International Journal of Safety and Security Engineering
Vol. 13, No. 3, June, 2023, pp. 509-517
Journal homepage: http://iieta.org/journals/ijsse

Enhanced SVM Model with Orthogonal Learning Chaotic Grey Wolf Optimization for Cybersecurity Intrusion Detection in Agriculture 4.0

Khaja Shareef Shaik[1], Naga Siva Kumar Thumboor[2], Siva Prasad Veluru[3], Naga Jagadesh Bommagani[4*], Dorababu Sudarsa[5], Ganesh Karthik Muppagowni[6]

[1] Department of Information Technology, MLR Institute of Technology, Hyderabad 500043, India
[2] Department of Artificial Intelligence and Data Science, Vishnu Institute of Technology, Bhimavaram 534202, India
[3] School of Computing, Mohan Babu University, Tirupati 517102, India
[4] School of Computer Science and Engineering, VIT-AP University, Vijayawada 522237, India
[5] Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur 522302, India
[6] Department of Computer Science and Engineering, GITAM School of Technology, GITAM University-Bengaluru Campus, Bengaluru 561203, India

Corresponding Author Email: nagajagadesh@gmail.com

https://doi.org/10.18280/ijsse.130313

Received: 21 May 2023
Accepted: 29 June 2023

Keywords:
smart agriculture, Agriculture 4.0, Internet of Things (IoT), cybersecurity, cyber-attacks, Distributed Denial of Service (DDoS), Enhanced Multiclass Support Vector Machine (EMSVM), Orthogonal Learning Chaotic Grey Wolf Optimization (OLCGWO), intrusion detection

ABSTRACT

Smart agriculture, also known as Agriculture 4.0, integrates cutting-edge technology with conventional farming practices through the agricultural Internet of Things (IoT). Despite its numerous advantages, Agriculture 4.0 introduces additional cybersecurity risks due to the widespread deployment of IoT-based devices. One significant threat is Distributed Denial of Service (DDoS) attacks, which can compromise the availability and integrity of agricultural systems. This paper proposes an Enhanced Multiclass Support Vector Machine (EMSVM) model for detecting DDoS attacks in Agriculture 4.0. To improve classification accuracy, the EMSVM model incorporates a novel optimization method called Orthogonal Learning Chaotic Grey Wolf Optimization (OLCGWO) for parameter selection. The performance of the proposed methodology is evaluated using two real-world traffic datasets, CIC-DDoS2019 and TON_IoT, which contain various DDoS attack scenarios. The results demonstrate the effectiveness of the EMSVM model in both binary and multiclass classification contexts.

1. INTRODUCTION

"Smart agriculture" [1, 2] is a new way of farming that emphasizes individualized customer care by making use of cutting-edge information technologies including the web, big data, the internet of things, and many others. In a word, the new method is an information technology–enabled, farm–specific answer. While modern information technology offers promising avenues for the enhancement of agricultural production, it also makes considerable stresses on safety in the context of smart farming [3]. Agriculture 4.0 is more than just a trend; it's the next technological step in the industry's evolution toward smarter, more efficient, and greener practices. The massive amounts of data generated by the supply chain every day [4]. This data was before wasted, but with the advent of big data and advanced farming techniques, it can be used to significantly advance the harvest and quality of any crop.

1.1 Advantages of Agriculture 4.0

The following sections discuss the results of practicing "smart agriculture".

Production volume: When used to farming, smart technology has the potential to greatly increase output. This helps meet the challenge of feeding an expanding population.

The quality of production has far-reaching consequences for the health and nourishment of Americans of all income levels. If a country's residents have access to higher-quality food, they will be healthier and live longer, increasing their economic output.

Efficiency in agricultural methods and material consumption: Smart technology can improve the efficiency of traditional farming practices. This, in turn, indorses better utilization of agricultural capitals [5].

The ideal production cost: It occurs when the methods employed strike a good balance between quantity, quality, and efficiency. The price of agricultural products goes up as a result.

Efforts to Lessen Waste The agricultural sector, a major contributor to the economy, is mostly to fault for the enormous amounts of food and other secondary resources that go to waste every year. This waste might be monitored and reduced with the help of modern technologies.

Ecological sustainability is achieved by direct reductions in environmental and ecological footprints as a result of decreased left-over and increased agricultural procedure efficiency.

Saving Time: Smart agriculture's timely distribution of required pesticides, fertilizers, and other consequence in timely and agricultural production with fewer wounded [6].

While implementing agricultural 4.0 technology, the farming may be at risk from the following:

More computational capabilities will be built into systems as gadgets and technology evolve [7]. The goal of this sort of

integration is to provide for the requirements of sustainable agriculture, mechanization, and farming methods [8]. Safeguarding sensitive information is another concern. Data privacy, data trustworthiness, and data accuracy from data production through decision making are all very vital.

- ❖ Theft of sensitive company and consumer information.
- ❖ By stealing resources managed by sensors and devices and destroying their targets.
- ❖ Reputational harm if sensitive information is leaked.

Damage to agricultural infrastructure, sensor failures in livestock breeding, scheme hacks in greenhouse farming are all potential threats to Agriculture 4.0 [9]. All of them have the potential to disrupt farming operations by damaging the hardware and software that makes up the IoT infrastructure. Malicious attacks, illegal access, privacy breaches, and other problems [10] also plague data-gathering technology.

The field has shifted, with researchers focusing on agricultural contexts including water management, livestock, and farmlands, as well as artificial intelligence and machine learning. Numerous water-saving and productivity-boosting monitoring, management, and decision-making options have been explored in the irrigation sector [11].

1.2 Research motivation

This piece was inspired by three main ideas:

1) As a response to the low output of conventional farming and the widespread adoption of information knowledge, "smart agriculture" is a novel paradigm that merges the two. It might be the breakthrough in farming that finally takes off. Therefore, outlining the existing production model and specific investigations is essential [12].
2) There has been less analysis of security issues in smart agriculture despite a lot of study in this area compared to industrial security solutions.
3) Thirdly, analyzing the worries in smart agricultural contexts is essential [13].
4) Given the aforementioned factors, it is impossible to provide an inclusive overview of the security issues posed by smart agriculture without leaving many gaps in our understanding.

These new technologies have seen extensive use in Industry 4.0, and it would be easy to adapt them for use in farming. Since the placement of IoT-based devices is in a public arena, the greatest difficulty in establishing technologies, but rather in the guarantee of privacy. The cyber security research community recommends (IDS), which are a skill for network safety that constantly monitors occurrences inside a system and assesses them in light of intrusion indication [14].

Anomaly-based intrusion detection systems (deep-learning techniques) are the subject of this study. However, in the area of Agriculture 4.0, there are eight significant obstacles to overcome: Challenges include [[IIoT data collection]], [[less training data]], [[non-representative training data]] [15]. These glitches are solved by our suggested model. Our article makes use of widely-used, up-to-date datasets that have been put to good use by the research community in the creation of intrusion schemes for IIoT systems.

Our aids in this work are:

We provide an evolutionary algorithm-based machine learning system for improving IDS models.

We analyze and compare several machine learning strategies for cyber safety in agriculture 4.0 and offer an evaluation of their efficacy.

Using the TON_IoT dataset, two brand-new real-world traffic datasets, we investigate the performance of each model across two categorization types (binary and multiclass).

We pay special attention to the ROC Curve, the False Acceptance Rate (FAR), the True Negative Rate (TNR), the Detection Rate (DR), and the Precision.

1.3 Organization of the paper

Here is how the rest of this piece is laid out. In Part 2, we'll examine the relevant literature. The use of IDSs is described in Section 3. A comparison of several IDS for Agriculture 4.0 is providing in Section 4. Section 5 accomplishes the paper.

2. RELATED WORKS

In order to detect cybersecurity threats in IoT cloud networks, Alrayes et al. [16] introduce an Enhanced Artificial Gorilla Troops Optimizer (EAGTO) that uses deep learning. IoT cloud environment threat detection is a fundamental part of the EAGTODL-CTD methodology given here. In order to identify malware via an image classification challenge, the suggested EAGTODL-CTD model prioritizes the transformation of input binary files into color pictures. To ensure compatibility, the EAGTODL-CTD model performs preliminary processing on the incoming data. Class labels are determined using a cascaded gated recurrent unit (CGRU) model for use in threat detection and classification. Our work's originality is demonstrated by the fact that we use the EAGTO method as a hyperparameter optimizer to fine-tune the CGRU's underlying parameters. The EAGTODL-CTD model's efficacy is measured using a dataset annotated with two classes, cancerous and benign. Enhanced accuracy of the EAGTODL-CTD model (99.47%) was indicated by the experimental results as the best.

Cyber-attack model built on (RFE) and multilayer perceptron (MLP) by Kilincer et al. [17]. The RFE method picked the best features by utilizing the kernel functions of (XGBRegressor). A hyperparameter optimization was used to fine-tune the MLP's parameters, and a 10-fold cross-validation procedure was used to assess the model's efficacy. Using Edith (The proposed methodology can be used to defend against cyberattacks on medical software.

There must be more research into cyber refuge and the prevention of cyber attacks, such as the deployment of intrusion detection as a preventative measure, as indicated by Jain et al. [18]. Most people today utilize at least one internet service. The term "cyber" is used to describe the realm of the internet, computers, and other technological services. New protocols and technology have allowed for significant development in the cyber realm. Every internet business must address the serious problem of cyber security. The foundations of any cyber defense system are intrusion detection systems (IDSs) like those used to monitor networks and computers for malicious activity. The NSL-KDD dataset is frequently used for the study and expansion of intrusion detection systems, as well as for algorithm research and verification. The purpose of this research was to create a neural network-based method for predicting potential threats in intrusion detection systems. In order to conduct the simulations in this article, the Python Spyder program is employed.

The FIEBB model, proposed by Wang et al. [19], integrates features and detects entities' boundaries. Our technique employs a recently developed pretrained language model called PERT to derive digital text word embeddings. And to combine the best of both graph neural networks and recurrent neural networks, a brand new neural network cell called GARU is created. To improve the quality of the hidden representation, this method syndicates the graph encoder with the gate apparatus. In addition, we contribute an entity boundary detection module for performing entity head and tail forecast as job, as there are many complicated entities in the field of cybersecurity. On cybersecurity datasets, we do comprehensive tests. The outcomes show that the suggested model outperforms the currently used approaches.

To this end, the elastic stack (ELK) architecture established by Folino et al. [20] is proposed to process and store log data in real time from various users and applications. Using the benefits of system produces an ensemble of models to categorize user behavior and identify abnormalities in real time. In addition, the users are sorted into groups based on the digital traces they've left behind, which are gleaned from a wide variety of data sources and analyzed with a distributed evolutionary algorithm. The approach's efficacy in detecting abnormalities in user behavior, dealing with missing data, and reducing false alarms has been experimentally validated on two real-world data sets.

Unsupervised Hunting of Anomalous Commands (UHAC) is a machine learning-based approach proposed by Kayhan et al. [21] for detecting text-based anomalous commands in security information and event management (SIEM) logs that are promising leads for threat hunting. Different from other approaches, the suggested one builds a feature set by augmenting document-term and document-character matrices, which is a novel step. Next, a custom loss function is used to this feature set as training data for an autoencoder-based detector. UHAC routinely achieves better results than competing feature sets and techniques, including one-class word-embedding based models like word2vec. If an anomaly exists in the top 10 percent of the data, the UHAC detector will find it. The results have ramifications for process auditing on endpoint devices, where cybersecurity analysts conduct threat hunting in SIEM logs.

Improved cyber-attack finding utilizing unlabeled data for ICS traffic nursing and identifying abnormal data transfers is presented by Dairi et al. [22]. Importantly, we developed two anomaly detection strategies based on semi-supervised hybrid deep learning for use in the intrusion detection of ICS traffic in a smart grid environment. Our first method is a Gated recurrent unit (GRU)-based stacked autoencoder (AE-GRU), and our second, which we term a GAN-RNN, is built on the GAN model but uses a (RNN) for both the generator and the discriminator. It is anticipated that including GRU and RNN into AE and GAN models would enhance their capability to learn temporal connections in multivariate data. Models like Support Vector Machine, and Elliptical Envelope are anomaly finding in cyber-attacks on power grids. In contrast to other methods for cyber-attack detection, these use simply regular events data for training, without designated attack kinds. On the IEC 60870-5-104 (also known as IEC 104) communication, which is frequently used for smart grids, the detection performance of different methods is proven. Among the approaches tested, those based on GAN-GRU and AE-GRU exhibited the greatest improvement in finding, with an average F1-score of 0.98.

3. PROPOSED SYSTEM

In this section, interest to security researchers who work to ensure the safety of the scheme. we suggest machine learning-based cyber-attacks that is explained in the following sub-sections.

3.1 Network model

Three levels, or "layers," make up the Agriculture 4.0 network model given here: (1) Agricultural sensors; (2) Fog computing; and (3) Cloud computing. Data collected by drones and other Internet of Things sensors is used in the agriculture sector. In the agricultural sensors layer, actuators are triggered by data that meets predetermined criteria. The agricultural sensors layer incorporates new energy technologies and smart grid design to power Internet of Things gadgets. Each fog node has an intrusion detection system powered by deep learning. Cloud computing nodes offer storage services, while the fog computing layer analyzes the IoT data with machine learning algorithms after receiving it from the agricultural sensors layer. Intrusion detection systems that rely on deep learning to process data do their calculations in the fog nodes. We assume that there is a malicious party intent on disrupting the network's operations in order to compromise food security, the efficacy of the agri-food supply chain, and agricultural production.

3.2 Dataset description

In particular, we used the CIC-DDoS2019 dataset [23] and the TON dataset [24]—two recently released real-world traffic datasets. There are three criteria for picking them: Agriculture 4.0 is like these networks because (1) they were designed for a TCP/IP communication stack, (2) they feature DDoS assaults, and (3) they are representative of the nature of the industry.

The TON_IoT dataset was developed to mimic the functioning of actual operational IoT/IIoT networks through the use of interacting network parts and IoT/IIoT systems across the Edge, Fog, and Cloud. To help with the administration of the interplay between these three levels, the NSX-VMware platform was used (NFV) technologies. The experiment is coded in Python 3 on a GPU using TensorFlow, and it is run on Google Colaboratory (https://colab.research.com). To be more specific, there are four stages to the method: There are four stages: (1) gathering datasets, (2) pre-processing, (3) training, and (4) testing.

3.3 Pre-processing of the CIC-DDos2019 dataset

There are a CIC-DDoS2019 dataset [23], including 50,006,249 rows related to DDoS assaults and 56,863 rows related to normal traffic. Each column has 86 characteristics. Table 1 summarizes the attack statistics for dataset. The training dataset includes 12 DDoS attacks, including NTP, DNS, LDAP, Microsoft SQL Server, NetBIOS, SNMP, SSDP, User Datagram TFTP.

- An NTP-based attack is a (DDoS) attack in makes advantage of a compromised Network Time Protocol (NTP) server to overwhelm a targeted client-server or other network with an excessive amount of UDP data traffic. This type of attack has the possible to render the target and its associated network unreachable to legitimate traffic.

- An attack that leverages the Domain Name System (DNS) to flood a target IP address with resolution requests is called a reflection-based DDoS assault.
- By sending queries to a publicly accessible susceptible LDAP server, an attacker can create huge replies (amplified), which are then reflected to a target server, causing a distributed denial of service (DDoS) attack.
- An MSSQL- DDoS attack in which the attacker forges an IP address to make scheduled requests seem to originate from the target server, thereby overwhelming its resources.
- An attacker conducting a reflection-based DDoS assault against NetBIOS can trick a target computer into rejecting all incoming NetBIOS communication by sending a faked "Name Release" or "Name Conflict" message.
- To jam the target's network pipes, an SNMP-based assault will produce attack volumes in the hundreds of gigabits per second using the Simple Network Management Protocol (SNMP).
- An SSDP-based assault is a reflection-based distributed denial of service attack in which the attacker uses UPnP protocols to send a victim an amplified quantity of data.
- Attacks based on User Datagram Protocol (UDP) lag try to disrupt the targeted host by flooding it with IP packets containing UDP datagrams.
- To compromise a Web server or application, a WebDDoS-based attack will use seemingly innocuous HTTP GET or POST requests.
- By mimicking the normal sending SYN-ACK (synchronize-acknowledge), and responding with an ACK (acknowledge), a SYN-based attack might starve the victim server of its resources and render it inoperable.
- The Trivial File Transfer Protocol (TFTP) may be exploited in this assault, which makes use of online TFTP servers. The attacker file, and the victim TFTP server returns the data to the attacker's target host.
- A port scan can be performed on a single computer or on a whole network as part of a port scan-based assault. Scanning is performed by inquiring as to what services are active on a remote server.

Table 1. Attack categories in CICDDoS2019 dataset

Attack Category	Flow Count
DDoS-NTP	1,202,653
DDoS_SNMP	5,159,821
Benign	56,863
DNS	5,071,421
LDAP	2,179,930
SSDP	2,610,611
SYN	1,582,289
TFTP	20,082,70
UDP	3,134,665
DDoS_UDP-Lag	366,461
DDoS_WebDDoS	459
DDoS_MSSQL	4,522,482
DDoS-NetBIOS	4,092,379

Classification problems including two classes and classification errands involving more than two classes (multi-class classification) are studied in order to assess the efficacy of machine learning algorithms. We generate data sets with class labels of Dataset_2_class, Dataset_7_class, and Dataset_13_class. Table 2 shows, for instance, the data used in training and testing the Dataset_2_class.

Table 2. Attack categories in Dataset_2_class

Category	Test	Training
Benign	17,146	56,101
Attack	314,716	997,054

3.4 Pre-processing of dataset

Data from networks, operating systems, and telemetry devices make up the TON_IoT dataset, a novel testbed for an IIoT network. IoT and IIoT sensor telemetry data is provided in 7 separate formats.

These files include the following information:

- Data types found in File 1: "Train_Test_IoT_Weather" include: Normal (35,000 rows), DDoS (5,000 rows), Injection (5,000 rows), Password (5,000 rows), Backdoor (5,000 displays the Internet of Things data from a networked weather sensor, including temperature, pressure, and humidity values.
- File 2: "Train_Test_IoT_Fridge" has the following categories of rows: Normal (35,000), DDoS (5,000), Injection (5,000), Password (5,000), Backdoor (5,000), Ransomware (2902), and XSS (2942). Data from a networked refrigerator sensor, including temperature readings and ambient variables, are presented in this file.
- File 3: "Train_Test_IoT_Garage_Door" includes the following categories of data: Normal (70,000 rows), DDoS (10,000 rows), (10,000 rows), Password. Data from a networked door sensor is shown in the file, showing whether the door is open or closed.
- File 4: "Train_Test_IoT_GPS_Tracker" includes the following categories of data: Normal (35,000 rows), DDoS (5,000 rows), Injection (5,000 rows), Password (5,000 rows), Backdoor (5,000 rows), Ransomware (2,833 rows), XSS (577 rows), and Scanning (550 rows). Data from a networked GPS tracker sensor, including its latitude and longitude coordinates, is presented in the file.
- You'll find the following data types in File 5: "Train_Test_IoT_Modbus": Normal. IoT data file containing Modbus function code for reading an input register.
- You'll find the following categories in File 6: "Train_Test_IoT_Motion_Light" and Scanning (3550 rows). The data in the file is the Internet of Things readings from a switched-on or -off light sensor.
- Included in File 7 "Train_Test_IoT_Thermostat" are the following types of data: Normal (35,000 rows), Injection (5,000 rows), Password (5,000 rows). The file contains Internet of Things data showing the current temperature as measured by a thermostat sensor on the network.

3.5 Classification using Enhanced multiclass SVM (EMSVM) model

To model intricate connections between variables, SVM is

well recognized as a statistical ML technique. The power to generalize and the capacity to deal with the curse of dimensionality are excellently combined in SVM. Typically, DM and ML algorithms suffer from the curse of dimensionality, which reduces their effectiveness. However, SVM has proven itself to be a gifted method that can achieve remarkable results despite the limited data available for training the algorithm. When used to non-linearly separable problems, kernel functions allow SVMs to effectively translate them into higher dimensions, where they may be separated with more ease. The vast majority of widely-applied models can have a common foundation thanks to kernel mapping. The training dataset's original dimension-space is transformed to higher dimensionality in order to map non-linear separable samples into separable ones That can be readily differentiated. Although SVM was originally designed for use in classification, it has now been shown to be effective in regression as well. The scenario studied here is categorized as a research challenge. It is well knowledge, however, that a model's generalizability increases as its ability to increase the margins between classes does. To achieve generalization in SVM, it is common practice to generate a collection of vectors that is sparse but yet capable of definitively distinguishing between classes. Boundary examples capture the information required to partition the classes, and hence may be used to categorize new data.

The primary objective of any classification problem is to establish a correlation between a given set of input features (also called predictors) and a given set of class variables (also called training instances) in a given input features (also called predictors) and m training instances. Note that whereas XR in regression issues, YR in classification difficulties. As an example, consider a classification problem with a training dataset.

T=x_ij, x_(i+1, j+1),...,x_nm,$c\_1, c\_2, \dots c\_t$, where t >=1. An "Optimal Separating Hyperplane (OSH)" is generated by minimizing are then applied to the training dataset to produce accurate results.

$$sgn\left(\sum_{i=1}^{n} y_i \propto_i . K\left(x_i . x_j\right) + b\right) \quad (1)$$

where, $x_j = 1,2,.3\ldots, Z$ are the so-called the " (SV)". The "lagrange dual equation," denoted by Eq. (2) and sometimes is used to get the coefficient _i and the bias b.

$$MAX\left(\sum_{i=1}^{n} \propto_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \propto_i \propto_j . y_i y_j . K\left(x_i, x_j\right)\right) \quad (2)$$

where:

$$\sum_{i=1}^{n} \propto_i y_i = 0 \quad (3)$$

Only if 0_iC, can we say that x_j is a support vector. In where C is issue that determines the misclassification error vs margin tradeoff. In other words, C controls the price at which you must choose between simplifying your model and reducing your training errors. It's worth noting that the SVM has a significant penalty for points, therefore if C is SVM could create an over-fitted model. However, a model may be under-fitted if C is too low. However, the dataset is transformed into hyperplanes via the kernel function K.

SVM may make use of a wide variety of kernel function types. Polynomial and Radial Basis Function (RBF) are the two most used methods. Degree d polynomial function is found using Eq. (4):

$$K\left(x_i, x_j\right) = \left(x_i^T . x_j + 1\right)^d \quad (4)$$

Note that the additive constant in the equation disappears when d = 1, transforming the polynomial function back into a linear one. This allows us to derive Eq. (5), which yields the linear kernel function:

$$K\left(x_i, x_j\right) = x_i^T x_j \quad (5)$$

However, the RBF (or Gaussian kernel) is determined by the following Eq. (6):

$$K\left(x_i . x_j\right) = exp\left(-\gamma x_i - x_j^d\right) \quad (6)$$

In which the Gaussian width is determined by. To put it another way, it functions similarly to d in Polynomial kernel in that it determines the degree of adaptability of the final classifier. Here, $\gamma = \frac{1}{2\sigma^2}$ and σ is a free parameter.

Sigmoid kernel is another type of Neural Network-related kernel function. This kernel function, which has been in use since 1995, may be computed with the help of the following Eq. (7):

$$K\left(x_i . x_j\right) = \tan h\left(\gamma x_i^T x_j + r\right) \quad (7)$$

where, γ and r are kernel influences.

3.5.1 Parameter settings in EMSVM

Research and finding the optimal standards of the most critical limits is necessary when constructing any classification model to ensure successful training and then reasonable testing of the created classification perfect(s). The present study employs a technical parameter exploration strategy to ensure that the best parameter values picked, as the parameters of an SVM model, " in QP" (k), and "the kernel" (K). An additional factor to think about is e ("Epsilon," for short), which determines the mistake margin before punishment kicks in. Choosing these criteria is typically a time-consuming and exhausting trial-and-error procedure. In this research, a "improved GWO" is used to the provided dataset in order to fine-tune all the aforementioned parameters necessary for developing a reliable SVM model.

3.5.2 Parameter tuning with chaotic grey wolf optimization (CGWO)

To avoid becoming stuck in a rut of local optimization, we provide the Optimization for feature learning. The traditional GWO takes its cues for its optimal selection of SVM models from the ranking and hunting methods used in this study.

Grey wolves may be divided into four distinct subspecies. Alpha (the leader), beta (one who contributes to decision making), delta (one who defers to alphas and omegas), and omega (the wolf pack's subservient member) are the four positions. The hunting group exhibits numerous social qualities in addition to the distinct ones associated with socioeconomic status. There are three distinct features of the hunting phase: First, there is the pursuit and approach; second, there is the pursuit and encirclement; and third, there is the attack to the target. The CGWO perfect is a mathematical framework inspired on the social structure of wolves.

3.5.3 Orthogonal learning-based CGWO (OLCGWO)

Orthogonal CGWO is an ‘intelligent drive mechanism’ for

making a provisional position like $r = (r_1, r_2, \ldots, r_n)^T$ and $h = (h_1, h_2, \ldots, h_n)^T$ for altogether particles. It is uttered as in Eqns. (8) & (9):

$$h_j = x_{i,j} + wv_{i,j} + c_1 r_{1,j}\left(x_{i,j}^{pbest} - x_{i,j}\right) \quad (8)$$

$$r_j = x_{i,j} + wv_{i,j} + c_2 r_{2,j}\left(x_j^{gbest} - x_{i,j}\right) \quad (9)$$

Social learning and individual cognition are represented by 'r' and 'h' in this context. Next position 'x' is obtained by applying the OLCGWO over 'r' and 'h', and the particle's velocity is calculated by taking the absolute value of the difference among the particle's present location and its future position, 'x'.

The OLCGWO mechanism for motion combines 'r' and 'h' data effectively to determine the next particle's location. Particles generate their current velocity and position via a moving mechanism throughout the searching process. The mobile mechanism for determining both the particle and population optimal solution. Particle searching is performed using x_iol, which is a replacement for x_ipbest and x_gbest in the orthogonal learning strategy's implementation of the moving mechanism. Eq. (10) represents a revised expression for the particles' velocities.

$$v_{i,j} = wv_{i,j} + c_1 r_{1,j}\left(x_{i,j}^{ol} - x_{i,j}\right) \quad (10)$$

In this case, the x_iol stores encouraging data, such as x_ipbest and x_gbest, to counteract the oscillation observed when learning from an exemplar with a lot of abrupt changes in direction. In terms of the generations required to get the optimal particle configuration, x_iol serves as a benchmark.

A new learning example is produced once the exemplar approaches the maximum mobility strategy. Stagnation is assumed to occur at generation k_i. When the particles' best-case scenario isn't changed, we add 1 to k_i. A new learning example is created whenever k_i > K. Best searching efficiency is improved by avoiding oscillation using this approach.

In order to keep the most important data from the particles intact, this study explores dimensional learning, which is motivated by OLCGWO. Here, the learning exemplar x_i_ol is built by transferring knowledge from x_i_pbest, which in turn was learned from x_gbest. This opens the door for xgbest to become the standard. An example is given to show how it works. Let's pretend we're trying to find the global minimum of a five-dimensional sphere using the function f(x) = x_12+x_22+x_32+x_42+x_52T. x_ipbest= (1,0,3,2,4)T is the best possible position. Similarly, xgbest = (2,4,2,0,T) is the best possible location right now. According to the illustration, x_ipbest = 30, and f(xgbest) = 28. Next, x_i_pbest uses the global features of x_gbest as a model for improvement. The xtemp vector is a temporary one, i.e., $x^{\text{temp}} = x_i^{pbest} = (1, 0, 3, 2, 4)^{\text{T}}$.

4. RESULTS AND DISCUSSION

Agriculture 4.0 entails incorporating cutting-edge technologies into conventional farming practices to boost output and quality. Some examples of these cutting-edge innovations are the computing, AI, NFV, and SDN (Software-Defined Networking). We used and chose cutting-edge data sets featuring DDoS attack scenarios against Agriculture 4.0 technologies based on these technologies.

4.1 Performance metrics

It is crucial to carefully choose performance criteria with which to compare various machine learning and deep learning tactics. Our analysis centres on the following key performance metrics: DR, FAR, precision, F-score, recall, and accuracy. In Table 3, we see examples of four potential classifications, two of which are incorrect.

$$TNR_{BENIGN} = \frac{TN\_BENIGN}{TN\_BENIGN + FP\_BENIGN} \quad (11)$$

$$FAR = \frac{FP\_BENIGN}{TN\_BENIGN + FP\_BENIGN} \quad (12)$$

$$Precision = \frac{TP\_Attack}{TP\_Attack * FP\_BENIGN} \quad (13)$$

$$Recall = \frac{TP\_Attack}{TP\_Attack * FN\_Attack} \quad (14)$$

$$DR\_Attack = \frac{TP\_Attack}{TP\_Attack + FN\_Attack} \quad (15)$$

$$F - score = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \quad (16)$$

$$Accuracy = \frac{TP\_Attack + TN\_BENIGN}{TP\_Attack + FN\_Attack + TN\_BENIGN + FP\_BENIGN} \quad (17)$$

$$DR\_Overall = \frac{\sum TP\_Each - Attack - Type}{\sum TP\_Each - Attack - Type + \sum FN\_Each - Attack - Type} \quad (18)$$

where, TP signifies a positive result, TN a negative result, FP a positive result, and FN a negative result. In this context, a True Negative (TN) refers to data that was correctly recognized as benign, whereas a False Positive (FP) shows data that was wrongly recognized as malicious. Data that has been accurately identified as an attack is denoted by the True Positive (TP). The FN represents malicious information that was wrongly categorized as non-threatening.

Table 3. Confusion matrix

		Forecast Class	
		Negative Class	Positive Class
Class	Negative session	(TN)	(FP)
	Positive session	(FN)	(TP)

Table 4 represents that the Overall classification results of EMSVM-CGWO model on dataset-1. In this comparison we used different class labels as class. In this analysis, initially we used 70% of training data has been used. In this ratio analysis, the Binary Class attained the accuracy as 94.87 and the precision value as 88.06 and the recall value as 67.72 and finally the F1 score value as 76.56 respectively. Another, Dataset_2_class attained the accuracy as 94.16 and the precision value as 85.37 and the recall value as 84.52 and finally the F1 score value as 84.94. Also, Dataset_7_class attained the accuracy as 92.12 and the precision value as 92.32 and the recall value as 96.46 and finally the F1 score value as 94.34. After that the Dataset_13_class attained the accuracy as 93.71 and the precision value as 88.58 and the recall value as

82.9 and finally the F1 score value as 85.28 respectively. In this analysis, initially we used 30% of testing data has been used. In this ratio analysis, the Binary Class attained the accuracy as 95.34 and the precision value as 87.98 and the recall value as 69.29 and finally the F1 score value as 77.52 respectively. Another, Dataset_2_class attained the accuracy as 94.49 and the precision value as 87.08 and the recall value as 83.69 and finally the F1 score value as 85.35. Also, Dataset_7_class attained the accuracy as 92.12 and the precision value as 92.56 and the recall value as 92.65 and finally the F1 score value as 94.75. After that the Dataset_13_class attained the accuracy as 94.13 and the precision value as 89.24 and the recall value as 83.31 and finally the F1 score value as 85.87 respectively.

In another set of training of 30%, the Binary Class attained the accuracy as 95.34 and the precision value as 87.98 and the recall of 69.29 and f1-score as 77.52. Another the Dataset_2_class attained the accuracy as 94.49 and the precision value as 87.08, and the recall value as 83.69 and finally the F1 score value as 85.35. After the Dataset_7_class attained the accuracy as 92.56 and the precision value as 92.65 and the recall value as 96.95 and finally the F1 score value as 94.75. After the Dataset_13_class attained the accuracy as 94.13 and the precision value as 89.24 and the recall value as 83.31 and finally the F1 score value as 85.87 respectively.

Table 4. Overall classification consequences of EMSVM-CGWO perfect on dataset-1

Class Labels	Accuracy	Precision	Recall	F-Score
Training Set (70%)				
Binary Class	94.87	88.07	67.72	76.56
Dataset_2_class	94.16	85.37	84.52	84.95
Dataset_7_class	92.12	92.32	96.46	94.34
Dataset_13_class	93.71	88.58	82.9	85.28
Testing Set (30%)				
Binary Class	95.35	87.98	69.29	77.52
Dataset_2_class	94.49	87.09	83.69	85.35
Dataset_7_class	92.56	92.65	96.96	94.75
Dataset_13_class	94.13	89.24	83.31	85.87

Table 5 represents that the Overall classification consequences of EMSVM-CGWO perfect on dataset-2. In this analysis, we used different class label as files. Also, we used the training and testing as 70-30% ratios. In 70% of training set we used the different files as File 1 reached the accuracy value as 98.96 and the precision value as 92.24 and a recall value as 89.38 and finally the f1-score value as 90.79 respectively. Another File 3 reached the accuracy value as 98.54 and the precision value as 99.25 and a recall value as 98.86 and finally the f1-score value as 99.05. After File 5 reached the accuracy value as 99.12 and the precision value as 96.07 and a recall value as 98.80 and finally the f1-score value as 97.42. Finally, the File 7 reached the accuracy value as 98.87 and the precision value as 95.85 and a recall value as 95.68 and finally the f1-score value as 95.75.

Table 6 represents that the comparison of various Machine Learning Models. In this analysis, we take two datasets to analysis the performance at the attacks and normal. In the initial dataset attack performance of the model as ELM model reaches the precision of 85.14 and also the recall value as 85.31 and another F1-score value as 85.20. After that the LR model reaches the precision of 85.29 and also the recall value as 87.92 and another F1-score value as 85.50. After that the MLP model reaches the precision of 84.92 and also the recall value as 83.98 and another F1-score value as 84.54. Then SVM model reaches the precision of 88.67 and also the recall value as 88.06 and another F1-score value as 85.63. Also, another, EMSVM 88.78 and also the recall value as 90.15 and another F1-score value as 87.54. Another proposed EMSVM-CGWO model reaches the precision of 92.65 and also the recall value as 96.95 and another F1-score value as 94.75 respectively.

Table 5. Overall classification consequences of EMSVM-CGWO perfect on dataset-2

Class Labels	Accuracy	Precision	Recall	F-Score
Training Set (70%)				
File 1	98.96	92.24	89.38	90.79
File 3	98.54	99.25	98.86	99.05
File 5	99.12	96.07	98.80	97.42
File 7	98.87	95.85	95.68	95.75
Testing Set (30%)				
File 1	98.99	93.22	89.12	91.12
File 3	98.86	99.37	99.15	99.26
File 5	99.14	96.24	98.71	97.46
File 7	99.00	96.28	95.66	95.95

Table 6. Comparison of various Machine Learning Models

Dataset	Method	Attacks			Normal		
		Precision	Recall	F1-Score	Precision	Recall	F1-Score
Dataset-1	ELM	85.14	85.31	85.21	65.46	70.04	68.75
	LR	85.29	87.92	85.50	70.11	63.40	65.91
	MLP	84.92	83.98	84.54	70.33	66.49	67.53
	SVM	88.67	88.06	85.63	72.00	74.63	76.28
	EMSVM	88.78	90.15	87.54	73.90	74.86	77.35
	EMSVM-CGWO	92.65	96.95	94.75	87.53	76.49	81.44
Dataset-2	ELM	74.66	91.22	85.29	93.92	92.93	94.40
	LR	84.83	81.53	81.02	92.18	92.37	95.07
	MLP	85.11	74.01	79.39	91.61	93.15	92.60
	SVM	82.54	88.99	85.96	94.02	92.92	93.17
	EMSVM	83.94	91.18	88.86	95.68	93.75	94.33
	EMSVM-CGWO	96.24	98.71	97.46	96.30	94.14	95.19

After that the normal section, the ELM reached the precision proportion of 65.45 and also the recall rate of 70.04 and the F1-score value as 68.75 respectively. Another LR model reached the precision value of 70.11 and the recall value of 63.40 and the F1-score value as 65.91. And the another, MLP model reached the precision value of 70.32 and also the recall value as 66.49 and the F1-score value as 67.53 respectively. And the another, SVM model reached the

precision value of 72.00 and also the recall value as 74.63 and the F1-score value as 76.28. And the another, EMSVM model reached the precision value of 73.90 and also the recall value as 74.86 and the F1-score value as 77.35. The proposed EMSVM-CGWO model reached the precision value of 87.53 and also the recall value as 76.49 and the F1-score value as 81.44 respectively.

In the second dataset in the normal category's performance of the model as ELM model reaches the precision of 74.66 and also the recall value as 91.22 and another F1-score value as 85.29. After that the LR model reaches the precision of 84.83 and also the recall value as 81.53 and another F1-score value as 81.02. After that the MLP model reaches the precision of 85.11 and also the recall value as 74.01 and another F1-score value as 79.39. Then SVM model reaches the precision of 82.54 and also the recall value as 88.99 and another F1-score value as 85.96. Also, another, EMSVM 83.94 and also the recall value as 91.18 and another F1-score value as 88.86. Another proposed EMSVM-CGWO model reaches the precision of 96.24 and also the recall value as 98.71 and another F1-score value as 97.46 respectively.

After that the normal section in dataset-2, the ELM reached the precision degree of 93.92 and also the recall proportion of 92.93 and the F1-score value as 94.40 respectively. Another LR model reached the precision value of 92.18 and the recall value of 92.37 and the F1-score value as 95.07. And the another, MLP model reached the precision value of 91.61 and also the recall value as 93.15 and the F1-score value as 92.60 respectively. And the another, SVM model reached the precision value of 94.02 and also the recall value as 92.92 and the F1-score value as 93.17. And the another, EMSVM model reached the precision value of 95.68 and also the recall value as 93.75 and the F1-score value as 94.33. The proposed EMSVM-CGWO model reached the precision value of 96.30 and also the recall value as 94.14 and the F1-score value as 95.19 respectively.

5. CONCLUSION

In this study, we introduce an improved machine learning approach to intrusion detection. This study refined the class assignment method by developing a superior multiclass SVM model to better support multiclass classification domains and help in choosing the optimal set of limits when constructing an SVM model. Many different types of machine learning were compared and analyzed with cyber security for Agriculture 4.0 in mind. We analyze the effectiveness of each model for classification using two recent, real-world traffic datasets: CICDDoS2019 and TON_IoT. The proposed model beats prior machine learning techniques across a wide range of performance metrics. These ROC Curve, and accuracy. Most machine learning IDS techniques were also surpassed by the EMSVM-CGWO model. These methods were put to the test on dataset. The standard dataset utilized in this study is one of the study's shortcomings; future studies might benefit from using data collected in a more realistic agricultural setting. Many farmers and agricultural workers may not be adequately aware of cybersecurity threats and best practices. They may not possess the necessary expertise to identify and respond to potential cyber intrusions effectively.

Limited Resources: Agricultural environments, especially in developing regions, often have limited resources for implementing robust cybersecurity measures. This can include a lack of investment in advanced intrusion detection systems, cybersecurity personnel, and up-to-date security infrastructure. Advanced algorithms can learn from large datasets to detect anomalies, identify patterns, and adapt to evolving threats in real-time.

REFERENCES

[1] Ferrag, M.A., Shu, L., Djallel, H., Choo, K.K.R. (2021). Deep learning-based intrusion detection for distributed denial of service attack in agriculture 4.0. Electronics, 10(11): 1257. https://doi.org/10.3390/electronics10111257

[2] Vatambeti, R., Divya, N.S., Jalla, H.R., Gopalachari, M.V. (2022). Attack detection using a lightweight blockchain based elliptic curve digital signature algorithm in cyber systems. International Journal of Safety and Security Engineering, 12(6): 745-753. https://doi.org/10.18280/ijsse.120611

[3] Smmarwar, S.K., Gupta, G.P., Kumar, S. (2022). Deep malware detection framework for IoT-based smart agriculture. Computers and Electrical Engineering, 104: 108410. https://doi.org/10.1016/j.compeleceng.2022.108410

[4] Zrelli, A., Nakkach, C., Ezzedine, T. (2022). Cyber-Security for IoT applications based on ANN algorithm. In 2022 International Symposium on Networks, Computers and Communications (ISNCC), Shenzhen, China, pp. 1-5. https://doi.org/10.1109/ISNCC55209.2022.9851715

[5] Friha, O., Ferrag, M.A., Shu, L., Maglaras, L., Choo, K.K.R., Nafaa, M. (2022). FELIDS: Federated learning-based intrusion detection system for agricultural Internet of Things. Journal of Parallel and Distributed Computing, 165: 17-31. https://doi.org/10.1016/j.jpdc.2022.03.003

[6] Al Asif, M.R., Hasan, K.F., Islam, M.Z., Khondoker, R. (2021). STRIDE-based cyber security threat modeling for IoT-enabled precision agriculture systems. In 2021 3rd International Conference on Sustainable Technologies for Industry 4.0 (STI), Dhaka, Bangladesh, pp. 1-6. https://doi.org/10.1109/STI53101.2021.9732597

[7] Yazdinejad, A., Zolfaghari, B., Azmoodeh, A., Dehghantanha, A., Karimipour, H., Fraser, E., Green, A.G., Russell, C., Duncan, E. (2021). A review on security of smart farming and precision agriculture: Security aspects, attacks, threats and countermeasures. Applied Sciences, 11(16): 7518. https://doi.org/10.3390/app11167518

[8] Macherla, H., Kotapati, G., Sunitha, M.T., Chittipireddy, K.R., Attuluri, B., Vatambeti, R. (2023). Deep learning framework-based chaotic hunger games search optimization algorithm for prediction of air quality index. Ingénierie des Systèmes d'Information, 28(2): 433-441. https://doi.org/10.18280/isi.280219

[9] Kumar, R., Kumar, P., Tripathi, R., Gupta, G.P., Gadekallu, T.R., Srivastava, G. (2021). SP2F: A secured privacy-preserving framework for smart agricultural Unmanned Aerial Vehicles. Computer Networks, 187: 107819. https://doi.org/10.1016/j.comnet.2021.107819

[10] Divya, N.S., Vatambeti, R. (2023). Detecting false data injection attacks in industrial internet of things using an optimized bidirectional gated recurrent unit-swarm

optimization algorithm model. Acadlore Transactions on AI and Machine Learning, 2(2): 75-83. https://doi.org/10.56578/ataiml020203

[11] Kumar, Y.P., Babu, B.V. (2022). Stabbing of intrusion with learning framework using auto encoder based intellectual enhanced linear support vector machine for feature dimensionality reduction. Revue d'Intelligence Artificielle, 36(5): 737-743. https://doi.org/10.18280/ria.360511

[12] Peppes, N., Daskalakis, E., Alexakis, T., Adamopoulou, E., Demestichas, K. (2021). Performance of machine learning-based multi-model voting ensemble methods for network threat detection in agriculture 4.0. Sensors, 21(22): 7475. https://doi.org/10.3390/s21227475

[13] Alahmadi, A.N., Rehman, S.U., Alhazmi, H.S., Glynn, D.G., Shoaib, H., Solé, P. (2022). Cyber-security threats and side-channel attacks for digital agriculture. Sensors, 22(9): 3520. https://doi.org/10.3390/s22093520

[14] Aldhyani, T.H., Alkahtani, H. (2023). Cyber security for detecting distributed denial of service attacks in agriculture 4.0: Deep learning model. Mathematics, 11(1): 233. https://doi.org/10.3390/math11010233

[15] Vatambeti, R., Mamidisetti, G. (2023). Routing attack detection using ensemble deep learning model for IIoT. Information Dynamics and Applications, 2(1): 31-41. https://doi.org/10.56578/ida020104

[16] Alrayes, F.S., Alotaibi, N., Alzahrani, J.S., Alazwari, S., Alhogail, A., Al-Sharafi, A.M., Othman, M. and Hamza, M.A. (2023). Enhanced gorilla troops optimizer with deep learning enabled cybersecurity threat detection. Computer Systems Science and Engineering, 45(3): 3037-3052. https://doi.org/10.32604/csse.2023.033970

[17] Kilincer, I.F., Ertam, F., Sengur, A., Tan, R.S., Acharya, U.R. (2023). Automated detection of cybersecurity attacks in healthcare systems with recursive feature elimination and multilayer perceptron optimization. Biocybernetics and Biomedical Engineering, 43(1): pp.30-41. https://doi.org/10.1016/j.bbe.2022.11.005

[18] Jain, J.K., Waoo, A.A. (2023). An artificial neural network technique for prediction of cyber-attack using intrusion detection system. Journal of Artificial Intelligence, Machine Learning and Neural Network (JAIMLNN), 3(2): 33-42. https://doi.org/10.55529/jaimlnn.32.33.42

[19] Wang, X., Liu, J. (2023). A novel feature integration and entity boundary detection for named entity recognition in cybersecurity. Knowledge-Based Systems, 260: 110114. https://doi.org/10.1016/j.knosys.2022.110114

[20] Folino, G., Godano, C.O., Pisani, F.S. (2023). An ensemble-based framework for user behaviour anomaly detection and classification for cybersecurity. The Journal of Supercomputing, 79: 11660-11683. https://doi.org/10.1007/s11227-023-05049-x

[21] Kayhan, V.O., Agrawal, M., Shivendu, S. (2023). Cyber threat detection: Unsupervised hunting of anomalous commands (UHAC). Decision Support Systems, 168: 113928. https://doi.org/10.1016/j.dss.2023.113928

[22] Dairi, A., Harrou, F., Bouyeddou, B., Senouci, S., Sun, Y. (2023). Semi-supervised deep learning-driven anomaly detection schemes for cyber-attack detection in smart grids. In: Haes Alhelou, H., Hatziargyriou, N., Dong, Z.Y. (eds) Power Systems Cybersecurity. Power Systems. Springer, Cham. https://doi.org/10.1007/978-3-031-20360-2_11

[23] Sharafaldin, I., Lashkari, A.H., Hakak, S., Ghorbani, A.A. (2019). Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In Proceedings of the 2019 International Carnahan Conference on Security Technology (ICCST), Chennai, India, pp. 1-8. https://doi.org/10.1109/CCST.2019.8888419

[24] TON_IOT DATASETS. https://ieee-dataport.org/documents/toniot-datasets.