Multidisciplinary Journal of Science Technology Education and Management
Journal homepage: https://www.mjstem.com

Enhanced Feature Selection with Autoencoders and Whale Optimization Algorithm for the Classification of DDoS Attacks using XGBoost based Model

Rukayya Umar[a*], Morufu Olalere[b], Ojeniyi Joseph Adebayo[a], Adebayo Surajuddeen[b], Ismaila Idris[a]

[a]Cyber Security Science, Federal University of technology, Minna, Niger, Nigeria;
[b]Cyber Security Science, National Open University of Nigeria, Abuja, FCT, Nigeria;

[*]Correspondence: Rukayya Umar (umarrukayya1@gmail.com).

Abstract

Distributed Denial of Service (DDoS) attacks continue to pose a serious and evolving threat to the stability, availability, and reliability of network systems. With the rapid growth of Internet-of-Things (IoT) devices, cloud infrastructures, and software-defined networks (SDN), the scale and sophistication of DDoS attacks have also grown significantly. Traditional detection methods often struggle to cope with high-dimensional traffic data, redundant attributes, and the nonlinear interactions inherent in modern attack patterns. These limitations result in reduced detection accuracy, susceptibility to false alarms, and poor generalization across diverse attack types. To mitigate these cyber threats, this research work on an approach that integrates Autoencoders (AE), the Whale Optimization Algorithm (WOA), and Extreme Gradient Boosting (XGBoost). The AE component compresses raw traffic flows into latent feature representations, capturing nonlinear relationships and reducing noise. However, AE alone produces a broad feature space (64 features), which includes redundancy and non-informative attributes. To refine this space, WOA is employed as a metaheuristic feature selector, guided by a fitness function that balances classification accuracy with feature compactness. This process reduces the dimensionality to 40 highly discriminative features, ensuring efficiency without compromising information richness. Finally, XGBoost is applied as the classifier due to its robustness, scalability, and ability to handle both detection and multi classification of DDoS attacks. The proposed model was evaluated using CICDDoS2019, which capture diverse attack scenarios and is widely used in intrusion detection research. Baseline experiments with AE–XGBoost achieved 98.57%. By contrast, the proposed AE+WOA-XGBoost achieved 99.89% accuracy, 99.96% precision, 99.87% recall, 99.91% F1-score, and an AUC of 1.000, representing a 1.32% gain over the baseline and near-perfect classification performance. Beyond raw accuracy, the optimization process reduced computational overhead, improved generalization, and demonstrated consistent effectiveness across both datasets.

Keywords: DDoS, Attacks, Autoencoder, Optimization, Classification.

Received November 05, 2025 Accepted January 01, 2026

1. Introduction

The rapid expansion of the internet and interconnected digital systems has significantly enhanced global communication, business operations, and access to services (Prieto & Durán Barroso, 2024). However, this growth has also increased exposure to cyber threats. Among the most disruptive are Distributed Denial of Service (DDoS) attacks, in which attackers overwhelm a target system with excessive traffic, making it unavailable to legitimate users (Ouhssini *et al.,* 2024). Such attacks can cause service outages, financial losses, data breaches, and reputational damage (Ortet *et al.,* 2021). As internet-based services become increasingly essential, effective and efficient DDoS detection mechanisms are critical.

Traditional detection approaches rely on signature- or rule-based methods (Liu *et al.,* 2023), which identify attacks by matching traffic against predefined patterns. These methods are limited in detecting novel or evolving attacks, as they cannot recognize previously unseen behaviors (Wei *et al.,* 2021). To address these challenges, machine learning (ML) techniques have been proposed, offering adaptive and intelligent detection capabilities that can improve the classification and identification of DDoS attacks (Gebremeskel *et al.,* 2023)

A major difficulty in detecting DDoS attacks lies in accurately differentiating legitimate network traffic from malicious traffic generated during an attack. The massive volume and rapid pace of network traffic during a DDoS attack can quickly overload traditional detection systems, leading to a surge in both false-positive and false-negative results (Can & Ha, 2021). False positives can lead to normal traffic being mistakenly identified as malicious, which may disrupt network services or cause unnecessary interruptions. Conversely, false negatives allow actual attacks to go unnoticed, potentially inflicting serious harm before mitigation measures are implemented (Xu *et al.,* 2021). Therefore, it is crucial to develop more sophisticated, adaptive, and real-time detection methods capable of identifying DDoS attacks accurately while reducing false detection rates.

Machine learning has attracted considerable attention in cybersecurity due to its ability to analyze large, complex datasets and uncover patterns that are difficult for humans to identify manually (Hadi *et al.,* 2024; Liu *et al.*, 2023). In DDoS attack detection, supervised learning models such as Extreme Gradient Boosting (XGBoost) have demonstrated high effectiveness in classifying network traffic as malicious or benign (Liu *et al.,* 2023). XGBoost is a scalable gradient boosting framework that sequentially constructs an ensemble of decision trees, with each tree aiming to correct the errors of its predecessor (Araújo *et al.,* 2021). Its strengths include high predictive accuracy, robustness in high-dimensional feature spaces, and the ability to handle imbalanced datasets, making it well-suited for network intrusion detection tasks (Liu *et al.,* 2023)

Despite these advantages, XGBoost depends heavily on manually engineered features derived from raw network traffic. Such features often require significant domain expertise and may fail to capture the full complexity of evolving traffic patterns and adversarial behaviors (Ouhssini *et al.,* 2024; Shaikh *et al.,* 2024). Autoencoders (AEs) provide a promising solution by learning

compact, meaningful representations directly from data, enabling models to adapt to complex, non-linear traffic behaviors without relying solely on predefined features (Chen & Guo, 2023).

AEs are neural networks designed for unsupervised learning, compressing input data into a latent space and reconstructing it to learn efficient representations (Chen & Guo, 2023). In network traffic analysis, they can capture underlying patterns of normal behavior, producing latent features suitable for further optimization rather than relying solely on reconstruction errors for anomaly detection(Ieracitano *et al.,* 2020)

To enhance the relevance and discriminative power of these features, the Whale Optimization Algorithm (WOA) a nature-inspired metaheuristic based on the bubble-net feeding strategy of humpback whales (Sihwail *et al.,* 2024; Wang *et al.,* 2024) is employed for feature selection. WOA has demonstrated effectiveness in complex optimization tasks, including feature selection and neural network tuning (Liu *et al.,* 2023).. The optimized AE-derived features are then input into an XGBoost classifier, leveraging its high accuracy, efficiency, and robustness in high-dimensional spaces. By combining AE for unsupervised feature extraction, WOA for metaheuristic optimization, and XGBoost for supervised classification, this approach aims to improve DDoS detection accuracy and generalization against evolving attack patterns.

The main contributions of this work are threefold: (1) it demonstrates the limitations of standalone AE and other existing models in handling high-dimensional DDoS traffic; (2) it introduces the integration of WOA into the AE feature space, resulting in a compact and highly discriminative feature subset; and (3) it validates the proposed hybrid AE–WOA + XGBoost framework on two benchmark datasets, achieving state-of-the-art results while maintaining efficiency and interpretability. Collectively, these results highlight the potential of using deep learning-based feature extraction with metaheuristic optimization to design scalable, adaptive, and production ready intrusion detection systems capable of mitigating the growing threat of DDoS attacks.

The remainder of this paper is as follows. Section 2 reviews existing research related to feature selection and DDoS detection techniques. Section 3 presents the proposed comparative framework, detailing the feature selection algorithms, classifiers, and the AE+WOA–XGBoost hybrid model used. Section IV describes the dataset, preprocessing steps, and the experimental methodology adopted for evaluation. Section V discusses the results and performance metrics, highlighting comparative outcomes across models. Finally, Section VI concludes the paper and outlines future research directions.

2. Literature Review

In this section, the review of related research on Distributed Denial of Service (DDoS) attack detection techniques and the analysis of the comparative performance of various classification algorithms explored by other researchers is presented.

A. Related Literature Based on Deep Learning Approaches

More recently, deep learning and hybrid models have dominated the DDoS detection landscape. (Panggabean *et al.,* 2024) presented a hybrid architecture that combined Gated Recurrent Units (GRUs) with Neural Turing Machines (NTMs) to capture both sequential dependencies and memory-based traffic patterns. Their model demonstrated strong generalization across datasets such as BoT-IoT and UNSW-NB15, achieving 98.7% accuracy and 98.1% recall. However, the lack of interpretability and explicit feature optimization limited its practical applicability. The present study addresses these gaps by incorporating Autoencoders (AE) for feature extraction and Whale Optimization Algorithm (WOA) for feature selection, thereby offering both adaptability and interpretability.

Efendi, (2025) proposed a multi-stage framework for DDoS detection that combined DBSCAN clustering, SMOTE oversampling, Artificial Neural Networks (ANN), XGBoost, and Particle Swarm Optimization (PSO). In simulated DDoS scenarios, this ensemble achieved 96.83% accuracy, 93.23% sensitivity, and 96.13% precision, demonstrating strong performance under controlled conditions. However, model's complexity introduced notable challenges in terms of scalability and computational demands, especially for real-time applications. Although the model delivered competitive accuracy, its multi-layer structure made it resource intensive. In contrast, the current AE+WOA-XGBoost method enhances this approach by integrating dimensionality reduction and optimization into a single, streamlined framework, providing greater efficiency and scalability.

In the same year, Liu *et al.,* (2025) focused on enhancing XGBoost for Named Data Networking (NDN), specifically targeting Interest Flooding Attacks (IFA) and Cache Pollution Attacks (CPA). Their modified XGBoost classifier demonstrated strong performance metrics, including high accuracy and robustness in detecting NDN-specific DDoS attacks. While the study showcased the adaptability of tree-based models in next-generation networks, it did not incorporate dimensionality reduction or feature optimization, which limited scalability under high-dimensional traffic data. In comparison, the present research addresses this gap by embedding Autoencoders for feature reduction and WOA for optimization, thus improving computational efficiency without sacrificing detection accuracy.

Alfatemi *et al.,* (2024) proposed a Deep Residual Neural Network (ResNet) with SMOTE to solve issues related to class imbalance in DDoS datasets. The model achieved 99.98% accuracy on the CICIDS dataset, showcasing exceptional performance in identifying attack traffic. Despite its accuracy, the complexity of ResNet architecture led to high computational costs, rendering the model impractical for real-time deployment in resource-constrained systems. By contrast, the present study opts for lightweight Autoencoders, which retain the ability to capture complex traffic patterns while ensuring computational efficiency.

Hacılar *et al.,* (2024) introduced a Deep Autoencoder paired with Artificial Bee Colony optimization to detect anomalies in DDoS traffic. When evaluated on the UNSW-NB15 dataset, their framework notably lowered false alarm rates and achieved over 98% accuracy, highlighting the effectiveness of integrating deep feature extraction with swarm intelligence

optimization. Nevertheless, the lack of a separate classification component restricted interpretability and overall scalability. To overcome these limitations, the present study employs an Autoencoder for feature extraction while incorporating XGBoost for classification, enabling both strong feature representation and transparent, interpretable decision-making.

B. Related Literature Based on Machine Learning Approaches

Berríos *et al.,* (2025) evaluated several machine learning models, including XGBoost, Random Forest, and LSTM, on the CIC-DDoS2019 and N-BaIoT datasets. XGBoost achieved the highest accuracy (over 97%) and F1-scores, outperforming other ensemble approaches, though no optimization techniques were applied. The current study addresses this limitation by incorporating WOA for feature refinement.

In Edge-IIoT environments, an ensemble-based approach scenarios (Laiq *et al.,* 2023) demonstrated that XGBoost outperformed traditional classifiers like SVM and Decision Tree in both accuracy and latency, making it a strong candidate for real-time DDoS detection in resource-limited settings. However, without deep feature extraction, its performance may decline when handling more complex traffic patterns.

Varghese & Muniyal, (2021) proposed a real-time statistical anomaly detection system integrated into the data plane of Software Defined Networking (SDN). Operating as part of an Intrusion Detection System (IDS), it relies on preset statistical thresholds to detect anomalies. While capable of near real-time detection, its lack of learning ability limits adaptability to new or sophisticated DDoS attacks. In contrast, the AE–WOA–XGBoost framework dynamically learns from evolving traffic patterns, providing improved adaptability and robustness against previously unseen attack scenario.

Pontes *et al.,* (2021) proposed an Energy-based Flow Classifier (EFC) that utilizes inverse statistical analysis of benign traffic to assign anomaly scores and classify DDoS attacks. While achieving a 97.5% F1-score, the model's generalization capacity remains limited as it depends heavily on the statistical distribution of known traffic types, and key performance metrics were not comprehensively reported. Compared to AE–WOA–XGBoost, EFC is less resilient in high-variability environments due to its static statistical basis, while the proposed hybrid model offers broader generalizability via data-driven feature learning.

Moustafa & Slay (2017) examined several machine-learning classifiers such as Decision Trees, Naive Bayes, and SVM for DDoS detection using the UNSW-NB15 dataset. Although SVM delivered strong accuracy, its performance deteriorated with high-dimensional feature spaces. In contrast, our work employs XGBoost, which is well suited for efficiently managing large and complex feature sets.

Oyelakin, (2024) investigated an ensemble XGBoost model for intrusion detection on the CICIDS2017 dataset, using processed traffic features and feature selection based on XGBoost's internal importance scores. Across eight dataset segments, the model reached about 98% accuracy, with precision, recall, and F1-score all at 0.98, and an AUC-ROC of 0.99—indicating robust detection capability. Unlike earlier methods that depend on manual or basic

feature-selection strategies, this study adopted classifier-guided feature reduction but did not integrate dynamic meta-heuristic optimization. Additionally, it did not explore real-time operational challenges such as latency or streaming analysis. While the results reinforce XGBoost's effectiveness for DDoS detection, our research advances this line of work by incorporating the Whale Optimization Algorithm to enable automatic, adaptive feature selection, thereby enhancing generalization and efficiency in real-world environments.

(Kaur *et al.,* 2024) conducted a study on cyberattack detection that focuses on enhancing Intrusion Detection System performance through feature selection. Their work evaluates ensemble machine-learning models—including XGBoost, Decision Trees, and Random Forest—using the NSL-KDD and CICIDS2018 datasets. By refining the feature set, the study demonstrates improved computational efficiency and reports consistently high accuracy levels between 98% and 99%, underscoring the strong capability of these methods for intrusion detection.

C. Related Literature Based on Hybrid Approaches

Wei *et al.,* (2021) introduced AE-MLP, a hybrid deep-learning framework that integrates an Autoencoder for automated feature extraction with an MLP classifier, achieving strong performance on the CICDDoS2019 dataset with an average accuracy of 98.34% and an F1-score of 98.18%. The model's effectiveness stems from its ability to learn latent feature representations without manual intervention and then classify specific DDoS attack categories such as SYN, UDP, MSSQL, NetBIOS, and LDAP. However, AE-MLP does not include an intermediate feature-optimization stage; the latent features produced by the Autoencoder are passed directly to the classifier, which may retain redundant or irrelevant information. Additionally, its evaluation is limited to CICDDoS2019, raising concerns about overfitting and the absence of cross-dataset validation (Wei et al., 2021).

Maseer *et al.,* (2021) assessed several classical machine-learning classifiers including KNN, Naive Bayes, Random Forest, and SVM using the CICDDoS2017 dataset, reporting high accuracy results ranging from 98.86% to 99.54%. Despite these strong outcomes, the study does not provide precision, recall, or F1-score metrics and lacks a multicass analysis, which may limit interpretability and generalization across different attack types.

Similarly, Ullah & Mahmoud, (2020)) applied Naive Bayes, Logistic Regression, Decision Tree, and Random Forest models to IoT-focused botnet datasets and achieved near-perfect F1-scores between 99.99% and 100%. However, the IoT-specific nature of the dataset restricts generalizability to broader network environments, and the absence of separately reported precision and recall further limits the evaluative depth of the study.

Another study used XGBoost for DDoS detection in an SDN environment with CICDDoS2019 and achieved 99.9% accuracy. Nonetheless, the study does not report precision, recall, or F1-scores and is constrained by its SDN-specific scope and the lack of multiclass or attack-type differentiation Alamri & Thayananthan, (2020). Likewise, Parfenov *et al.,* (2020) evaluated Gradient Boosting and CatBoost on CICDDoS2019, obtaining F1-scores of 96.8% and 96.9%,

respectively, but without providing precision or recall values and without conducting multiclass or real-time testing.

Shieh *et al.,* (2021) combined Bi-directional LSTM with Gaussian Mixture Models for intrusion detection on the CICDDoS2019 dataset, achieving 98% accuracy. However, precision, recall, and F1 scores were not reported, and the study did not consider multiclass classification or real-time performance. Similarly, (Rehman et al., 2021) proposed using Gated Recurrent Units (GRU) for IDS on CICDDoS2019, achieving accuracies between 99.69% and 99.94%, yet precision, recall, F1 scores, and multiclass evaluation were not addressed.

Samom *et al.,* (2021) applied a Multi-Layer Perceptron (MLP) for IDS on CICDDoS2019, reporting 99.92% accuracy. Like the previous studies, it lacked precision, recall, F1 metrics, and did not provide attack-type-specific or multiclass results. Niyaz *et al.,* (2015) employed a Sparse Autoencoder for IDS on the NSL-KDD dataset, attaining 88.39% accuracy, but did not report other performance metrics; additionally, reliance on an older dataset may reduce relevance to contemporary threats.

Agarwal *et al.,* (2022) tested their FS-WOA–DNN approach using a cloud storage dataset, integrating Whale Optimization Algorithm for feature selection with a Deep Neural Network for classification. The model achieved 95.35% accuracy, but precision, recall, and F1 scores were not provided. The absence of validation on public datasets and detailed performance metrics limits generalizability and benchmarking.

Singh & De, (2017) evaluated an ensemble feature selection method with an MLP classifier on the CAIDA 2007 dataset, reaching 98.3% accuracy. However, the study did not include precision, recall, or F1 metrics, relied on an older dataset, and did not conduct multiclass or real-time testing, restricting its applicability to modern attack scenarios.

Chanu *et al.,* (2023) employ various public benchmark datasets (not individually named) to test their voting-based hybrid feature selection and MLP-GA classifier, reporting 98.8% accuracy, 0.6% false positive rate, but not precision, recall, or F1 scores. The main limitation is the absence of detailed per-class metrics and potential overfitting due to lack of cross-dataset validation.

Zhou *et al.,* (2022) introduce SAFE, tested on datasets including IoT traffic, but do not specify the dataset name. The system achieves high accuracy and efficiency, but precision, recall, and F1 scores are not provided. Limitations include lack of detailed metric reporting and potential dataset bias.

Shohan *et al.,* (2024) proposed a live DDoS detection framework using 1D-CNN for feature extraction and Random Forest and MLP for classification. Their model demonstrated real-time performance compatibility. However, the absence of metaheuristic tuning is improved upon in this current research through WOA.

Zhou *et al.,* (2021) worked on a DDoS detection technique that combined AE for feature reduction and Support Vector Machines for classification. While promising, the SVM model

lacked scalability for real-time analysis. Our study improves upon this by replacing SVM with XGBoost, which is both faster and more accurate for large-scale classification tasks.

Wang *et al.*, (2024) proposed a hybrid IDS using metaheuristic optimization (WOA) for feature selection, paired with deep neural networks. The WOA effectively reduced redundant features and improved detection speed and accuracy. However, the model did not leverage any encoder-based feature extraction, which could further refine performance. By integrating AE and WOA before classification with XGBoost, the current study seeks to achieve superior feature engineering and classification performance.

3. Data and Methodology

This study's phase consists of five stages: input, preprocessing, autoencoder training, feature optimization with WOA, and optimized feature output, each constituting its own component. These stages are discussed below in detail.

3.1 Input Layer

In the first stage of this process, the CICDDoS2019 was employed. It contains a sample of 88 features with data types as follows float64 (45), int64 (37) & object (6) and 500 63112 data points (containing benign and simple service discovery protocol (SSDP) type DDoS attacks. The details of the dataset are presented in Table 1. Furthermore, it has been validated and used by numerous researchers. This stage serves as the foundation of the framework and provides the raw traffic features to the preprocessing stage.

Table 1: Description of the Dataset

Dataset	Total number of features	Benign	Malicious	Total
CICDDoS2019	**88**	**113,828**	**70,313,809**	**70,427,637**

3.2 Data Preprocessing Stage

The data underwent three sequential processes in this stage: data cleaning, transformation, and preliminary feature engineering.

i. Data Cleaning: Missing values, redundant entries, and inconsistent records were removed or corrected to improve the quality of the dataset.

ii. Transformation: To ensure that the features were comparable, the dataset was normalized using a Min-Max Scaler that scaled values between 0 and 1:

$$\text{MinMaxScaler}(v_i') = \frac{x_i - \min_A}{\max_A - \min_A} \cdot \left(\text{new}_{\max\ A} - \text{new}_{\min\ A} \right) + \text{new}_{\min\ A} \qquad (1)$$

where x_i represents the i^{th} feature value, $\min_A$ and $\max_A$ denote the minimum and maximum values of a feature, while $new\_\max_A = 1$ and $new\_\min_A = 0$.

iii. Feature Engineering: Before deep feature learning, irrelevant or constant-value features were removed to reduce redundancy and computational burden. The preprocessed dataset was then divided into 80% training and validation and 20% testing for subsequent modeling.

3.3 Autoencoder (AE) Stage

At this stage, an autoencoder was employed to learn compact, latent feature representations from the high-dimensional network traffic data.

i. Encoder Operation: The encoder compresses the input data into a lower-dimensional latent representation:

$$h = f(W_e X + b_e) \tag{2}$$

where W_e and b_e represent the encoder weights and bias, respectively, and f is the ReLU activation function.

ii. Decoder Operation: The decoder reconstructs the input data from the latent space:

$$\hat{X} = g(W_d h + b_d) \tag{3}$$

where W_d and b_d are decoder parameters, and g is the activation function (Sigmoid).

iii. Loss Function: The AE was trained by minimizing reconstruction error using Mean Squared Error (MSE):

$$L = \frac{1}{n}\sum_{i=1}^{n} \left\| X_i - \hat{X}_i \right\|^2 \tag{4}$$

This stage ensures that the network traffic data is represented in a compressed but information-rich feature space.

3.4 Whale Optimization Algorithm (WOA) Stage

While AE extracts deep latent features, not all of them contribute equally to detection. Therefore, the Whale Optimization Algorithm (WOA) was integrated for feature selection and optimization. WOA mimics the bubble-net hunting strategy of humpback whales, alternating between exploitation (encircling prey) and exploration (searching for new prey).

i. Encircling Prey: Whales update their positions towards the best feature subset:

$$X(t+1) = X^*(t) - A \cdot |C \cdot X^*(t) - X(t)| \tag{5}$$

ii. Spiral Updating (Bubble-net): With probability p, whales update their position using:

$$X(t+1) = D' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t) \tag{6}$$

where $D' = |X^*(t) - X(t)|, b$ is a constant, and l is a random number in $[-1,1]$.

iii. Fitness Function: The best feature subset was determined by maximizing accuracy while reducing the number of selected features:

$$\text{Fitness } = \alpha \cdot \text{ Accuracy } - \beta \cdot \frac{|S|}{|F|} \tag{7}$$

where S is the selected feature subset, F is the full set of features, and α, β are balancing.

The complete workflow, summarized in Algorithm 1, begins with loading the CICDDoS2019 dataset, which contains 70,313,809 traffic records and 88 features. The raw data undergoes preprocessing, where all numerical features are normalized to the range [0, 1] to ensure stable gradient updates, and nonpredictive identifiers are removed The dataset is then partitioned using a 70% training, 15% validation, and 15% testing split, with stratified sampling applied to preserve class distributions. To address class imbalance in the dataset, class weighting is incorporated during XGBoost training to prevent bias toward majority attack categories

In the first stage, an Autoencoder (AE) is configured with an encoder–decoder architecture, latent dimension, learning rate, and training epochs. The AE is trained in mini-batches, where each batch is compressed and reconstructed. After exploring several architectures we selected an AE with one hidden layer of 32 neurons and a 64-dimensional latent bottleneck because this configuration offered a good trade-off between reconstruction accuracy and compression. The resulting compressed latent matrix serves as the input to the second stage, where the WOA performs optimized feature selection (Table 2).

Table 2: AE-WOA for feature selection

Algorithm 1: AE-WOA for feature selection	
Input:	Input Training data $X = \{x_1, x_2, ..., x_L\}$, Labels $Y = \{y_1, y_2, ..., y_L\}$
Output:	Optimized latent feature matrix Z^*
Steps	
1	Begin
2	Preprocessing: Normalize the input features in X using Min-Max scaling: $x'_{i,j} = \frac{x_{i,j} - \min(x_j)}{\max(x_j) - \min(x_j)}$
3	Autoencoder Initialization: Initialize encoder $E\phi$, decoder D_θ, latent dimension m, learning rate α, and number of epochs T

3.1 AE Training:

For each mini batch

Encode: $z_i = E_\phi(x_i')$
Decode: $\hat{x}_i = D_\theta(z_i)$
Compute loss: $\mathcal{L}_{AE} = \frac{1}{k}\sum \|x_i - \hat{x}_i\|^2$
Update ϕ and θ using gradient descent

3.2 Latent Feature Extraction: After AE is trained, obtain latent matrix

$\mathbf{Z} = \{E_\phi(x_1'), \ldots, E_\phi(x_n')\}$

4 WOA Initialization:

Set whale population P , number of iterations I , and objective function $J(.)$

4.1 Feature Optimization (WOA Loop):

For each whale solution $S_i \in \{0,1\}^m$

5 Select optimal subset: Choose the best solution

$\mathbf{S}^* = \arg\max J(\mathbf{S}_i)$

6 Generate Output: Compute optimized features

$\boldsymbol{Z}^* = \mathbf{Z} \odot \mathbf{S}^*$

7 End:

Return Z^* to the classier

In the second stage, the WOA is applied to refine these latent features. Each whale encodes a binary feature mask representing a candidate subset of the latent space. For every candidate, the selected features are evaluated using a fitness function based on validation accuracy and feature sparsity. Through WOA's spiral bubble-net search process, the population iteratively explores and improves candidate subsets. The whale with the best fitness value defines the optimal feature mask, which is applied to the latent matrix to generate the optimized feature set Z^*.

In the final stage, the optimized feature set $Z^*$is fed into the XGBoost classifier for DDoS detection. Because WOA operates only during the offline optimization phase, the deployed system requires only the trained encoder and the XGBoost model, ensuring fast inference suitable for real-time detection environments. Figure 1 shows AE-WOA framework for optimized feature selection.

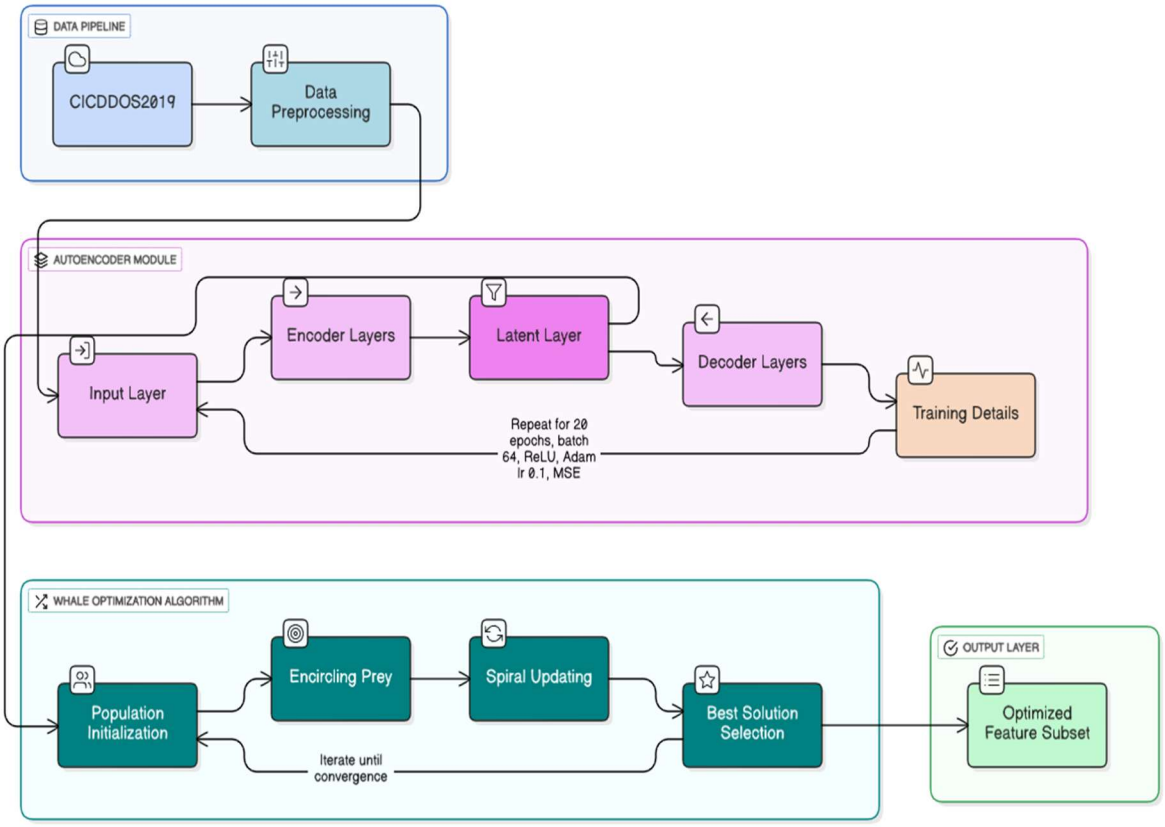

Figure 1: AE-WOA Framework for Optimized Feature Selection

3.5 Detection and Classification Phase with Xgboost

During training, the preprocessed CICDDoS2019 dataset cleaned of missing values, duplicates, and irrelevant identifiers is label-encoded and normalized using Min–Max scaling, The dataset is then partitioned using a 70% training, 15% validation, and 15% testing split, with stratified sampling applied to preserve class distributions. An Autoencoder (AE) is trained on the 88 input features to learn compact representations, using an encoder–decoder architecture with a 64-dimensional latent space; after training, only the encoder is retained. The Whale Optimization Algorithm (WOA) is then applied to the latent matrix, where each whale represents a binary mask over the 64 latent features. Candidate subsets are evaluated using a fitness function based on validation accuracy and feature sparsity, and through iterative refinement WOA produces an optimized subset of 40 discriminative features. These selected features are used to train the XGBoost classifier, incorporating class weighting to address class imbalance. The final model consisting of the encoder and XGBoost classifier is saved for deployment. During testing and real-time detection, unseen traffic samples undergo the same preprocessing and are encoded via the trained encoder; the WOA derived feature mask is applied to obtain the optimized feature vector, for onward classification task. Table 3 represents the algorithm of the developed model while Figure 2 shows the overall framework and each phase explained below.

Table 3: AE-WOA+XGBoost Algorithm

Pseudocode: AE-WOA-XGBoost for DDoS Detection and Classification

Input Training dataset $X = \{x_1, x_2, \dots, x_n\}$

Testing dataset $X' =$ Figureg labels $Y =$ Figureing labels $Y' = \{y'_1, y'_2, \dots, y'_m\}$

Encoder E_ϕ, Decoder D_θ, Whale Optimization Algorithm $\mathcal{W}$, XGBoost classifier $\mathcal{X}$

Output Predicted output labels $\hat{Y}'$:Including detection (attack vs. normal) and multi-class classification of attack type

Phase1 Begin

Phase 2 *//Feature Extraction using Autoencoder*

$\phi, \theta \leftarrow$ Initialize parameters

for each training iteration

do

Sample mini-batch $\{X_1, \dots, X_k\} \subset X$

Compute reconstruction loss

$$V = \frac{1}{k}\sum_{i=1}^{k} \left\| X_i - D_\theta\left(E_\phi(X_i)\right)\right\|^2$$

Update ϕ, θ using gradient descent on V

end for

Phase 3 *//Feature Optimization using WOA*

For each $x \in X$

encode features $a = E_\phi(x)$

Construct feature set $A = \{a_1, a_2, \dots, a_n\}$

Optimize features $A^* = \mathcal{W}(A, Y)$

Phase 4 *//Detection and Classification using XGBoost*

$\mathcal{X}$ *on* A^* *and* Y

Phase 5 *//Testing and Evaluation*

for each test sample $x' \in X'$

do

Encode features $a' = E_\phi(x')$

Optimize $a'^* = \mathcal{W}(a')$

Predict label $\hat{y}' = \mathcal{X}(a'^*)$

end for

It Return predicted labels $\hat{Y}' = \{\hat{y}'_1, \dots, \hat{y}'_m\}$

Phase 6 End

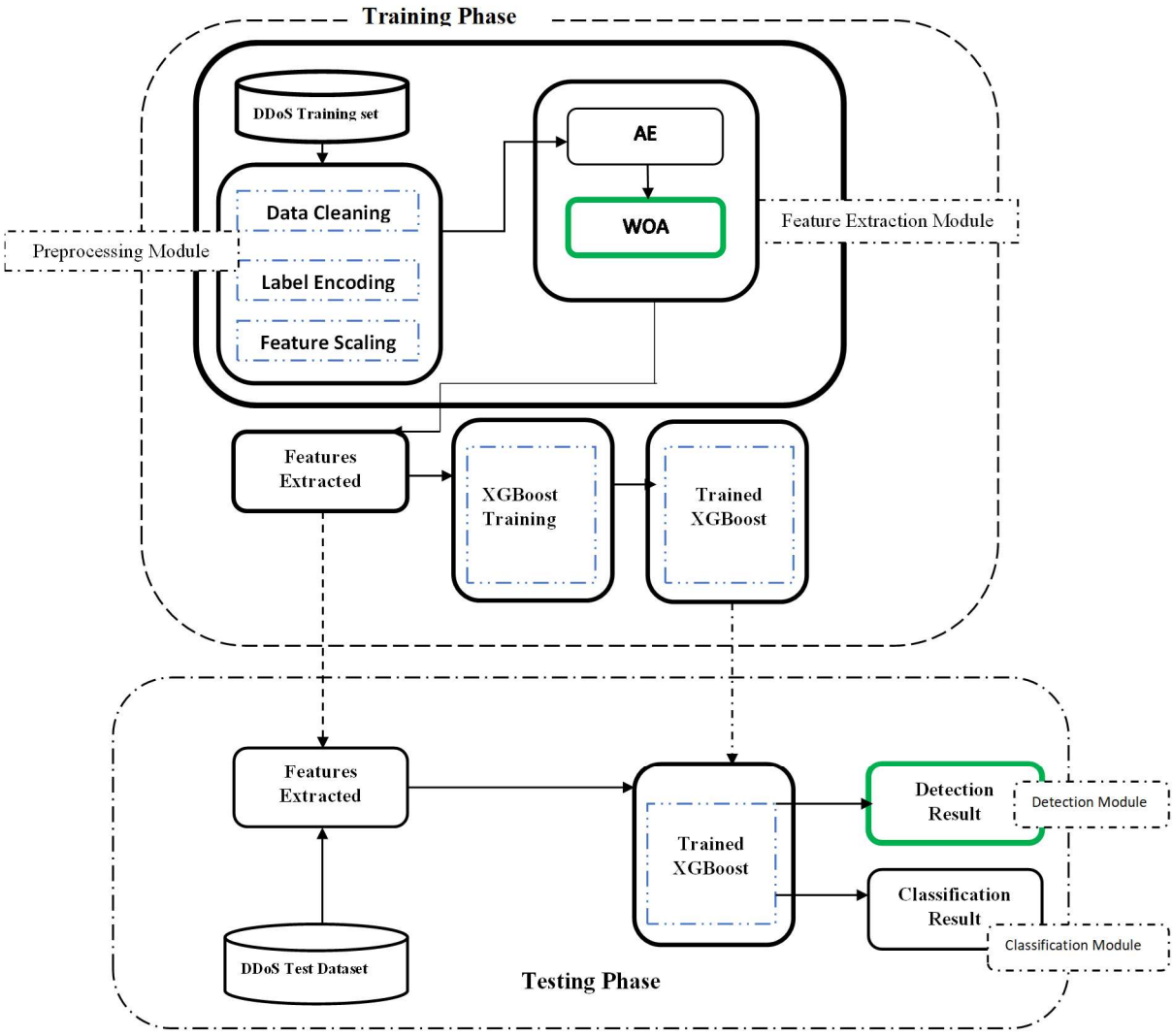

Figure 2: AE-WOA+XGBOOST Framework for Detection and Classification of DDoS Attacks

3.4.1 Training phase

The system trains on the CICDDoS2019 dataset containing both DDoS and normal traffic samples.

3.4.1.1 Preprocessing module

To ensure clean and consistent data, the dataset was preprocessed through

i. Data Cleaning: Removing missing, duplicate, or irrelevant records.

ii. Label Encoding: Converting categorical labels (“Attack”, “Normal”) to numeric form.

iii. Feature Scaling: Normalizing features using min–max normalization.

3.4.1.2 Feature Extraction Module

A two-stage hybrid process was employed to reduce dimensionality and enhance feature quality:

i. Autoencoder (AE): Trained in unsupervised mode for dimensionality reduction. The optimal architecture uses 88 input features, a hidden layer with 32 neurons, and a 64-feature latent space.

ii. Whale Optimization Algorithm (WOA): Refines AE's 64 latent features to 40 optimal ones by mimicking humpback whale bubble-net hunting behavior, selecting the most informative subset.

3.4.1.3 XGBoost Training

The 40 optimized features are used to train the XGBoost classifier to differentiate DDoS from normal traffic.

3.4.1.4 Trained Model

The final trained XGBoost model was saved for deployment in the testing phase.

3.4.2 Testing Phase

In this phase, the unseen traffic data is processed and evaluated.

3.4.2.1 Preprocessing & Feature Extraction

The same preprocessing and AE+WOA feature extraction steps were applied to ensure consistency.

3.4.2.3 Detection & Classification Modules

i. Detection Result: Identifies whether the sample is a DDoS attack.

ii. Classification Result: Specifies the attack type among five DDoS categories from the CICDDoS2019 dataset.

4. Experimental Setup and Performance Evaluation

The experiments were implemented in Python 3.10 using Jupyter Notebook. The dataset CICDDoS2019 was preprocessed and evaluated using a combination of deep learning and machine learning libraries. The experiments were conducted on a workstation equipped with an Intel Core i7 CPU, 32 GB RAM, and an NVIDIA RTX 3060 GPU (12 GB VRAM) to accelerate training. The data were split into training, validation and testing subsets, with preprocessing handled by StandardScaler for feature normalization and LabelEncoder for categorical encoding. The Autoencoder was trained for 20 epochs with a batch size of 64, using the Adam optimizer (learning rate = 0.1) and mean squared error (MSE) loss function. Table 4 shows the experimental parameters.

Table 4: Experimental Parameters

Algorithm	Hyperparameters	Values
AE	Activation Function	Relu
	Epochs	20
	Batch size	64
	Optimizer	Adam
	Learning rate	0.01
	Loss	MSE
WOA	Polpulation Size	30
	Max iterations	100
	Coeeficients Scalar (a)	Linearly decreases 2 → 0
	Spiral constant (b)	1
	Probability (p)	Random in [0.1
	Fitness function	Classification error + feature penalty]
XGBoost	Learning rate	0.01
	Max depth	6
	Objective function	Multi:softmax
	Number of classes	5
	Evaluation metrics	Classification error

To evaluate the performance of our proposed model, we use the following performance metrics.

Precision: it represents the ratio of correctly classified DDoS attack types to the total number of instances the model labeled as that attack type. Equation 7 calculates the precision of the model.

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

Recall: This measures how many of the true DDoS attack instances in the dataset were successfully identified by the model.

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

Accuracy quantifies the overall correctness of the prediction algorithm by comparing the number of correct class predictions to the total number of true class labels in the dataset. Equation 10 provides the formula used to compute this metric

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (10)$$

The F1-Score integrates precision and recall by calculating their harmonic mean, providing a single metric that balances both aspects of the classifier's performance.

$$F1 - score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \quad (11)$$

5. Experimental Results

This section presents the results of the combination of the Autoencoder (AE) with the Whale Optimization Algorithm (WOA) for feature selection and classification using XGBoost on the

CICDDoS2019 dataset. The Autoencoder is employed to learn compressed representations of the original feature set, effectively reducing dimensionality while retaining critical information. Subsequently, WOA is applied to identify and select the most significant features from the encoded data. Figure 3(a) illustrates the importance scores of the 64 features generated by the Autoencoder, while Figure 3(b) shows the 40 features selected from the extracted features through the AE–WOA process. Features such as Feature 30, Feature 39, and Feature 10 exhibit the highest importance scores, exceeding 0.8, indicating their key role in accurately detecting and classifying DDoS attacks. Mid-range features, including Feature 4, Feature 15, and Feature 23, contribute moderately, whereas lower-scoring features, such as Feature 24 and Feature 22, have minimal influence. This ranking highlights the effectiveness of the AE + WOA feature selection framework in isolating a smaller, more relevant set of features, ultimately enhancing the XGBoost classification performance while reducing computational complexity and eliminating redundant information.

(a) (b)

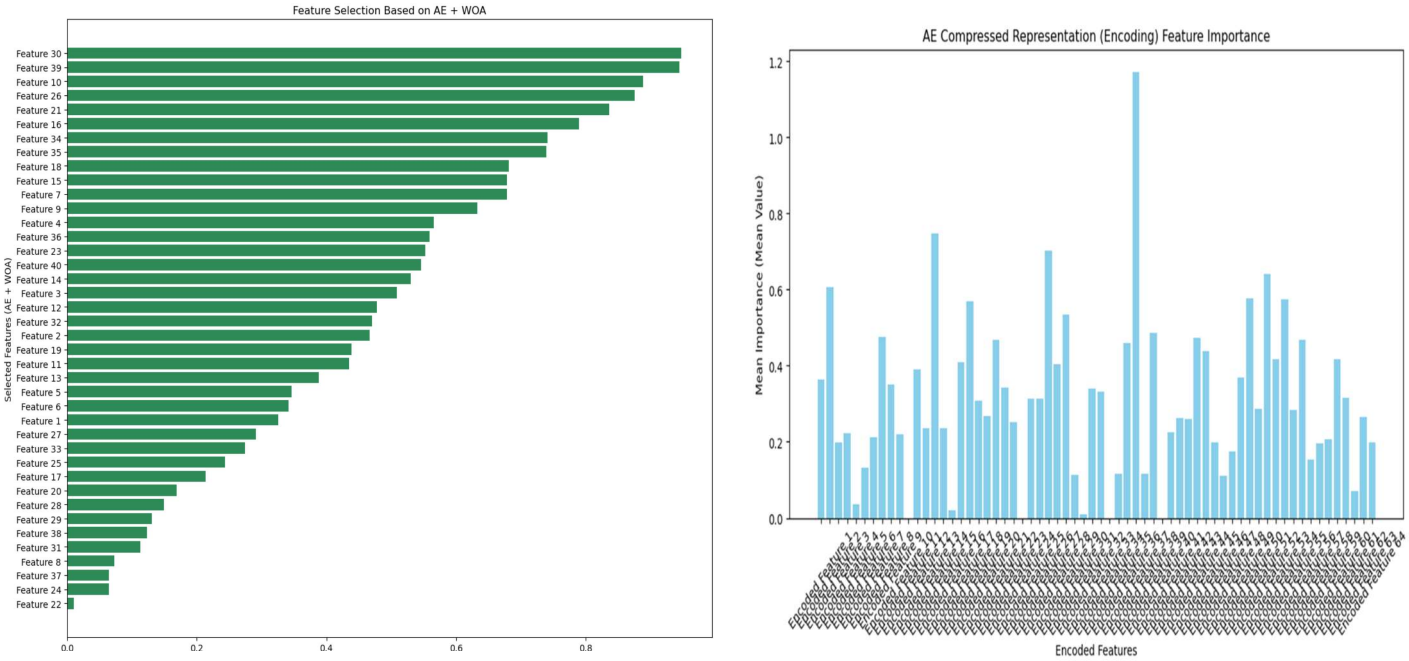

Figure 3(a) and (b): Feature Representation with AE and Feature Selection with AE+WOA

The performance of the XGBoost model is assessed using optimized features derived through the combined use of an Autoencoder (AE) and the Whale Optimization Algorithm (WOA). The model was trained and tested using these reduced-dimensional features to ensure faster computation and improved generalization. Table 5 shows the model performance of five DDoS attacks types.

Table 5: Classification Performance of DDoS attacks using optimized features from the AE-WOA

Class	Accuracy	Precision	Recall	F1 Score	AUC
NetBIOS	99.97	98.53	98.53	98.53	99.99
LDAP	99.99	95.76	93.91	94.83	99.96
MSSQL	99.15	94.71	97.57	96.12	99.86
Syn	99.88	99.91	99.90	99.90	1.000
UDP	99.40	99.40	97.96	98.68	99.99

Overall, the model exhibits outstanding performance across all evaluated attack classes, NetBIOS, LDAP, MSSQL, SYN, and UDP, achieving consistently high scores in accuracy, precision, recall, F1-score, and AUC.

For NetBIOS attacks, the model achieves a near-perfect accuracy of 99.97%, with precision and recall both at 98.53%, and an AUC of 99.99%. These results indicate that the model is highly effective in accurately identifying NetBIOS-related traffic. In the case of LDAP attacks, the accuracy rises slightly to 99.99%, with an impressive AUC of 99.96%. However, precision and recall drop to 95.76% and 93.91%, respectively, suggesting a few more misclassifications compared to other classes.

The model also performs well in detecting MSSQL attacks, recording an accuracy of 99.15%. It achieves a high recall of 97.57%, demonstrating its ability to identify most MSSQL attacks, while a precision of 94.71% reflects a higher false positive rate. Nonetheless, the F1-score of 96.12% and AUC of 99.86% affirm its strong overall performance in this category.

SYN attacks are detected with exceptional accuracy (99.88%), and the model attains nearly identical precision, recall, and F1-score values of 99.90%. Notably, the AUC reaches a perfect score of 1.000, underscoring the model's reliability and robustness in identifying SYN-based threats. For UDP attacks, the model maintains a high accuracy of 99.40% and a precision of 99.40%. Although the recall is slightly lower at 97.96%, the model still achieves a strong F1-score of 98.68% and an AUC of 99.99%.

Figure 4(a) displays the progression of training and validation loss across 20 epochs, highlighting the model's efficient learning and convergence. At the start, there is a steep drop in training loss, indicating that the model quickly adapts to the training data. By around the third epoch, both training and validation losses have significantly decreased and begin to level off near zero. There is close alignment between the two loss curves which suggests strong generalization and minimal overfitting. With loss values consistently remaining below 0.001, the model demonstrates high performance and accuracy, indicating that it has effectively learned the patterns in the dataset. This outcome points to both a well-optimized model and a clean, structured dataset. While Figure 4(b) depicts the training and validation accuracy over 20 epochs, illustrating the model's learning progression. Initially, the training accuracy increases rapidly, surpassing 90% by the 5th epoch and continuing to improve, reaching nearly 100% by the final epoch. The validation accuracy also shows an upward trend, though with more noticeable fluctuations between epochs 4 and 7, possibly indicating variability in generalization or sensitivity to specific data batches. Despite these fluctuations, both curves converge towards the end, demonstrating excellent accuracy and strong generalization performance. The close alignment in the final epochs suggests the model effectively learned from the training data without overfitting.

(a) (b)

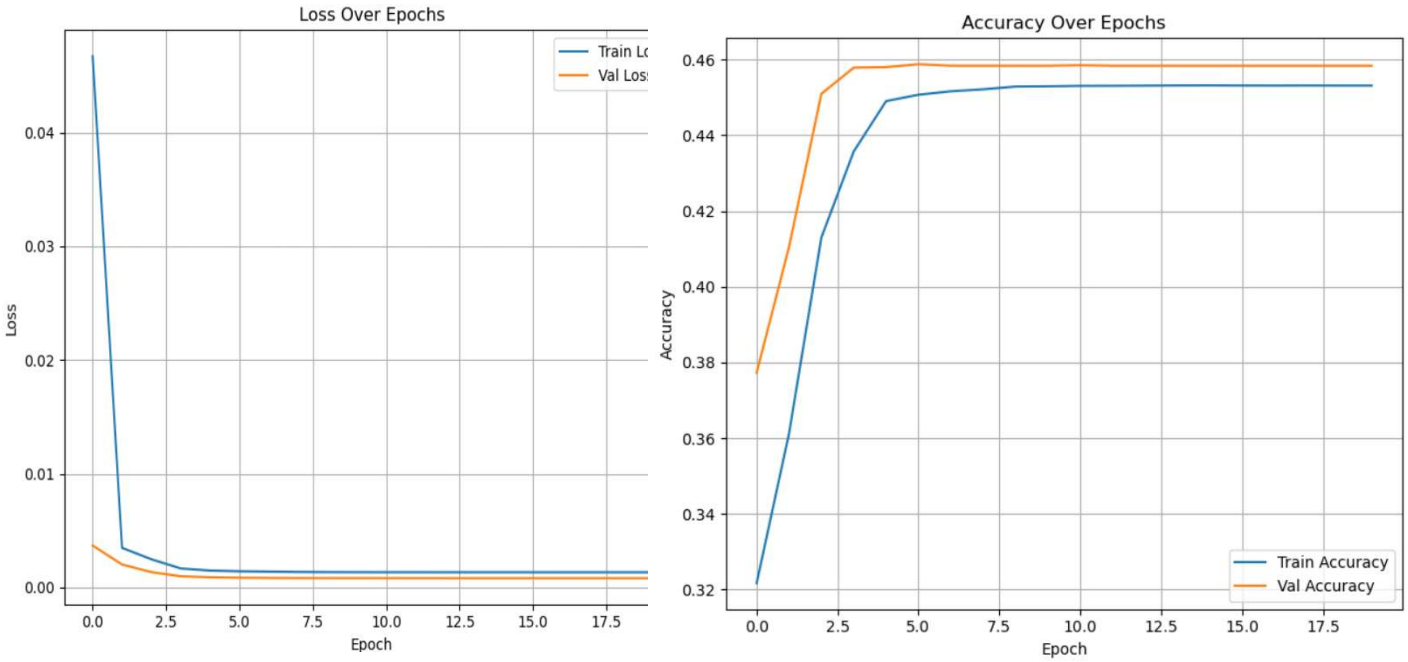

Figure 4 (a)and (b): Validation & Training Loss Over Epochs and Accuracy over Epochs for XGBoost DDoS Attack Classification Model

Table 6 shows the comparison between Wei *et al.* (2021) and the current study which demonstrates a clear improvement in classification performance across all evaluated attack types; LDAP, MSSQL, NetBIOS, SYN, and UDP. Overall, the current study achieves better accuracy, precision, recall, and F1-scores, indicating a more robust and reliable detection model. Specifically, LDAP accuracy improved slightly from 98.35% to 99.99%, though precision and recall saw minor decreases, suggesting a small trade-off in classification balance. MSSQL and NetBIOS attacks show consistent performance gains across all metrics, with the latter achieving a notable F1-score increase from 96.42 to 98.53, reflecting enhanced detection consistency. The most significant advancement is observed in SYN attacks, where the model achieves near perfect performance with an F1-score of 99.9, indicating exceptional robustness. Similarly, UDP attack detection improved substantially, with accuracy rising from 97.49% to 99.4% and F1-score from 95.4 to 98.68.

Table 6: Classification Performance of DDoS attacks using the optimized features from the AE-WOA

Attack Type / Class	Accuracy (Wei et al., 2021)	Accuracy (This study)	Precision (Wei et al., 2021)	Precision (This study)	Recall (Wei et al., 2021)	Recall (This study)	F1-score (Wei et al., 2021)	F1-score (This study)
LDAP	98.35	99.99	96.63	95.76	94.74	93.91	95.57	94.83
MSSQL	98.19	99.15	93.59	94.71	97.91	97.91	95.47	96.12
NetBIOS	97.62	99.97	96.71	98.53	96.21	98.53	96.42	98.53
SYN	98.36	99.88	97.5	99.91	96.37	99.9	96.88	99.9
UDP	97.49	99.4	93.76	99.4	97.39	97.96	95.4	98.68

Further comparison of the model accuracy of various intrusion detection models across multiple studies as shown in Figure 5 was carried out. The results show a steady improvement in performance over time. Earlier models, such as WOA+DNN by Wang et al. (2021) and AE+SVM by Zhou et al. (2021), achieved accuracies of 97.2% and 97.5%, respectively. Later approaches, like GA+DNN (97.8%) and AE+XGBoost (99.1%), demonstrated notable gains. The current study, using the AE+WOA-XGBoost hybrid model, achieved the highest accuracy of 99.89%, indicating a significant enhancement in detection capability and model optimization compared to previous methods

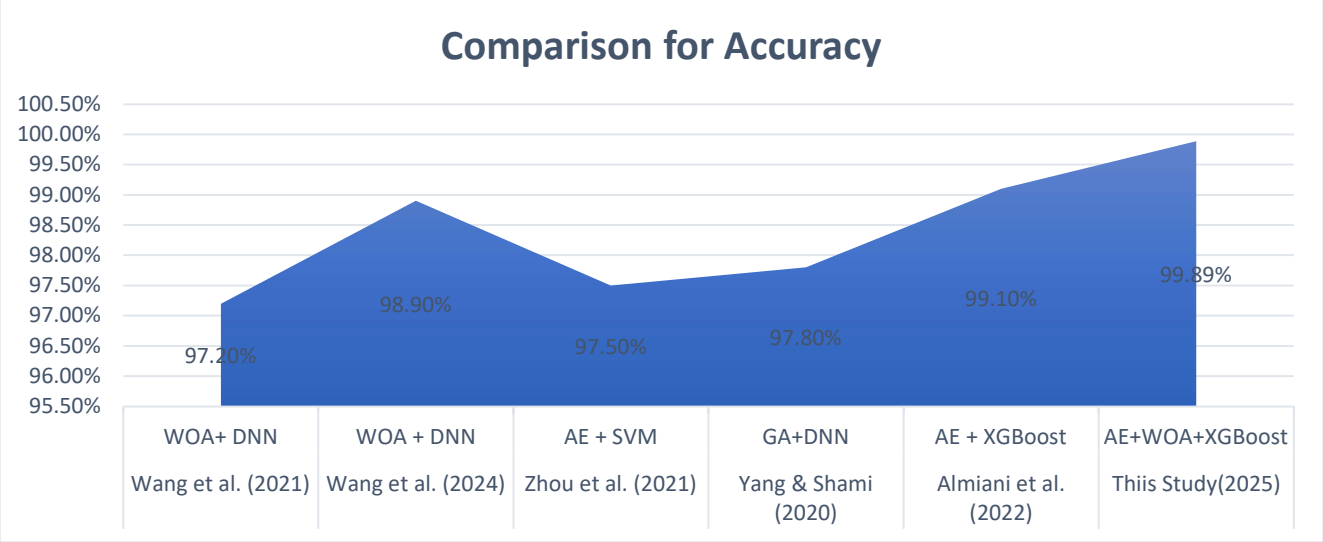

Figure 5: Comparison of the performance measure of the Proposed AE-WOA-XGBoost Model with other existing studies

6. Conclusions and Future Work

This paper tackles the ongoing challenge of classifying Distributed Denial of Service (DDoS) attacks, which traditional methods struggle to manage due to their dynamic and large-scale nature. It introduces the AE+WOA-XGBoost model, which combines an Autoencoder (AE) for extracting meaningful data features, the Whale Optimization Algorithm (WOA) for selecting the most relev features, and XGBoost for accurate and scalable classification. Evaluated on the CIC-DDoS2019 and CICIDS2017 datasets, the model achieved exceptional performance with 99.89% accuracy, 99.96% precision, 99.87% recall, and 99.91% F1-score, significantly outperforming the baseline and several other existing models. The findings confirm the power of metaheuristic optimization in enhancing detection accuracy and reducing false alarms. For future work, we recommend adopting the AE+WOA-XGBoost model in critical infrastructure systems, integrating metaheuristic feature selection in deep learning models, validating models across multiple datasets to ensure robustness, and implementing continuous learning mechanisms for sustained performance in evolving network environments.

References

Agarwal, A., Khari, M., & Singh, R. (2022). Detection of DDOS Attack using Deep Learning Model in Cloud Storage Application. *Wireless Personal Communications*, *127*(1), 419–439. https://doi.org/10.1007/s11277-021-08271-z

Alamri, H. A., & Thayananthan, V. (2020). Bandwidth control mechanism and extreme gradient boosting algorithm for protecting software-defined networks against DDoS attacks. *IEEE Access*, *8*, 194269–194288. https://doi.org/10.1109/ACCESS.2020.3033942

Alfatemi, A., Rahouti, M., Amin, R., ALJamal, S., Xiong, K., & Xin, Y. (2024). *Advancing DDoS Attack Detection: A Synergistic Approach Using Deep Residual Neural Networks and Synthetic Oversampling*. http://arxiv.org/abs/2401.03116

Berríos, S., Garcia, S., Hermosilla, P., & Allende-Cid, H. (2025). A Machine-Learning-Based Approach for the Detection and Mitigation of Distributed Denial-of-Service Attacks in Internet of Things Environments. *Applied Sciences (Switzerland)*, *15*(11). https://doi.org/10.3390/app15116012

Can, H. Q. L., & Ha, Q. T. (2021). Detection of distributed denial of service attacks using automatic feature selection with enhancement for imbalance dataset. *ACIIDS*.

Chanu, U. S., Singh, K. J., & Chanu, Y. J. (2023). A dynamic feature selection technique to detect DDoS attack. *Journal of Information Security and Applications*, *74*. https://doi.org/10.1016/j.jisa.2023.103445

Chen, S., & Guo, W. (2023). Auto-Encoders in Deep Learning—A Review with New Perspectives. In *Mathematics* (Vol. 11, Issue 8). MDPI. https://doi.org/10.3390/math11081777

Efendi, R. (2025). Optimizing Neural Network Architecture for Detecting DDOS Attacks using ANN and XGBoost in Imbalanced Networks. *Engineering, Technology and Applied Science Research*, *15*(3), 22518–22526. https://doi.org/10.48084/etasr.9909

Gebremeskel, T. G., Gemeda, K. A., Krishna, T. G., & Ramulu, P. J. (2023). DDoS Attack Detection and Classification Using Hybrid Model for Multicontroller SDN. *Wireless Communications and Mobile Computing*, *2023*, 1–18. https://doi.org/10.1155/2023/9965945

Hacılar, H., Dedeturk, B. K., Bakir-Gungor, B., & Gungor, V. C. (2024). Network anomaly detection using Deep Autoencoder and parallel Artificial Bee Colony algorithm-trained neural network. *PeerJ Computer Science*, *10*. https://doi.org/10.7717/PEERJ-CS.2333

Hadi, H. J., Cao, Y., Li, S., Xu, L., Hu, Y., & Li, M. (2024). Real-time fusion multi-tier DNN-based collaborative IDPS with complementary features for secure UAV-enabled 6G networks. *Expert Systems with Applications*, *252*, 124215. https://doi.org/https://doi.org/10.1016/j.eswa.2024.124215

Ieracitano, C., Adeel, A., Morabito, F. C., & Hussain, A. (2020). A novel statistical analysis and autoencoder driven intelligent intrusion detection approach. *Neurocomput.*, *387*(C), 51–62. https://doi.org/10.1016/j.neucom.2019.11.016

Kaur, N., Bansal, M., & Sran, S. S. (2024). Analyzing Cyber Attacks and Optimizing Performance Metrics through Feature Selection in Intrusion Detection Systems. *International Journal of Intelligent Systems and Applications in Engineering*, *12*, 2162–2175. https://doi.org/10.17762/ijisae.v12i22s.7648

Laiq, F., Al-Obeidat, F., Amin, A., & Moreira, F. (2023). DDoS Attack Detection in Edge-IIoT using Ensemble Learning. *2023 7th Cyber Security in Networking Conference, CSNet 2023*, 204–207. https://doi.org/10.1109/CSNet59123.2023.10339784

Liu, L., Yu, W., Wu, Z., & Peng, S. (2025). XGBoost-Based Detection of DDoS Attacks in Named Data Networking. *Future Internet*, *17*(5). https://doi.org/10.3390/fi17050206

Liu, Z., Wang, Y., Feng, F., Liu, Y., Li, Z., & Shan, Y. (2023). A DDoS Detection Method Based on Feature Engineering and Machine Learning in Software-Defined Networks. *Sensors*, *23*(13). https://doi.org/10.3390/s23136176

Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., & Foozy, C. F. M. (2021). Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset. *IEEE Access*, *9*, 22351–22370. https://doi.org/10.1109/ACCESS.2021.3056614

Moustafa, N., & Slay, J. (2017). *A hybrid feature selection for network intrusion detection systems: Central points*. https://doi.org/10.4225/75/57a84d4fbefbb

Niyaz, Q., Sun, W., Javaid, A. Y., & Alam, M. (2015). A deep learning approach for network intrusion detection system. *EAI International Conference on Bio-Inspired Information and Communications Technologies (BICT)*. https://doi.org/10.4108/eai.3-12-2015.2262516

Ortet L. I., Zou, D., Ruambo, F. A., Akbar, S., & Yuan, B. (2021). Towards Effective Detection of Recent DDoS Attacks: A Deep Learning Approach. *Security and Communication Networks*, *2021*. https://doi.org/10.1155/2021/5710028

Ouhssini, M., Afdel, K., Agherrabi, E., Akouhar, M., & Abarda, A. (2024). DeepDefend: A comprehensive framework for DDoS attack detection and prevention in cloud computing. *Journal of King Saud University - Computer and Information Sciences*, *36*(2). https://doi.org/10.1016/j.jksuci.2024.101938

OYELAKIN, A. M. (2024). A Learning Approach for The Identification of Network Intrusions Based on Ensemble XGBoost Classifier. *Indonesian Journal of Data and Science*, *4*(3). https://doi.org/10.56705/ijodas.v4i3.88

Panggabean, C., Venkatachalam, C., Shah, P., John, S., Renuka Devi, P., & Venkatachalam, S. (2024). Intelligent DoS and DDoS Detection: A Hybrid GRU-NTM Approach to Network Security. *Proceedings of the 5th International Conference on Smart Electronics and Communication, ICOSEC 2024*, 658–665. https://doi.org/10.1109/ICOSEC61587.2024.10722438

Parfenov, D., Kuznetsova, L., Yanishevskaya, N., Bolodurina, I., Zhigalov, A., & Legashev, L. (2020). Research application of ensemble machine learning methods to the problem of multiclass classification of DDoS attacks identification. *2020 International Conference Engineering and Telecommunication, En and T 2020*. https://doi.org/10.1109/EnT50437.2020.9431255

Pontes, C. F. T., De Souza, M. M. C., Gondim, J. J. C., Bishop, M., & Marotta, M. A. (2021). A New Method for Flow-Based Network Intrusion Detection Using the Inverse Potts Model. *IEEE Transactions on Network and Service Management*, *18*(2), 1125–1136. https://doi.org/10.1109/TNSM.2021.3075503

Prieto, J., & Durán Barroso, R. J. (2024). Emerging Technologies in Edge Computing and Networking. *Sensors*, *24*(4). https://doi.org/10.3390/s24041271

Rehman, S. ur, Khaliq, M., Imtiaz, S. I., Rasool, A., Shafiq, M., Javed, A. R., Jalil, Z., & Bashir, A. K. (2021). DIDDOS: An approach for detection and identification of Distributed Denial of Service (DDoS) cyberattacks using Gated Recurrent Units (GRU). *Future Generation Computer Systems*, *118*, 453–466. https://doi.org/10.1016/j.future.2021.01.022

Samom, P. S., Taggu, A., & Taggu E-Mail:, A. (2021). Distributed denial of service (Ddos) attacks detection: A machine learning approach. *Lecture Notes in Networks and Systems*, *187*, 75–87. https://doi.org/10.1007/978-981-33-6173-7_6

Shaikh, J., Awais Butt, Y., Fatima Naqvi, H., & Author, C. (2024). *Effective Intrusion Detection System Using Deep Learning for DDoS Attacks Chronicle Abstract*. *4*. https://doi.org/10.62019/abbdm

Shieh, C. S., Lin, W. W., Nguyen, T. T., Chen, C. H., Horng, M. F., & Miu, D. (2021). Detection of unknown ddos attacks with deep learning and gaussian mixture model. *Applied Sciences (Switzerland)*, *11*(11). https://doi.org/10.3390/app11115213

Shohan, N. J., Tanbhir, G., Elahi, F., Ullah, A., & Sakib, M. N. (2024). Enhancing Network Security: A Hybrid Approach for Detection and Mitigation of Distributed Denial-of-Service Attacks Using Machine Learning. *Communications in Computer and Information Science*, *2091 CCIS*, 81–95. https://doi.org/10.1007/978-3-031-64064-3_7

Sihwail, R., Ghamri, M. Al, & Ibrahim, D. (2024). An Enhanced Model of Whale Optimization Algorithm and K-nearest Neighbors for Malware Detection. *International Journal of Intelligent Engineering and Systems*, *17*(3), 606–621. https://doi.org/10.22266/ijies2024.0630.47

Singh, K., & De, T. (2017). Efficient Classification of DDoS Attacks Using an Ensemble Feature Selection Algorithm. *Journal of Intelligent Systems*, *29*, 71–83.

Ullah, I., & Mahmoud, Q. H. (2020). A Technique for Generating a Botnet Dataset for Anomalous Activity Detection in IoT Networks. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, *2020-Octob*, 134–140. https://doi.org/10.1109/SMC42975.2020.9283220

Varghese, J. E., & Muniyal, B. (2021). An Efficient IDS Framework for DDoS Attacks in SDN Environment. *IEEE Access*, *9*, 69680–69699. https://doi.org/10.1109/ACCESS.2021.3078065

Wang, Z., Li, Y., Wu, L., & Guo, Q. (2024). A Nonlinear Adaptive Weight-Based Mutated Whale Optimization Algorithm and Its Application for Solving Engineering Problems. *IEEE Access*, *12*, 40225–40254. https://doi.org/10.1109/ACCESS.2024.3350336

Wei, Y., Jang-Jaccard, J., Sabrina, F., Singh, A., Xu, W., & Camtepe, S. (2021). AE-MLP: A

Hybrid Deep Learning Approach for DDoS Detection and Classification. *IEEE Access*, *9*, 146810–146821. https://doi.org/10.1109/ACCESS.2021.3123791

Xu, W., Jang-Jaccard, J., Singh, A., Wei, Y., & Sabrina, F. (2021). Improving Performance of Autoencoder-Based Network Anomaly Detection on NSL-KDD Dataset. *IEEE Access*, *9*, 140136–140146. https://doi.org/10.1109/ACCESS.2021.3116612

Zhou, J., Qiu, Y., Zhu, S., Armaghani, D. J., Li, C., Nguyen, H., & Yagiz, S. (2021). Optimization of support vector machine through the use of metaheuristic algorithms in forecasting TBM advance rate. *Engineering Applications of Artificial Intelligence*, *97*. https://doi.org/10.1016/j.engappai.2020.104015

Zhou, L., Zhu, Y., Zong, T., & Xiang, Y. (2022). A feature selection-based method for DDoS attack flow classification. *Future Generation Computer Systems*, *132*, 67–79. https://doi.org/10.1016/j.future.2022.02.006