Alexandria Engineering Journal 114 (2025) 312–330

Contents lists available at ScienceDirect

Alexandria Engineering Journal

journal homepage: www.elsevier.com/locate/aej

Original article

# Interrelated dynamic biased feature selection and classification model using enhanced gorilla troops optimizer for intrusion detection

Appalaraju Grandhi, Sunil Kumar Singh *

*School of Computer Science and Engineering, VIT-AP University, Near Vijayawada, Andhra Pradesh 522237, India*

ARTICLE INFO

*Keywords:*
Intrusion Detection System
Optimization
Feature Extraction
Feature Selection
Enhanced Gorilla Troops Optimization
Feature Vector
Classification

ABSTRACT

An Intrusion Detection System (IDS) is a valuable tool for network security since it can identify attacks, intrusions, and other types of illegal access. Excessive and irrelevant data slows down the classification process and eventually weakens the system's capacity to make informed decisions when IDS is monitoring a huge volume of network traffic. Innovative approaches are utilized to create large amounts of data and a lot of network traffic in order to test its effectiveness. A vital step in machine learning is feature selection. By determining which features are most essential for describing the dataset and its initial attributes, feature selection seeks to improve intrusion prediction performance while simultaneously delving deeper into the stored data. Making a feature selection is similar to fixing an optimization problem without a clear definition when users don't know where to begin. Finding the fewest characteristics required to describe the dataset, the original features, and to conduct classification is the primary purpose of feature selection, which also aims to improve prediction performance and acquire a deeper understanding of the stored data. As a result, researchers have been focusing on feature-selection issues recently, especially in light of the massive growth in available databases. Metaheuristic algorithms using a learning model have been the subject of studies to optimize feature selection difficulties. This research uses Enhanced Gorilla Troops Optimizer (EGTO), for enhancing the feature selection process and then performing classification. This research presents a Interrelated Dynamic Biased Feature Selection Model using Enhanced Gorilla Troops Optimizer (IDBFS-EGTO) for generation of feature vector set for intrusion detection. Despite its apparent success in handling a wide range of practical problems, it risks getting mired in local optima and premature convergence when faced with more difficult optimization challenges that is overcome with EGTO. The EGTO approach, which uses a collection of operators to strike a more steady equilibrium between exploitation and exploration. The proposed model generates relevant feature subset for machine learning model for accurate detection and classification of intrusions in the network. The proposed model achieved 98.4 % accuracy in intrusion detection and 98.6 % accuracy in EGTO optimization classification. The proposed model is improved by 3.8 % in feature weight allocation accuracy and 1.2 % in detection accuracy levels. The proposed model is compared with the traditional models and the results represent that the proposed model performance is high.

## 1. Introduction

Intrusion Detection Systems (IDSs) have been increasingly popular in recent years because of how reliable they are. Integral threat detection systems are designed to detect possible dangers within a certain area. A intrusion detection system (IDS) consists of a monitoring agent, an analysis engine that detects intrusion attempts and triggers alarms, and a return module that processes the analysis engine's output. Although intrusion detection systems (IDSs) have improved in reliability and effectiveness over the years, cybercriminals have found ever-more-intricate ways to circumvent them [2]. On top of that, conventional intrusion detection systems can't cope with the complexity of the multiple network layers. It has been suggested that distributed IDSs can make use of ML techniques such as Artificial Neural Networks (ANN), Reinforcement Learning (RL), and deep learning (DL) [3]. Even though AI has come a long way, traditional ANNs aren't yet up to the challenge of intrusion detection systems (IDSs). Therefore, it is necessary to build technologies to overcome those shortcomings in order to fully utilize IDSs in applications [5].

With the arrival of the Big Data era, the amount and variety of data at

* Corresponding author.
*E-mail addresses:* rajugrandhi.21phd7068@vitap.ac.in (A. Grandhi), sunil.singh@vitap.ac.in (S.K. Singh).

https://doi.org/10.1016/j.aej.2024.10.100
Received 8 November 2023; Received in revised form 7 September 2024; Accepted 25 October 2024
Available online 3 December 2024

our fingertips has grown exponentially. Problems with handling large volumes of high-dimensional data are becoming more prevalent in several areas, including ML, TM, and DA. When superfluous or irrelevant attributes are added to the dimension, making it more complicated and making proper categorization more difficult, the algorithm produces bad results [6]. In a similar vein, the IDS finds and handles illegal resource usage and implements network transmission controls using massive amounts of data [7]. Improving the qualities of IDS is vital, however, because maintaining high detection accuracy is presently a developing challenge that must be addressed. The IDS model is shown in Fig. 1.

Feature selection (FS) is becoming increasingly used as a means to reduce data dimensionality. Reducing data complexity by removing useless information is a key capability of IDS [8]. By eliminating extraneous details, FS approaches lower the dimensionality of the network data. We have reduced the computational overhead of IDS and speed up detection times [9]. Among the activities involved in preparing data for IDS, FS rates highly due to its effect on detection precision. The four separate steps that comprise the FS are the following: search, evaluation, case and investigation, results. The search phase lays the framework and provides the strategy [10]. Following processing of the initial feature set, the search module generates the matching feature subset. In order to evaluate the feature subsets, suitable evaluation criteria are created [11]. The feature subset that was provided is output when the FS procedure reaches its termination condition. At the same time, we are testing how well the feature-picking algorithm works. The accuracy and overall performance of the system were enhanced across different classifiers when FS approaches were incorporated into IDS.

Network security has become an extremely pressing issue due to the enormous volume of online applications and the rapid expansion of electronic communication methods, such as the Internet. For the purpose of protecting modern networks, numerous cybersecurity procedures and protection systems have been developed, including firewalls, authentication methods, cryptography techniques, and IDSs [12]. Internet security systems search data from networks for indications of malicious or suspected cyber activity. The administrator of the network is alerted by intrusion detection systems whenever they detect unusual activity. We will also take the necessary steps to deal with this incident and make sure it doesn't happen again. As their patterns are unknown, signature-based IDSs overlook entirely novel, zero-day threats, even though they excel at detecting known attacks. By contrast, anomaly-based intrusion detection systems distinguish between unknown attacks and pre-defined usual processes in an effort to discover zero-day attacks. Their performance is often surpassed by signature-based intrusion detection systems when it comes to identifying known threats.

Hybrid intrusion detection systems may detect both known and unknown threats because they combine signature-based and anomaly-based detection methods. Machine learning (ML) techniques have given rise to a new breed of IDSs. To automatically discover, evaluate, and extract patterns from data, ML uses mathematical equations [13]. Providing machine learning models with accurate data allows them to improve their prediction and judgment abilities. Deep Learning (DL) algorithms, K-Nearest Neighbors (KNN), and Decision Tree (DT) based models are all examples of supervised learning ML techniques that train with labeled data to convert input variables into a target variable. On the other hand, unsupervised learning algorithms do not rely on labels to guide their learning process. Unsupervised learning techniques include programs like GMM, k-means, and random forests. Patterns in unlabeled data can be discovered using these methods. Supervised learning algorithms are commonly used in signature-based IDSs to train on labelled network datasets, whereas unsupervised learning techniques are employed in anomaly-based IDSs to separate normal from abnormal data. In today's ever-changing networks, the identification of cyber-attacks is a major challenge for managers and operators of these systems. Ensemble learning, Hyper-Parameter Optimization (HPO), and Transfer Learning (TL) are a few of the advanced ML approaches that may be utilized to build intrusion detection systems (IDSs), which in turn improves the accuracy of intrusion detection and increases defenses against more attacks. The goal of ensemble learning techniques—which include voting, bagging, stacking, and others—is to enhance model learning performance by combining the results of numerous individual ML algorithms used as base models.

Filtering out unnecessary characteristics and selecting the best subset of features is what FS is all about when it comes to better defining patterns that belong to distinct classes. FS approaches are often categorized as either filters or wrappers based on whether they use learning algorithms for selection. For the purpose of determining the weight of feature connections, filter algorithms employ an external metric. Wrapper algorithms, on the other hand, use targeted learning approaches to assess a subset of features. Because the proposed subset of attributes is explicitly assessed using input obtained during learning, wrapper procedures outperform filter techniques in terms of classification accuracy [14]. Since their effectiveness is greatly dependent on learning, they are computationally more costly than filters but produce better results. The general process for feature selection and extraction is shown in Fig. 2.

Finding the almost ideal subset of attributes is another important

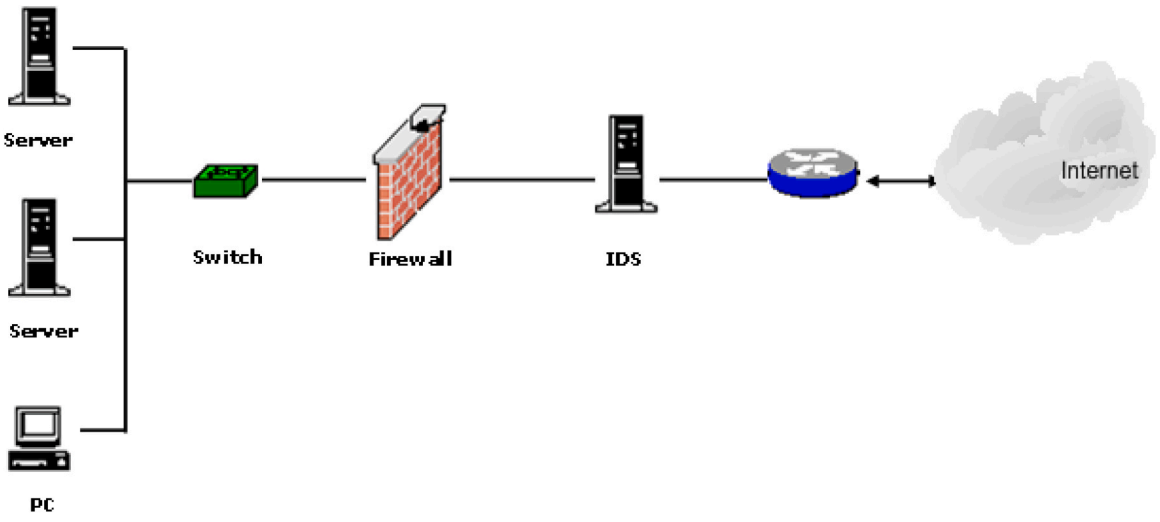


**Fig. 1.** IDS Model.

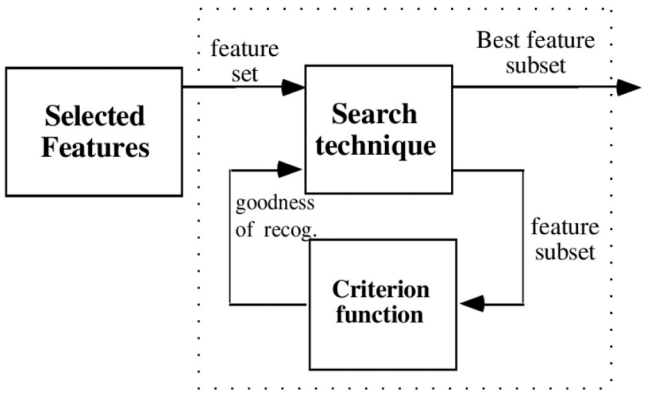


**Fig. 2.** Feature extraction and selection process.

issue when developing an FS approach [15]. Using conventional comprehensive methods such as width searches, depth inquiries, and others to find the optimal subset of attributes in huge datasets is a challenge [16]. Building and evaluating subsets of a dataset with N characteristics becomes computationally costly when using a wrapper-based approach, like a neural network. This is why FS has been labeled as an NP-hard optimization problem by a number of academics [17]. With its help, we can lessen the burden on features while simultaneously increasing classification accuracy [18]. A popular way to solve this challenge in feature selection is to formulate FS as either a multi-objective optimization problem and identify trade-off resolutions between the two competing aims [19] or as a single-objective optimization problem and combine both of these goals.

As a whole, intrusion detection systems (IDS) are conceptually designed to increase detection accuracy while decreasing false positive alerts [20]. Consequently, any intrusion detection system (IDS) must take that strategy into account when it is being developed and deployed. Recently, ML-based IDSs have emerged as the leading contenders in the intrusion detection research sector. Machine learning has the potential to help computers improve with use [21]. What this means is that ML-based software development does not necessitate human engineering. They are capable of learning independently. There are primarily two schools of thinking when it comes to machine learning: supervised and unsupervised. Training models on labelled data is an integral part of supervised ML. To train models, unsupervised machine learning makes use of unstructured data [22]. This research considers supervised methods, with an emphasis on tasks involving binary and multiclass classifications. Classification refers to the process of assigning a discrete value prediction job to a supervised model [23]. Models are trained in this system utilizing massive datasets that have rich feature dimensions [24]. Due to this dimensionality, supervised model training and testing can be quite time-consuming. The overall number of features required for training and testing can be decreased by the implementation of feature engineering procedures [25].

In classification, a model, or classifier, is learned from a training set of data examples that have labels applied to them. Then, a test instance is placed into one of the classes using the learned model. Methods for detecting anomalies based on classification follow a similar two-stage process. The training process involves creating a classifier with the help of the annotated training data. To find out if a test instance is normal or unusual, the classifier is utilized in the testing step. Classification is a subset of supervised machine learning that teaches computers to group data according to its original input structure. The act of sorting information into distinct groups is known as classification. Both structured and unstructured data can be processed with it. With the internet's meteoric rise, ensuring the safety of data transmissions is a top priority for every computer network infrastructure. There has been a steady increase in the frequency of network attacks. The term "intrusion" refers to direct assaults on computer networks. Protecting a system's data and network against harmful incursion attempts is the main goal of an intrusion detection system. In order to keep tabs on and analyze massive amounts of network data and to categorize this data as either normal or pathological, data mining techniques are employed. There are a lot of sources for data, which means that network traffic is high. Data mining methods like as clustering and classification are utilized in the construction of intrusion detection systems. This study offers a structure for organizing intrusion detection systems (IDS) and a range of data mining methods and datasets to help identify patterns linked to malicious and non-malicious network activities more efficiently.

The research presented here first suggested a machine learning model for feature extraction from datasets; then it suggested a novel FS model based on the naturally inspired Gorilla Troop Optimization (GTO) algorithm. Complex issues can be solved with minimum configuration using the GTO. Going into the unknown, socializing with other gorillas, and then returning to a familiar area are the three stages of exploration. And there are two parts to the exploitation process: luring the female gorillas and luring the silverbacks. Quickly discovering the ideal response is GTO's main advantage when compared to other models and more sophisticated algorithms. The general working process of GTO model is shown in Fig. 3.

This research proposes a enhanced GTO algorithm (EGTO) for accurate classification of intrusions in the network. Using the machine learning algorithm as a way to derive features from the input datasets, implementing the EGTO algorithm as a new FS method, assessing the performance of the suggested approach and comparing the results with current techniques on large datasets are the main contributions of this research. To strike a more consistent equilibrium between exploitation and exploration, the EGTO model makes use of a group of operators. As with every metaheuristic algorithm, the GTO method is susceptible to local optima and population variety while being one of the most efficient and fastest populations-based optimization algorithms. In order to

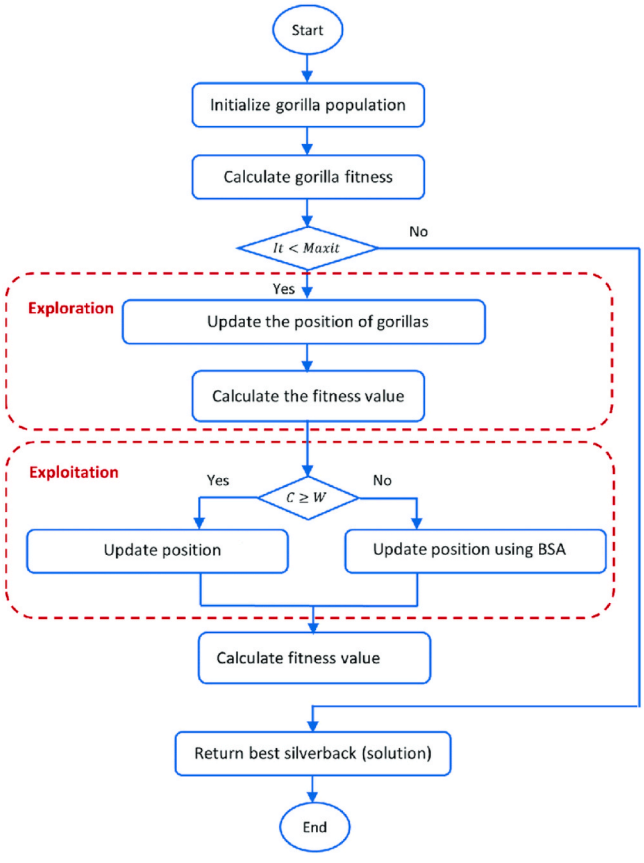


**Fig. 3.** General working process of GTO.

address the shortcomings of the original GTO method, this research proposes a new approach, EGTO for accurate feature selection. This research uses Enhanced Gorilla Troops Optimizer (EGTO), for enhancing the feature selection process. The results suggest that the EGTO algorithm outperforms the other optimization strategies in terms of convergence speed and its ability to avoid local optimum point trapping. It has also demonstrated a great deal of accuracy. But just like other swarm intelligence algorithms, the EGTO algorithm suffers from an imbalance between exploration and exploitation as a result of strong randomness in optimization. This imbalance causes issues like low convergence and the algorithm's propensity to easily fall into the optimal local solution. This research presents a Interrelated Dynamic Biased Feature Selection Model using Enhanced Gorilla Troops Optimizer for generation of feature vector set for intrusion detection.

## 2. Literature survey

As the Internet Of Things (IoT), cloud computing, and other forms of remote data storage have advanced, new security threats have emerged. Since the existing security procedures do not offer effective answers, cyber attacks have also been expanding swiftly alongside these improvements. Several AI-based solutions have been developed as of late for use in various intrusion detection and other security contexts. In this research, Fatani et al. [1] presented an effective Artificial Intelligence (AI)-based mechanism for IDS in IoT systems. The author took advantage of recent developments in deep learning and metaheuristics (MH) algorithms, which have been shown to be effective at resolving difficult engineering problems. The author presented a method for extracting useful features using Convolutional Neural Networks (CNNs). The author also created a novel feature selection approach by modifying the Transient Search Optimization (TSO) algorithm with operators from the DE algorithm to create TSODE. To achieve a better equilibrium between the exploitation and exploration phases, the suggested TSODE makes advantage of the DE.

With the exponential growth of cyber threats, experts, professionals, and specialists in the field of security have developed more reliable protection systems, such as efficient IDS mechanisms that are able to increase accurately detected threats while limiting erroneously detected threats at the same time. The success of a IDS architecture, however, hinges on the accuracy with which it extracts features from network traffic and uses those features to categorize that traffic as either abnormal or normal. The primary goal of this research is to improve IDS performance in network environments by developing a two-stage framework to strengthen and, ultimately, improve detection rate while reducing false alarm designed by Ghanem et al. [2]. The first step makes use of a multi-objective BAT algorithm (MOBBAT) based on a developed method for proficient wrapper-approach-based feature selection. The next phase makes use of the information gathered in the first to classify the traffic using the enhanced BAT algorithm (EBAT) for training multilayer perceptron (EBATMLP), which in turn enhances the performance of the IDS. The method that emerged from these considerations is called the (MOB-EBATMLP).

Malicious network attacks can be detected with the use of network intrusion detection systems (NIDSs). Many scientists have dedicated time and energy to creating NIDSs that use ML methods to identify different kinds of attacks. By analyzing the properties of a huge dataset, ML methods may automatically uncover the key distinctions between typical and atypical data. The computational cost of accomplishing this goal often rises due to the extraction of a large number of attributes without discriminating. Then, to boost the efficiency of ML-based detection approaches, a feature selection method is used to pick out a subset of features from the full feature set. The salp swarm algorithm (SSA) is an efficient optimization approach that has been shown to successfully reduce the processing difficulties typically associated with optimizing feature selection problems. Alsaleh et al. [3] examined the effect of the SSA on the performance of several ML classifiers, including the extreme gradient boosting (XGBoost) and Naive Bayes (NB) algorithms, in the context of enhancing ML-based network anomaly detection.

Cyber-physical systems (CPS) integrate digital and physical components for smart and efficient app management. However, new security concerns are being presented by the growing interconnection and complexity of CPS, making intrusion detection a crucial part of preserving the integrity and dependability of these systems. The increasing sophistication of AI methods is useful in solving the security issues that plague CPS settings. As a result, Almutairi et al. [4] suggested a Quantum Dwarf Mongoose Optimization with Ensemble Deep Learning Based Intrusion Detection (QDMO-EDLID) approach in a CPS setting. The purpose of the given QDMO-EDLID method is to identify intrusions through the use of FS and ensemble learning. In the QDMO-EDLID method, the QMO algorithm is used to select feature subsets. Also, for the classification of intrusions, an ensemble of Convolution Residual Networks (CRN), Deep Belief Networks (DBN), and Deep Autoencoder (DAE) models is used.

Tao et al. [5] proposed a hybrid strategy improved sparrow search algorithm (HSISSA) for use in intrusion detection's feature selection and model optimization to address the issue of Sparrow Search Algorithm (SSA) getting stuck in local optima and having a slow convergence speed. The spiral search method in the vulture search algorithm is combined with Levy's flight formula to update the positions of the discoverer and scouter, respectively, to increase the search range and search capability; and the simplex method and pinhole imaging method are used to optimize the position of the population. Using these techniques, the algorithm's performance was improved. For verification, the algorithm's improved convergence speed and accuracy were tested on ten standard benchmark functions and integrated with Wilcoxon rank-sum test analysis. Finally, it was used for intrusion detection's feature selection and model optimization. Accuracy levels of 99.5 % and 96.01 % were attained on the CIC-IDS2017 and UNSW-NB15 datasets, respectively, with an average retention rate of 7.6 and 10.1 features, respectively.

Today, cloud computing is focused on transforming the digital era, and as a result, customers have legitimate worries about the safety of their cloud-stored data in the face of sophisticated and regular cyber-attacks. Therefore, it is crucial for both individuals and businesses to have an effective IDS that can monitor network packets, identify malicious behavior, and identify attack types. IDS systems that use machine learning to identify threats to a network perform admirably. However, the efficiency of such systems degrades when dealing with data sets with a high number of dimensions. In order to avoid affecting the categorization process or losing information, it is crucial to develop an appropriate feature selection technique. Additionally, the false positive rate (FPR) increases and the detection rate (DR) decreases when training ML models on imbalanced datasets. In this paper, Bakro et al. [6] proposed a hybrid approach for feature selection that combines three techniques: information gain (IG), chi-square (CS), and particle swarm optimization (PSO), and the author presented a cloud IDS that incorporates the SMOTE to address the imbalanced data issue. Accuracy in excess of 98 % and 99 % were achieved in the multi-class classification scenario while testing the proposed system on the UNSW-NB15 and Kyoto datasets, respectively. Overview of research gaps and limitations are in Table 1.

Unpredictable faults on the feeders are a common issue for microgrids (MGs) due to a wide variety of causes. These flaws have the potential to disrupt the MG's steady operation and cause harm to its parts. Fault kinds, locations, resistances, MG modes, DG penetration, load changes, and system topologies are only some of the uncertainties that might affect the MG's response to faults. As a result, MGs rely heavily on fault detection, classification, and placement to facilitate speedy restoration and safeguard the components. For the purpose of the future renewable electric energy delivery and management (FREEDM) system, Hatata et al. [7] suggested an adaptive protection (AP) strategy. Convolutional neural networks (CNNs) form the basis of the proposed

**Table 1**
Research gaps and limitations.

| Author Name | Year of Publication | Proposed Model | Research Gaps/ Limitations |
|---|---|---|---|
| Fatani et al. [1] | 2021 | The author presented an effective Artificial Intelligence (AI)-based mechanism for IDS in IoT systems. The author took advantage of recent developments in deep learning and metaheuristics (MH) algorithms, which have been shown to be effective at resolving difficult engineering problems. | The Enhanced Transient Search Optimization proposed experiences local optima that need to be resolved. The optimization accuracy is impacted with the initial values considered. The computational complexity levels also need to be reduced using a suitable optimization model. |
| W. A. H. M. Ghanem et al. [2] | 2022 | The primary goal of this research is to improve IDS performance in network environments by developing a two-stage framework to strengthen and, ultimately, improve detection rate while reducing false alarm designed. | The false alarm rate is still in normal level that need to be considered. The iterations considered are high that increases the time complexity levels. |
| A.Alsaleh et al. [3] | 2021 | The author examined the effect of the SSA on the performance of several ML classifiers, including the extreme gradient boosting (XGBoost) and Naive Bayes (NB) algorithms, in the context of enhancing ML-based network anomaly detection | Because of its complexity, XGBoost is not always easy to understand. Because it has so many hyperparameters, XGBoost's training time can be somewhat long. Overfitting is a potential issue with XGBoost if it is not calibrated correctly. XGBoost is not a good fit for older or less powerful computers due to its memory requirements. When faced with the "zero-frequency problem," the Naive Bayes Algorithm struggles. |
| L. Almutairi et al. [4] | 2023 | The author suggested a Quantum Dwarf Mongoose Optimization with Ensemble Deep Learning Based Intrusion Detection (QDMO-EDLID) approach in a CPS setting. The purpose of the given QDMO-EDLID method is to identify intrusions through the use of FS and ensemble learning. In the QDMO-EDLID method, the QMO algorithm is used to select feature subsets. | The pattern changes and similarity levels are not considered in identification of relevant features for intrusion detection. More features are considered that need to be reduced. |
| L. Tao et al. [5] | 2023 | The author proposed a hybrid strategy improved sparrow search algorithm (HSISSA) for use in intrusion detection's feature selection and model optimization to address the issue of Sparrow Search Algorithm (SSA) getting stuck in local optima and having a slow convergence speed. | The results are improved when compared to other intelligent algorithms, but the method still suffers from issues including poor convergence speed, inadequate solution precision, and the tendency to slip into local optimums. |
| M. Bakro et al. [6] | 2023 | The author proposed a hybrid approach for feature selection that combines three techniques: information gain (IG), chi-square (CS), and particle swarm optimization (PSO), and the author presented a cloud IDS that incorporates the SMOTE to address the imbalanced data issue. | Feature dimensionality need to be performed and ranking model can be selected for best feature set generation. |

approach, which uses multidimensional array processing of the observed current and voltage at buses to perform identification and classification of images. The suggested CNN has been optimized by the gorilla troops optimization (GTO) method, which has been utilized to acquire the optimal architecture and hyperparameters. Three CNN-GTO protection scheme models are suggested and used to detect system faults, categorize fault types, and pinpoint fault locations. Data, information, and tripping signals are transferred between the various devices in the FREEDM system via a communication channel. The suggested technique is put through its paces by simulating various faults in a FREEDM microgrid system.

There has been a dramatic increase in cyber-attacks due to the development and refinement of hacking techniques, stressing the need for more robust methods of network protection. Machine learning-based network intrusion detection systems are becoming increasingly important in the field of network security. However, there remains persistent worry about how to best structure a network intrusion detection system. In this research, Siddiqi et al. [8] offered a top-notch architecture for an image processing-based NIDS. The framework combines an improved feature selection flow with an image enhancement and modification technique. In order to maximize efficiency, the suggested framework first minimizes the amount of available features. The information that isn't an image gets converted into pictures later. A deep-learning classifier is then used to improve the modified photos for efficient anomaly identification. Three different intrusion detection benchmark datasets are used to test the proposed approach. Recent works on image-processing based network intrusion detection systems are contrasted with the proposed framework to demonstrate its efficacy.

## 3. Proposed model

Due to the widespread usage of computer networks, data security has become an emerging issue in many fields. Network security is crucial due to the proliferation of online communication and the ease with which intruders can access networks using various technologies. The data in databases is not adequately protected by the current security policies [26]. There are a variety of security solutions, such as firewalls, encryption, and authorization processes, yet they are all vulnerable to assaults because of vulnerabilities in the underlying systems. In this research, we design and build a new feature selection model using EGTO model for designing novel intrusion detection system [27]. Internet use has rapidly grown commonplace in recent years. Internets based on

information processing systems nowadays are vulnerable to a wide range of threats that can cause expensive damage in a number of ways [28]. As a result, the significance of information security is rapidly changing. The most fundamental objective of network security is the creation of defensive networking systems that are immune to disclosure, disruption, alteration, and destruction by malicious actors. Network security also reduces threats to the three primary tenets of information security: privacy, integrity, and availability [29].

Improving performance, alleviating the curse of dimensionality, fostering generalization abilities, speeding up by depreciating computational power, growing model strength, and saving money by avoiding expensive features are just some of the many applications of dimensionality reduction [30]. To address the high dimensionality issue and eliminate superfluous or irrelevant features, feature selection is typically used. Feature selection has many benefits, including a shorter training period, improved generalization performance, higher quality classification results, and a highly interpretable model. There is no guarantee of improved results even when the best classifiers are utilized for the classification process [31]. Therefore, the likelihood of success increases when strong feature selectors are used in tandem with competent classifiers. Unlike feature extraction, which creates new features by merging the existing features, feature selection focuses on a subset of those features. While Feature Selection is in charge of keeping working prototypes of the chosen features [32], which are applicable to various industries. Pre-Processing approaches, applied to the normalized training dataset, typically result in a smaller feature set being selected.

The typical steps in a feature selection approach are creating the subset, assessing the subset, determining when to stop, and verifying the results. The first step of the method is subset construction, which employs a careful exploration tactic to discover potential subsets of features to analyze. A precise estimate approach is also used to compare each potential subset to the best ones that came before. If it is more powerful than the others, it will replace the one that came before it. This process of establishing and evaluating subsets is repeated, and it is terminated when a predetermined threshold is reached. One heuristic method for weighing the benefits of a feature subset is the correlation-based method. According to this line of thinking, a feature may only be considered exceptional if it has a strong relationship with the class as a whole, rather than with the other features individually. Therefore, it is necessary to establish a connection between aspects, which stand in for crucial and highly efficient qualities.

When developing a machine learning model, feature selection is crucial because it increases the model's effectiveness. The features must be scaled before any feature selection process is carried out so that their potentially huge values do not adversely affect the final output. After being scaled, features have a mean of zero and a standard deviation of one after being averaged. Feature selection is the process of narrowing down a dataset to its essential components in preparation for model building. This reduces the amount of time needed to calculate results and also helps prevent models from being overfit.

Refining a dataset for use in machine learning model training is called feature engineering. To guarantee that the generated machine learning model will meet its business use case, data scientists expertly modify the training data by adding, removing, merging, or altering variables within the data set. To ensure that the input data set is optimal for the business objective for which the machine learning algorithm was designed, data scientists employ feature engineering. One technique is to deal with intrusions. Predictions can be thrown off by intrusions because they are so far beyond the norm. Trimming is a frequent method for dealing with extreme values. Outlier values can be filtered away with a simple trim operation, protecting the integrity of the training set.

When it comes to feature engineering, feature extraction is a subset. When data in its raw form is useless, data scientists resort to feature extraction. Extracting features from raw data prepares it for use in machine learning techniques. Text documents are a frequent source of raw data, and by identifying properties, data scientists can develop new features that are well-suited for machine learning. The process of selecting features is similar. Feature selection is the process of determining which features will improve the quality of the prediction variable or output, as opposed to feature extraction and feature engineering, which both require producing new features. Feature selection helps build easier-to-understand machine learning models by weeding out irrelevant data.

When it comes to feature selection, a high-dimensional dataset is like a treasure trove of potentially misleading, incorrect, or redundant features. Not only that, but it also increases the search space dimension and tries to get the dataset ready for learning. Consequently, we must extract the feature subdivision from the initial dataset. Finding the best characteristics from the initial set is the job of Feature Selection (FS) techniques. In order to pick the features, feature selection algorithms use computational principles. There are two stages to the feature selection algorithm, and they are as follows: (i) Generating Subsets:(ii) In Subset Evaluation, we must determine if the subset that was formed from the input dataset is optimal. This process is known as subset generation. There are three possible approaches to these operations: filtering, wrapping, and embedding. The usual criterion for feature gathering by arrangement is the utilization of filter methods and feature rank. From the original feature dataset, it extracts more correlation by selecting just relevant features.

The classification technique is used to build a subset of features using wrapper approaches. When using a wrapper method, think of the interpreter as an opaque entity. It maximizes subset evaluations using searching methods and objective functions. There are two main categories of wrapper methods: i) heuristic search algorithms and ii) sequential selection algorithms. Starting with an empty set, the SSA adds features from the dataset until it reaches the desired output. A criterion was chosen to permeate the decision process; this criterion increases the goal function gradually until the maximum is obtained with the fewest features.

Classification refers to the act of organizing data into distinct groups, such as normal and abnormal, where normal refers to conformity to a known pattern and abnormal to its departure. It is primarily useful for detecting abuse, although it can also be used to spot anomalies. The data sets were classified into predetermined categories. As a kind of intrusion detection, it pales in comparison to clustering. This procedure can be categorized as either a network intrusion detection system (NIDS) or a host-based intrusion detection system (HIDS), depending on the location of the intrusion. The NIDS analyzes network data and combines the results with those of other technologies to increase the speed with which threats can be identified and forecast. Particularly, using artificial neural networks, intrusion detection systems can better evaluate enormous data sets because of increased pattern recognition for intrusions. Meanwhile, the HIDS monitors the device's critical system files and packet traffic in both directions, sounding alarms if anything out of the ordinary is found.

Classification methods can be broken down into two major groups: those that rely on signature features and those that use anomaly features. Signature-based methods scour networks for predetermined byte sequences or malware for predefined sequences of known hazardous instructions. While anomaly-based techniques excel at identifying previously unknown attacks, they fall short when faced with previously unknown attacks. Because malicious code evolves so quickly, anomaly-based techniques are essential for classifying new attacks. Machine learning is used to construct a reliable model, which is then tested for its efficacy. This method has the potential for both identifying previously undetected assaults and producing false positives. Use an efficient feature selection method to raise the quality of your classifications.

The gorilla swarm behavior inspired by GTO, which models five alternative approaches. Some of these strategies include: travelling to an unknown location, migrating for other gorillas, migrating towards a recognized region, continually tracking the silverback, and then competing for mature females. They are used as models to demonstrate

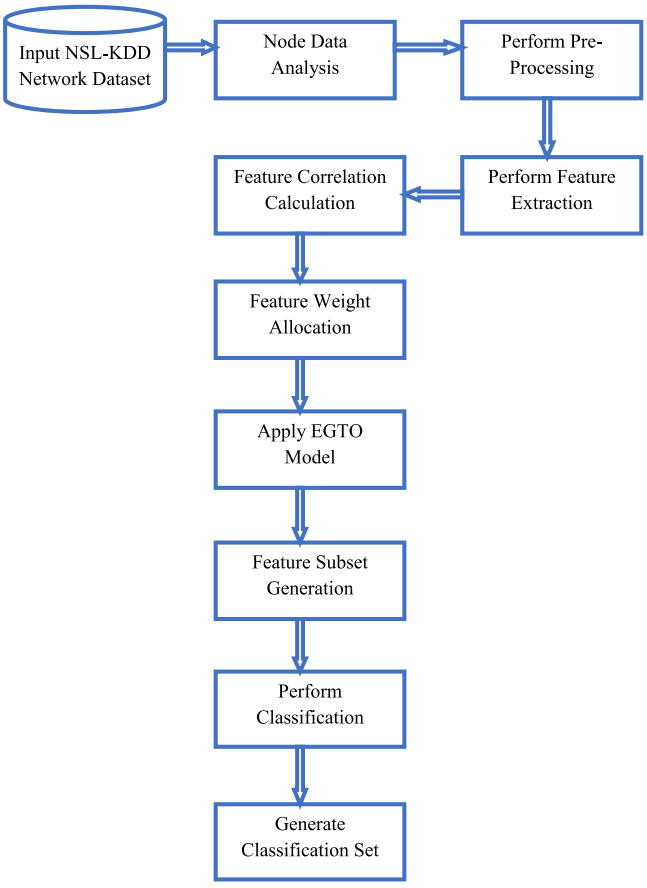


**Fig. 4.** Proposed model framework.

the exploration and exploitation stages of the optimization process. In the exploratory phase, one can take one of three paths: toward an unknown place, toward the remaining members, or toward a recognized region. There are two strategies utilized during exploitation: first, following the silverback, and subsequently, competing for adult females. The proposed model framework is shown in Fig. 4.

It is quite probable that the proposed framework depicting the steps involved in doing a classification task with the NSL-KDD network dataset within an NIDS. A detailed description of the process is provided here:

At the initial stage, researchers typically incorporate the widely-used NSL-KDD dataset, a resource for studying network intrusion detection into the system. In order to derive useful insights from the data, the next step is to do granular analysis, or look at the data at the level of individual nodes or data points. In order to get the data ready for analysis, pre-processing involves cleaning it up in various ways, such dealing with missing values, standardizing it, and maybe even encoding category features. The first stage is to extract features from the dataset. These features will be utilized to construct a set of important variables that will be used in the next steps. The goal of this procedure is to find the categorization features that are the most useful.

Finding Duplicate or Highly Correlated Features involves calculating the correlation between various features to get insight into their relationships with one another. In Feature Weight Allocation stage, characteristics are given weights according to their significance or relevance, which can be determined by their correlation or other statistical metrics. Optimize the feature selection process using the EGTO model. EGTO probably stands for a particular model or algorithm. This may necessitate a task-specific heuristic or evolutionary strategy. The EGTO model is used to generate a subset of features that are most suited for the classification task based on the data. This improves the model's performance by reducing the dimensionality.

In order to classify the data, a classification model is trained using the chosen subset of features. This model is usually used in intrusion detection to distinguish between normal and attack behaviors. The end outcome is a collection of findings that have been classed, most likely showing which data points have been labeled as normal and which may indicate possible intrusions or anomalies. The significance of data preparation, feature analysis, and model application in obtaining accurate classification outcomes is highlighted in this architecture, which overall demonstrates a methodical approach to feature selection and classification using the NSL-KDD dataset.

The most significant drawback of GTO is its propensity toward local optimization, a problem shared by many optimization methods. Scholars have thus addressed the GTO's faults and made improvements to it. At each stage of the optimization process, the GTO might designate the optimal option as the silverback gorilla. Moving toward an unknown region with the intention of increasing GTO exploration ability, reducing the search area by troubling the spaces between gorillas to strike a balance between exploration and extraction, and finally moving toward a known region with the intention of increasing GTO inspection of the region, were all employed during the exploration stage. Although it is one of the most effective and fast metaheuristic algorithms, the GTO approach is still vulnerable to local optima and population diversity. This research presents a novel method, EGTO, for precise feature selection to overcome the drawbacks of the traditional GTO method.

Everyone knows that the silverback gorilla relies on group communication for vital decisions. In order to improve the algorithm's capability for exploration, a contraction control factor fusion technique that mimics this connection. To avoid the original algorithm's failure to thoroughly explore and develop the local space of the optimal solution a problem related to partial investigation of the entire space we shifted one solution to the place of another optimal solution. We also enabled the algorithm to jump out of the local optimal solution and find a more reasonable and effective one by using randomization to transfer a solution to the exploration space that the algorithm does not reach. So, to make the answer better, we mimicked the gorilla's random moving method. U represents the amount of experience that the gorillas currently have. We include a contraction factor CAN to allow gorillas to explore more unfamiliar places when they are inexperienced, that is, when $U > 1$. When venturing into uncharted territory, there are two main approaches. If the absolute value of CAN is greater than or equal to 0.5, then the way gorillas investigate unfamiliar areas is accurately replicated based on their cognitive abilities. To further lessen the blindness of discovery, gorillas engage in a second kind of exploration involving unknown positions, which occurs after they share their experiences with one another. The parameter CAN determines the gorillas' preference between the two approaches, which broadens the algorithm's search space and the gorillas' ability to explore uncharted territories. Even with enough life experience, gorillas still need to socialize with one another to avoid being blinded by a lack of information and to make sure their knowledge pool is complete, accurate, and consistent. The fact that EGTO converges on multimodal functions suggests that its hybrid operation and mechanism for high and low velocity ratios play a significant role in escaping local optima and improving the algorithm's overall performance and durability. This research presents a Interrelated Dynamic Biased Feature Selection Model using Enhanced Gorilla Troops Optimizer for generation of feature vector set for intrusion detection.

**Algorithm.** IDBFS-EGTO

{

**Input:** Network Dataset {NDset}

**Output:** Classification Set {$Class_{set}$}

**Step-1:** The dataset records are considered and analyzed for calculating the maximum and minimum range values in the network dataset. The record analysis helps in considering each attribute values and to perform analysis that is done as

$$RecordProcess[L] = \sum_{r=1}^{L} \frac{getattr(r,r+1)}{L} + maxrange\big(attr(r \in L)\big) + minrange\big(attr(r \in L)\big) + \sum_{r=1}^{L} \sum attr(r, r+1) \quad (1)$$

Here getattr(r,r+1) model is used to extract the records from the dataset with L records. maxrange() and minrange() is used to calculate the maximum and minimum values in the record set.

**Step-2:** The goal of data preparation is to get the raw network data ready for use in an automated learning model. Therefore, the most important part of developing a machine-learning model is the preprocessing of data. As data is pooled from disparate sources utilizing mining and warehousing methods, it often contains errors, noise, partial information, and missing values. Because of them, machine learning relies on preparing data that need to be clean. The pre-processing is applied as

$$Rprocess[L] = \prod_{r=1}^{L} \frac{RecordProcess(r)}{\max\,(attr(r,r+1))} + \sum_{r=1}^{L} \max\big(diff(r+1, L-1)\big) + \sum_{r=1}^{L} \lim_{r \to L} \left(maxrange\big(attr(r \in L)\big) + \frac{minrange\big(attr(r \in L)\big)}{\mu(r)}\right)^{n} \quad (2)$$

$$Normalization[L] = \prod_{r=1}^{L} \frac{getattr\big(RProcess(\mathrm{r})\big) - \mathrm{minrange(r)}}{maxrange(r) - minrange(r)} (newmaxrange(r) - newminrange(r) + \sum_{r=1}^{L} \frac{\sum_{i=r}^{L} attr(i) + minrange(r)}{len(Rprocess)} \quad (3)$$

Here diff() model is used to identify the dissimilarity levels in the records for removing the duplicates and new range values are generated. μ considers the unwanted symbols records in the dataset.

**Step-3:** The goal of feature extraction is to improve the classifier's performance by identifying the smallest, most informative set of features for intrusion detection. In order to achieve trustworthy classification, feature extraction is used to extract features from the initial signal. The effectiveness and efficiency of the proposed machine learning model can be greatly enhanced by the use of feature extraction. The feature extraction process is performed as

$$FeatExtr[L] = \sum_{r=1}^{L} \frac{Max(Normalization(r,r+1))}{L} + \prod_{r=1}^{L} \max_{r \le x \le L} newmaxrange(r) * Th + \sum_{r=1}^{L} \omega(getattr(\frac{maxrange(r)}{newmaxrange(r)}) - \min\,(Rprocess(r))) \quad (4)$$

Here ω is the model for extracting the features in the maximum and minimum range attributes. The features with minimum range attributes are excluded from the set.

(*continued on next page*)

(*continued*)

**Step-4:** The level of linear relationships between sets of data can be evaluated with the help of correlation coefficients. Use Pearson's correlation coefficient if the two variables have similar normal distributions, one feature can be removed from the feature set. In machine learning, correlation is frequently used in the context of data analysis and mining. It can identify the most problematic features in a dataset and eliminate them before they affect the model fitting process. The correlation calculation is performed as

$$FeatCorrSet[L] = \sum_{r=1}^{L} \frac{\sum(attr(r)-\overline{attr(r)})*(attr(r+1)-\overline{attr(r+1)})}{\sqrt{\sum((attr(r)-\overline{attr(r)})^2}*\sqrt{\sum((attr(r+1)-\overline{attr(r+1)})^2}} + \sqrt{\frac{maxrange(r)}{minrange(r)}} \quad (5)$$

**Step-5:** The correlations of the features are considered and the highest correlation features are removed and the features that are independent are considered and weight allocation is considered to the feature set. The weight allocation is considered as

$$Walloc[L] = \sum_{l=1}^{L} \frac{\max(FeatCorrSet(r,r+1))}{len(FeatCorrSet)} + \max(attr(r)) \begin{cases} Walloc = k++ \ if \ (FeatCorrSet(r)) > Th \\ 0 \qquad\qquad\qquad\qquad Otherwise \end{cases} \quad (6)$$

**Step-6:** Inspired by the ingenuity of wild gorilla groups as a whole, the GTO is a cutting-edge metaheuristic algorithm. Despite its apparent success in handling a wide range of practical problems, it may nevertheless struggle to get beyond local optima and prematurely converge when faced with more difficult optimization challenges. The optimization structures of the GTO are typically changed by embedding various components from other optimization methods due to the complexity and roughness of search regions for real-world optimization problems. Injecting the adaptable mutation mechanism, which boosts exploration, and the Gorilla memory-saving approach, which boosts exploitation are two examples of how the GTO is improved. The EGTO optimization is applied on the weighted feature set as

**Input:** Iteration I, Size of population $P_s$, Parameters δ and τ for position updating

**Output:** Best Fitness Value $Fit_{val}$ and Location L

Perform random population size initialization $P_s$={s=1,2,….,M}

Allocate a initial attribute parameter range for δ and τ

While (Stop_Cond_Check == True) do

$$\delta[P_s] = \sum_{p=1}^{M} \max\left(Size(P_s, P_{s+1})\right) + \delta(P_s) \quad (7)$$

$$L[P_s] = \sum_{p=1}^{M} \min\left(\tau(P_s, P_{s-1})\right) - \delta(P_{s-1}) \quad (8)$$

$$for\ each\ P_s\ from\ s \in M$$

do

$$\tau[P_s] = \lim_{p \to M} \left(\max(range(L, L-1)) + \frac{\delta(P_{s+1}-P_s)}{M}\right)^2 \quad (9)$$

$$Fit_{val[M]=\sum_{p=1}^{M} \frac{\max(L(\delta(P_{S+1}\delta(P_{S-1})}{\tau_{L-1}+\tau_L}} \quad (10)$$

end for

(*continued on next page*)

(*continued*)

$if\ (Fit_{val[M] \geq Th})$

$$for\ each\ P_s\ from\ s \in M$$

$$if\ (\delta[P_s]) < \delta[P_{s+1}]$$

$$Fit_{val[M]=\sum_{p=1}^{M} \frac{\max\ (L(\delta(P_{S+1}\delta(P_S)}{\tau_L}} \tag{11}$$

End for

End if

End while

Return

$Fit_{val[M]}\ and\ L_s$

**Step-7:** Feature selection is to lower the dimensionality of their data. IDS relies heavily on its ability to reduce data complexity by excluding irrelevant information. By filtering out unnecessary information, FS methods reduce the network data's dimensionality. The computational burden of IDS has been decreased, and detection times have been sped up. Because of its impact on detection precision, FS ranks high among the tasks involved in data preparation for IDS. The feature subset is generated as

$$FeatSet[L] = \sum_{i=1}^{L} \frac{maxrange(Fit_{val[M]}\ (P_s,P_{s+1})}{\tau[P_s]} + \sum_{i=1}^{L} \frac{\delta(P_{s+1}-P_s)}{M} + maxrange(\delta[P_s], \delta[P_{s-1})] + \max\big(Walloc(i,i+1)\big) - \min\ (Walloc(i-1,i)) \tag{12}$$

**Step-8:** The purpose of an intrusion detection system is to safeguard a system's data and network from malicious intrusion attempts. Massive amounts of network data are monitored and analyzed using the proposed model, and the results are then sorted into "anomalous" and "normal" categories. Network traffic is heavy because information originates from so many different places. In order to create an intrusion detection system with better classification, machine learning technique like feature set classification is applied. The process of attack classification is performed as

$$Attkclass[L] = \prod_{i=1}^{L} \frac{\max\ (FeatSel(i,i+1))}{len(FeatSet)} + maxrange(\delta[P_s]) + minrange(\delta[P_{s+1}]) + \frac{\max\ (FeatSet(Walloc(\tau[P_s])))}{\max\ (Fit_{val[M]})} \tag{13}$$

$$Attkclass[L] = \begin{cases} Attkclass(i) \leftarrow attack\ \ if(\delta[P_s]) > Th\ \ and\ \ \delta[P_{s+1}] < Th + \max\ (Walloc(i)) \\ Attkclass(i) \leftarrow normal \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad Otherwise \end{cases} \tag{14}$$

**Step-9:** Classifying data is a common application of machine learning methods. Machine learning techniques are utilized to classify normal and abnormal data from big datasets, as demonstrated by numerous existing studies of the IDS. The proposed work enhances the efficiency of a classifier that detects and categorizes aberrant associations with a high degree of precision and a comparatively low rate of false positives. The proposed model attack classification set is generated as

$$Class_{set}[L] = \sum_{i=1}^{L} simm(Attkclass(\delta[P_s], \delta[P_{s+1}]) + \lim_{i \to L}(maxrange(Attkclass(i)) + \frac{minrange(Attkclass(i+1)}{\max\ (Fit_{val[M]})}) \begin{cases} Class \leftarrow 1\ \ if\ Attkclass\big(FeatSel(i,i+1)\big) == attack \\ Class \leftarrow 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad Otherwise \end{cases} \tag{15}$$

}

**Table 2**
KDD Cup'99 dataset description.

| Dataset | Class | Total Samples | Sample Percent (%) |
|---|---|---|---|
| | Normal | 97277 | 19.6 |
| | Probe | 4107 | 0.8 |
| KDD Cup'99 | DoS | 391458 | 79.2 |
| | U2R | 52 | 0.01 |
| | R2L | 1126 | 0.2 |
| **Total** | | **494021** | **100 %** |

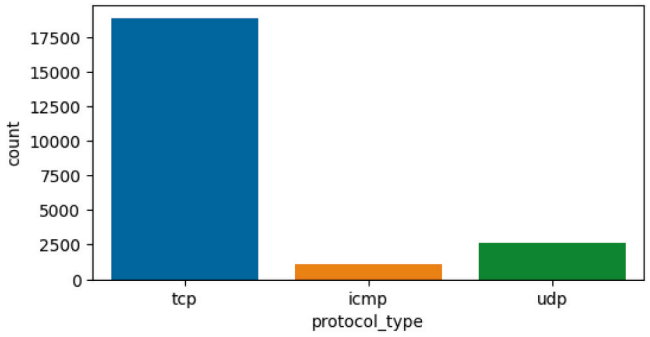


**Fig. 5.** Protocols used.

**Table 3**
Dataset record analysis time levels in milliseconds.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 7.2 | 13.5 | 17 |
| 40000 | 7.3 | 14 | 17.3 |
| 60000 | 7.5 | 14.5 | 17.6 |
| 80000 | 7.6 | 15 | 18 |
| 100000 | 7.8 | 15.5 | 18.3 |
| 120000 | 8 | 16 | 18.5 |

## 4. Dataset description

Out of a total of 494021 records, 97278 are considered normal in the 10 % KDD Labeled Training Dataset, which is a subset of KDD Cup'99. Among the 31,1029 records included in the KDD Cup'99 dataset, 60,593 were deemed normal; this subset, known as the Corrected KDD, underwent additional modifications to remove redundant traffic data packets. There are 5 million records in this dataset, which includes both labelled training and test data as well as entire data that has not been labelled. There are 41 characteristics in every piece of traffic data. The types of attacks considered are discussed. The Table 2 shows the features of the considered dataset.

1. A Denial of Service (DoS) attack occurs when an attacker or attackers overwhelm a target computer with unnecessary requests, preventing it from processing valid requests from users. Some examples include apache2, back, land, mailbomb, smurf, syslogd, teardrop, udpstorm, and SYN Flood.
2. A User to Root Attack (U2R) occurs when an attacker gains root access by first gaining access to the system through a normal user

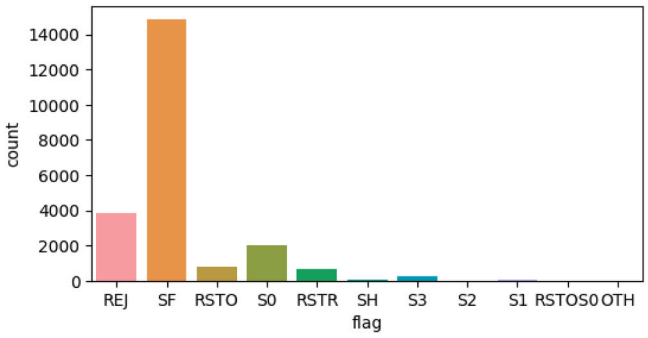


**Fig. 7.** Proposed model flags.

**Table 4**
Dataset pre-processing accuracy levels.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 97.7 | 93.6 | 91 |
| 40000 | 97.9 | 93.8 | 91.2 |
| 60000 | 98 | 94 | 91.4 |
| 80000 | 98.2 | 94.2 | 91.6 |
| 100000 | 98.4 | 94.7 | 91.8 |
| 120000 | 98.6 | 95 | 92 |

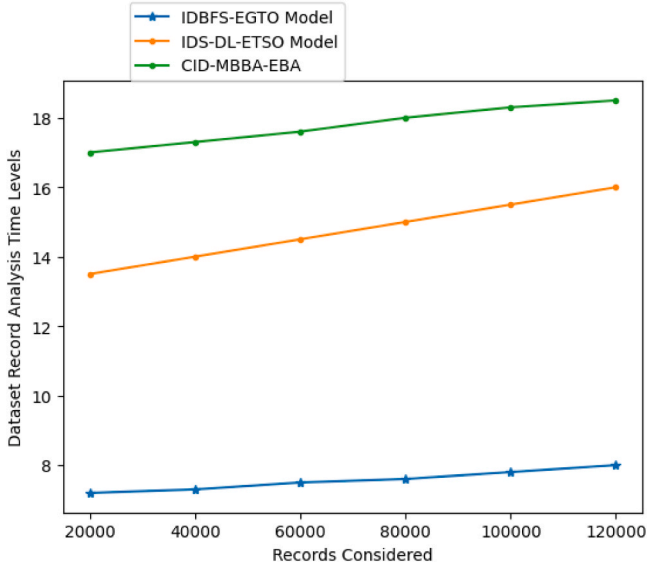


**Fig. 6.** Dataset record analysis time levels in milliseconds.

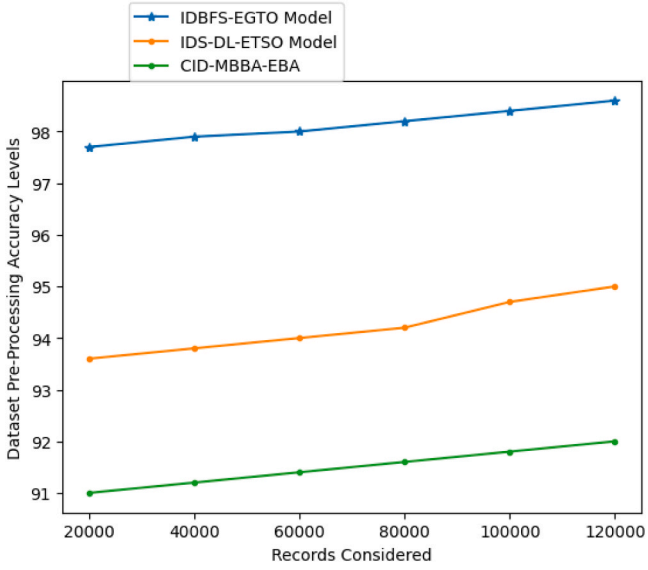


**Fig. 8.** Dataset pre-processing accuracy levels.

**Table 5**
Feature extraction accuracy levels.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 97 | 94.6 | 92.8 |
| 40000 | 97.3 | 94.8 | 93.1 |
| 60000 | 97.6 | 95 | 93.4 |
| 80000 | 97.8 | 95.2 | 93.6 |
| 100000 | 98 | 95.5 | 93.8 |
| 120000 | 98.2 | 96 | 94 |

account. The following programs are vulnerable to this kind of attack: Eject, Ffbconfig, Fdformat, loadmodule, perl, PS, and Xterm.

3. 3.Remote to Local Attack (R2L) is another name for user to superuser attacks. In order to obtain local user access to a machine, the attacker sends packets to the machine without the proper authorization. Some examples include xlock, xsnoop, named, phf, imap, ftp_write, xlock, and guest.
4. Probe: An attacker or attackers sweep a network to gather details about the computers in that network in order to get past its security measures. Saint, Satan, ipsweep, Mscan, and Nmap are a few examples.

A total of fourteen types of test data and twenty-four types of training attacks make up the dataset. Here, it's worth noting that the training set and test set do not come from the same distribution. Notably absent from the training data are some attack types that were present in the test set. While some attacks are included in both the test and training sets, some are exclusive to the test set.

When it comes to the digital realm, cyber security is paramount. Nowadays, one of the biggest problems is making sure all the data and information is secure. Cybercrime is the first thing that comes to mind when cybersecurity is concerned, and it's growing at a tremendous rate. The purpose of instituting an anti-virus policy in IDS is to lessen the likelihood of malware and other harmful code attacks on company computers, networks, and technological systems. The goal of this policy is to protect user software, data, files, and hardware from harm. Computers may be protected from harmful software like viruses and worms with antivirus software. This program can detect these threats, stop them, and even remove them. Antivirus software often has an auto-update function that allows it to download virus profiles automatically, allowing it to scan for newly found viruses immediately. Every computer should have antivirus software installed.The corporate/company's computer devices should be used appropriately, according to the Acceptable Use of Data Systems Policy. Both the authorized user and the business are safeguarded by these regulations. Risks like as malware assaults, compromised network systems and services, and legal concerns can befall the company due to inappropriate use.

## 5. Results analysis

In analyzing the network data, a IDS must contend with irrelevant and redundant features, as well as false positives. These characteristics not only slow down detection but also require extensive computational resources. To improve precision; training and testing accuracy, it is crucial to have a process for selecting the best characteristics. By determining which features are useful, feature selection aids in resolving a number of IDS's more typical issues. Pertinent features contain crucial data that significantly aids in classification. Feature selection in IDS is crucial since it decreases the amount of time spent processing data, requires less space for storage, and improves comprehension of the test results. The proposed model considered NSL-KDD dataset from the link https://www.kaggle.com/datasets/hassan06/nslkdd. This updated version of the KDD'99 data collection is known as NSL-KDD. The proposed model is implemented in Python and executed in Google Colab.

In this research, how to choose best features from network traffic data is suggested in order to identify possible attacks. This research initially started with elucidating the salient features of network traffic data that aid in the identification of a malicious signals. Information technology workers can use the characteristics of hostile signals uncovered by feature selection techniques to better secure their networks.

**Table 6**
Feature correlation calculation time levels in milliseconds.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 16.3 | 19.7 | 22.6 |
| 40000 | 16.6 | 20 | 23 |
| 60000 | 16.8 | 20.2 | 23.3 |
| 80000 | 17.2 | 20.5 | 23.5 |
| 100000 | 17.6 | 20.7 | 23.7 |
| 120000 | 18 | 21 | 24 |

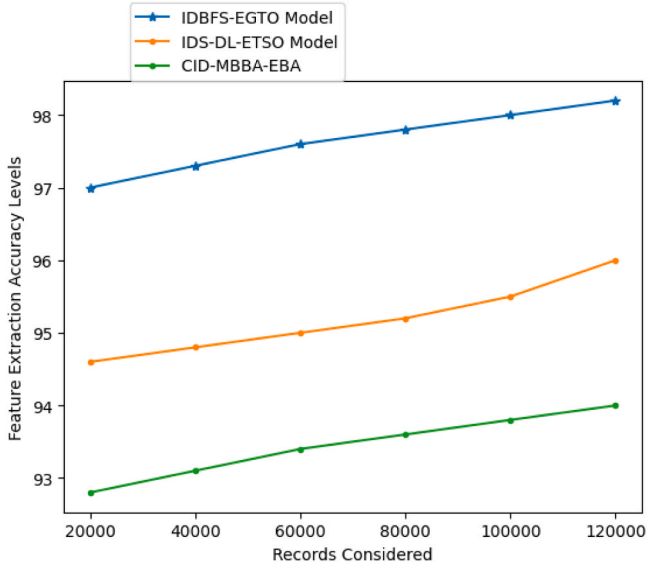


**Fig. 9.** Feature extraction accuracy levels.

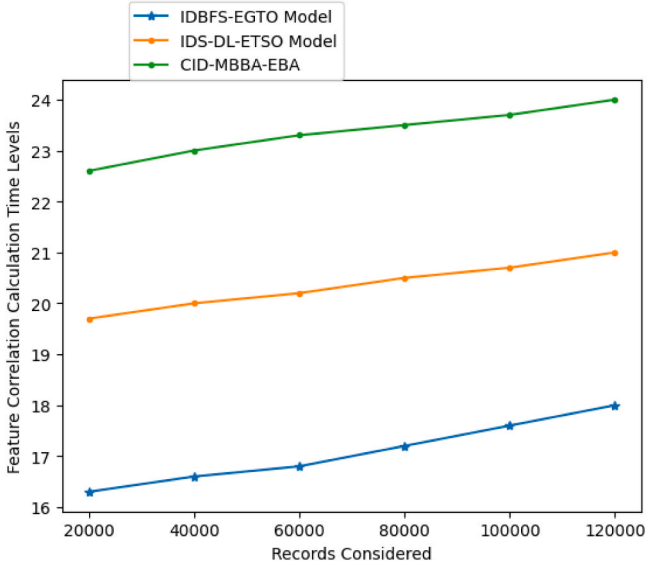


**Fig. 10.** Feature correlation calculation time levels in milliseconds.

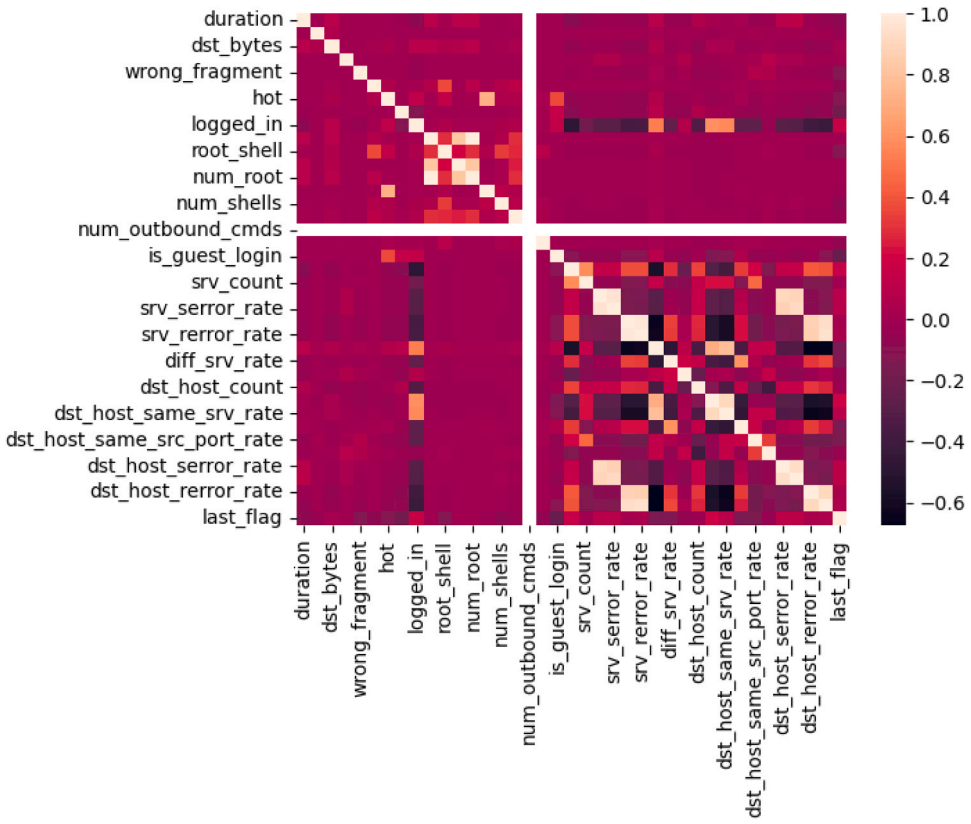


**Fig. 11.** Correlation matrix.

**Table 7**
Feature weight allocation accuracy levels.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 97 | 92 | 91.7 |
| 40000 | 97.2 | 92.4 | 92 |
| 60000 | 97.3 | 92.8 | 92.3 |
| 80000 | 97.5 | 93 | 92.5 |
| 100000 | 97.6 | 93.5 | 92.8 |
| 120000 | 97.8 | 94 | 93 |

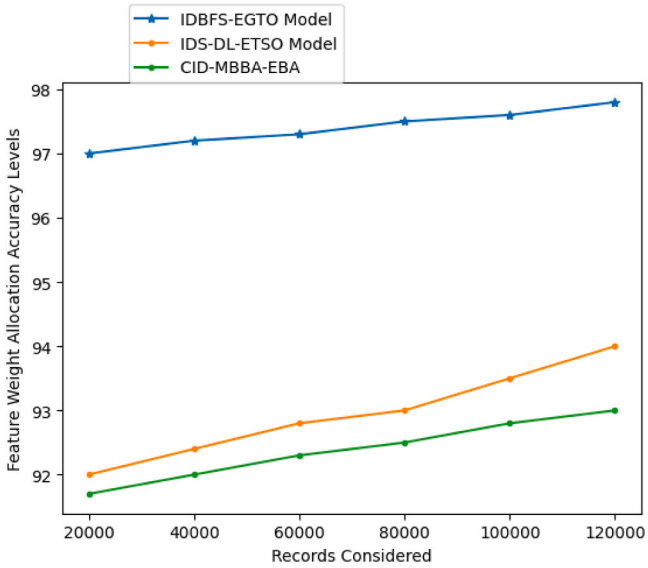


**Fig. 12.** Feature weight allocation accuracy levels.

Second, effective machine learning-based IDS is developed by feature selection. Using feature selection techniques, a number of detection algorithms with impressive detection and false alarm rates are detected. Differentiating between continuous and discrete features can be difficult, so a new forward search technique that uses correlation and mutual information to choose the best features is proposed in this research. The final feature score is derived by dividing the feature's significance by its redundancy and weighting the difference. This research presents a Interrelated Dynamic Biased Feature Selection Model using Enhanced Gorilla Troops Optimizer (IDBFS-EGTO) for generation of feature vector set for intrusion detection. The proposed model is compared with the traditional IoT Intrusion Detection System Using Deep Learning and Enhanced Transient Search Optimization (IDS-DL-ETSO) and Cyber Intrusion Detection System Based on a Multiobjective Binary Bat Algorithm for Feature Selection and Enhanced Bat Algorithm for Parameter Optimization in Neural Networks (CID-MBBA-EBA) models. The results represent that the proposed model performance in feature selection is high than the traditional models.

Companies really do look at the technical and financial aspects of IDS systems to see how well they worked. Therefore, when deciding on the best IDS product, there are a plethora of factors that go into the overall assessment of any IDS deployment. When considering intrusion detection from a practical standpoint, the cost-effectiveness, or cost-benefit trade-off, is an essential but frequently overlooked aspect. It is common for security risk management to drive a well-informed decision to implement a security mechanism like an intrusion detection system. In this way, the goal of intrusion detection systems is to safeguard valuable yet vulnerable information assets. Because it should not cost more than the anticipated amount of loss from intrusions, an intrusion detection system (IDS) must be cost-effective. This necessitates that an intrusion detection system (IDS) take into account the cost-benefit analysis of

**Table 8**
EGTO optimization accuracy levels.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 97.3 | 93.8 | 92.4 |
| 40000 | 97.4 | 94 | 92.7 |
| 60000 | 98 | 94.3 | 93 |
| 80000 | 98.4 | 94.5 | 93.2 |
| 100000 | 98.5 | 94.8 | 93.7 |
| 120000 | 98.6 | 95 | 94 |

**Table 9**
Feature subset generation accuracy levels.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 98 | 94 | 92.5 |
| 40000 | 98.1 | 94.2 | 92.8 |
| 60000 | 98.2 | 94.3 | 93 |
| 80000 | 98.4 | 94.6 | 93.1 |
| 100000 | 98.5 | 94.8 | 93.3 |
| 120000 | 98.6 | 95 | 93.5 |

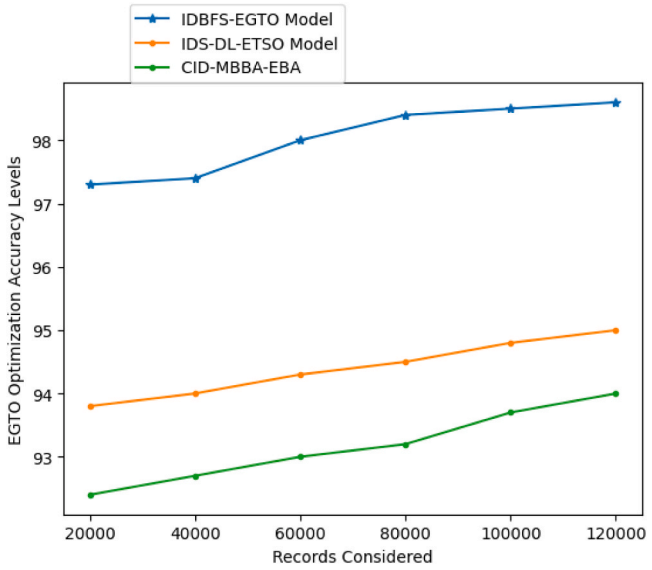


**Fig. 13.** EGTO optimization accuracy levels.

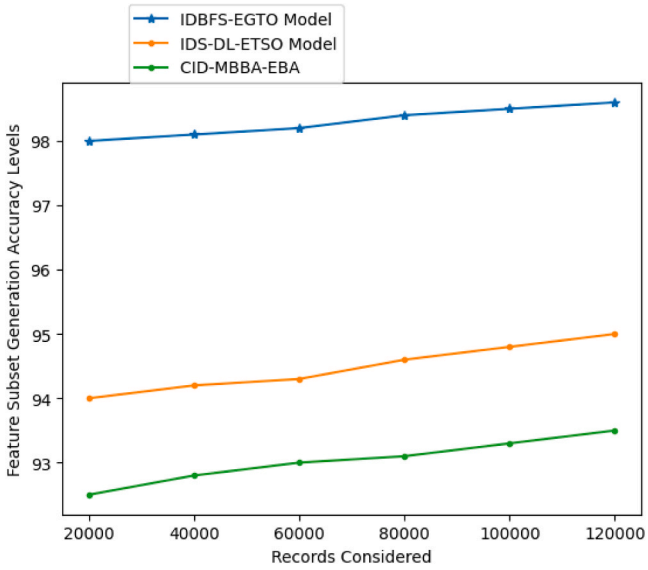


**Fig. 15.** Feature subset generation accuracy levels.

```
------------ Gorilla score after Exploration   1 -------------
0.867811918258667
2953/2953 [==============================] - 6s 2ms/step - loss: 101.9720 - accuracy: 0.8678
------------Gorilla score after Exploitation  1 -------------
0.867811918258667
2953/2953 [==============================] - 5s 2ms/step - loss: 101.9720 - accuracy: 0.8678
------------ Gorilla score after Exploration   2 -------------
0.867811918258667
2953/2953 [==============================] - 5s 2ms/step - loss: 101.9720 - accuracy: 0.8678
------------Gorilla score after Exploitation  2 -------------
0.867811918258667
2953/2953 [==============================] - 9s 3ms/step - loss: 101.9720 - accuracy: 0.8678
------------ Gorilla score after Exploration   3 -------------
0.867811918258667
2953/2953 [==============================] - 8s 3ms/step - loss: 101.9720 - accuracy: 0.8678
------------Gorilla score after Exploitation  3 -------------
0.867811918258667
2953/2953 [==============================] - 7s 2ms/step - loss: 101.9720 - accuracy: 0.8678
------------ Gorilla score after Exploration   4 -------------
0.867811918258667
2953/2953 [==============================] - 9s 3ms/step - loss: 101.9720 - accuracy: 0.8678
------------Gorilla score after Exploitation  4 -------------
0.867811918258667
2953/2953 [==============================] - 7s 2ms/step - loss: 101.9720 - accuracy: 0.8678
------------ Gorilla score after Exploration   5 -------------
0.867811918258667
2953/2953 [==============================] - 11s 4ms/step - loss: 101.9720 - accuracy: 0.8678
------------Gorilla score after Exploitation  5 -------------
0.867811918258667
The best so-far Fitness value obtained by GTO at iteration  5 == 101.97203826904297
```

**Fig. 14.** GTO phases and fitness calculation process.

**Table 10**
Intrusion detection accuracy levels.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 97.5 | 93 | 93.5 |
| 40000 | 97.7 | 93.4 | 93.7 |
| 60000 | 97.8 | 93.8 | 94 |
| 80000 | 98.1 | 94 | 94.2 |
| 100000 | 98.2 | 94.3 | 94.5 |
| 120000 | 98.4 | 94.8 | 95 |

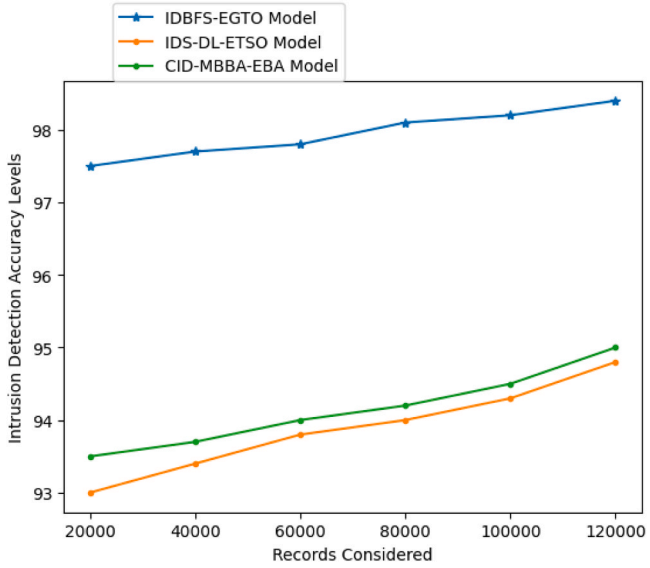


**Fig. 16.** Intrusion detection accuracy levels.

**Table 11**
Classification accuracy.

| Records Considered | Models Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model | IDS-DL-ETSO Model | CID-MBBA-EBA Model |
| 20000 | 97.6 | 92.4 | 93.7 |
| 40000 | 97.8 | 92.5 | 93.9 |
| 60000 | 98 | 92.7 | 94 |
| 80000 | 98.1 | 93 | 94.1 |
| 100000 | 98.3 | 93.2 | 94.2 |
| 120000 | 98.6 | 93.5 | 94.4 |

various factors, such as development cost, damage cost, cost of manual or automatic response to an intrusion, and operational cost, which measures resource and time constraints.

Building an intelligent room lighting system and an Internet of Things (IoT)-based fire alarm system allows for the creation of a real-time testbed. One system uses an ESP8266 microcontroller, while the other uses NodeMCU. Data was sent to a local server using the NodeMCU-enabled WiFi module. In order to sniff the network and collect the packets in real-time, the packet capture algorithm runs continually. We have executed denial-of-service and man-in-the-middle attacks on the same network from two separate systems using the script in order to test the model's performance in real-time.

An example of a swarm intelligence algorithm is the EGTO, which draws inspiration from the social and behavioral patterns observed in gorilla troops. These troops typically include an adult male, or silverback, multiple adult females, and their young. When a silverback reaches maturity, it starts to grow hair on its back that is the origin of its

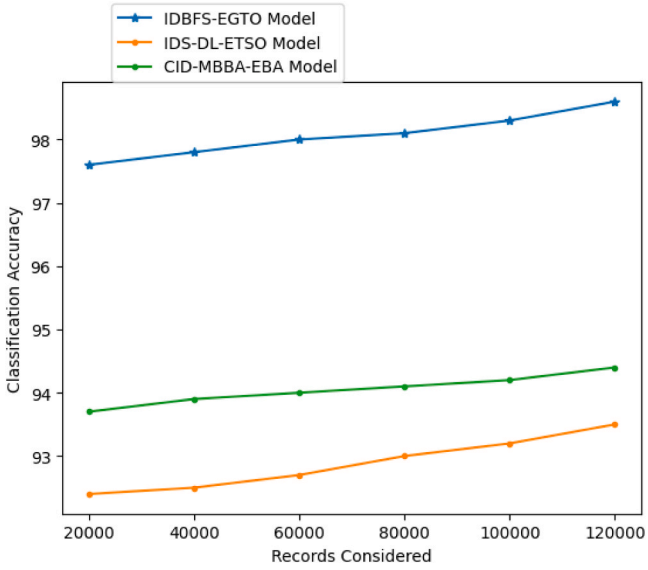


**Fig. 17.** Classification accuracy.

**Table 12**
Evaluation metrics.

| Evaluation Metric | Results Achieved |
|---|---|
| Accuracy | 98.6 |
| Precision | 98.2 |
| Recall | 94 |
| F1-Score | 92 |
| Loss Rate | 0.04 |

name. Male gorillas often split off from their existing groups to establish new ones, a process that involves courting females who have previously relocated. Male gorillas sometimes remain with the family group from which they originated; this is called the silverback group. In order to protect themselves from predators and have a successful breeding season, gorilla females form strong relationships with males. In the absence of silverback gorillas, baby gorillas are more likely to be victims of infanticide and may try to find refuge with other groups. The center of attention among them is Silverback. The gorilla pride is in charge of everything: making decisions, resolving disagreements, directing the troop to food sources, and making sure everyone stays safe. Young male gorillas known as "blackbacks" accompany the silverbacks and offer additional protection when needed. The EGTO model is compared with different datasets and in different domains.

The protocols included in the model has their own performance levels in data transmission. The types of protocols used in the model are depicted in Fig. 5.

The network dataset is considered that contains normal and malicious transactions. The network data is analyzed and the values and their ranges are calculated. The Dataset Record Analysis Time Levels in milliseconds of the existing and proposed models are shown in Table 3 and Fig. 6.

The proposed model uses different flag variables to check the range of attributes for detection of a attack. These flags play a key role in the detection of a attack accurately. The change of attributes in a specific flag type is used for attack detection. The flags used in the porposed model are represented in Fig. 7.

Pre-processing is used to describe the operations performed on network data before it is sent into the algorithm. In order to get useful information out of raw data, a method called data preprocessing must be performed. In other words, data collection from multiple sources typically results in raw data that cannot be easily analyzed. The preprocessing allows to clear the data before performing feature

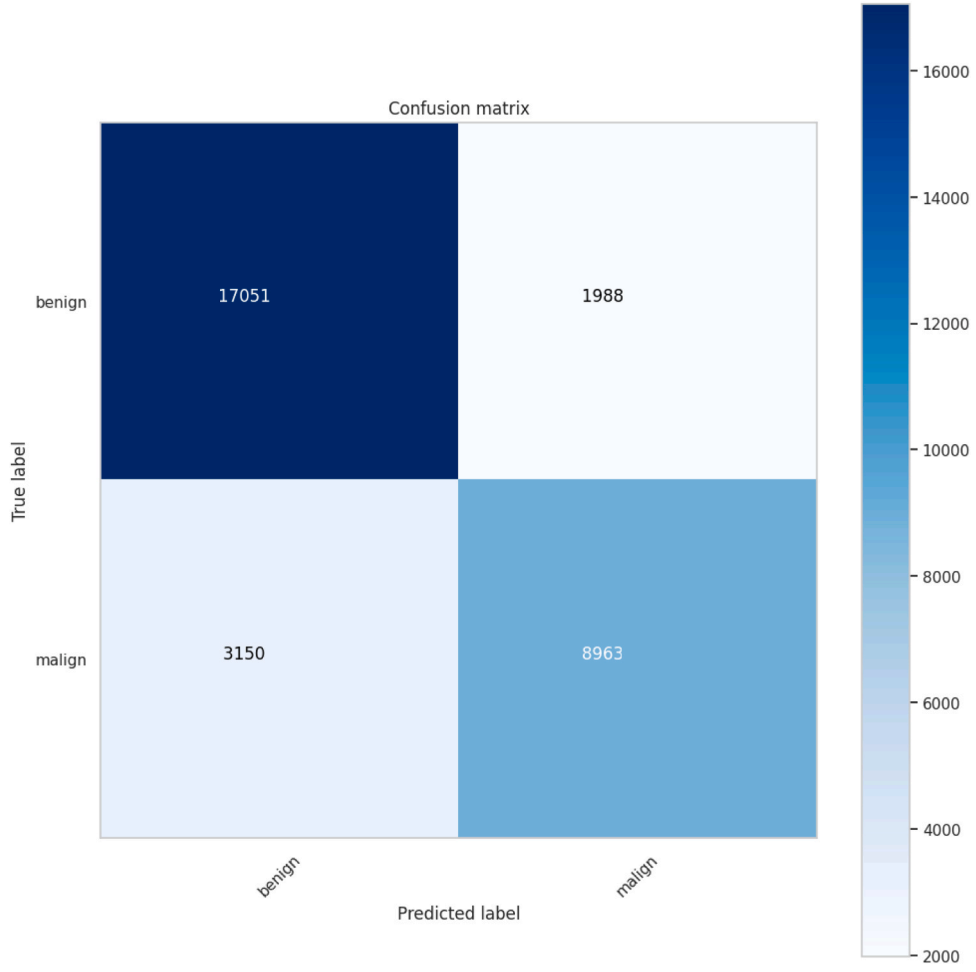


**Fig. 18.** Confusion matrix.

**Table 13**
EGTO accuracy with same datasets.

| Records Considered | Datasets Considered | | |
|---|---|---|---|
| | IDBFS-EGTO Model NSL-KDD (Proposed) | OID-DL [10] NSL-KDD | LSTM-FCNN [11] NSL-KDD |
| 20000 | 97.3 | 95.6 | 96.3 |
| 40000 | 97.4 | 96.4 | 96.7 |
| 60000 | 98.1 | 96.8 | 97.2 |
| 80000 | 98.4 | 97 | 97.3 |
| 100000 | 98.5 | 97.2 | 97.7 |
| 120000 | 98.6 | 97.4 | 97.8 |

**Table 14**
EGTO accuracy with different datasets.

| Records Considered | Datasets Considered | | |
|---|---|---|---|
| | NSL-KDD | UNSW-NB15 | CICIDS2017 |
| 20000 | 97.3 | 97 | 96.8 |
| 40000 | 97.4 | 97.3 | 97.1 |
| 60000 | 98 | 97.6 | 97.3 |
| 80000 | 98.4 | 97.9 | 97.5 |
| 100000 | 98.5 | 98 | 97.6 |
| 120000 | 98.6 | 98.2 | 97.9 |

extraction. The Dataset Pre-Processing Accuracy Levels of the proposed and existing models are shown in Table 4 and Fig. 8.

Dimensionality reduction, or feature extraction, is the process by which a large amount of raw data is broken down into smaller, more manageable chunks. The sheer number of variables in such data sets necessitates the use of powerful computers. To reduce the amount of data that needs to be processed while still providing an accurate and complete description of the original data set, a number of techniques have been developed under the feature extraction. This research presents a model for accurate feature extraction of the complete dataset. The Table 5 and Fig. 9 show the Feature Extraction Accuracy Levels of the proposed and traditional models.

Correlation coefficients and reciprocal data are computed between features and the variable that is dependent. Using this representation, features having a strong correlation or big shared data with the variable that is dependent can be isolated for further analysis during the feature selection process. The Feature Correlation Calculation Time Levels of the proposed and traditional models are shown in Table 6 and Fig. 10.

**Table 15**
EGTO accuracy in different domains.

| Records Considered | Domains Considered | | |
|---|---|---|---|
| | Intrusion Detection | Tumor Prediction | Cloud Scheduling |
| 20000 | 97.6 | 97 | 96.7 |
| 40000 | 97.8 | 97.4 | 97 |
| 60000 | 98 | 97.6 | 97.2 |
| 80000 | 98.1 | 98.1 | 97.6 |
| 100000 | 98.3 | 98.4 | 97.9 |
| 120000 | 98.6 | 98.7 | 98.3 |

The statistical method of examining the connection between two variables in a data collection is called a correlation matrix. Each cell in the matrix represents a correlation coefficient, with 1 denoting a highly significant positive relationship, 0 indicating no significant relationship, and −1 a highly significant negative relationship. The correlation matrix of the proposed model is represented in Fig. 11.

Methods for enhancing classification accuracy and decreasing data complexity include selecting and weighting relevant features. Only data with features that are redundant or unimportant should use feature selection, whereas data with variable relevance should use feature weighting. Each feature's importance to the model's ability to make predictions is quantified by its weight. The significance of the values of the coefficients and the variation of the variables that are provided are used to determine them. The Table 7 and Fig. 12 shows the Feature Weight Allocation Accuracy Levels of the existing and proposed models.

EGTO mimics the social behavior of gorillas as a meta heuristic optimization method. Gorilla behavior, such as moving to a known location, moving to an unknown location, moving toward other gorillas, following silverbacks, and vying with silverbacks for females, served as inspiration for its design. Local optimality, lack of diversity, uneven utilization, etc. are all problems that the EGTO shares with other meta heuristic algorithms. This study suggests a EGTO to enhance the GTO's performance. The EGTO Optimization Accuracy Levels of the proposed and traditional models are shown in Table 8 and Fig. 13.

The prime instance of a metaheuristic optimization algorithm that attempts to mimic the social dynamics of gorillas is the artificial GTO algorithm. The typical size of a troop is 9, though it can range from 2 to 12 members. Their activity patterns are extremely coordinated with one another. If there are more than one silverback in a group, they are likely the male offspring of the leader. The GTO phases and the fitness value calculation process is indicated in Fig. 14.

Finding and picking out the best features to use from a large network dataset is what feature subset selection is all about. Its goals are to make models more efficient, cut down on over fitting, and make them easier to understand. Feature subset selection involves finding and discarding as much superfluous data as feasible. This decreases the data's dimensionality, making it easier for learning algorithms to function quickly and efficiently. The Feature Subset Generation Accuracy Levels of the proposed and traditional models are shown in Table 9 and Fig. 15.

To keep monitoring network or systems for malicious behavior or policy violations, users need an intrusion detection system. Security information and monitoring systems often collect reports of intrusion activity and violations and either forward them to an administrator or store them centrally. An IDS analyzes network traffic seeking for suspicious activities and recognized hazards, putting up notifications when it finds such items. Intrusion detection has been a mainstay of corporate cyber security for quite some time, and while it is still very important, it may not be sufficient on its own in the modern business. The Intrusion detection accuracy levels of the proposed and existing models are represented in Table 10 and Fig. 16.

IDS are the backbone of network safety. IDSs monitor the behavior and activities of the system to look for signs of intrusion. The detection of misuse and anomaly detection are only two examples of IDS methods that can be used to spot assaults of any sophistication. The detection accuracy of anomaly detection is low for both known and unknown attacks, while that of abuse detection is high exclusively for known attacks. The proposed model accurately detects the intrusions in the network and the classification is performed. The Classification accuracy of the proposed and existing models is depicted in Table 11 and Fig. 17.

The proposed model classification rate is high when compared to tha traditional models. The evaluation metrics like precision, recall are calculated and the confusion matrix is also generated. The Table 12 shows the evaluation metrics when compared to the traditional models and the confusion matrix is shown in Fig. 18. The proposed model performs accurate analysis of the evaluation metrics. The evaluation metrics are calculated based on the given equations.

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

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall}$$

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

Here TP indicates True Positive, FP indicates False Positive, TN indicates True Negative, FN indicates False Negative

The Table 13 shows the EGTO model comparison with other models that used the same dataset and Table 14 shows the comparison of EGTO model applied with different datasets and Table 15 shows the comparison of EGTO model applied in different domains.

## 6. Limitations and discussions

It might be tough to deploy dynamic biased feature selection methods in real-time applications due to the complexity they inject into the feature selection process. The system's responsiveness and efficiency during actual threat detection could be compromised due to the higher computing overhead caused by the adaptability necessary for dynamic feature selection.Since class imbalance is so common in datasets like KDD Cup'99, utilizing dynamic biased feature selection inside EGTO for IDS has several significant limitations. When there are a lot fewer instances in the minority class (attacks) compared to the majority class (regular traffic), biased feature selection might not work as well. As a result, the classification model may start to favorably forecast the majority class, which might lead to subpar generalization.

Optimal feature selection with the EGTO relies on a balance between global and local search capabilities, which can be difficult to achieve with complicated datasets. Failure to thoroughly investigate the feature space can lead to feature subsets that are less than ideal and fail to capture the diversity of infiltration patterns. Optimization of the EGTO settings is a challenging task, which adds another layer of complexity to its implementation.Clearly defined and exhaustive evaluation metrics are required for the efficient usage of EGTO and dynamic biased feature selection in intrusion detection systems. Important metrics for effectiveness evaluation include detection accuracy (DA), detection rate (DR), precision, and F-score. However, when used in isolation, these traditional metrics might miss the complexity of feature selection and classification performance in the real world. Performance evaluations could be made more accurate with more complex measures that consider the application environment. Researchers can strengthen intrusion detection systems' ability to withstand and counteract new cyber threats by discussing how to incorporate dynamic biased feature selection and EGTO-based classification models and finding solutions to these limitations.

For efficient training of deep learning models, Graphics Processing Units (GPUs) are crucial due to their capacity to execute parallel calculations on massive data sets. Depending on the model and features, GPUs can cost anywhere from a few hundred to several thousand dollars. As an example, deep learning applications frequently utilize NVIDIA Tesla or A100 GPUs, and their pricing mirror their computing capabilities. In order to host the intrusion detection systems and enable real-time data processing, a solid server infrastructure is necessary. Both cloud-based solutions and servers hosted on-premises are available to organizations. While cloud services charge recurring monthly fees according to use and capacity, the initial investment in on-premises servers can be quite large, frequently reaching tens of thousands of dollars. Jupyter notebooks and integrated development environments (IDEs) are examples of tools used for training, monitoring, and evaluating models;

nevertheless, these tools can be expensive, particularly in professional contexts when capabilities like collaboration are required.

As more data becomes available and new patterns of attacks arise, deep learning models must undergo ongoing training. This involves performing routine maintenance and updates on both hardware and software. Data scientists and machine learning engineers may be a continuing expense due to the time and effort required to oversee and retrain the models. There is a substantial quantity of electricity consumed by high-performance computer systems. Powering servers and cooling systems to keep them at ideal operating temperatures is an expense that organizations need to consider.

## 7. Conclusion

Intrusion Detection Systems have developed into a crucial component of modern security infrastructure due to their role in risk management. When looking at a IDS via the lens of a pattern recognition system, feature extraction becomes a crucial first step in the process. Two steps, feature construction and feature selection, make up feature extraction. The efficiency of a IDS is heavily reliant on the quality of the algorithms used in its feature building and feature selection processes. Improving the IDS's performance as a whole requires reducing the quantity of irrelevant traffic features without sacrificing classification accuracy. Identifying and stopping malicious network activity is what intrusion detection is all about. It plays a crucial role in preventing hackers from gaining access to the network. To detect incursion activities, a collection of procedures and methods must be applied. Thus, any program with the ability to monitor for and react to suspicious behavior is technically an intrusion detection system. When developing machine learning models, feature selection is crucial. The model's accuracy and the time required to train it both suffer when unnecessary data elements are included. When developing a IDS, feature selection plays a crucial role.

### Findings

In this research, a novel IDS using EGTO is designed for enhancing system security. Classification is the task of assigning class labels to the feature set. This research presents a Interrelated Dynamic Biased Feature Selection Model using Enhanced Gorilla Troops Optimizer for generation of feature vector set for intrusion detection and then performing classification. It has been noted that finding and categorizing records using all 41 attributes of the KDD'99 cup data set requires a significant amount of computational time. The suggested feature selection algorithm chooses just the most useful features, speeding up the record detection and classification processes. Inspired by the ingenuity of wild gorilla groups as a whole, the EGTO is a cutting-edge metaheuristic algorithm introduced in this research. Despite its apparent success in handling a wide range of practical problems, it risks getting mired in local optima and premature convergence when faced with more difficult optimization challenges that is overcome with EGTO. The EGTO approach, which uses a collection of operators to strike a more steady equilibrium between exploitation and exploration. The proposed model generates relevant feature subset for machine learning model for accurate detection and classification of intrusions in the network. The proposed model achieved 98.4 % accuracy in intrusion detection and 98.6 % accuracy in EGTO optimization classification. The proposed model is improved by 3.8 % in feature weight allocation accuracy and 1.2 % in detection accuracy levels. A new innovation in network and system security known as an IDS was proposed in this research. Many professionals in the field of cybersecurity stress the significance of IDS in bolstering system defensive capabilities through the detection and alerting of hostile activity and attacks.

### Declaration of Competing Interest

None

### Acknowledgement

The authors would like to acknowledge VIT-AP University and its management for supporting the financial assistance during the research work and for publishing the research paper, enabling our research to be openly accessible to all.

#### *Research limitations*

There are benefits and drawbacks to using IDS. Despite its usefulness, it encounters problems including false positives, overloaded alerts, insufficient resources, insufficient people experience, evasion strategies, complicated integration, and limited signatures and encryption. The potential for intrusion detection systems to mistakenly flag harmless changes in network behavior as worrisome anomalies increases the likelihood of false positives.

#### *Recommendations for future research*

In future, dimensionality reduction techniques can be further applied to still reduce the feature subset dimensions and hybrid optimization techniques can be applied for better accurate and security levels.

### References

[1] M.Abd Elaziz Fatani, A. Dahou, M.A.A. Al-Qaness, S. Lu, IoT intrusion detection system using deep learning and enhanced transient search optimization, IEEE Access 9 (2021) 123448–123464, https://doi.org/10.1109/ACCESS.2021.3109081.
[2] W.A.H.M. Ghanem, et al., Cyber intrusion detection system based on a multiobjective binary bat algorithm for feature selection and enhanced bat algorithm for parameter optimization in neural networks, IEEE Access 10 (2022) 76318–76339, https://doi.org/10.1109/ACCESS.2022.3192472.
[3] A. Alsaleh, W. Binsaeedan, The influence of salp swarm algorithm-based feature selection on network anomaly intrusion detection, IEEE Access 9 (2021) 112466–112477, https://doi.org/10.1109/ACCESS.2021.3102095.
[4] L. Almutairi, R. Daniel, S. Khasimbee, E.L. Lydia, S. Acharya, H.-I. Kim, Quantum dwarf mongoose optimization with ensemble deep learning based intrusion detection in cyber-physical systems, IEEE Access 11 (2023) 66828–66837, https://doi.org/10.1109/ACCESS.2023.3287896.
[5] L. Tao, M. Xueqiang, Hybrid strategy improved sparrow search algorithm in the field of intrusion detection, IEEE Access 11 (2023) 32134–32151, https://doi.org/10.1109/ACCESS.2023.3259548.
[6] M. Bakro, et al., An improved design for a cloud intrusion detection system using hybrid features selection approach with ML classifier, IEEE Access 11 (2023) 64228–64247, https://doi.org/10.1109/ACCESS.2023.3289405.
[7] A.Y. Hatata, M.A. Essa, B.E. Sedhom, Adaptive protection scheme for FREEDM microgrid based on convolutional neural network and gorilla troops optimization technique, IEEE Access 10 (2022) 55583–55601, https://doi.org/10.1109/ACCESS.2022.3177544.
[8] M.A. Siddiqi, W. Pak, Tier-based optimization for synthesized network intrusion detection system, IEEE Access 10 (2022) 108530–108544, https://doi.org/10.1109/ACCESS.2022.3213937.
[9] M.A. Alohali, F.N. Al-Wesabi, A.M. Hilal, S. Goel, D. Gupta, A. Khanna, Artificial intelligence enabled intrusion detection systems for cognitive cyber-physical systems in industry 4.0 environment, Cogn. Neurodyn 16 (2022) 1–13.
[10] Agalit Mohamed Amine, Optimization of intrusion detection with deep learning: a study based on the KDD Cup 99 database, Int. J. Saf. Secur. Eng. 14 (2024) 1029–1038, https://doi.org/10.18280/ijsse.140402.
[11] A. Chakrawarti, S.S. Shrivastava, Intrusion detection system using long short-term memory and fully connected neural network on Kddcup99 and NSL-KDD dataset, Int. J. Intell. Syst. Appl. Eng. 11 (9s) (2023) 621–635.
[12] F. Ullah, G. Srivastava, S. Ullah, K. Yoshigoe, Y. Zhao, NIDS-VSB: network intrusion detection system for VANET using spark-based big data optimization and transfer learning, IEEE Trans. Consum. Electron. 70 (1) (Feb. 2024) 1798–1809, https://doi.org/10.1109/TCE.2023.3328320.
[13] K. Zhang, et al., Intrusion detection model for internet of vehicles using GRIPCA and OWELM, IEEE Access 12 (2024) 28911–28925, https://doi.org/10.1109/ACCESS.2024.3368392.
[14] D. Jayalatchumy, R. Ramalingam, A. Balakrishnan, M. Safran, S. Alfarhood, Improved crow search-based feature selection and ensemble learning for IoT intrusion detection, IEEE Access 12 (2024) 33218–33235, https://doi.org/10.1109/ACCESS.2024.3372859.

[15] M. Sharma, H. Elmiligi, F. Gebali, A novel intrusion detection system for RPL-based cyber–physical systems, IEEE Can. J. Electr. Comput. Eng. 44 (2) (2021) 246–252.
[16] J. Ali, Intrusion detection systems trends to counteract growing cyber-attacks on cyber-physical systems, Proc. 22nd Int. Arab Conf. Inf. Technol. (ACIT) (2021) 1–6.
[17] M.M. Althobaiti, K.P.M. Kumar, D. Gupta, S. Kumar, R.F. Mansour, An intelligent cognitive computing based intrusion detection for industrial cyber-physical systems, Measurement 186 (2021).
[18] R.F. Mansour, Artificial intelligence based optimization with deep learning model for blockchain enabled intrusion detection in CPS environment, Sci. Rep. 12 (1) (2022) 12937.
[19] A.K. Dutta, R. Negi, S.K. Shukla, Robust multivariate anomaly-based intrusion detection system for cyber-physical systems, Proc. Int. Symp. Cyber Secur. Cryptogr. Mach. Learn. (2021) 86–93.
[20] P.F. de Araujo-Filho, G. Kaddoum, D.R. Campelo, A.G. Santos, D. Macêdo, C. Zanchettin, Intrusion detection for cyber–physical systems using generative adversarial networks in fog environment, IEEE Internet Things J. 8 (8) (Apr. 2021) 6247–6256.
[21] V.S.F. Enigo, K.T. Ganesh, N.N.V. Raj, D. Sandeep, Hybrid intrusion detection system for detecting new attacks using machine learning, Proc. 5th Int. Conf. Commun. Electron. Syst. (ICCES) (2020) 567–572.
[22] S.A.A. Ghaleb, M. Mohamad, S.A. Fadzli, W.A.H.M. Ghanem, Training neural networks by enhance grasshopper optimization algorithm for spam detection system, IEEE Access 9 (2021) 116768–116813.
[23] M.S. Haghighi, F. Farivar, A. Jolfaei, A machine learning-based approach to build zero false-positive IPSs for industrial IoT and CPS with a case study on power grids security, IEEE Trans. Ind. Appl. (2020).
[24] O. Almomani, A hybrid model using bio-inspired metaheuristic algorithms for network intrusion detection system, Comput. Mater. Contin. 68 (1) (2021) 409–429.
[25] Z. Liu, Y. Shi, A hybrid IDS using GA-based feature selection method and random forest, Int. J. Mach. Learn. Comput. 12 (2) (2022) 43–50.
[26] M. Ajdani, H. Ghaffary, Introduced a new method for enhancement of intrusion detection with random forest and PSO algorithm, Secur. Privacy 4 (2) (2021).
[27] R.O. Ogundokun, J.B. Awotunde, P. Sadiku, E.A. Adeniyi, M. Abiodun, O.I. Dauda, An enhanced intrusion detection system using particle swarm optimization feature extraction technique, Proc. Comput. Sci. 193 (2021) 504–512.
[28] F.H. Almasoudy, W.L. Al-Yaseen, A.K. Idrees, Differential evolution wrapper feature selection for intrusion detection system, Proc. Comput. Sci. 167 (2020) 1230–1239.
[29] W.A.H.M. Ghanem, A. Jantan, Training a neural network for cyberattack classification applications using hybridization of an artificial bee colony and monarch butterfly optimization, Neural Process. Lett. 51 (1) (2020) 905–946.
[30] S.A.A. Ghaleb, M. Mohamad, S.A. Fadzli, W.A.H.M. Ghanem, E-mail spam classification using grasshopper optimization algorithm and neural networks, Comput. Mater. Contin. 71 (3) (2022) 4749–4766.
[31] S.A.A. Ghaleb, M. Mohamad, E.F.H.S. Abdullah, W.A.H.M. Ghanem, An integrated model to email spam classification using an enhanced grasshopper optimization algorithm to train a multilayer perceptron neural network, Proc. Int. Conf. Adv. Cyber Secur. (2020) 402–419.
[32] H. Zhang, J.-L. Li, X.-M. Liu, C. Dong, Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection, Future Gener. Comput. Syst. 122 (2021) 130–143.