96.400	98.300	99.300	99.400
Sensitivity	96.000	97.600	98.200	99.100	99.400
Specificity	96.200	95.400	98.400	99.500	99.500
Precision	95.800	95.000	98.200	99.400	99.400
FPR	3.800	4.600	1.600	0.500	0.500
FNR	4.000	2.400	1.800	0.900	0.600
NPV	96.200	95.400	98.400	99.500	99.500
FDR	4.200	5.000	1.800	0.600	0.600
F1-Score	95.900	96.300	98.200	99.200	99.400
MCC	92.300	92.900	96.600	98.600	98.900

6.6 Performance Assessment of the Implemented Anomaly Detection and Intrusion Prevention Protocol

The performance assessment on the implemented IGOA-ADCapsNet-based anomaly detection and privacy preservation protocol is made by contrasting it against several conventional algorithms and detection techniques, and the assessment results are provided in Tables 4 and 5, respectively. The performance assessment is carried out by considering an epoch count of 400. The performance is assessed by utilizing both Type 1 and Type 2 measures for all the datasets. Out of various metrics, the accuracy evaluation is considered in this section. When dataset 1 is taken into consideration, the accuracy of the implemented IGOA-ADCapsNet-based anomaly detection and privacy preservation protocol is 7.33%, 5.52%, 2.07%, and 0.54% higher than the DTCN, 1DCNN, MobileNet, and Deep CapsNet models, respectively. Similarly, the implemented IGOA-ADCapsNet-based anomaly detection and privacy preservation protocol is 8.57%, 6.12%, 2.63%, and 2.4% more accurate than the MBO-ADCapsNet, OSMA-ADCapsNet, ESOA-ADCapsNet, and GOA-ADCapsNet

algorithms, respectively. These enhanced values of accuracy prove the enhanced quality outcomes achieved by the implemented IGOA-ADCapsNet-based anomaly detection and privacy preservation protocol than several existing approaches. Thus, the accurate detection outcomes made by means of the implemented IGOA-ADCapsNet-based anomaly detection and privacy preservation protocol make it possible to implement effective intrusion prevention measures on the IoT platform.

6.7 ROC Evaluation of the Suggested Anomaly Detection and Intrusion Prevention Approach

The ROC evaluation of the suggested anomaly detection and intrusion prevention approach is done, and the results are provided as follows in Fig. 10. The ROC computation is carried out by modifying the FPR with the True Positive Rate (TPR). For FPR of 0.2, the TPR of the suggested IGOA-ADCapsNet-based anomaly detection and privacy preservation approach is 7.14%, 8.43%, 18.42%, and 32.35% more than the Deep CapsNet,

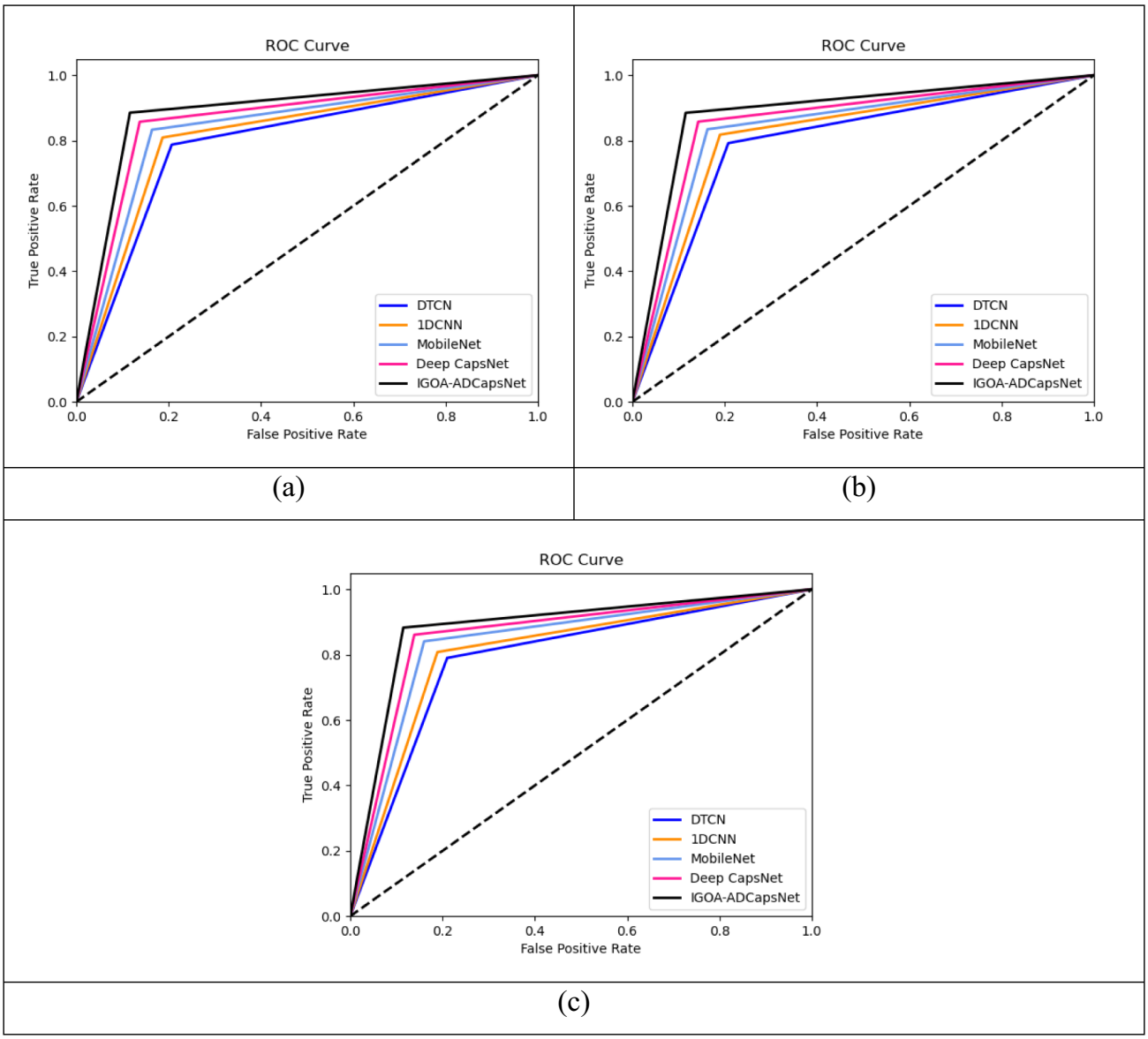

Fig. 10 ROC of suggested anomaly detection and privacy prevention scheme for**a** Dataset 1, **b** Dataset 2, and **c** Dataset 3

MobileNet, 1DCNN, and DTCN models, respectively, regarding Dataset 1. Thus, the enhanced performance of the suggested IGOA-ADCapsNet-based anomaly detection and privacy preservation approach is proven.

6.8 Statistical Validation of the Designed Anomaly Detection and Intrusion Prevention Technique

The statistical validation of recommended anomaly detection and intrusion prevention techniques is carried out, and the results are given in Table 6. The statistical computation is conducted by considering the accuracy measure. The accuracy is evaluated using statistical measures such as median, mean, standard deviation, best, and worst. The mean value of the implemented IGOA-ADCapsNet-based anomaly detection and privacy preservation approach is 11.98%, 5.83%, 3.17%, and 2.83% better than the MBO-ADCapsNet, OSMA-ADCapsNet, ESOA-ADCapsNet, and GOA-ADCapsNet algorithms, respectively, regarding dataset 1. Thus, an enhanced detection rate on anomaly detection is provided by the implemented IGOA-ADCapsNet-based anomaly detection and privacy preservation approach than other techniques.

6.9 Performance Assessment of the Implemented Model over the Existing Model

In the realm of IoT security, the integration of advanced protocols for anomaly identification and privacy protection is paramount. Table 7 delineates the comparative performance of the recently implemented IGOA-ADCapsNet-based anomaly identification and confidentiality preservation protocol against several traditional models. Specifically, when evaluating the CICIDS 2017 KNN dataset, the IGOA-ADCapsNet-based protocol demonstrates superior performance, surpassing the OMP model and the CPP model by 3.33% and 5.52%, respectively. This notable improvement underscores the efficacy of the IGOA-ADCapsNet approach in enhancing intrusion prevention measures on the IoT platform. By leveraging this advanced protocol, IoT systems can achieve heightened levels of security and privacy, addressing the critical challenges posed by potential intrusions.

Table 6 Statistical validation of the designed anomaly detection and intrusion prevention technique

Algorithms/terms	MBO-ADCapsNet [31]	OSMA-ADCapsNet [32]	ESOA-ADCapsNet [33]	GOA-ADCapsNet [26]	IGOA-ADCapsNet
Dataset 1					
Best	86.300	88.300	91.300	91.500	93.700
Worst	88.500	87.500	91.200	91.200	92.900
Mean	84.300	89.200	91.500	91.800	94.400
Median	84.500	88.700	91.200	91.500	94.200
Standard Deviation	15.700	10.800	8.500	8.200	5.600
Dataset 2					
Best	90.000	90.600	93.100	95.600	97.000
Standard Deviation	90.000	90.600	93.100	95.600	97.000
Mean	89.800	88.800	94.000	96.200	96.900
Worst	90.300	92.600	92.100	95.000	97.100
Median	88.900	88.200	93.300	95.800	96.600
Dataset 3					
Mean	94.100	95.400	98.100	99.700	99.500
Standard Deviation	5.900	4.600	1.900	0.300	0.500
Best	94.000	95.900	98.000	99.700	99.400
Median	93.400	94.900	97.900	99.700	99.400
Worst	93.900	96.400	97.900	99.700	99.400

Table 7 Classifier-based performance assessment of the implemented anomaly detection and intrusion prevention model

Algorithms/terms	OMP [36]	CPP [37]	IGOA-ADCapsNet
Dataset 1			
FPR	7.809984	7.407407	5.600
Sensitivity	91.81933	92.7669	92.900
FDR	6.256046	5.895546	5.800
MCC	0.83796	0.852044	82.700
FNR	8.18067	7.233102	7.100
Precision	93.74395	94.10445	94.200
NPV	89.83915	90.94504	94.400
Accuracy	91.9823	92.69027	93.700
F1-Score	92.77166	93.43089	93.500
Specificity	92.19002	92.59259	94.400
Dataset 2			
FDR	4.60474	3.122651	3.400
Specificity	96.99501	97.9736	96.900
FNR	2.979109	2.022852	2.900
MCC	0.937388	0.957601	94.000
Accuracy	97.00513	97.97499	97.000
FPR	3.004987	2.0264	3.100
NPV	98.06732	98.69251	96.900
Precision	95.39526	96.87735	96.600
F1-Score	96.20121	97.42414	96.900
Sensitivity	97.02089	97.97715	97.100
Dataset 3			
MCC	0.945471	0.96518	96.000
FDR	2.097902	1.362916	2.100
Specificity	97.343	98.26892	98.100
FPR	2.657005	1.731079	1.900
Sensitivity	97.28364	98.29438	97.900
FNR	2.716361	1.705622	2.100
Precision	97.9021	98.63708	97.900
NPV	96.5655	97.83567	98.100
Accuracy	97.30973	98.28319	98.000
F1-Score	97.59189	98.46543	97.900

6.10 Performance Analysis on Compression Techniques Considering Lightweight IoT Devices

In this phase, the performance of various compression models considering lightweight IoT devices overdeveloped IGOA-ADCapsNet is observed and discussed in Table 8. Recently, compression schemes considering lightweight IoT devices used in the validation are SZ4IoT, OCFSDA, GCIGNN-ENet, and 1D CNN-KD. In this phase, multiple experiments were executed to verify the performance of the developed IGOA-ADCapsNet model over different performance measures. Accuracy developed IGOA-ADCapsNet-based anomaly detection and intrusion prevention technique gained superior performance as 11.07%, 8.95%, 6.90%, and 4.80% than the recent compression procedures considering lightweight IoT like SZ4IoT, OCFSDA, GCIGNN-ENet, and 1DCNN-KD, respectively, in dataset 1. In the FNR computation, the implemented IGOA-ADCapsNet gained fewer errors than other frameworks like SZ4IoT, OCFSDA, GCIGNN-ENet, and 1DCNN-KD. Reducing the errors in the suggested IGOA-ADCapsNet technique helps to accomplish superior anomaly detection efficiency than other models.

 https://doi.org/10.1007/s44196-025-00940-2

Table 8 Performance analysis on compression techniques considering lightweight IoT devices

Algorithms/terms	SZ4IoT [41]	OCFSDA [42]	GCIGNN-ENet [43]	1DCNN-KD [44]	IGOA-ADCapsNet
Dataset 1					
Accuracy	84.35792	85.99727	87.648	89.40118	93.700
Sensitivity	72.86639	75.51082	78.10308	80.80839	92.900
Specificity	91.60787	92.3937	93.33333	94.4254	94.400
Precision	84.56284	85.8265	87.46585	89.44672	94.200
FPR	8.392128	7.606305	6.666667	5.574599	5.600
FNR	27.13361	24.48918	21.89692	19.19161	7.100
NPV	84.25546	86.08265	87.73907	89.37842	94.400
FDR	15.43716	14.1735	12.53415	10.55328	5.800
F1-Score	78.28011	80.33887	82.51974	84.90841	93.500
MCC	0.666109	0.698782	0.732965	0.770085	87.300
Dataset 2					
Accuracy	85.89481	87.77322	89.37842	91.18852	97.000
Sensitivity	75.33753	78.10794	80.79654	83.80414	97.100
Specificity	92.35003	93.58367	94.39134	95.39121	96.900
Precision	85.7582	87.97814	89.37842	91.18852	96.600
FPR	7.649972	6.416332	5.608656	4.608789	3.100
FNR	24.66247	21.89206	19.20346	16.19586	2.900
NPV	85.96311	87.67077	89.37842	91.18852	96.900
FDR	14.2418	12.02186	10.62158	8.811475	3.400
F1-Score	80.21083	82.74976	84.87109	87.34053	96.900
MCC	0.696753	0.736437	0.769517	0.807705	94.000
Dataset 3					
Accuracy	89.03689	90.77869	92.56603	94.41029	99.400
Sensitivity	80.37094	83.07308	86.19154	89.4209	99.400
Specificity	94.08903	95.19885	96.11771	97.11927	99.500
Precision	88.79781	90.84699	92.52049	94.39891	99.400
FPR	5.910975	4.801147	3.88229	2.880731	0.500
FNR	19.62906	16.92692	13.80846	10.5791	0.600
NPV	89.15642	90.74454	92.5888	94.41598	99.500
FDR	11.20219	9.153005	7.479508	5.601093	0.600
F1-Score	84.37449	86.7863	89.24395	91.8425	99.400
MCC	0.761871	0.799145	0.836976	0.876702	98.900

6.11 Performance Analysis on Intrusion Prevention Techniques

Different analyses executed in the suggested IGOA-ADCapsNet by considering the recent intrusion prevention techniques are offered in Table 9. In this phase, various intrusion prevention models Like BSNN-BC, H-DOCTOR, HB-IDP, and CCMA-BC techniques are employed to verify the intrusion prevention performance of the developed model. In the FDR computation, the developed IGOA-ADCapsNet technique had a minimal error than other techniques. Reducing the errors in the suggested technique helps to detect different attacks in the early phases and also improves the visibility of the network. In the sensitivity validation, the recommended IGOA-ADCapsNet-based intrusion prevention model gained superior efficiency as 17.72%, 26.3%, 21.8% and 7.17% better than the recent intrusion detection techniques BSNN-BC, H-DOCTOR, HB-IDP and CCMA-BC, respectively, in dataset 2. Accomplishing higher sensitivity and accuracy in the developed IGOA-ADCapsNet helps to tackle the data breaching issues and also easily identify the malicious activities in the network.

Table 9 Analysis of intrusion prevention models over implemented IGOA-ADCapsNet

Algorithms/terms	BSNN-BC [45]	H-DOCTOR [46]	HB-IDP [47]	CCMA-BC [48]	IGOA-ADCapsNet
Dataset 1					
Accuracy	95.40073	85.18898	87.02186	91.73497	93.700
Sensitivity	91.21489	74.118	77.09091	84.70996	92.900
Specificity	97.64069	92.09019	92.99781	95.70563	94.400
Precision	95.38934	85.38251	86.88525	91.76913	94.200
FPR	2.359315	7.909813	7.002188	4.294369	5.600
FNR	8.785108	25.882	22.90909	15.29004	7.100
NPV	95.40642	85.09221	87.09016	91.7179	94.400
FDR	4.610656	14.61749	13.11475	8.230874	5.800
F1-Score	93.25543	79.35248	81.69557	88.09836	93.500
MCC	0.898204	0.683082	0.720058	0.819369	87.300
Dataset 2					
Accuracy	90.39162	86.94217	88.67259	95.08197	97.000
Sensitivity	82.4813	76.87896	79.69278	90.59857	97.100
Specificity	94.94261	93.03601	93.95912	97.49562	96.900
Precision	90.36885	86.9877	88.5929	95.11612	96.600
FPR	5.057389	6.963992	6.040875	2.504378	3.100
FNR	17.5187	23.12104	20.30722	9.401431	2.900
NPV	90.40301	86.9194	88.71243	95.06489	96.900
FDR	9.631148	13.0123	11.4071	4.88388	3.400
F1-Score	86.24511	81.62154	83.90749	92.8024	96.900
MCC	0.790802	0.718833	0.754565	0.891315	94.000
Dataset 3					
Accuracy	93.57923	89.8224	91.63251	96.34563	99.400
Sensitivity	87.88462	81.44712	84.62267	92.97725	99.400
Specificity	96.7161	94.7027	95.58483	98.12207	99.500
Precision	93.64754	89.95902	91.53005	96.31148	99.400
FPR	3.283898	5.297297	4.415168	1.877934	0.500
FNR	12.11538	18.55288	15.37733	7.02275	0.600
NPV	93.54508	89.7541	91.68374	96.3627	99.500
FDR	6.352459	10.04098	8.469945	3.688525	0.600
F1-Score	90.6746	85.49172	87.94094	94.615	99.400
MCC	0.858869	0.779111	0.816968	0.918834	98.900

6.12 Scalability Evaluation of the Suggested Anomaly Detection and Intrusion Prevention Approach

The scalability evaluation of the proposed anomaly detection and intrusion prevention approach has been conducted, with the results illustrated in Fig. 11. This evaluation focuses on the relationship between data size and accuracy. As depicted in the figure, there is a positive correlation between the increase in data size and the accuracy of the anomaly detection system. This validation effectively enhances the overall efficiency of the network by handling enormous data and also provides superior outcomes while learning the features. This trend indicates that the suggested approach not only scales effectively with larger datasets but also enhances its performance, thereby ensuring robust anomaly detection and intrusion prevention in expansive IoT environments.

 https://doi.org/10.1007/s44196-025-00940-2

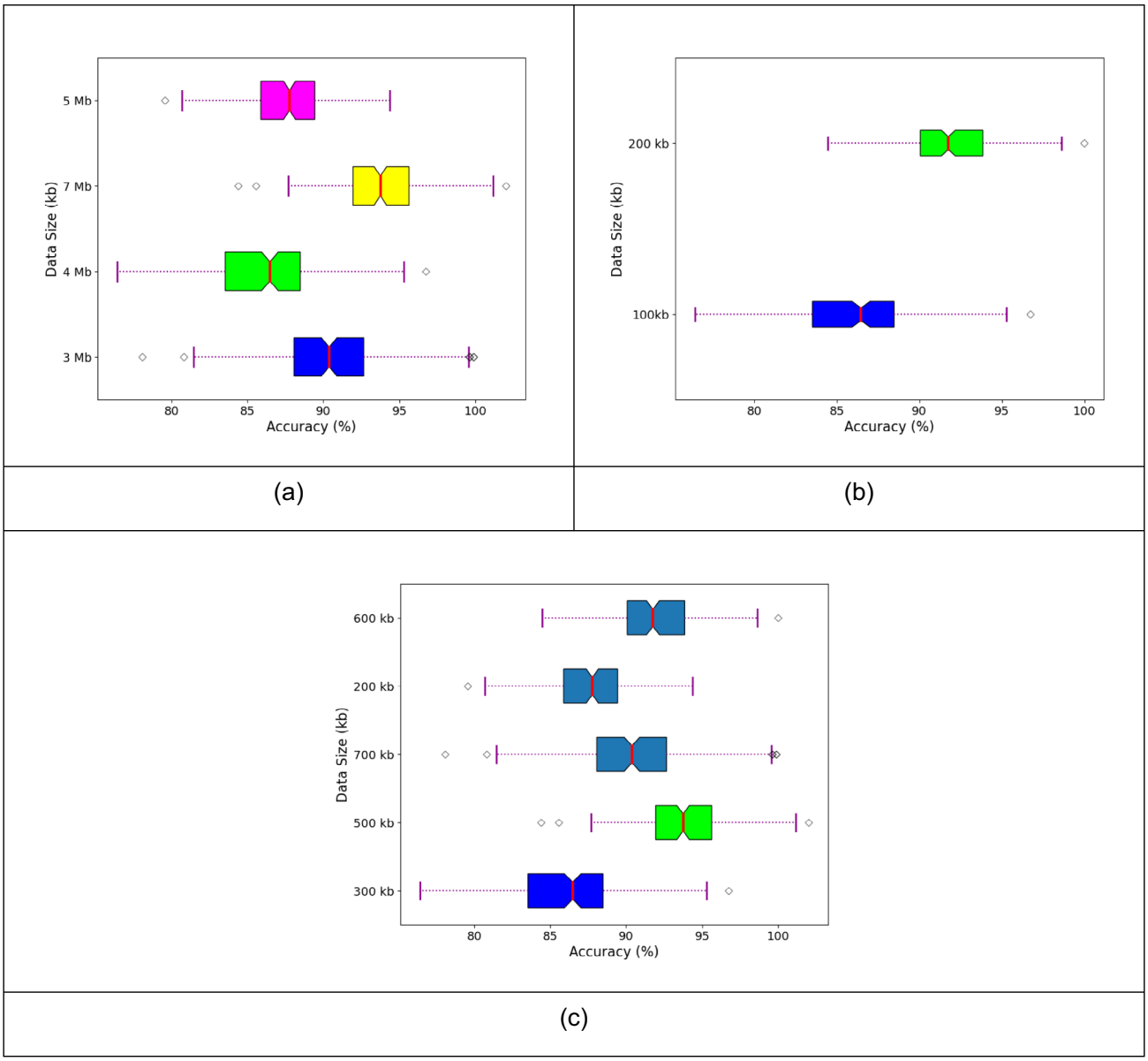

Fig. 11 Scalability evaluation of the suggested anomaly detection and intrusion prevention scheme over **a** Dataset 1, **b** Dataset 2 and **c** Dataset 3

6.13 Analysis on Training Time

Training time validation carried out in the developed IGOA-ADCapsNet-based anomaly detection and intrusion detection model is provided in Table 10. This process supports computing the time required for training the developed network IGOA-ADCapsNet over classical schemes. Training time of the suggested IGOA-ADCapsNet is 13.6 (Mins) while training the network, which is less than the classical schemes. Attaining minimal training time supports to verify the overall efficiency of the developed technique. This computation supports observing the overall efficiency of IGOA-ADCapsNet and then displays several issues that arise in the training phase. Moreover, it assigns the essential resources to the respective network and also enhances the scalability. Reduced training time helps to enhance the processing speed and also offers superior outcomes

Table 10 Analysis on training time in implemented IGOA-ADCapsNet-based anomaly detection and intrusion prevention models

Analysis of heuristic models	
Heuristic techniques	Time (Mins)
MBO-ADCapsNet [31]	16.4
OSMA-ADCapsNet [32]	14.7
ESOA-ADCapsNet [33]	18.4
GOA-ADCapsNet [26]	14.9
IGOA-ADCapsNet	13.6
Anomaly detection and intrusion prevention techniques	Time (Mins)
DTCN [28]	19.3
1DCNN [29]	17.3
MobilNet [30]	18.4
Deep CapsNet [27]	14.98
IGOA-ADCapsNet	13.6

Table 11 Computational cost validation in implemented IGOA-ADCapsNet-based anomaly detection and intrusion prevention models

Techniques	Computational cost
MBO-ADCapsNet [31]	$O\left[M_{it}+2+N_p+3+C_h+2\right]$
OSMA-ADCapsNet [32]	$O\left[M_{it}+3+N_p+1+C_h+1\right]$
ESOA-ADCapsNet [33]	$O\left[M_{it}+2+N_p+2+C_h+2\right]$
GOA-ADCapsNet [26]	$O\left[M_{it}+1+N_p+1+C_h\right]$
IGOA-ADCapsNet	$O\left[M_{it}+N_p+C_h\right]$

without any delay. Thus, attaining reduced training time enables the developed IGOA-ADCapsNet to accomplish reliable results without any misclassifications.

6.14 Analysis on Computational Cost

Computational cost validation carried out in the developed IGOA-ADCapsNet-based anomaly detection and intrusion prevention model is offered in Table 11. In this phase, chromosome length is indicated as C_h, population count is given as N_p and maximum iteration count is specified as M_{it}. Here, the computational cost validation is carried out to verify the memory spaces required for the network while detecting the anomalies and preventing the intrusions. This computation supports obtaining faster training by reducing the validation time. Moreover, enhances the training and also provides higher generalization by eliminating the network complexity. In this validation efficiency of the developed IGOA-ADCapsNet is computed over different scenarios. Computational cost validation supports to obtain feasible outcomes in a particular range by resolving several limitations.

6.15 Discussions

6.15.1 Handling Concept Drift

The process employed to manage and adapt a novel technique for modifying the data distributions over time in machine learning is called Handling concept drift. This process mainly focuses on the relationship between the target shift and features, and in some cases, it affects the accuracy when the system is updated. The handling concept drift helps to maintain better performance when the data patterns are used, and also it needs a novel mechanism to execute the continuous monitoring. Here, the statistical properties presented in the data are employed to train

the network over time, and this procedure is termed as concept drift. Moreover, the drift concept may affect the accuracy and also slow down the prediction procedure for the new data presented with various patterns. This technique also requires a network performance detection procedure for accuracy and errors over time. Finally, using the handling drift concept in the developed IGOA-ADCapsNet-based anomaly detection with privacy preservation technique helps to maintain better accuracy and also enhances the applicability over real-world systems.

6.15.2 Online Model Retraining

In this process, the updating process happens continuously when the new real-time data arrives in the machine learning technique. This process permits the developed framework to accept the changing conditions without any retraining from the initial stages using enormous data samples. Moreover, this technique upgrades the parameters incrementally due to the presence of novel data. Classical batch retaining techniques perform retaining procedures periodically with the novel dataset. But, in the online model retraining, instantaneous updating is performed once the new data is received. This online model retraining has the efficiency to handle the incoming data streams, process them, and then learn the significant details from entirely new data. This technique also has better-adjusting efficiency, which did not affect the performance of the developed network. Adapting an online model retraining helps the developed IGOA-ADCapsNet-based anomaly detection with a privacy preservation technique to accept the modified data patterns, which helps to enhance the accuracy and also rectify the overall performance degradation.

6.15.3 Integrating Proposed Technique with Existing IoT Platforms

In the initial stage, the developed framework requires finding the appropriate data for the IoT devices, and it needs to execute the processing procedure. Later, a suitable deep learning mechanism is selected according to the task and then applied to the developed framework for the appropriate edge computing infrastructure with continuous monitoring, when the new data is available. Here, the developed framework uses ADCapsNet to execute the recognition task, and also executes training in the developed ADCapsNet model improves the recognition efficiency. To enhance the recognition efficiency, the developed framework tunes the hyperparameters of ADCapsNet. Moreover, these techniques help to acquire the complicated features for real-world analysis. Integrating the classical IoT models with developed techniques aids in offering real-world analysis with good decision-making.

6.15.4 Creating User-Friendly Interfaces for Administrators

In IoT devices, user-friendly interfaces are created by considering and understanding the user requirements. Most of the IoT platforms are simple, with easy access policies and clarity. These IoT devices easily identify the basic actions of the administrator needs to execute. Most of the IOT platforms need more experience in accessing information. Designing a customizable dashboard helps to offer more focus on the respective details. Moreover, collecting feedback regularly from the user helps to improve the performance of the IoT platform according to their needs.

6.15.5 Generalizing Real-World IoT Networks in Different Scenarios

In this section, generalizng the real-world IoT network in various scenarios is discussed as follows.

6.15.5.1 Imbalanced Traffic Patterns In most cases, the IoT uses uneven traffic patterns over different classes. Arising imbalance issues generate huge impacts in the anomaly detection and intrusion prevention phase. Moreover, the imbalanced traffic patterns are prone to increasing the false alarm rate and also fail to identify the attacks more precisely. In some cases, they lead to inaccurate outcomes and also generate more difficulties in detecting the anomalies in the network, which leads to a

lack of network security. Due to these difficulties, network efficiency is lagging in terms of latency, bandwidth and throughput, which makes the traffic classification process complex.

6.15.5.2 Adversarial Attacks This attack generates duplicate samples to trick the network, and it makes the network to generate incorrect outcomes and leads to more difficulties. Adversarial attacks lead to more vulnerability while carrying out specific tasks. Moreover, these attacks permit the attacker to access the sensitive information that leads to data breach issues as well as unauthorized access. In addition, they lead to incorrect decision-making, which affects the overall security of the network in dynamic conditions.

6.15.5.3 Noisy Data Using noisy data in the real-world IoT model malfunction issues in the network, and also leads to more errors while transmitting the data. In the real world, IoT noisy data refers to irrelevant, erroneous or inaccurate data that generates more complications while processing the samples. Noisy data in the IoT creates more fluctuation issues and also leads to failure in the hardware. In some cases, more errors arise while transferring the data and which leads to inaccurate outcomes. In addition, the reliability of the system is affected, which creates more impacts in the decision-making phase about the anomalies and intrusions.

6.15.6 Potential Biases in Artificial Intelligence-Based Intrusion Detection Models

Generally, the artificial intelligence-based intrusion detection models are prone to various biases, which create significant impacts in their outcomes. The artificial intelligence model uses various data to execute the training, which leads to some accuracy-related issues and also generates traffic in the network. Artificial intelligence models lead to sensitive issues while analyzing certain patterns and are subject to several losses. While analyzing the performance, it leads to several losses, which creates network traffic in the real-world systems.

6.15.6.1 Evasion Attacks These kinds to attacks affect the system logs, network packets through malicious inputs. Here, the attacker may add noise to the payloads and display the nodes as malicious. This generates a huge impact while modifying the timings and also leads to network traffic. Evision attack failed to detect the real anomalies.

6.15.6.2 Poisoning Attacks In this attack, the attackers may corrupt the training data utilized to design the intrusion detection system. Here, the malicious information is injected that makes the network to study the incorrect patterns in the stored information. This attack generates misclassification in the network and also fails to detect the actual attacks along which leads to more errors.

7 Conclusion

An approach for network intrusion prevention and anomaly detection with deep learning in the IoT sector was developed. The designed model performed three major steps: data collection, feature extraction, and anomaly detection. Initially, the required data were gathered from the publicly available data resource. Subsequently, the features were extracted from the input data. The weight of these features was then optimally tuned by the IGOA. The resultant weights were multiplied by the weights that were optimally selected, and these weighted features were then given as input to the ADCapsNet system to detect the anomaly present in the network. To improve the detection performance, the hyperparameters in the ADCapsNet model were optimized by the developed GOA. Finally, the developed model was compared with multiple metrics to find the efficiency of the implemented model. On experimentation, it was seen that the accuracy of the implemented IGOA-ADCapsNet-based anomaly detection and privacy preservation protocol was 7.33%, 5.52%, 2.07%, and 0.54% higher than the DTCN, 1DCNN, MobileNet, and Deep CapsNet models, respectively, for Dataset 1. The experimental result showed that the developed model outperformed conventional outcomes in terms of anomaly detection and intrusion prevention in the IoT network than other conventional models, and thus can be used in various real-time applications.

7.1 Future Work

In the future, out-of-distribution detection mechanisms will be considered in the developed framework for eliminating errors while analyzing the inputs. In the upcoming research work, the developed model will be designed to handle multi-modal data and detect correlated anomalies across devices. Moreover, in the forthcoming research work, new IoT threats like side-channel and AI-generated attacks will be considered, and also various analyses will be executed to observe the efficiency of the developed framework. Furthermore, the transfer learning concept will be considered for handling the large and labelled dataset to enhance the network performance. Also, the explainability techniques will be integrated to increase transparency. In addition, the data augmentation procedure will be suggested to be incorporated in the upcoming research work to enhance the training efficiency of data, along with it transfer learning mechanism will be suggested to enhance the network performance in different scenarios. Additionally, hybrid deep learning techniques such as transformers, GAN and federated learning models will be considered in forthcoming research works for the detection of anomalies.

7.1.1 Real-Time Implementation Strategies Using Edge Computing and 5G-Enable IoT Network

In the forthcoming works, real-time implementation techniques will be considered to reduce the latency and also support offering higher bandwidth for quickly processing the data associated with the edge computing models. Using the real-time techniques in the edge computing models supports achieving real-time monitoring and controlling, along with better decision making. 5G-based IoT network supports executing network slicing and also widely supports use in the ultra-reliable low latency communication models and massive machine type communication models. Designing a novel framework will offer huge support to use in healthcare, smart cities, industrial automation and public safety.

Author Contributions Weiwei Hu, Jafar A. Alzubi, and Shreyas J: Conceptualization, Methodology, Software Data curation, Writing- Original draft preparation, Reviewing and Editing Software, Validation. Yasser A. Ali, Muna Al-Razgan, and Karthikayaen A: Writing- Reviewing and Editing, Visualization, Investigation.

Funding Open access funding provided by Manipal Academy of Higher Education, Manipal. This research work did not receive any funding from the university or any external organization.

Data Availability The Dataset underlying in the article are taken from the following link: Dataset-1: (CICIDS 2017 KNN) "https://www.kaggle.com/code/saqlainhussainshah/cicids-2017-knn/data" access date: 2023-12-20 Dataset-2: (NSL-KDD dataset) "https://www.unb.ca/cic/datasets/nsl.html" access date: 2023-12-20 Dataset-3: (Kddcup99) "https://datahub.io/machine-learning/kddcup99" access date: 2023-12-20

Declarations

Conflict of interest The authors declare no competing interests.

Ethical Approval Not applicable.

Consent for Publication Not applicable.

References

1. Yin, X., Wang, L., Jia, W., Jin, C.: Semi-supervised transformation and deep embedding-based anomaly identification for agricultural internet of things. IEEE Sens. J. **21**(22), 24959–24966 (2021)
2. Sahingoz, O.K., Cekmez, U., Buldu, A.: Internet of things (IoTs) security: intrusion detection using deep learning. J. Web. Eng. **20**(6), 1721–1760 (2021)
3. Alshahrani, H., Maray, M., Aljebreen, M., Alymani, M., Elfaki, M.A., Al Duhayyim, M., Balaji, P., Gupta, D.: Energy aware routing with optimal deep learning based anomaly detection in 6G-IoT networks. Sustain. Energy. Technol. Assess. **60**, 103494 (2023)
4. Kumar, P., Gupta, G.P., Tripathi, R.: Design of anomaly-based intrusion detection system using fog computing for IoT network. Autom. Control. Comput. Sci. **55**, 137–147 (2021)
5. Keserwani, P.K., Govil, M.C., Pilli, E.S., Govil, P.: A smart anomaly-based intrusion detection system for the internet of things (IoT) network using GWO–PSO–RF model. J. Reliab. Intell. Environ. **7**, 3–21 (2021)
6. Hao, Xu., Sun, Z., Cao, Y., Bilal, H.: A data-driven approach for intrusion and anomaly detection using automated machine learning for the internet of things. Soft. Comput. **27**, 14469–14481 (2023)
7. Nimmy, K., Dilraj, M., Sankaran, S., Achuthan, K.: Leveraging power consumption for anomaly detection on IoT devices in smart homes. J. Ambient. Intell. Humaniz. Comput. **14**, 14045–14056 (2023)
8. Moustafa, N., Koroniotis, N., Keshk, M., Zomaya, A.Y., Tari, Z.: Explainable intrusion detection for cyber defences in the internet of things: opportunities and solutions. IEEE. Commun. Surv. Tutor. **25**(3), 1775–1807 (2023)
9. Jeyanthi, D.V., Indrani, B.: IoT-based intrusion detection system for healthcare using RNNBiLSTM deep learning strategy with custom features. Soft. Comput. **27**, 11915–11930 (2023)
10. Nizamudeen, S.M.T.: Intelligent intrusion detection framework for multi-clouds – IoT environment using swarm-based deep learning classifier. J. Cloud. Comput. **12**, 134 (2023)
11. Gopalakrishnan, B., Purusothaman, P.: A new design of intrusion detection in IoT sector using optimal feature selection and high ranking-based ensemble learning model. Peer-to-Peer Netw. Appl. **15**, 2199–2226 (2022)
12. Shirafkan, M., Shahidinejad, A., Ghobaei-Arani, M.: An Intrusion detection system using deep cellular learning automata and semantic hierarchy for enhancing RPL protocol security. Clust. Comput. **26**, 2443–2461 (2023)
13. Ullah, I., Mahmoud, Q.H.: Design and development of a deep learning-based model for anomaly detection in IoT networks. IEEE Access **9**, 103906–103926 (2021)
14. Xu, L., Ding, X., Peng, H., Zhao, D., Li, X.: ADTCD: an adaptive anomaly detection approach toward concept drift in IoT. IEEE Internet Things J. **10**(18), 15931–15942 (2023)
15. Savic, M., Lukic, M., Danilovic, D., Bodroski, Z., Bajović, D., Mezei, I., Vukobratovic, D., Skrbic, S., Jakovetić, D.: Deep learning anomaly detection for cellular IoT with applications in smart logistics. IEEE Access **9**, 59406–59419 (2021)
16. Osen, A., Moustafa, N., Creech, G., Sohrabi, N., Strelzoff, A., Tari, Z., Linkov, I.: An explainable deep learning framework for resilient intrusion detection in IoT-enabled transportation networks. IEEE Trans. Intell. Transp. Syst. **24**(1), 1000–1014 (2023)
17. Ullah, I., Mahmoud, Q.H.: A framework for anomaly detection in IoT networks using conditional generative adversarial networks. IEEE Access **9**, 165907–165931 (2021)
18. Ali, A., Yousaf, M.M.: Novel three-tier intrusion detection and prevention system in software defined network. IEEE Access **8**, 109662–109676 (2020)
19. Sharma, B., Sharma, L., Lal, C., Roy, S.: Anomaly-based network intrusion detection for IoT attacks using deep learning technique. Comput. Electr. Eng. **107**, 108626 (2023)
20. Saba, T., Rehman, A., Sadad, T., Kolivand, H., Bahaj, S.A.: Anomaly-based intrusion detection system for IoT networks through deep learning model. Comput. Electr. Eng. **99**, 107810 (2022)
21. Ntizikira, E., Wang, L., Chen, J., Saleem, K.: Honey-block: edge assisted ensemble learning model for intrusion detection and prevention using defense mechanism in IoT. Comput. Commun. **214**, 1–17 (2023)
22. Bakhsh, S.A., Khan, M.A., Ahmed, F., Alshehri, M.S., Ali, H., Ahmad, J.: Enhancing IoT network security through deep learning-powered intrusion detection system. Internet. Things. **24**, 100936 (2023)
23. Sáez-de-Cámara, X., Flores, J.L., Arellano, C., Urbieta, A., Zurutuza, U.: Clustered federated learning architecture for network anomaly detection in large-scale heterogeneous IoT networks. Comput. Sec. **131**, 103299 (2023)
24. Singh, K.P., Kesswani, N.: An anomaly-based intrusion detection system for IoT networks using trust factor. SN Comput. Sci. **3**, 168 (2022)

25. Bacha, S., Aljuhani, A., Abdellafou, K.B., Taouali, O., Liouane, N., Alazab, M.: Anomaly-based intrusion detection system in IoT using kernel extreme learning machine. J. Ambient. Intell. Humaniz. Comput. **15**, 231–242 (2022)
26. Pan, J.-S., Zhang, L.-G., Wang, R.-B., Snasel, V., Chu, S.-C.: Gannet optimization algorithm: a new metaheuristic algorithm for solving engineering optimization problems. Math. Comput. Simul **202**, 343–373 (2022)
27. Singh, B., Joshi, M., Kumar, A., Senthilkumar, S.: Hybrid deep learning algorithm for heart disease analysis based on diabetes. Int. J. Intell. Syst. Appl. Eng. **12**(3s), 363–368 (2024)
28. Zhao, W., Gao, Y., Ji, T., Wan, X., Ye, F., Bai, G.: Deep temporal convolutional networks for short-term traffic flow forecasting. IEEE Access **7**, 114496–114507 (2019)
29. Wang, H., Liu, Z., Peng, D., Qin, Y.: Understanding and learning discriminant features based on multiattention 1DCNN for wheelset bearing fault diagnosis. IEEE Trans. Industr. Inf. **16**(9), 5735–5745 (2019)
30. Li, Y., Huang, H., Xie, Q., Yao, L., Chen, Q.: Research on a surface defect detection algorithm based on MobileNet-SSD. Appl. Sci. **8**(9), 1678 (2018)
31. Sadollah, A., Bahreininejad, A., Eskandar, H., Hamdi, M.: Mine blast algorithm: a new population-based algorithm for solving constrained engineering optimization problems. Appl. Soft Comput. **13**(5), 2592–2612 (2013)
32. Naik, M.K., Panda, R., Abraham, A.: Adaptive opposition slime mould algorithm. Soft. Comput. **25**(22), 14297–14313 (2021)
33. Chen, Z., Francis, A., Li, S., Liao, B., Xiao, D., Ha, T.T., Li, J., Ding, L., Cao, X.: Egret swarm optimization algorithm: an evolutionary computation approach for model-free optimization. Biomimetics **7**(4), 144 (2022)
34. Rajkumar, P.V., Sandhu, R.: Safety decidability for pre-authorisation usage control with identifier attribute domains. IEEE Trans. Dependable Secur. Comput. **17**(3), 465–478 (2020)
35. Rajkumar, P.V., Sandhu, R.: Safety decidability for pre-authorisation usage control with finite attribute domains. IEEE Trans. Dependable Secur. Comput. **13**(5), 582–590 (2016)
36. Raghavan, K., Desai, M., Rajkumar, P.V.: Multi-step operations strategic framework for ransomware protection. SAM Adv. Manag. J. **85**, 16 (2020)
37. Rajkumar, P.V., Raghavan, K., Desai, M.: Cyber Security and Hybrid Work Environments. SAM Adv. Manag. J. **88**, 44 (2023)
38. Huang, W., Deng, X.: GeoEkuiper: a cloud-cooperated geospatial edge stream processing engine for resource-constrained IoT devices with higher throughput. IEEE Internet Things J. **11**(18), 30094–30113 (2024)
39. Li, K., Wang, X., He, Q., Wang, J., Li, J., Zhan, S.: Computation offloading in resource-constrained multi-access edge computing. IEEE Trans. Mob. Comput. **23**(11), 10665–10677 (2024)
40. Lu, D., Zhai, Y., Shen, J., Fahmideh, M., Wu, J., Tchaye-Kondi, J.: TreeNet based fast task decomposition for resource-constrained edge intelligence. IEEE Trans. Serv. Comput. **16**(3), 2254–2266 (2023)
41. Idrees, S.K., Azar, J., Couturier, R., Idrees, A.K., Gechter, F.: SZ4IoT: an adaptive lightweight lossy compression algorithm for diverse IoT devices and data types. J. Supercomput. **81**, 392 (2025)
42. Otokwala, U., Petrovski, A., Kalutarage, H.: Optimized common features selection and deep-autoencoder (OCFSDA) for lightweight intrusion detection in internet of things. Int. J. Inf. Secur. **23**, 2559–2581 (2024)
43. Begum, M.B., Yogeshwaran, A., Nagarajan, N.R., Rajalakshmi, P.: Dynamic network security leveraging efficient CoviNet with granger causality-inspired graph neural networks for data compression in cloud IoT devices. Knowl.-Based Syst. **309**, 112859 (2025)
44. Mnif, M., Sahnoun, S., Saad, Y.B., Fakhfakh, A., Kanounm, O.: Combinative model compression approach for enhancing 1D CNN efficiency for EIT-based hand gesture recognition on IoT edge devices. Internet. Things. **28**, 101403 (2024)
45. Sarveshwaran, V., Pandiaraj, S., Bindu, G., Ganesan, V., Swamidason, I.T.J.: Binarized spiking neural network with blockchain-based intrusion detection framework for enhancing privacy and security in cloud computing environment. Appl. Soft Comput. **154**, 111218 (2024)
46. Amal, M.R., Venkadesh, P.: H-DOCTOR: honeypot-based firewall tuning for attack prevention. Measurement. Sens. **25**, 100664 (2023)
47. Ntizikira, E., Wang, L., Chen, J., Saleem, K.: Honey-block: edge assisted ensemble learning model for intrusion detection and prevention using defense mechanism in IoT. Comput. Commun. **214**, 1–17 (2024)
48. Ntizikira, E., Wang, L., Chen, J., Lu, X.: Attention-based ResNet for intrusion detection and severity analysis using sliding window blockchain and firewall in IoT. Clust. Comput. **27**(7), 10025–10051 (2024)
49. Khalaf, O.I., Ashokkumar, S.R., Algburi, S., Anupallavi, S., Selvaraj, D., Sharif, M.S., Elmedany, W.: Federated learning with hybrid differential privacy for secure and reliable cross-IoT platform knowledge sharing. Secur. Priv. **7**(3), e374 (2024)
50. Putra, M.A.P., Alief, R.N., Rachmawati, S.M., Sampedro, G.A., Kim, D.-S., Lee, J.-M.: Proof-of-authority-based secure and efficient aggregation with differential privacy for federated learning in industrial IoT. Internet. Things. **25**, 101107 (2024)
51. Wang, R., Lai, J., Li, X., He, D., Khan, M.K.: RPIFL: reliable and privacy-preserving federated learning for the internet of things. J. Netw. Comput. Appl. **221**, 103768 (2024)
52. Zhang, T., Xu, D., Hu, Y., Vijayakumar, P., Zhu, Y., Tolba, A.: Deep fingerprinting data learning based on federated differential privacy for resource-constrained intelligent IoT systems. IEEE Internet Things J. **11**(15), 25744–25756 (2024)

53. Yao, H., Luo, W., Lou, J., Yu, W., Zhang, X., Qiang, Z.: Scalable industrial visual anomaly detection with partial semantics aggregation vision transformer. IEEE Trans. Instrum. Measurement. **73**, 1–17 (2024)
54. Kwon, D., Yu, J.: RaViT-AE: unsupervised anomaly detection for intelligent cultural heritage monitoring using region-attentive ViT autoencoder. IEEE Access **12**, 180767–180780 (2024)
55. Yao, H., Luo, W., Yu, W., Zhang, X., Qiang, Z., Luo, D.: Dual-attention transformer and discriminative flow for industrial visual anomaly detection. IEEE Trans. Autom. Sci. Eng. **21**(4), 6126–6140 (2024)
56. Li, Y., Peng, X., Zhang, J., Li, Z., Wen, M.: DCT-GAN: dilated convolutional transformer-based GAN for time series anomaly detection. IEEE Trans. Knowl. Data Eng. **35**(4), 3632–3644 (2023)
57. Chen, Z., Duan, J., Kang, L., Qiu, G.: Supervised anomaly detection via conditional generative adversarial network and ensemble active learning. IEEE Trans. Pattern Anal. Mach. Intell. **45**(6), 7781–7798 (2023)
58. Cheng, W., Li, Y., Ma, T.: S-KDGAN: series-knowledge distillation with GANs for anomaly detection of sensor time-series data in smart IoT. IEEE Sens. J. **24**(15), 24344–24354 (2024)

Authors and Affiliations

WeiWei Hu[1] · Jafar A. Alzubi[2] · J. Shreyas[3] · Muna Al-Razgan[4] · Yasser A. Ali[5] · A. Karthikayan[6]

✉ J. Shreyas
shreyas.j@manipal.edu

[1] Henan Kaifeng College of Science Technology and Communication, Kaifeng 475000, Henan, China

[2] Faculty of Engineering, Al-Balqa Applied University, Salt 19117, Jordan

[3] Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India

[4] Department of Software Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia

[5] Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia

[6] Department of Electronics and Communication Engineering, Saveetha School of Engineering, SIMATS, Chennai 602105, Tamilnadu, India