Computers & Security 137 (2024) 103587

Contents lists available at ScienceDirect

Computers & Security

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

M-MultiSVM: An efficient feature selection assisted network intrusion detection system using machine learning

Anil V Turukmane [a], Ramkumar Devendiran [b,*]

[a] Professor, School of Computer Science & Engineering, VIT-AP University, Amaravati, Vijayawada, Andhra Pradesh 522237, India
[b] Assistant Professor Senior Grade, School of Computer Science & Engineering, VIT-AP University, Amaravati, Vijayawada, Andhra Pradesh 522237, India

ARTICLE INFO

Keywords:
Null value handling
Min-max normalization
Synthetic minority oversampling
Singular value decomposition
Northern Goshawk Optimization
Multilayer SVM
Mud ring

ABSTRACT

The intrusions are increasing daily, so there is a huge amount of privacy violations, financial loss, illegal transferring of information, etc. Various forms of intrusion occur in networks, such as menacing networks, computer resources and network information. Each type of intrusion focuses on specified tasks, whereas the hackers may focus on stealing confidential data, industrial secrets and personal information, which is then leaked to others for illegal gains. Due to the false detection of attacks in the security and changing environmental fields, limitations like data lagging on actual attacks and sustaining financial harms occur. To resolve this, automatic abnormality detection systems are required to secure the required computing ability and to analyze the attacks. Hence, an efficient automated intrusion detection system using machine learning methodology is proposed in this research paper. Initially, the data are gathered from CSE-CIC-IDS 2018 and UNSW-NB15 datasets. The acquired data are pre-processed using Null value handling and Min-Max normalization. Null value handling is used to remove missing values and irrelevant parameters. Min-Max normalization adjusted the unnormalized data in the pre-processing stage. After pre-processing, the class imbalance problem is reduced by using the Advanced Synthetic Minority Oversampling Technique (ASmoT). ASmoT aims to balance the class and reduce imbalance class problems and overfitting issues. The next phase is feature extraction, which is performed by Modified Singular Value Decomposition (M-SvD). M-SvD extracts essential features such as basic features, content features and traffic features from the input. The extracted features are optimized by the Opposition-based Northern Goshawk Optimization algorithm (ONgO). These optimal features are able to produce optimal output. After feature selection, the different types of attacks are classified by a hybrid machine learning model called Mud Ring assisted multilayer support vector machine (M-MultiSVM) and finally, the hyperparameters are tuned by the Mud Ring optimization algorithm. Thus, the proposed M-MultiSVM model can efficiently detect intrusion in the network. The performance metrics show that the proposed system achieved 99.89 % accuracy by using the CSE-CIC-IDS 2018 dataset; also, the proposed system achieved 97.535 % accuracy by using the UNSW-NB15 dataset.

1. Introduction

Artificial intelligence (AI) is one of the most popular technologies to solve problems effectively, detecting human error and outperforming human abilities. Using AI, businesses are protected from Internet threats, and malware detection increases the recovery and prevention procedures and fixes the security standards setting. In Cyber Security, AI plays an important role and is helpful for AI applications and their technology (Jia et al., 2020). The security research's major aim is to develop the system by monitoring unauthorized persons' access to the network and enhancing the system. With the advancement of technology, businesses are affected by several cyber-attacks that decrease the performance of businesses and applications. Due to the improvement of network services or applications, network traffic data is increased and difficult to process (Gupta and Kulariya, 2016). The several connectives between the applications make the instant analysis for cyber security breaches. These breaches can affect business operations and create financial losses. Due to this reason, cyber security issues are becoming the highest priority in business (Kaja et al., 2019).

Cybersecurity protects computer and program exposure from modification, attacks, destruction and unauthorized access. The cybersecurity system comprises antivirus techniques, firewalls and an Intrusion

* Corresponding author.
E-mail address: ramkumar.d@vitap.ac.in (R. Devendiran).

https://doi.org/10.1016/j.cose.2023.103587
Received 4 August 2023; Received in revised form 10 October 2023; Accepted 6 November 2023
Available online 10 November 2023

Detection System (IDS) (Wong et al., 2017; Morris et al., 2015). Using this IDS, unwanted modifications and attacks are detected in the system. These intruders are of two types: internal and external intruders (Rekha et al., 2020). In cyberspace, breaches are increasing in different fields like medicine, finance, and industry, generating serious problems in cyber security. An effective and strong IDS must detect the intrusion attack in network traffic with energy and intelligence to overcome these issues. To prevent and reduce cyberattacks, awareness of cyberattacks is needed to secure or react against attackers (Alom et al. 2017). Denial of service (DoS) attacks, malware, as well as phishing are some attacks that IDS detects in specific network environments. Furthermore, this system can effectively classify unwanted changes in the network. IDS is only designed to detect the vulnerabilities presented outside the network infrastructure. But it doesn't act as the actual communication passage between the receiver and sender of data. The copy of inline traffic streams is analyzed using SPAN and TAP ports, and through this copy, a pre-trained algorithm detects the attacks (Vigneswaran et al., 2018).

Intelligent intrusion detection can be developed by using various Machine learning (ML) algorithms. Anomalies attacks are detected in cyber security by using ML techniques like SVM, Naïve Bayes (NB) Models, k-nearest neighbor (k-NN) algorithms, Artificial neural networks (ANN), fuzzy c-means clustering, Restricted Boltzmann machine, Recurrent neural network (RNN), Decision Trees and Logistic Regression (LR) (Al-Omari et al., 2021). In intrusion detection, the feature selection step is important to reduce the number of variables and hold the most important features by removing irrelevant features, performed by the Least Absolute Shrinkage and Selection Operator (LASSO). This selection algorithm would consume more amount of time to select optimal features (Magán et al. 2020). Here, the intrusion can be detected by a Bayesian network with two-step verification algorithms. Pre-process the messages on the Controller Area Network (CAN) and analyzed by the Bayesian network (Pascale et al., 2021).

Hesitant fuzzy-based- (AHP-TOPSIS) Analytical Hierarchical Process and Technique for Order of Preferences by Similarity to Ideal-Solutions has the capability to solve MCDM (Multi-criteria decision-making) problems, which are caused by imprecise and uncertain data (Alharbi et al., 2021), it enhances secure and reliable intrusion detection model and effectively analyze the types of attacks in the network. (Haider et al., 2021) developed Real-Time Sequential Deep Extreme Learning Machine Cybersecurity Intrusion Detection System (RTS-DELM-CSIDS) to classify several types of attacks in intrusion networks; this model can suppress security issues. More attributes are collected from both open-source datasets and real-time data. The feature selection can be performed by the combination of filter and wrapper method called cuttlefish algorithm (CFA) used for removing undesirable and irrelevant features from the original dataset (Mohammadi et al., 2019).

1.1. Motivation

Several existing models are analyzed, which are related to intrusion detection mechanisms. The existing models provide better performances, but it is unable to detect intrusion accurately due to some limitations. The limitations in existing models include consuming more time to detect the types of attacks, not applicable for large-scale AMI networks, and the model using low dimensional security data. These issues are motivated by this proposed model to suppress the limitations. Thus, an Efficient Feature Selection Assisted Network Intrusion Detection System using Machine Learning is used to improve security and avoid attacks in networks.

1.2. Contribution

The existing models, like traditional SmoT (Synthetic Minority Oversampling Technique), don't have the capability to solve two minority nearest neighbor samples placed at a long distance and the outliers are removed. The SVD (Singular Value Decomposition) model extracts basic features from the input data. Several types of attacks are not classified easily by the single classifier model, like LR, RF and SVM with large amounts of datasets. These limitations are suppressed by providing a novel hybrid ML algorithm. *Some of the major contributions of the proposed model are described as follows.*

- To remove missing values, irrelevant parameters, and unnormalized data can be adjusted by Null value handling and value standardization using Min-Max normalization in the pre-processing stage.
- To solve the class imbalance problem in the pre-processed data using the Advanced Synthetic Minority Oversampling Technique (ASmoT).
- To extract the features from balanced data using Modified Singular value decomposition (M-SvD),
- To select optimal features and reduce feature dimensionality issues using the Opposition-based Northern Goshawk optimization (ONgO) algorithm.
- To construct a hybrid machine learning model called Mud ring assisted multilayer support vector machine (M-MultiSVM) to classify the different attacks.

The traditional SmoT algorithm has solved the minority class problems, but there are some limitations, like the two minority nearest neighbor samples are far and the outliers are removed. Here, ASmoT enhances the data samples and provides overfitting issues by removing overfitting problems, producing an efficient intrusion detection classification model. From the balanced data obtained from ASmoT, the next stage is to collect the required features using the SVD method. This model doesn't extract the required features. Thus, the M-SvD model extracts the required features accurately, which is used for the classification stage. Opposition-based learning is used to improve speed in the convergence analysis, and the optimization algorithm NGO selects optimal features from the input data. Hybrid ML model SVM and MLP are utilized in the classification phase. SVM helps to improve performances and reduce error rates in classification. MLP can handle large amounts of data and predict the training set quickly. Finally, the hyperparameters are tuned by the Mud Ring optimizer to improve the overall accuracy of the model.

The considerations in the research paper are systematically examined to find the answer for the following questions.

a. Which type of pre-processing technique is used to remove irrelevant parameters and missing values?
b. How many datasets are collected to classify types of intrusion in the network?
c. How do you select optimal features from the extracted features?
d. What are the performance metrics analyzed for the proposed M-MultiSVM classifier model?
e. How to solve the class imbalance problem in the pre-processed data input?

In this paper, five sections are prearranged as surveys: Section 1 depicts basic information about the technologies used for intrusion detection. Section 2 deliberates various related works of Network Intrusion Detection System techniques. Section 3 explains the ML-based proposed methodology for an automatic intrusion detection system. Section 4 contains the result and discussion; Section 5 describes the overall conclusion, future work and references.

2. Related works

Some recent research on intrusion detection systems using machine learning methodology is surveyed as follows.

Dahiya et al. (2022) developed the Regularized Long Short-Term Memory (HT-RLSTM) framework to control and manage security in cyber security. HT-RLSTM framework is designed based on hyperparameter tuning. The method was trained and tested to detect the

various attacks. Poor scaling, overlapped data, and missing values caused incomplete data; these mistakes were addressed, and these missing values were controlled by Kernelized Robust Scaler (KRS). The framework was evaluated using different datasets and attained better specificity (94.72 %) and a low false rate to detect the attacks. The HT-RLSTM framework took a long computational time to detect the intrusion and was not able to prevent physical tampering and theft by opponents.

Mishra et al. (2022) developed the Optimized Gradient Boost Decision Trees (OGBDTs) and Genetic Algorithms (GAs) framework to create the IDSs. Using the African Buffalo Optimizations (EABOs), the classification was increased. Standardization, pre-processing, feature selection and data exploration were the modules of this framework. OGBDTs and GAs framework increased the intrusion detection in cyber security, which was evaluated by the CICIDS2018, UNBS-NB 15 and KDD 99 datasets. The framework provides better accuracy of intrusion detection, such as 99.8 %, 98.6 % and 99.1 % in KDD 99, UNBS-NB 15 and CICIDS2018 datasets, respectively. This method could not predict the types of attacks, and very few security parameters were analyzed.

Hemanand et al. (2023) proposed the Logistics Decision Support Vector (LDSV) system to detect network cyber-attacks. LDSV method analyses the behavior features and finds the type of cyber-attack. The LDSV method used the KDD Cup 99 dataset to generate intrusion detection; then, the data was pre-processed after the feature selection. Finally, cyber-attacks were classified using this method. This method was measured based on accuracy, recall, precision and F-Measure, and it attained 80 % accuracy and 74 % specificity in cyber security. The limitation of an LDSV method was less accuracy in induction detection.

Sarker et al. (2021) proposed the cyber security model with the fusion of feature selection and security model based on ML. In this Cyber Learning, binary classification was used to detect the anomalies and to analyze the different kinds of attacks; the multi-class classification model was used in this method. Ten classification models provided the security and security model based on artificial neural networks. The model is tested on the NSL-KDD and UNSW-NB15 datasets, where the KNN classifier achieves an accuracy of 99 %. The limitation of this model was that it used low-impacted security data.

Sun et al. (2020) developed a two-phase model for cyber intrusion protection techniques to detect cyber-attacks in Advanced Metering Infrastructures (AMI). In the first phase, an SVM identified the doubtful behaviours inside the smart meter. The second phase event attack route was created using a Temporal Failure Propagation Graph (TFPG). The model was trained and tested using normal and attack data from several datasets, and it achieved an accuracy of 98.7 %. The limitation of this model was unsuitable for large-scale AMI networks. Table 1 describes the various existing methods of surveys.

The review of current approaches reveals shortcomings, minimum accuracy, less capacity in feature learning, and require more computational time. Additionally, the system's overall performance was poor, and a hybrid machine-learning model was used to classify the networks. Therefore, the suggested model offers a new technique to resolve the automatic intrusion detection system.

Table 1
Survey of existing approaches.

Author name and Reference	Techniques used	Dataset used	Performance	Demerits
Dahiya et al. (2022)	HT-RLSTM	NSL-KDDCUP99 and ISCX intrusion detection dataset.	Specificity-94.7% Sensitivity-95.4%	• More computational time. • Not able to prevent physical tampering and theft by opponents.
Mishra et al. (2022)	OGBDTs-IDs	CICIDS2018, UNBS-NB 15 and KDD 99 dataset	Accuracy 99.8%-KDD 99 98.6%-UNBS-NB 15 99.1%-CICIDS2018	• Cannot find the type of cyber-attack. • Very small security attributes are analyzed.
Hemanand et al. (2023)	LDSV	KDD Cup 99 dataset	80% of accuracy 74% of the specificity	• Less accuracy of 70% can obtained.
Sarker et al. (2021)	Fusion of feature selection and security model based on ML	NSL-KDD and UNSW-NB15 dataset	Accuracy-99%	• Used low-dimensional security data.
Sun et al. (2020)	SVM and TFPG	Real-time	Accuracy-98.7%	• Not applicable to large-scale AMI network

3. Proposed methodology

Automatic abnormality detection systems are required to secure computing ability and analyze the attacks. Hence, an efficient automatically detecting intrusion in the system using machine learning methodology is proposed in this research work. The workflow of a proposed efficient automated intrusion detection system using machine learning methodology is depicted in Fig. 1. The novelty of the research work is explained in the following. The class imbalance problem has been reduced by ASmoT. The traditional SmoT algorithm has solved the minority class problems, but there are some limitations, like the two minority nearest neighbor samples are far and the outliers are removed, thus enhancing the imbalance class problems by introducing the ASmoT algorithm. The SVD method is used to extract features from data-balanced input data. This model doesn't extract the required features; thus, the exact features can be selected from the input using M-SvD. The feature selection can be performed efficiently by combining the Northern Goshawk algorithm with Opposition-based learning. Opposition-based learning is used to speed up the convergence rate. Thus, the combined opposition-based Northern Goshawk optimization (ONgO) algorithm selects optimal features to provide better classification results. In the classification phase, hybrid machine learning is used to improve results. SVM and MLP are integrated to provide accurate results in the classification for various types of attacks. The single classifier model is not able to handle large datasets and doesn't predict the attacks accurately. Thus, the hybrid model is used to overcome the issues and enhance the performance of the models. Several attacks are classified by a single classifier model, which is very complex. These issues are suppressed by hyperparameter tuning using the Mud Ring Optimization algorithm.

Initially, the data are collected from CSE-CIC-IDS 2018 and UNSW-NB15 datasets. The acquired data are pre-processed using Null value handling and value standardization using Min-Max normalization. The missing values and irrelevant data are reduced by Null values handling, and the unnormalized data are normalized by Min-Max normalization. Class imbalance problems in the input data are reduced by the ASmoT technique for reducing overfitting issues in the classifier model. Here, the various types of features like basic, content, and traffic features have been extracted by M-SvD. These extracted features are optimized by the ONgO algorithm. After feature selection, different types of attacks in the network are classified by a hybrid machine learning algorithm called M-MultiSVM, and the hyperparameters in the model have been tuned by the Mud Ring optimization algorithm. The novel M-MultiSVM model is used to classify the several types of attacks.

3.1. Pre-Processing

Data preparation is a technique that prepares raw data that should be

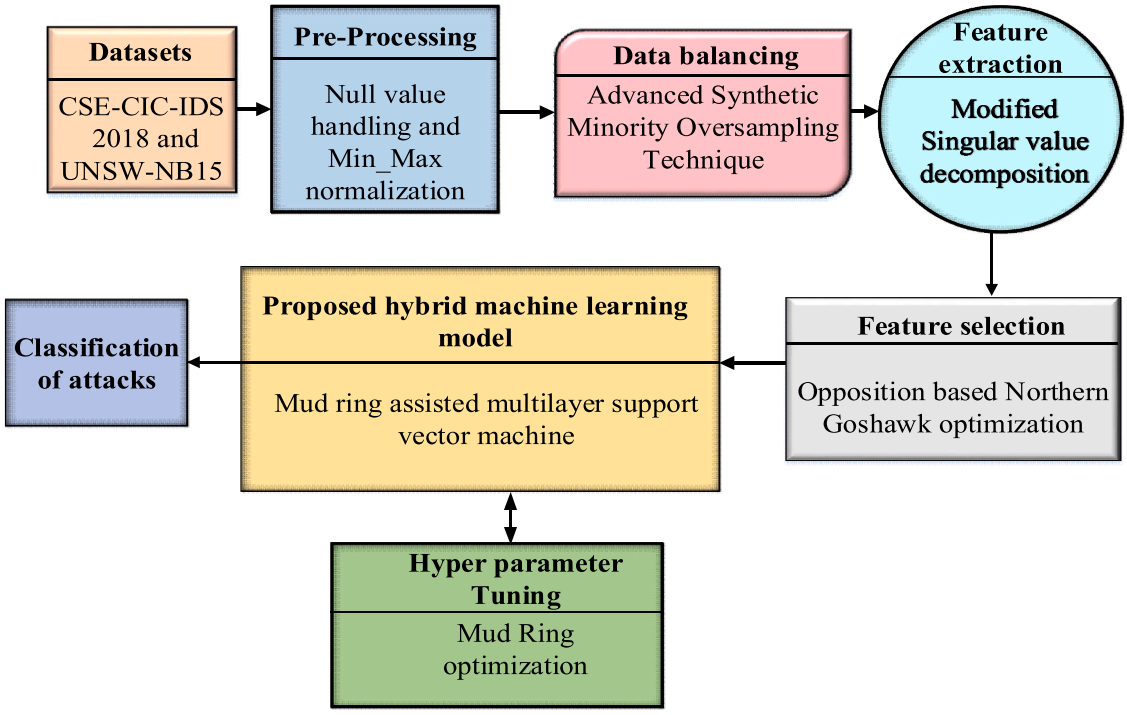

Fig. 1. Architecture of proposed model.

acceptable for the ML model. The main important step is to build a machine learning model. Building an ML model is the first and most important stage. The proposed method uses the following techniques to pre-process the given dataset.

3.1.1. Null value handling

By removing missing values, irrelevant parameters, etc., data is cleaned and filtered. The secret to reducing the dimensions of a data set is data cleaning. As the dimension increases, more processing time and power are required (Lalwani et al., 2022).

3.1.2. Value standardization using min-max normalization

Min-Max standardization is a broadly recognized method to standardize data. The technique linearly adjusts the unnormalized data to predetermined lower and higher boundaries. Each component of the extreme value is changed into 1, the base estimation of an element is changed to 0, and the decimal values are altered between 0 and 1 for other values (Raju et al., 2020).

$$S = \frac{(S - S_{\min})}{(S_{\max} - S_{\min})} \tag{1}$$

Here, $S_{\min}$represents the minimum value of S feature and $S_{\max}$denotes the maximum value for Sfeature.

3.2. Advanced Synthetic Minority Oversampling Technique (ASmoT)

The SMOTE algorithm improves the number of low-frequency samples by conducting a random and linear interpolation between low-frequency samples and their congeneric closest neighbor (Alshamy et al., 2021). The SMOTE technique not only ignores the issue of handling outliers and the spatial distribution dispersion of the low-frequency sample but also raises the total amount of low samples, causing some computational resources to be wasted.

Step 1:

a) If $(K_n = K)$it denotes no congeneric examples in the K nearest neighbors of the low-frequency sampley. It is known as the Independent Point Region.
b) If $(K/2 < K_n < K)$ it suggests that in the closest neighbors, there are more high-frequency samples than low-frequency samples. If the low-frequency samples in this phase are dangerous, then it is called a Dangerous Point Region.
c) If$(0 < K_n < K)$ it shows fewer high-frequency samples than low-frequency samples in the closest neighbor. The low-frequency samples in this phase are secure points called Safety Point Region.

Step 2:

a) Contrary to applications like image processing, low-frequency samples from IPR should not be overlooked in intrusion detection as noise. Consequently, like the SMOTE algorithm, a new sample is produced between the low-frequency sample and its closest neighbor high-frequency samples by equation (2),

$$y_{new} = y + v_{[0,1]}\,(y_n - y) \tag{2}$$

where y_{new} is the newly generated low-frequency sample, $y_h$is the high-frequency sample in IPR, and $v_{[0,1]} \in [0,1]$is a random number.

a) As a result, the spatial dispersion of a low-frequency sample set is high, and low-frequency samples that are part of the DPR are mixed with high-frequency samples. The original low-frequency samples are used to create the new samples, and the new samples in the IPR and DPR have a linear relationship. The generation rules for creating frequency samples are described in Eqs. (3) and (4):

$$y_m = \frac{1}{K - K_n} \sum_{i=1}^{K-K_n} y_i \tag{3}$$

$$y_{new} = y + v_{[0,1]}\,(y_m - y) \tag{4}$$

Here, y_m denotes all low-frequency samples in DPR.

Step 3:

The temporal relation between new and original images is unrealistic when using the ASmoT algorithm based on the actual samples. The traditional SmoT algorithm had the capability to reduce minority class problems, but it has some limitations, like two minority nearest neighbor samples are far, and the outliers are removed. Thus, the ASmoT model is used to balance the classes in input by decreasing the overfitting problems in the model instead of the traditional SmoT algorithm. ASmoT algorithm is used to adjust the imbalance ratio in the dataset using majority values by using majority values to change the minor values to set the class samples in a balanced ratio. The dataset samples can be balanced by ASmoT, which reduces the overfitting issues in classification and improves the performance of the model.

3.3. Feature extraction

The conversion of original raw data into numerical features while preserving the information in the original dataset is called feature extraction. This paper used the well-known M-SvD technique.

3.3.1. Modified Singular Value Decomposition (M-SvD)

The M-SVD theory has been developed in several papers, including this paper. Calculating kurtosis from Hilbert transform (HT) using SEK indicator and efficient reconstruction order of SVD is chosen (Ugwu et al., 2021).

The original discrete data is$A = [a\ (1), a\ (2).....,a\ (n)]$, and the data is subjected to M-SVD. Then, $\lambda_1 \geq \lambda_2 \geq .... \geq \lambda_k \geq 0$can be calculated and next called the reconstructed order$g\ (g = 1,\ 2.....k)$. The following are the steps: The envelope of the first $i-$ thorder reconstructed data is obtained first, followed by the envelope kurtosis of this first i $-$ thorder, and finally, the envelope kurtosis value of each reconstructed order $i-th$ is equated to determine the optimal reconstructed order with the highest envelope kurtosis value. The following equation envelopes the first i $-$ thorder rebuilt data.

$$a\,t_i^g = \left|\frac{1}{\pi p}.a_i^g\ (p)\right| \quad g = 1, 2, .k \tag{5}$$

Here, the first$g$-order data's reconstructed data is $a_i^g\ (t)$ and $a\,t_i^g$ is the absolute value. Logically, the first$g$-order reconstructed data's envelope kurtosis value can be determined as,

$$dek_i\ (g) = \frac{D(at_i^g - \mu(at_i^g))^4}{\sigma(at_i^g)^4}. \quad g = 1, 2, ...k \tag{6}$$

Here, $\mu(a\,t_i^g)$ stand for the $a\,t_i^g$mean and $\sigma(a\,t_i^g)$ are the standard deviation.

$$SEK_g = \frac{dek_i(g) - dek_i(g+1)}{dek_i(g+1)} \quad (g = 1, 2, \ldots k-1) \tag{7}$$

Here, $dek_i\ (g)$ is an envelope kurtosis value of the reconstructed data of first$g$ -order and k is the number of singular values. Then, for each order$g$, calculateSEK_g and SEK_g, so the highest absolute value can be obtained. It is presumed that the first g-order reconstructed data yields the maximum absolute value of SEK_g and which is recordable as the following equation

$$SEK_{g.\max} = \frac{dek_i(g) - dek_i(g+1)}{dek_i(g+1)} \quad (g = 1, 2, ...k-1) \tag{8}$$

Equation (8) is described as possible to assume that the best way to determine the M-SVD model's optimal order SEK_g is denoted as an absolute maximum$SEK_{g.\max}$. The following two cases must be discussed to explain the SEK better to select the reconstructed order. The resulting formula is shown below as,

$$SEK_{g.\max} = \begin{cases} \dfrac{dek_i(g) - dek_i(g+1)}{dek_i(g+1)} \\ \dfrac{dek_i(g) - dek_i(g+1)}{dek_i(g+1)} \end{cases} \quad (g = 1, 2, ...k-1). \tag{9}$$

Further developments of Eq. (9) are as follows: 1) If $dek_i(g) > dek_i(g+1)$, or $dek_i(g) - dek_i(g+1) > 0$ then, the answer is positive. The positive value of Eq. (8) illustrates how the$SEK_{g.\max}$ arrives, and it denotes that the greatest abrupt drop occurs at the gth order. As a result, the gth order is regarded as the best-reconstructed order. 2) If $dek_i(g) < dek_i(g+1)$ that is$dek_i(g) - dek_i(g+1) > 0$. The negative value yields the$SEK_{g.\max}$, as shown in Eq. (9), and it means that the first $g+1$th order is chosen as the ideal order because the envelope kurtosis value of the ith order is significantly smaller than that of the$g+1$ th order. Using the SEK indicator, the optimal reconstructed order of SVD can be effectively calculated, i.e., the M-SVD is achieved. The SVD method is used to extract features from input data. This model doesn't extract the required features; thus, the exact features can be selected from input using M-SvD. Thus, M-SvD is used instead of SVD in the feature extraction phase. M-SvD model is used to extract basic features, content features and traffic features from the balanced data.

3.4. Feature selection

Feature Selection is a technique used in the Opposition learning-based Northern Goshawk optimization (ONgO) to reduce the model's input variable and extract optimal features from the extracted features.

3.4.1. Opposition-based Northern Goshawk Optimization (ONgO)

The Northern Goshawk Optimization (NGO) algorithm is used to select relevant features from the input data for effective intrusion detection. (Helliwell et al., 2022; Desuky et al., 2022). The suggested NGO algorithm determines the population matrix.

$$G = \begin{bmatrix} G_1 \\ \vdots \\ G_2 \\ \vdots \\ G_3 \end{bmatrix}_{M\times n} = \begin{bmatrix} g_{1.1} & \cdots & g_{1.j} & \cdots & g_{1.m} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ g_{t,1} & \cdots & g_{t.j} & \cdots & g_{t.m} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ g_{N.1} & \cdots & g_{N.j} & \cdots & g_{N.m} \end{bmatrix}_{U\times v} \tag{10}$$

Here, G is denoted as the population of hawks, g_t is denoted as ith offered solution, g_{t-J} and $g_{t.i}$ is denoted as jth proposed solution. Population size is denoted as U, and v is the number of variables in the problem. Consequently, it is possible to evaluate the problem's objective function for each member of the population, and it is mathematically expressed in the following equation.

$$K(G) = \begin{bmatrix} K_1 & = & K(G_1) \\ & \vdots & \\ K_t & = & K(G_t) \\ & \vdots & \\ K_U & = & K(G_U) \end{bmatrix}_{U\times 1} \tag{11}$$

Here, K is a values vector for the objective function and K_t is the value of an objective function obtained by ani^{th}proposed solution. The primary behaviours of goshawks are (i) prey identification and attack and (ii) chase and escape operation, which were replicated twice in this strategy.

3.4.2. Prey identification and attack

The initial phase concepts are arithmetically represented using the following equations,

$$f_i = G_k,\ i = 1, 2, ...U, d = 1,\ 2, ...i-1, i+1....U. \tag{12}$$

$$g_{i,j}^{mewf1} = \begin{cases} g_{i,j} & + & w\left(f_{i,j} - If_{i,j}\right),\ K_{f1} & < & F_i \\ g_{i,j} & + & w\left(g_{i,j} - f_{i,j}\right),\ K_{f1} & \geq & F_i \end{cases} \tag{13}$$

$$G_i = \begin{cases} G_i^{new.f1} & K_i^{new.f1} & < & K_i, \\ g_i & K_i^{new.f1} & \geq & K_i, \end{cases} \tag{14}$$

Here, f_i is the position of prey for i th goshawk, K_{f1} is the objective function value, l is a random natural number in the interval $[1, U]$, $g_j^{new.f1}$ is a new status for i th proposed solution, and $K_i^{new.f1}$ is an objective function value based on the first phase of NGO, r *is* the random number in the interval [0,1], I is a random number that can be 1 or 2. Random numbers are generated using the search and update parameters r and I to create a random NGO character.

3.4.3. Chase and escape operation

In the suggested GO algorithm, using the radius A to hunt is presumably close to an attack position. The second phase's models are represented mathematically in the following equation.

$$g_{i,j}^{new.f2} = g_{i,j} + L(2r-1)g_{i,j}, \tag{15}$$

Here, $L = 0.02\left(1 - \frac{t}{T}\right)$

$$G_i = \begin{cases} G_i^{new.f2}, & K_i^{new.f2} & < & K_i \\ G_i, & K_i^{new.f2} & \geq & K_i \end{cases} \tag{16}$$

Where, t is an iteration counter, T is the maximum number of iterations, $g_i^{new.f2}$ is a new status for i th proposed solution, $g_j^{new.f2}$ is jth dimension, and the objective function value based on the second phase of NGO. The NGO promotes opposition-based learning (OPL), a technique that improves and increases the likelihood of locating a potentially successful area.

The initial solutions are typically generated randomly when the optimization OPL algorithm tries to solve a problem with the best solution. Optimization aims to minimize a mismatch between the optimal and starting solutions. Let $r$is a real number in the range of m and$n$ $(m \leq n)$, the opposite number of $\breve{e}$and e can be presented in the subsequent equation;

$$\breve{e} = m + n - e \tag{17}$$

Higher dimensions can also use the definition of the Contract $z = \{r_1, r_2, ...r_c\}$ be an C-dimensional search space point and $z_x (1 \leq x \leq H)$ is the range of $m_x$andn_y. The following equation can be used to calculate the opposite point $\breve{z} = \{\breve{e}_1, \breve{e}_2, ..\breve{e}_c\}$.

$$\breve{r}_x = m_x + n_y - m_x \tag{18}$$

Eqs. (7) and (8) state that the random guess and its inverse can generate a higher possibility. The ONgO algorithm has been described in Table 2.

Here, the feature selection can be performed by combining NGO and Opposition based learning. The NGO is used to select optimal features which are required for attack classification. Opposition-based learning has improved the speed of the convergence. The optimal features are gathered accurately at low consumption of time if the convergence rate is increased. Thus, the Combined ONgO algorithm chooses optimal features and removes irrelevant data.

3.5. Mud ring-assisted multilayer support vector machine (M-MultiSVM)

The M-MultiSVM classifier worked based on the patch size pixel-wise classification. In this method, the MultiSVM neural network contains three hidden layers, each consisting of 500, 350 and 250 neurons, respectively. The final hidden layer of MLP consists of many important

Table 2
ONgO algorithm.

Start
Enter the details of the optimization problem.
Decide on the population's size (U) and how many iterations there are (T).
Beginning of the northern goshawk positions and assessment of the objective function.
For *t=1:T*
For *i=1:U*
Step 1: investigation stage
Select the prey at using a random Eq. (11).
For j=1:m
Calculate using equation, the new status of the *j*th dimension (12).
End j=1:m
Update *i*th population member using Eq. (13).
Step 2: phase of exploitation
Update L using Eq. (14).
For *j=1:m*
Calculate new status of *j*th dimension using Eq. (15).
End for *i=1:m*
Update *i*th populace participant utilizing Eq. (16).
End for *i=1:U*
Save best proposed solution so far.
End for *t=1:T*
Using the OBL algorithm, you can increase the likelihood of discovering a potentially promising region.
Update the best position in Eqs. (17) and (18)
Return *C**(the best position)
Stop

features for every training data, which is used to attain a more accurate outcome. Using these features, the SVM further classify to improve the classification performance. Fig. 2 describes the proposed multilayer support vector machine architecture.

Traditional SVM techniques need the quadratic programming (QP) package to overcome the QR problem, but it takes a long time and numerical analysis skills are required for huge memory. r_k and $s_k \in \{-1,\ +1\}$ are an input vector and binary label, the corresponding classification problem of a dataset is represented as (r_1, s_1), (r_h, s_h). QR problem dual form is mentioned below:

$$\max_{\alpha} \sum_{k=1}^{h} \alpha_k - \frac{1}{2} \sum_{k=1}^{h} \sum_{j=1}^{m} q_k q_j G\left(p_k,\ p_j\right) \alpha_k \alpha_j \tag{19}$$

$$0 \leq \alpha_k \leq S, FOR\ \ k = 1,\ 2,\ ....., h, \quad \sum_{k=1}^{h} q_k \alpha_k = 0 \tag{20}$$

The consecutive minimal optimization requires creating a way for QR and neglecting to work through numerical QR. This problem is reduced by using the α_1 and α_2 multipliers.

$$S \geq \alpha_1,\ \alpha_2 \geq 0 \tag{21}$$

$$r_1 \alpha_1 + r_2 \alpha_j = D \tag{22}$$

Using the detection of less one-dimensional quadratic function, the reduced problem is resolved analytically. In the equality constraint, the negative sum of the remaining terms is assigned to a fixed variable and used in every iteration. The phases of this algorithm are given below:

1 Discovery of a first Lagrange multiplier (α_1), which does not follow the condition of the Karush–Kuhn–Tucker (KKT) optimization problem.
2 Choice the $\alpha_2$multiplier and enhance the ($\alpha_1,\ \alpha_2$) pair.
3 Process 1 and 2 is repeated till convergence.

The QP problem is removed when the Lagrange multiplier fulfills the KKT condition. The multipliers pair is chosen by heuristic measure and guarantees convergence, which increases the convergence rate.

MLP algorithm of an artificial neural network has been employed as a classification technique. Three layers are present in MLP classifiers: input, hidden, and output. There are many neurons in each layer, and

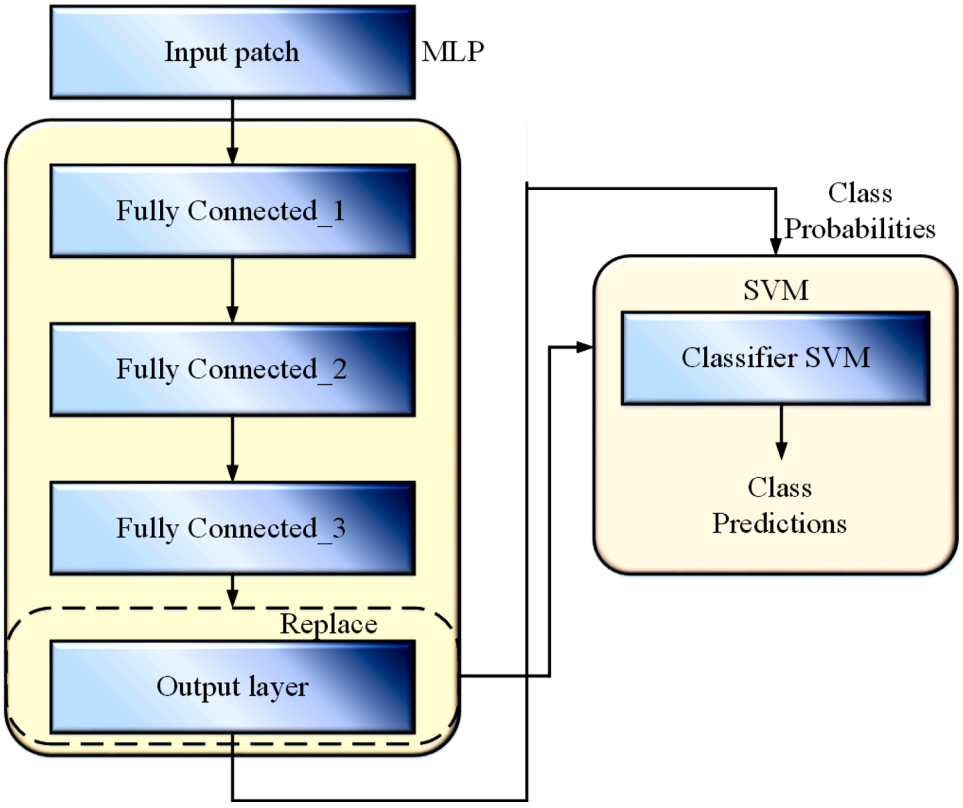

Fig. 2. Architecture of multilayer support vector machine.

the direct learning process of MLP provides several classes. The training process of backpropagation is used to compute the optimal weight. Fig. 3 represents the architectures of MLP for intrusion detection.

Mud ring techniques take the recent best solution as the target prey since the search space optimal design location is not identified earlier. The dolphins updated their positions and moved to the location of the best dolphin after choosing the best search agent. The dolphin behaviours are described using the following equation:

$$\vec{L} = \left| \vec{P}\,\vec{Q}^{*s-1} - \vec{A}^{s-1} \right| \vec{Q}^{*} \tag{23}$$

$$\vec{Q}^{s} = \vec{Q}^{*s-1} \cdot \sin(2\pi r) - \vec{U} \cdot \vec{L} \tag{24}$$

where, the recent time step and the random number are represented as s and $r$respectively. Coefficient vectors are represented as $\vec{P}$ and $\vec{U}$. The

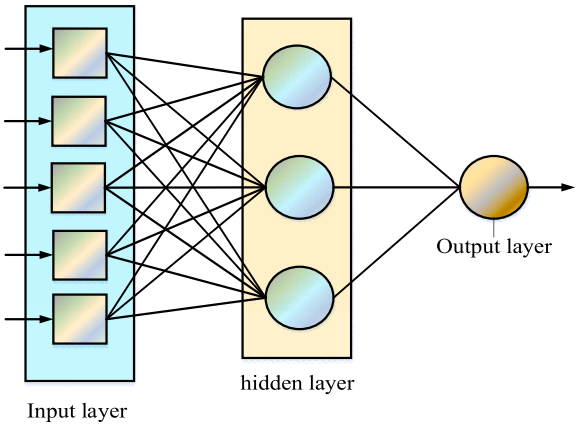

Fig. 3. MLP architecture for intrusion detection.

Position vector of the dolphin is denoted as $\vec{Q}$ and yet the best dolphin location is represented using position vector such as $\vec{Q}^{*}$. In every time step, the $\vec{Q}^{*}$ is monitored and modified to obtain a better position. The other dolphins circle the prey as the best dolphin rapidly waves its tail in the sand to create a sine wave that emits a plume. The coefficient vector such as $\vec{P}$ considered using the following Calculation:

$$\vec{P} = 2.\vec{w} \tag{25}$$

The Pseudo code of a Mud Ring optimization algorithm is shown in Table 3.

In the search area, any position is obtained by defining the random

Table 3
Mud ring optimization algorithm.

Start
Initially fixed the random population of dolphins, $Q_j, j \in [1, 2, \ldots\ldots, m]$ and velocity v_j
Calculate each dolphin's Fitness Function
Assign the greatest dolphin location to Q^{*}.
While $(s < S\_\max)$
for $j = 1$ *to* m
Change P, U, r *and* l
If $|\vec{U}| \geq 1$then
Change the velocity v_j and creating the new solution
Else
Update the recent dolphin location through Eq. (25)
End if
End for
Change the bounds for outside dolphins from the search space
Achieve the Fitness Function of Dolphins
Update existence Better position using Q^{*}
Set $s \rightarrow s + 1$
End while
Return best position Q^{*}
Stop

vector $\vec{w}$. Eq. (25) refers to the encircling of prey and justifies its location. The mud ring begins its search procedure with a collection of random outcomes (location of dolphins). Dolphins justify their places based on the best location or with a randomly selected dolphin in each time step. Accordingly, the parameter based on the time step is transmission among the exploitation and exploration. The dolphin's best position and random dolphin are selected when $|\vec{U}| < 1$ and $|\vec{U}| \geq 1$. The random dolphin is useful to validate the dolphin's location and the parameter P and U is only required to modify in the Mud ring algorithm. Through the Mud ring optimization algorithm, high efficiency is attained in classification. The novelty of the classification model is combining two ML algorithms and adding optimization to improve the efficiency of the model. The traditional single classifier model is not able to classify the attacks accurately and is too complex to handle a large amount of data. To overcome these issues, a new M-MultiSVM is proposed to classify the types of attacks. Here, Multi-SVM is used to identify the several types of attacks, and the hyperparameters are tuned by the Mud Ring optimizer.

4. Results and discussion

This division describes the various investigational analyses for existing methods and proposed methods. This section analyses and compares various performance metrics with other existing systems. The proposed method uses two types of datasets from open source, CSE-CIC-IDS 2018 and UNSW-NB15, to identify the types of attacks such as Brute-force, Heart Bleed, Botnet, DoS, Distributed DoS (DDoS), Web attacks, network infiltration, Fuzzers, Analysis, Backdoors, Exploits, Generic, Reconnaissance, Shell code and Worms in any network. Various performance metrics such as accuracy, Detection Rate (DR.....), False Alarm Rate (FAR), precision, F1Score, training time, specificity and Mathews Correlation coefficient (MCC) are measured for the CIC-IDS 2018 dataset and Accuracy, Precision, Recall, F1Score, False Alarm Rate (FAR), Training time and False Positive Rate (FPR) are analyzed for UNSW-NB15. These performance metrics are analyzed for various existing systems such as Decision Tree (DT), Logistic Regression (LR), Random Forest (RF), K-Nearest Neighboring (KNN), AdaBoost, XGBoost (Extreme Gradient Boosting), Light Gradient Boosting Machine (Light GBM), Multilayer Perceptron (MLP), HBGD (Histogram Gradient Boosting) ET (Extra Trees) and proposed method based on CIC-IDS 2018 dataset. For the UNSW-NB15 dataset, the performance metrics are analyzed for various existing systems such as DT, AdaBoost, MLP, Long Short Term Memory Network (LSTM), Gated Recurrent Units (GRU) and the proposed method. These performances are analyzed and compared with various existing and proposed methods discussed in this section. The Hyperparameter details are described in the following Table 4.

Table 4
Hyperparameter details.

Hyperparameters	Values
Number of CNN layers	22
Epochs	300
Batch size	(256,256,3)
Learning Rate	0.0001
Batch size	16
Activation function	Sigmoid, ReLU
Loss function	Mean Absolute Error
Dropout	0.5
Optimizer	Mud ring, Opposition-based Northern Goshawk optimization algorithm
MLP hidden layer size	(50,50,50)
Solver	Adam

4.1. Dataset description

The proposed method uses two datasets, CIC-IDS 2018 and UNSW-NB15, whereas the description is as follows.

4.1.1. CIC-IDS 2018

Seven different attack scenarios, including brute force, Heart Bleed, Botnet, DoS, DDoS, Web attacks, and network infiltration, are included in the CIC-IDS 2018 dataset. Brute force can be conducted by the following tools, namely Hydra, Medusa, Ncrack, Nmap NSE (N-map Scripting Engine) scripts and Metasploit modules. In Heart Bleed attacks, the Heart leech tool is used to scan systems vulnerable to the bug. The web server performance is reduced by the Slowloris tool in the Dos attack. DDOS attack can be performed by HOIC (High Orbit Ion Cannon) using four different computers. Web attacks are conducted by Damn Vulnerable Web App (DVWA) is a type of PHP/MySQL web application which is vulnerable to web servers. The victim association has five departments, each with 420 machines and 30 servers, while the attacking infrastructure consists of 50 machines. This dataset includes network traffic that has been recorded, system logs for each machine, and 80 features that CICFlowMeter-V3 take out from the recorded traffic. The overall samples of the CICIDS 2018 dataset are split into the ratio of 80 % for training and 20 % for data testing, before and after sampling of the data sample, which is represented in the following Table 5.

The sample distribution of the dataset is described in the following Table 6.

4.1.2. UNSW-NB15

The IXIA perfect storm tool in the Cyber Range Lab of UNSW Canberra produced the UNSW-NB15 dataset to produce a hybrid of real contemporary normal activities and synthetic contemporary attack behaviours. Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shell code, and Worms are the nine types of attacks in the UNSW-NB15 dataset. Fuzzers are used to bring a program or network to a halt by feeding it randomly generated data. Analysis attack contains a variety of port search, malware and HTML file penetration attacks. Backdoors provide a method of gaining unauthorized access to a device by circumventing a system authentication process invisibly. Dos performs an interruption or halt to the services of a host connecting to the networks. Using the Argus and Bro-IDS tools, twelve algorithms are accustomed to creating features with class labels. The overall samples of the dataset are split into the ratio of 80 % for training and 20 % for testing. The data sample numbers for before and after oversampling are described in the following Table 7.

The distribution of the UNSW-NB 15 dataset is described in the following Table 8.

4.2. Performance metrics

The proposed method maintains various performance metrics are analyzed, such as Accuracy, Precision, Recall, F1 Score, FRP, MCC, DR..... and FAR, using both datasets CIC-IDS 2018 and UNSW-NB15 to improve and enhance the system to high quality. These metrics' mathematical equations are provided below:

4.2.1. Accuracy

Accuracy is the parameter used to measure class performance across various models. It is determined by dividing the number of accurate predictions by the total number of predictions.

Table 5
Before and after oversampling.

Dataset	Actual		Balanced	
CICIDS 2018	Benign 808,204	Malicious 217,489	Benign 808,204	Malicious 786,310

Table 6
Sample distribution.

Type	Number of Samples	
	Training	Testing
Benign	646,603	161,601
Bot	86,811	21,709
DoS attacks-SlowHTTPTest	34	7
DoS attacks-Hulk.	87,117	21,811

Table 7
Before and after oversampling.

Dataset	Actual		Balanced	
UNSW-NB 15	Benign	Malicious	Benign	Malicious
	93,000	164,673	136,905	164,673

Table 8
Sample distribution.

Type	Number of records	
	Training	Testing
Normal	56,000	37,000
Analysis	2000	677
Backdoor	1746	583
DoS	12,264	4089
Exploits	33,393	11,132
Fuzzers	18,184	6062
Generic	40,000	18,871
Reconnaissance	10,491	3496
Shell code	1133	378
Worms	130	44

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{26}$$

4.2.2. Precision

The precision parameter is used to measure the performance of a model by evaluating how often the perdition of a model is validated. When an instance is positively predicted, it can be mathematically expressed as in the below equation:

$$\frac{TP}{(TP + FP)} \tag{27}$$

4.2.3. Recall

The recall parameter is used to correctly detect true positive instances by the Machine Learning (ML) model. It also measures the accuracy and how to identify relevant data.

$$recall = \frac{TP}{(TP + FN)} \tag{28}$$

4.2.4. F1 score

F1 Score measures trade-off values between Recall and Precision, which considers FP and FN. It measures the overall accuracy of an ML model, and it can be expressed in the beneath equation;

$$F_1 Score = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{29}$$

4.2.5. FPR (False positive rate)

FRP parameter is measured as the ratio of misclassification using false positive and false negative. This can be represented in the below equation:

$$FPR = \frac{FP}{(FP + FN)} \tag{30}$$

4.2.6. MCC (Matthews correlation coefficient)

The MCC parameter summarizes the confusion matrix or error matrix with four entities: True Positive, True Negative, False Positive and False Negative.

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \tag{31}$$

4.2.7. DR..... (Detection rate)

DR..... parameter is used to compare the performance using the fault detection mechanism

$$DR = \frac{TP}{TP + FN} \tag{32}$$

4.2.8. FAR

FAR is used to measure many false alarms per total number of warnings or alarms expressed in the equation below;

$$FAR = \frac{FP}{FP + TN} \tag{33}$$

4.2.9. Specificity

The proportionality of the negative value is denoted as specificity. The specificity is calculated using the below equation.

$$Specificity = \frac{TN}{TN + FP} \tag{34}$$

4.3. Performance analysis

The performance is analyzed for various existing and proposed methods based on two datasets, namely CIC-IDS 2018 and UNSW-NB15.

4.3.1. Evaluation of CIC-IDS 2018 dataset (Zhang et al., 2023)

The performance metrics such as Detection Rate (DR.....), False Alarm Rate (FAR), precision, F1Score and Mathews Correlation coefficient (MCC) are analyzed and compared with various existing systems such as DT, LR, RF, KNN, AdaBoost, XGBoost, Light GBM and MLP and proposed method by using CIC-IDS 2018 dataset. Figs. 4(a) and (b) represent the accuracy and DR..... values in percentage to compare proposed and existing methods.

Fig. 4(a) compares various existing methods with the proposed model. The accuracy is analyzed in both existing and proposed methods. The overfitting issues that occur in the existing models lead to a decrease in the performance of the existing model. Thus, the existing can obtain less performance than the proposed model. The proposed method can obtain enhanced performance than other existing approaches. An accuracy of 99.89 % was obtained in the proposed method. In Fig. 4(b), the parameter DR..... is measured and compared with various existing systems. The proposed method can obtain 99.125 % of DR....., and it provides better results when compared to other existing methods. The existing models can obtain less accuracy because it takes more time to compute problems. Figs. 5(a) and (b) show that FAR and Precision values are measured for existing and proposed methods.

In Fig. 5(a), the proposed method can obtain a first-rank position over other existing models. The existing model can obtain higher values in analyzing error rates by the FAR parameter. Analyzing the proposed method's error rate using this parameter can obtain fewer values than other systems. Fig. 5(b) analyses the precision parameter for existing and proposed methods. The proposed method can obtain 99.914 % of precision values, and the existing method can obtain less than 97 %. Fig. 6(a) and (b) shows the F1 Score and MCC parameter analysis for existing and proposed methods.

Fig. 6(a) analyses the F1 Score parameter for existing systems and proposed models. The existing system can obtain fewer outcomes than the proposed model. The proposed method can obtain 99.241 % by

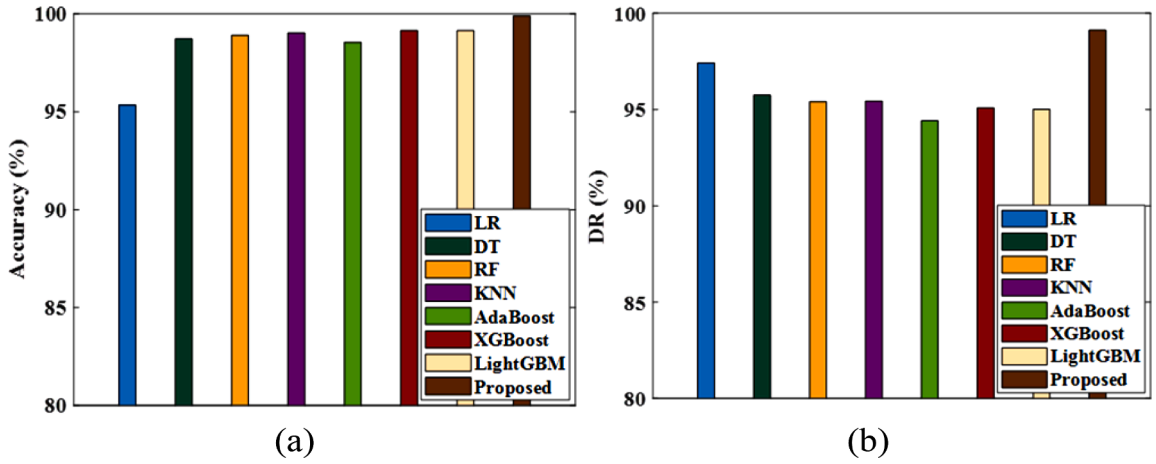

Fig. 4. Performance analysis (a) Accuracy (b) DR.

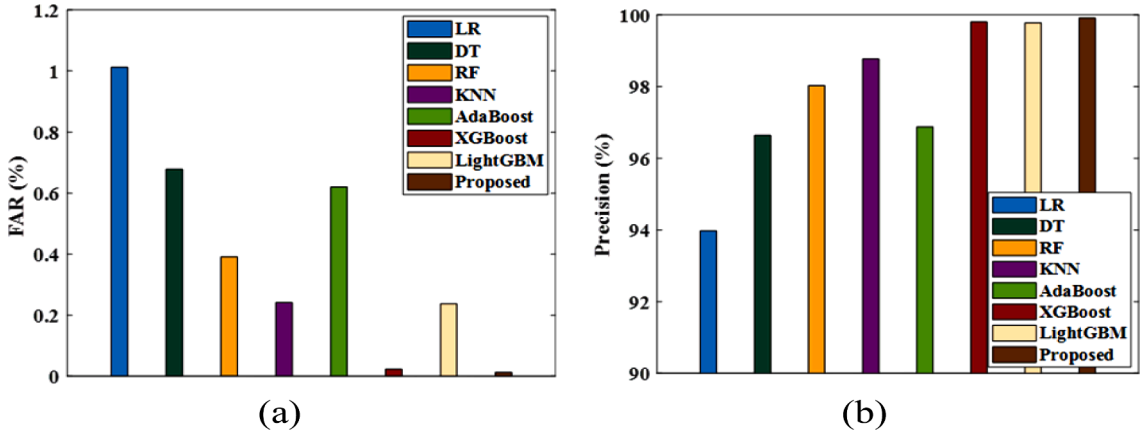

Fig. 5. Performance comparison (a) FAR, (b) precision.

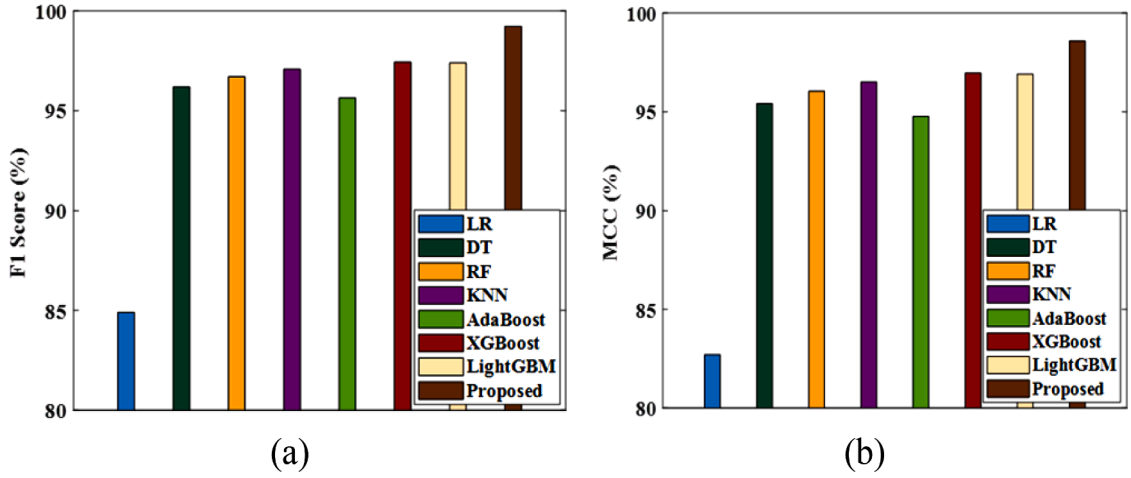

Fig. 6. Performance outcomes (a) F1Score (b) MCC.

using this F1 Score parameter. The better rate of an F1 Score leads to improved quality and performance in the proposed method. The existing system cannot identify attacks in dynamic situations because it obtains fewer values in the F1 Score. Fig. 6(b) measures the MCC parameter for various existing and proposed methods. The proposed method can obtain 98.584 % of MCC values. This parameter provides a huge amount of security and less maintenance cost. Thus, the proposed method has obtained a more efficient and high-quality system. Fig. 7(a) and (b) represents the specificity and training time for the several existing and proposed models.

Fig. 7(a) shows that the Specificity parameter is analyzed for the existing and proposed model. The existing models, such as HBGD (Histogram Gradient Boosting), ET (Extra Trees) and RF, can obtain values of 99.36 %, 90.26 % and 91.271 %. The proposed model can attain

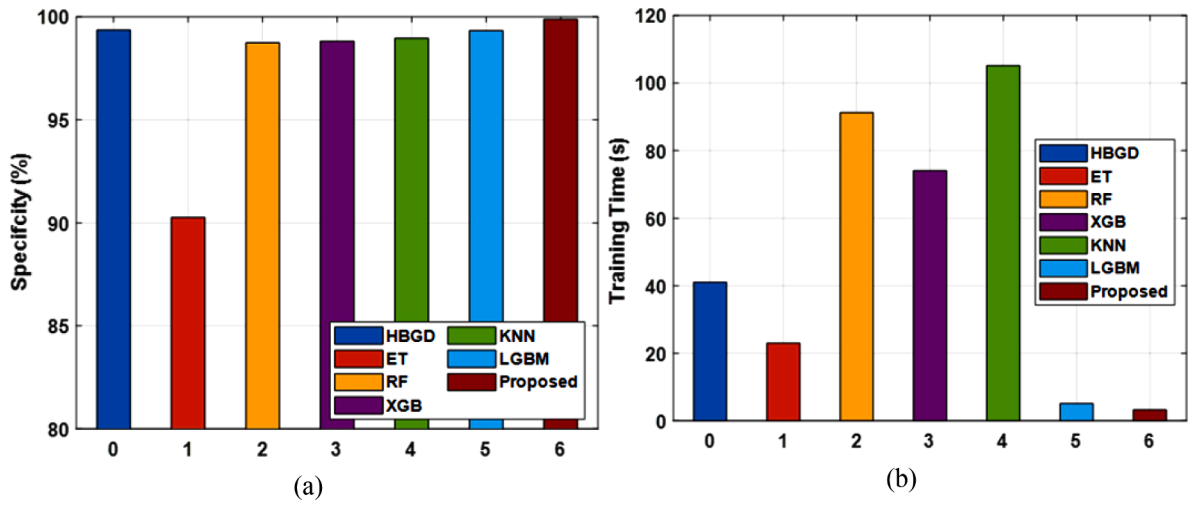

Fig. 7. Performance outcomes (a) specificity (b) training time (Seth et al., al.2021).

specificity values of 99.88 %. Fig. 7(b) denotes the outcomes of training time. The proposed can attain less amount of time is 3.21502541. In this analysis, the proposed model can obtain better results than other existing models. Comparison analysis for several existing models in machine learning algorithms is described in the following Table 9.

The above table describes the analysis of several performance metrics such as accuracy, precision, recall, f1-measure and prediction latency for existing models like HBGD, ET, RF, XGB, KNN, and LGBM (Light Gradient Boosting Machine) and proposed M-MultiSVM model. Thus, the suggested model can attain better accuracy than other existing ones.

4.3.2. Evaluation of UNSW-NB15 dataset (Disha et al. 2022)

The performance metrics such as Accuracy, Precision, Recall, F1Score, and False Positive Rate (FPR) are analyzed and compared with various existing methods such as DT, AdaBoost, MLP, Long short-term memory Network (LSTM), Gated Recurrent Units (GRU) and proposed method using UNSW-NB15 datasets. Figs. 8(a) and 7(b) represent accuracy and precision values calculated for existing and proposed methods.

In Fig. 8(a), the proposed method can obtain an accuracy of 97.535 %; the proposed method can obtain better accuracy than other existing methods. The existing system takes more time to solve a problem or does not handle complex problems. Due to the limitations in this existing system, it can obtain less accuracy value. Fig. 8(b) calculates the precision values for the existing system and the proposed method. The proposed method can obtain 97.674 % of precision values, and the existing system can obtain less than 95 %. Thus, the M-Multi-SVM can obtain an efficient system with these high values of parameters. The precision and accuracy values enhance the proposed system. Figs. 9(a) and 8(b) show the recall and F1 score analysis for existing systems and proposed methods.

Table 9
Comparison analysis of machine learning models for CIC-IDS 2018 dataset (Seth et al., al.2021).

Classifier models	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Prediction latency
HBGD	97.80	99.36	96.04	97.67	0.300503
ET	91.98	90.26	90.26	91.72	0.345334
RF	97.34	98.74	98.74	97.18	0.33835
XGB	96.97	98.82	98.82	96.77	0.182613
KNN	97.68	98.96	98.96	97.54	186.4265
LGBM	97.72	99.43	99.33	97.57	0.138008
Proposed	99.89	99.914	99.125	99.214	0.1264300

In Fig. 9(a), recall parameter values are analyzed and compared with the existing and proposed method. The proposed method can obtain 98.945 % of recall values, and other existing systems obtain values slightly different from the proposed method. The existing models are tuned properly or do not balance the class accurately; thus, they can obtain lower performances. Thus, the proposed method can develop an efficient system by obtaining high values in these parameters. Fig. 9(b) shows that the proposed method can attain 97.995 % of F1Score values. The existing models obtain lower performances due to high computational costs and take more time to detect intrusion in networks. The proposed method can achieve better performance and a more efficient system than existing models. Fig. 10 represents the performance outcomes of FAR (False Alarm Rate) and training time.

Fig. 10(a) represents the outcomes for FAR analyzed for an existing model, such as SVM obtaining fourth position in FAR analysis, DT attaining values of 1.1, RFC (Random Forest Classifier) obtaining 0.21 in FAR, SMO—HPSO (Spider Monkey Optimization-Hierarchical Particle Swarm Optimization) can obtain values of 0.15. The proposed model can obtain values of 0.105. In this analysis, the proposed model can obtain the first position by consuming less amount of values in FAR. Fig. 10(b) shows training time for existing and proposed models. The proposed model can attain a training time of 0.00954. The proposed model consumes very little training time. Fig. 11 represents the FRP parameter analyzed and compared with existing and proposed models.

The proposed model can achieve 5.089 FRP parameters and obtain a first-rank position compared to existing models. The DT algorithm achieves the second rank, and the MLP algorithm occupies the last position. Thus, the proposed method can provide an efficient system with better performance than existing models. Comparison analysis for several existing models in machine learning algorithms is described in the following Table 10. The kappa parameter is analyzed for several existing, and the proposed model is described in the following Table 11.

The above table mentions the parameters such as accuracy, precision-recall, kappa score and so on, which are analyzed for existing models and compared with the proposed model. By using this analysis, can show that the proposed models obtain better performances and are highly efficient than the existing ones. There are several existing models used for intrusion detection that provide better results, but there are some limitations in the existing ones. The limitations are computational time, high maintenance cost and complexity in detecting intrusion for large-size AMI networks in the SVM model. Thus, these limitations are suppressed in the proposed M-MultiSVM model. Thus, the proposed M-MultiSVM model can accurately detect intrusion in the network.

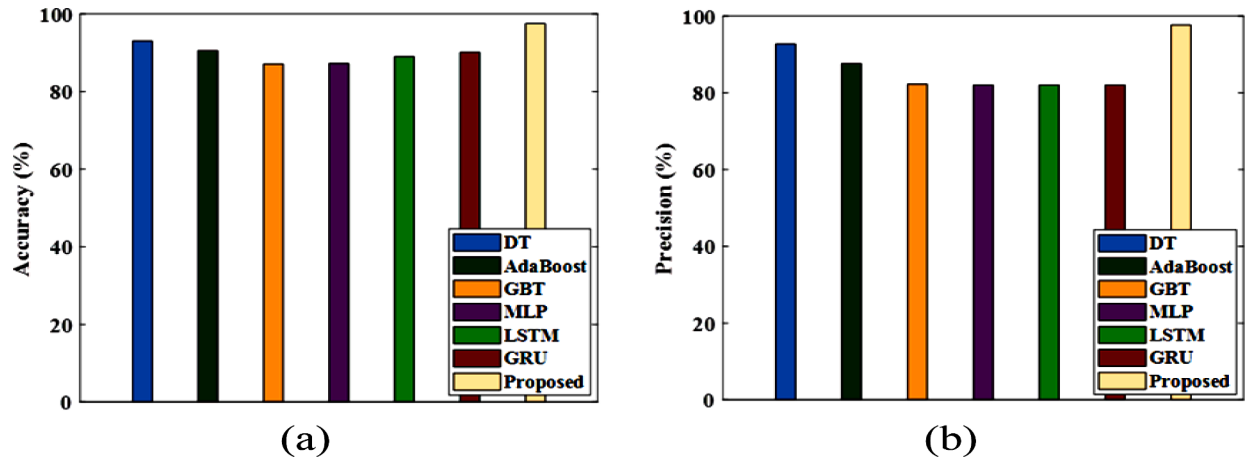

Fig. 8. Performance comparison (a) accuracy (b) precision.

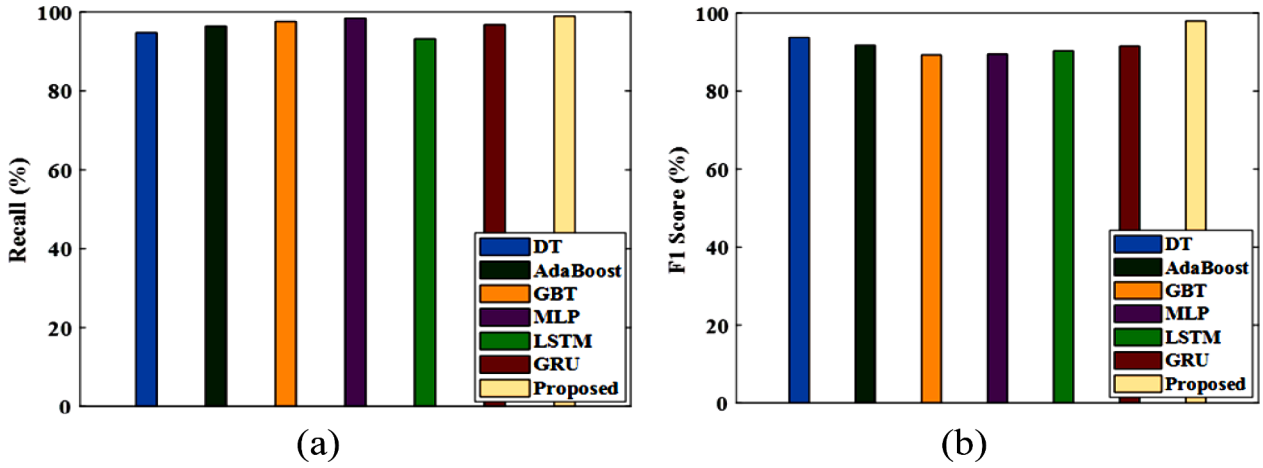

Fig. 9. Performance analysis (a) recall (b) F1 score.

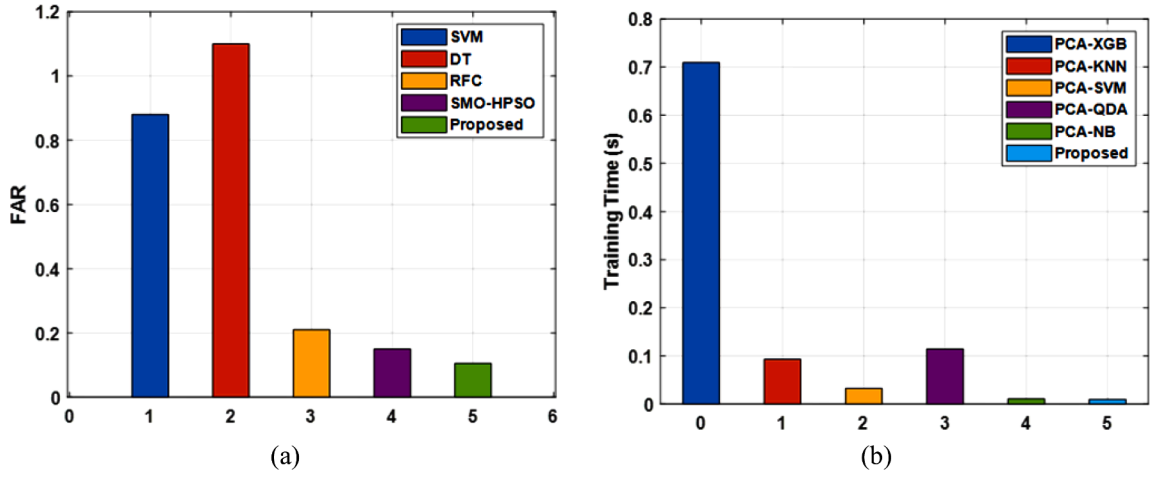

Fig. 10. Performance evaluation (a) FAR (Ethala et al. 2022) (b) training time (Saheed et al., al.2022).

4.4. Discussion

An intrusion detection system is used to identify the intrusion and several types of attacks in various types of applications like cloud environments and network environments. Early detection of intrusion is a very significant challenge. Thus, early detection should avoid cyber-attacks, virus threats and other attacks in a network environment. In the proposed model, several performance metrics such as accuracy, precision, recall, f1-score, FAR, FPR, MCC, specificity and training time are analyzed. The accuracy is measured for the proposed M-MultiSVM model to analyze the security of the model, which provides better accuracy and a better intrusion detection system. The existing models can

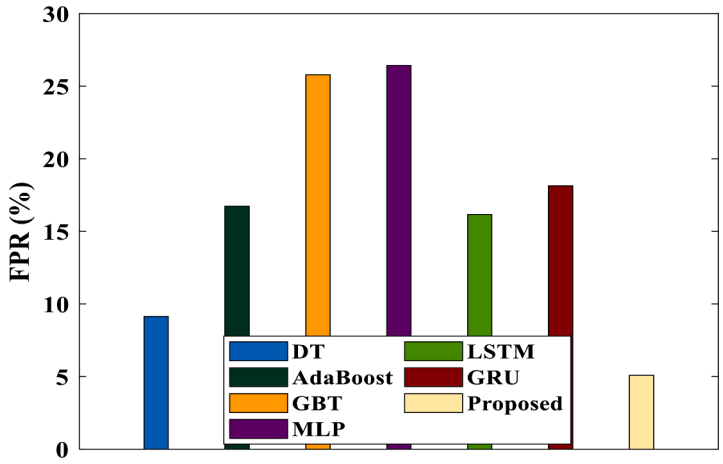

Fig. 11. Performance analysis for FPR.

Table 10
Performance analysis for UNSW-NB 15 dataset (Ethala et al. 2022).

Models	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC (%)	MCC (%)
SVM	85.88	89.36	85.93	88.73	94.25	0.43
DT	90.40	90.77	88.85	90.40	95.53	0.11
RFC	91.84	92.57	91.33	91.84	98.44	0.22
SMO—HPSO	94.12	94.19	93.32	94.12	99.78	0.19
Proposed	97.535	97.674	98.945	97.995	99.84	0.16

Table 11
Comparison of performances for UNSW-NB 15 dataset (Saheed et al., al.2022).

Classifier models	Kappa (%)
PCA-XGB	99.97
PCA-CB	99.97
PCA-KNN	99.96
PCA-SVM	99.96
PCA-QDA	99.94
PCA-NB	93.28
Proposed	99.99

obtain less amount of accuracy, when compared to the suggested M-MultiSVM model.

4.4.1. Limitations

In existing research works, intrusion detection can be performed by various ML and DL algorithms such as HT-RLSTM (Dahiya et al., 2022), OGBDTs-IDs (Mishra et al. 2022), LDSV (Hemanand et al. 2023), several types of attacks are classified by ML algorithms (Sarker et al. 2021), SVM and TFPG (Sun et al., 2020). HT-RLSTM consumes more computation time to detect attacks and is not able to prevent physical tampering theft by opponents. OGBDTs-IDs were not able to find the type of cyber-attacks, and very few security parameters were analyzed. Less amount of accuracy can be obtained in the LDSV model. Sarker et al. 2021 use low dimensional security data for attack classification. SVM and TFPG models are capable of identifying the types of attacks in large-scale AMI networks. To enhance the intrusion detection model by overcoming these existing issues, a hybrid machine-earning model is able to produce an efficient outcome in the intrusion detection system by classifying several types of attacks. The proposed model enhances the system by suppressing these issues by adding some extra efficient opposition-based learning, optimization algorithms and hybrid machine learning algorithms.

Some limitations in existing models were present in the existing models, like consuming more time to predict the model, complex to identifying the attacks in the large-scale AMI networks and so on. The existing models are not tuned properly and aren't able to balance the classes accurately. Thus, it can obtain less performance in the comparison analysis. The proposed model suppresses these issues and develops an efficient model with an accurate intrusion detection system using optimization strategies and a hybrid ML model with oversampling techniques. In recent research, works have not applied attention mechanisms before or after the stage of classification. The attention mechanism helps to focus the relevant parts of the input and provides better detection in any network environment. Several types of attacks can be identified by the proposed M-MultiSVM model using input data. To improve the proposed model with the capability to perform efficiently on multiple datasets. In the future, an attention mechanism will be added to the proposed model to improve performance. Create the model by skillfully managing many datasets and offering a mathematical prediction technique that might be used to predict optimal solutions using enhanced optimization strategies.

5. Conclusion

A novel intrusion detection system using machine learning technology is proposed in this research work. This proposed work is to enhance the performance and efficiency of the intrusion detection system. This novel technique performs detection of abnormalities in various networks, such as menacing networks, computer resources and network information. Automatic detection of abnormality is used to secure the required computing ability and analysis of attacks. Here, Min-Max normalization is used to remove null values from input in a preprocessing stage, and imbalance data is converted into balanced data by ASmoT. Various features are extracted and selected from the input using M-SvD, and the ONgO reduces the complexity of feature extraction and selection. Using these extracted features, the classification of intrusion is performed by the M-MultiSVM model optimized by the Mud ring algorithm. The evaluation of the given model resulted in 99.89 % of accuracy, 99.125 % of DR....., 0.013 of FAR, 99.914 % of precision, 99.214 % of F1 Score, Specificity of 99.88 %., training time is 3.21502541 and 98.584 % of MCC in the CSE-CIC-IDS dataset. Further, the proposed model is also evaluated using dataset UNSW-NB 15, which resulted in an accuracy of 97.535 %, precision of 997.674 %, recall of 98.945 %, F1-Score of 97.9955, FRP of 5.089, FAR of 0.105 and training time of 0.00954.

Compliance with ethical standards

Funding: No funding is provided for the preparation of manuscript.

Ethical approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Consent to participate

All the authors involved have agreed to participate in this submitted article.

Consent to publish

All the authors involved in this manuscript give full consent for publication of this submitted article.

CRediT authorship contribution statement

Anil V Turukmane: Conceptualization, Methodology, Software, Writing – original draft, Visualization. **Ramkumar Devendiran:** Formal analysis, Supervision, Project administration.

Declaration of Competing Interest

No conflict of Interest

Data availability

No data was used for the research described in the article.

References

Alharbi, A., Seh, A.H., Alosaimi, W., Alyami, H., Agrawal, A., Kumar, R., Khan, R.A., 2021. Analyzing the impact of cyber security related attributes for intrusion detection systems. Sustainability 13 (22), 12337.

Alom, M.Z., Taha, T.M., 2017. Network intrusion detection for cyber security on neuromorphic computing system. In: In2017 International Joint Conference on Neural Networks (IJCNN), pp. 3830–3837. IEEE.

Al-Omari, M., Rawashdeh, M., Qutaishat, F., Alshira'H, M., Ababneh, N, 2021. An intelligent tree-based intrusion detection model for cyber security. J. Netw. Syst. Manag. 29, 1–8.

Alshamy, R., Ghurab, M., Othman, S., Alshami, F., 2021. Intrusion detection model for imbalanced dataset using smote and random forest algorithm. In: In Advances in Cyber Security: Third International Conference, ACeS 2021, Penang, Malaysia, August 24–25, 2021, Revised Selected Papers 3. Springer Singapore, pp. 361–378.

Dahiya, M., Nitin, N., Dahiya, D, 2022. Intelligent cyber security framework based on SC-AJSO feature selection and HT-RLSTM attack detection. Appl. Sci. 12 (13), 6314.

Desuky, A.S., Cifci, M.A., Kausar, S., Hussain, S., El Bakrawy, L.M, 2022. Mud Ring Algorithm: a new meta-heuristic optimization algorithm for solving mathematical and engineering challenges. IEEE Access 10, 50448–50466.

Disha, R.A., Waheed, S., 2022. Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. Cybersecurity 5 (1), 1.

Ethala, S., Kumarappan, A., 2022. A hybrid spider monkey and hierarchical particle swarm optimization approach for intrusion detection on Internet of Things. Sensors 22 (21), 8566.

Gupta, G.P., Kulariya, M., 2016. A framework for fast and efficient cyber security network intrusion detection using apache spark. Procedia Comput. Sci. 93, 824–831.

Haider, A., Khan, M.A., Rehman, A., Ur Rahman, M., Kim, H.S, 2021. A real-time sequential deep extreme learning machine cybersecurity intrusion detection system. Comput. Mater. Contin. 66 (2).

Helliwell R., Hartley S., Pearce W. NGO perspectives on the social and ethical dimensions of plant genome-editing. InRethinking Food System Transformation. Cham: Springer Nature Switzerland 2022; 129–141.

Hemanand, D., Vallem, R.R., 2023. Cyber security system based on machine learning using logistic decision support vector. Mesop. J. CyberSecur. 2023, 64–72.

Jia, G., Lam, H.K., Ma, S., Yang, Z., Xu, Y., Xiao, B, 2020. Classification of electromyographic hand gesture signals using modified fuzzy C-means clustering and two-step machine learning approach. IEEE Trans. Neural Syst. Rehabil. Eng. 28 (6), 1428–1435.

Kaja, N., Shaout, A., Ma, D., 2019. An intelligent intrusion detection system. Appl. Intell. 49, 3235–3247.

Lalwani, P., Mishra, M.K., Chadha, J.S., Sethi, P., 2022. Customer churn prediction system: a machine learning approach. Computing 1–24.

Magán-Carrión, R., Urda, D., Díaz-Cano, I., Dorronsoro, B., 2020. Towards a reliable comparison and evaluation of network intrusion detection systems based on machine learning approaches. Appl. Sci. 10 (5), 1775.

Mishra, S., 2022. An optimized gradient boost decision tree using enhanced African buffalo optimization method for cyber security intrusion detection. Appl. Sci. 12 (24), 12591.

Mohammadi, S., Mirvaziri, H., Ghazizadeh-Ahsaee, M., Karimipour, H., 2019. Cyber intrusion detection by combined feature selection algorithm. J. Inf. Secur. Appl. 44, 80–88.

Morris T.H., Thornton Z., Turnipseed I. Industrial control system simulation and data logging for intrusion detection system research. 7th annual southeastern cyber security summit. 2015; 3–4.

Pascale, F., Adinolfi, E.A., Coppola, S., Santonicola, E., 2021. Cybersecurity in automotive: an intrusion detection system in connected vehicles. Electronics 10 (15), 1765.

Raju, V.G., Lakshmi, K.P., Jain, V.M., Kalidindi, A., Padma, V., 2020. Study the influence of normalization/transformation process on the accuracy of supervised classification. In: 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), pp. 729–735. IEEE.

Rekha, G., Malik, S., Tyagi, A.K., Nair, M.M., 2020. Intrusion detection in cyber security: role of machine learning and data mining in cyber security. Adv. Sci. Technol. Eng. Syst. J. 5 (3), 72–81.

Saheed, Y.K., Abiodun, A.I., Misra, S., Holone, M.K., Colomo-Palacios, R, 2022. A machine learning-based intrusion detection for detecting internet of things network attacks. Alex. Eng. J. 61 (12), 9395–9409.

Sarker, I.H., 2021. CyberLearning: effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks. Internet Things 14, 100393.

Seth, S., Singh, G., Chahal, K.K., 2021. A novel time efficient learning-based approach for smart intrusion detection system. J. Big Data 8 (1), 1–28.

Sun, C.C., Cardenas, D.J., Hahn, A., Liu, C.C., 2020. Intrusion detection for cybersecurity of smart meters. IEEE Trans. Smart Grid 12 (1), 612–622.

Ugwu, C.C., Obe, O.O., Popoọla, O.S., Adetunmbi, A.O., 2021. A distributed denial of service attack detection system using long short term memory with singular value decomposition. In: In2020 IEEE 2nd International Conference on Cyberspac (CYBER NIGERIA), pp. 112–118.

Vigneswaran, R.K., Vinayakumar, R., Soman, K.P., Poornachandran, P., 2018. Evaluating shallow and deep neural networks for network intrusion detection systems in cyber security. In: In2018 9th International conference on computing, communication and networking technologies (ICCCNT), pp. 1–6. IEEE.

Wong, K., Dillabaugh, C., Seddigh, N., Nandy, B., 2017. Enhancing Suricata intrusion detection system for cyber security in SCADA networks. In: In 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), pp. 1–5.

Zhang, H., Zhang, B., Huang, L., Zhang, Z., Huang, H., 2023. An efficient two-stage network intrusion detection system in the Internet of Things. Information 14 (2), 77.

Dr. Anil Vitthalrao Turukmane, Working as a Professor in the School of Computer Science & Engineering at VIT-AP University (Vellore Institute of Technology, Andhra Pradesh), Amaravati, Vijayawada, Andhra Pradesh 522237, India.

Dr. Ramkumar Devendiran, Working as an Assistant Professor Senior Grade in the department of School of Computer Science & Engineering at VIT-AP University (Vellore Institute of Technology, Andhra Pradesh), Amaravati, Vijayawada, Andhra Pradesh 522237, India.