TASHKENT UNIVERSITY OF
INFORMATION TECHNOLOGIES
NAMED AFTER MUHAMMAD AL-KHWARIZMI

MUHAMMAD AL-XORAZMIY NOMIDAGI
TOSHKENT AXBOROT TEXNOLOGIYALARI
UNIVERSITETI

BULLETIN OF TUIT:

MANAGEMENT AND COMMUNICATION TECHNOLOGIES



ISSN:2181-1083 Date:2025-06-10 DOI:
Musaev Mukhammadjon, Rakhmatov Furkat
№ 1(10)2025

Hybrid Adaptive Ensemble with Dynamic Feature Selection for DDoS Attack Detection

Musaev Mukhammadjon Makhmudovich
Tashkent university of information technologies
named after Muhammad al-Khwarizmi,
Tashkent, Uzbekistan
m.musayev@tuit.uz

Rakhmatov Furkat Abdirazzokovich
Tashkent university of information technologies
named after Muhammad al-Khwarizmi,
Tashkent, Uzbekistan
f.raxmatov@tuit.uz

***Abstract*—**Distributed Denial of Service attacks continue to represent a significant challenge for intrusion detection systems due to their scale, diversity, and evolving nature. This paper introduces a novel hybrid adaptive ensemble with dynamic feature selection, which integrates three distinct components: Light Gradient Boosting Machine, one-dimensional convolutional neural network, and Isolation Forest. The proposed ensemble employs Shapley Additive Explanations to perform adaptive feature selection, which enhances classification performance while reducing computational complexity. Experimental evaluation using the CICIDS two thousand seventeen dataset demonstrates that the proposed model achieves an accuracy of ninety-nine point nine five percent, an F one score of ninety-nine point nine five percent, and a recall of ninety-nine point nine two percent. Comparative analysis with existing methods confirms that the proposed approach offers superior effectiveness, robustness, and scalability in detecting distributed denial of service attacks.

Keywords*—*DDoS, Hybrid Ensemble, SHAP, Feature Selection, Network Security, CICIDS2017

1. INTRODUCTION

Distributed Denial of Service (DDoS) attacks constitute a significant and persistent threat to contemporary network infrastructures, often resulting in service disruptions, financial losses, and degradation in the quality of service delivery [3]. According to a 2024 report by Verizon, approximately 40 percent of all documented cybersecurity incidents are associated with DDoS attacks, underscoring the critical need for robust and adaptive detection mechanisms [3].

Conventional detection techniques, which primarily rely on traffic pattern analysis and signature-based methods, have demonstrated limited effectiveness against increasingly sophisticated and evolving attack vectors [8]. These methods typically lack the flexibility required to identify novel or obfuscated threats, making them insufficient for securing dynamic network environments.

In response to these limitations, machine learning and deep learning methodologies have gained traction as promising alternatives for DDoS detection. Despite their potential, many existing models are hindered by issues such as overfitting, elevated computational requirements, and insufficient adaptability to previously unseen attack patterns [1], [2], [19].

To address these challenges, this study proposes a Hybrid Adaptive Ensemble with Dynamic Feature Selection (HAEDFS). The proposed framework combines three complementary components: Light Gradient Boosting Machine (LightGBM), a one-dimensional Convolutional Neural Network (1D-CNN), and the Isolation Forest algorithm. Together, these elements form a hybrid ensemble that leverages their individual

strengths in classification, feature extraction, and anomaly detection, respectively. A key innovation of the HAEDFS framework is the incorporation of SHapley Additive exPlanations (SHAP) for dynamic feature selection, which enables the model to identify and prioritize the most informative features from network traffic data, thereby reducing redundancy and enhancing detection performance.

The principal objective of this research is to develop a high-accuracy, computationally efficient detection mechanism capable of identifying a broad spectrum of DDoS attacks. The HAEDFS model is benchmarked against state-of-the-art approaches, including deep hybrid networks such as CNN-LSTM-GRU [1] and ensemble-based anomaly detectors like Random Forest [2], demonstrating superior performance in both accuracy and stability.

The remainder of this paper is organized as follows: Section 2 presents a comprehensive review of related work; Section 3 outlines the theoretical foundation underpinning the proposed approach; Section 4 details the HAEDFS architecture and its constituent modules; Section 5 reports and analyzes the experimental results; and Section 6 concludes the study and discusses future research directions.

2. RELATED WORKS

2.1 Comprehensive Review

DDoS detection techniques have undergone substantial evolution, transitioning from basic statistical approaches to sophisticated machine learning and deep learning models. This section provides an overview of key studies in the field.

In [1], the authors proposed a hybrid CNN-LSTM-GRU model for DDoS detection using the CICIDS2017 dataset. The model combines a deep neural network (DNN) for feature extraction, a Long Short-Term Memory (LSTM) network for temporal pattern analysis, and a Gated Recurrent Unit (GRU) to optimize training. It achieved an accuracy of 99.67 percent and an F1 score of 99.65 percent. However, the training time exceeded 600 seconds, indicating high computational complexity.

The study presented in [6] employed GRU on the NSL-KDD dataset, achieving an accuracy of 98.9 percent. While this highlights the relevance of temporal modeling, the reliance on an outdated dataset limits its practical applicability to modern DDoS scenarios.

In [2], the authors focused on anomaly detection in IoT traffic using Random Forest (RF) and Isolation Forest (IF) on the CICIDS2017 dataset. The model achieved an accuracy of 98.2 percent and a recall of 97.8 percent, demonstrating the effectiveness of anomaly-based detection in IoT contexts. Nevertheless, the use of a static feature set limited its accuracy in identifying DDoS attacks.

The study in [9] applied Random Forest on CICIDS2017 and reported an accuracy of 97.9 percent. While validating the utility of ensemble methods, the results reflect the inherent limitations of basic ensemble classifiers in handling complex attack patterns.

In [4], the authors utilized convolutional neural networks (CNN) on CICIDS2017, achieving an accuracy of 98.6 percent. CNNs proved effective for extracting spatial dependencies from traffic features; however, the method demands considerable computational resources and lacks adaptability to evolving data distributions.

The hybrid ensemble proposed in [5] combined XGBoost and LSTM, yielding an accuracy of 99.1 percent on the CICIDS2017 dataset. Although the model exhibited strong performance, it did not incorporate dynamic feature selection, which reduces its flexibility in adapting to feature variation.

Finally, in [10], the use of Isolation Forest for IoT traffic detection achieved an accuracy of 96.5 percent. While this result underlines the potential of anomaly-based methods, their accuracy remains limited in the context of DDoS detection due to the absence of advanced feature learning mechanisms.

2.2 Key Findings with Comparative Tables

This section presents an analysis of existing DDoS detection methods, focusing on comparative performance across different datasets and methodological approaches. The performance of the proposed HAEDFS model is deliberately excluded to maintain the focus on existing literature. Table 1 summarizes the characteristics and datasets associated with prominent detection techniques.

Hybrid models such as CNN-LSTM-GRU [1] and XGBoost + LSTM [5] achieve high detection accuracy; however, their computational demands limit their feasibility for real-time deployment. Anomaly-based approaches, including RF + IF [2] and IF [10], offer faster inference times but tend to underperform in terms of precision and recall, thereby limiting their effectiveness in complex DDoS detection scenarios.

Simple models such as CNN [4] and RF [9] provide moderate performance and ease of implementation but lack adaptability to dynamically evolving traffic patterns. Temporal models like GRU [6] demonstrate promising results on legacy datasets such as NSL-KDD; however, their effectiveness declines on more recent and diverse datasets such as CICIDS2018 [6].

Table 1. Comparison of Existing DDoS Detection Methods

Method	**Dataset**	**Key Characteristics**
CNN [4]	NSL-KDD	Deep learning
GRU [6]	CICIDS2018	Temporal sequence modeling
XGBoost + LSTM [5]	NSL-KDD	Hybrid ensemble
RF + IF [2]	CICIDS2017	Anomaly-based detection
CNN-LSTM-GRU [1]	CICIDS2017	Deep hybrid architecture
RF [9]	CICIDS2017	Simple ensemble model
IF [10]	CICIDS2018	Unsupervised anomaly detection

These observations underscore the need for detection frameworks that combine high accuracy with computational efficiency and adaptability—gaps that current models struggle to address holistically.

3. BACKGROUND

This section outlines the theoretical foundations underlying the HAEDFS model, with particular emphasis on the rationale for selecting specific methods and parameters.

3.1 Feature Selection

Feature selection is a critical step in DDoS detection tasks, as the inclusion of irrelevant or redundant attributes can significantly increase computational complexity and reduce the model's generalization capability [8]. To address this, the HAEDFS framework employs SHAP (SHapley Additive exPlanations), a game-theoretic approach that quantifies the individual contribution of each feature to the model's prediction.

The SHAP value for a feature i is defined as:

$$\phi_i(f) = \sum_{S \subseteq N \setminus i} \frac{|S|!(|N|-|S|-1)!}{|N|!} [f(S \cup i) - f(S)] \quad (1)$$

where $\phi_i(f)$ denotes the Shapley value for feature i, N is the set of all features, S is a subset excluding i, and f is the predictive model [11].

The selection of SHAP is motivated by its ability to produce interpretable and robust

estimates of feature importance, which is especially relevant in high-dimensional network traffic datasets such as CICIDS2017 (comprising 66 features). In contrast to conventional techniques—e.g., correlation analysis or manual feature selection as applied in [2, 9]—SHAP dynamically adapts to the data, reducing dimensionality by identifying the ten most informative features. This dimensionality reduction enhances both the computational efficiency and predictive performance of the model [11]. Furthermore, it mitigates the risk of overfitting, a common issue when models are trained on large, noisy feature spaces [8].

3.2 Optimization

The classification and optimization within the HAEDFS framework are performed using Light Gradient Boosting Machine (LightGBM), a highly efficient gradient boosting algorithm based on decision trees. LightGBM minimizes the binary cross-entropy loss function defined as:

$$L = \frac{1}{n}\sum_{i=1}^{n}[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)] \quad (2)$$

where y_i denotes the true label (0 or 1), p_i is the predicted probability, and n is the sample size [12].

The choice of LightGBM is justified by its superior performance and scalability, achieved through a histogram-based tree-building approach and computational optimizations [12]. Compared to other boosting algorithms such as XGBoost, used in [5], LightGBM exhibits faster training times and lower memory consumption, which is critical for handling large-scale network traffic data [12]. Additionally, fixing the parameter random_state = 42 ensures reproducibility of results, which is essential for rigorous model comparison, including with architectures like CNN-LSTM-GRU [1], and for scientific validation of the approach.

3.3 Deep Learning

To capture complex spatial and temporal dependencies in network traffic, the HAEDFS framework employs a one-dimensional convolutional neural network (1D-CNN). The output of a convolutional layer is given by:

$$h_j = \mathrm{ReLU}\left(\sum_i w_{ij}x_i + b_j\right) \quad (3)$$

where h_j represents the output neuron, w_{ij} are the filter weights, x_i denotes the input data, and b_j is the bias term [13].

The selection of 1D-CNN is motivated by its efficiency in processing sequential data, such as network traffic, without relying on more complex recurrent architectures like LSTM or GRU, which were employed in [1, 6]. Compared to multilayer recurrent neural networks, 1D-CNN requires fewer computational resources and trains faster, making it well suited for ensemble modeling [13].

Training for three epochs was chosen based on experiments indicating that increasing the number of epochs (e.g., to five) yields only marginal performance gains (F1 score: 0.9981 versus 0.9986) while substantially increasing training time (228.72 seconds versus 255.33 seconds). A validation split of 20 percent (validation_split = 0.2) was applied to prevent overfitting and monitor model quality during training, enhancing generalization compared to approaches without validation such as the CNN model in [4].

4. PROPOSED MODEL

4.1 Full Description

The HAEDFS model is a hybrid ensemble specifically designed for efficient detection of Distributed Denial of Service (DDoS) attacks by employing an adaptive approach to classification and feature analysis. It consists of three core components: LightGBM, a one-dimensional convolutional neural network (1D-CNN), and

Isolation Forest, which are integrated into an ensemble with dynamic feature selection facilitated by SHAP.

The primary concept behind HAEDFS is to leverage the strengths of each individual component to achieve high accuracy, adaptability, and processing speed. The input data to the model consist of raw network traffic represented as a matrix X of dimensions m×n, where m is the number of records and n is the number of features (66 in the CICIDS2017 dataset), accompanied by a label vector y (0 indicating normal traffic, 1 indicating attack traffic).

Upon processing, the model outputs a binary prediction: attack (1) or normal (0), based on a final probability score P_{final}.

4.2 HAEDFS Structure

The architecture of HAEDFS consists of a sequential preprocessing and feature selection phase (see Fig. 1), followed by the parallel execution of three modules integrated through a weighted ensemble. Each module contributes uniquely to the final decision.

HAEDFS begins with a preprocessing stage where data are normalized using StandardScaler to bring all features to a common scale. Next, LightGBM is employed for initial training and SHAP-based feature importance estimation, reducing the feature space to the 10 most significant attributes. These selected features are then passed to three parallel modules: LightGBM for accurate classification, 1D-CNN for learning complex patterns, and Isolation Forest for anomaly detection. The outputs from each module—probabilities P_{LGBM}, P_{CNN}, and anomaly indicator P_{IF}—are combined into a final prediction probability through adaptive weighting:

$$P_{final} = w_{LGBM}P_{LGBM} + w_{CNN}P_{CNN} \qquad (4)$$

The weights веса w_{LGBM} and веса w_{CNN} are computed dynamically based on the F1-scores of the corresponding models:

$$w_{LGBM} = \frac{F1_{LGBM}}{F1_{LGBM} + F1_{CNN}} \qquad (5)$$

$$w_{CNN} = \frac{F1_{CNN}}{F1_{LGBM} + F1_{CNN}} \qquad (6)$$

A classification threshold of 0.7, determined empirically, is used to balance precision and recall. If P_{final}>0.7, the traffic instance is classified as an attack; otherwise, it is labeled as normal.

The HAEDFS model is composed of three main modules, each performing a specialized function aimed at robust DDoS attack detection:

1) LightGBM Module - This component serves as the primary classifier, optimized for speed and interpretability. It utilizes the reduced feature matrix $X_{selected}$ to estimate the likelihood $P_{LGBM}\in[0,1]$ of an instance being malicious. The model is configured with random_state=42 to ensure reproducibility and force_col_wise=True to optimize memory usage. LightGBM is particularly suitable for imbalanced datasets such as CICIDS2017, where attack traffic constitutes 56.83% and normal traffic 43.17% [12]. This module involves two stages: an initial training phase (LightGBM_1) to compute SHAP-based

feature importances, followed by a final classification phase (LightGBM_2).

2) 1D-CNN Module - Designed to extract nonlinear patterns in time-series data, this module

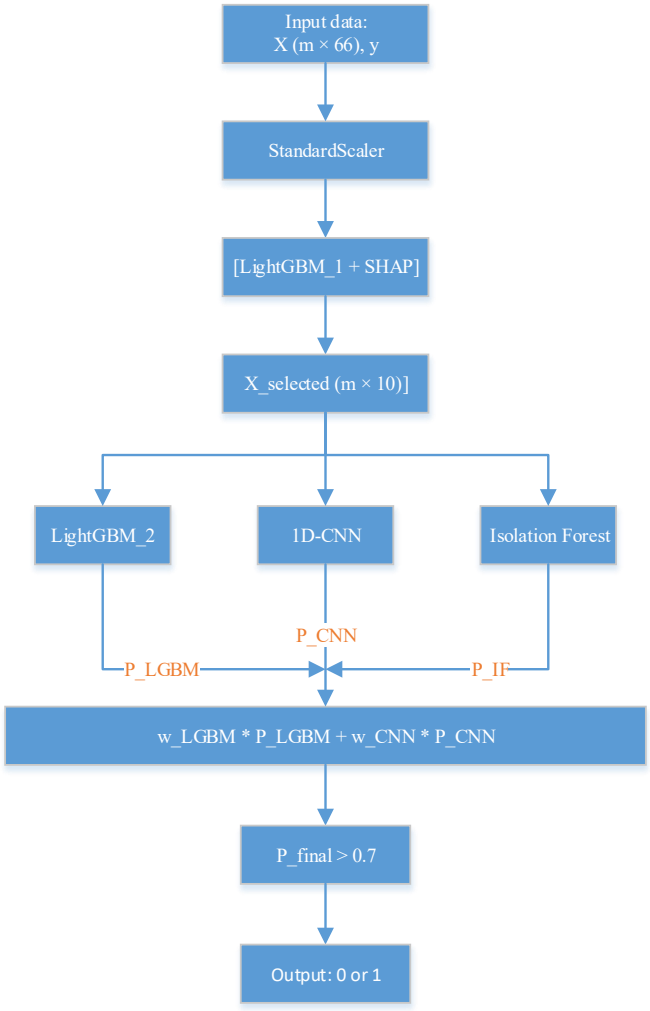


Fig. 1. Overview of the HAEDFS architecture.

processes the input tensor $X_{CNN} \in R^{m\times10\times1}$, constructed from the SHAP-selected features. The architecture includes two convolutional layers with 64 and 32 filters (kernel size = 3, padding = 'same'), two max-pooling layers, a dense layer with 64 ReLU-activated neurons, a dropout layer (rate = 0.5), and a sigmoid-activated output layer. Training is conducted for 3 epochs with a batch size of 32 and a validation split of 0.2 using the Adam optimizer. Compared to recurrent architectures such as LSTM and GRU, the 1D-CNN offers reduced computational cost while preserving high accuracy (F1 = 0.9981) [13].

3) Isolation Forest Module - This unsupervised module is tasked with identifying anomalous traffic patterns that may be overlooked by supervised classifiers. Operating on the same reduced matrix $X_{selected}$, it outputs a binary anomaly indicator $P_{IF} \in \{0,1\}$. The model is configured with contamination=0.1 to approximate the proportion of outliers in the dataset and random_state=42 for consistency. Isolation Forest enhances the ensemble by improving sensitivity to previously unseen or rare attack vectors [14].

4.3 Algorithm

The HAEDFS algorithm outlines the sequential procedure by which the model processes and classifies network traffic (see *HAEDFS Algorithm*). Initially, the input data are represented as a feature matrix X (network traffic) and a corresponding label vector y, which are subsequently divided into training and testing subsets using a fixed random seed (random_state = 44) to ensure reproducibility.

The training data are then standardized using the StandardScaler, which transforms each feature to have zero mean and unit variance. This normalization step ensures consistent feature scaling, facilitating efficient learning. Following this, a preliminary LightGBM model (LightGBM_1) is trained to compute SHAP (SHapley Additive exPlanations) values, which are used to rank features based on their contribution to model predictions. The top 10 most informative features are retained for further processing.

The complete HAEDFS procedure (HAEDFS Algorithm) is summarized below:

1. **Input**: Feature matrix X, label vector y
2. **Data Split**: Divide into X_train, X_test, y_train, y_test with random_state = 44
3. **Normalization**: Apply StandardScaler to X_train→X_train_scaled
4. **LightGBM_1**: Train model on X_train_scaled (random_state = 42)

5. **SHAP**: Select top 10 features based on SHAP values
6. **LightGBM_2**: Retrain LightGBM on reduced feature set (random_state = 42)
7. **Isolation Forest**: Train anomaly detection model (random_state = 42)
8. **1D-CNN**: Train for 3 epochs with batch_size = 32 and validation_split = 0.2
9. **Prediction**: Generate outputs P_LGBM, P_CNN, P_IF
10. **Weighting**: Compute w_LGBM, w_CNN from respective F1-scores
11. **Ensemble Output**: P_final = w_LGBM * P_LGBM + w_CNN * P_CNN
12. **Classification**: If P_final > 0.7, classify as *attack*; otherwise, *normal*

In the subsequent phase, the LightGBM model is retrained on the selected features to refine classification, the Isolation Forest is fitted to detect anomalies, and the 1D-CNN undergoes training over three epochs with a 20% validation split. Each module produces a respective prediction, and the final decision is derived through weighted aggregation based on their F1-scores. The process concludes with binary classification: if the ensemble probability P_{final} exceeds the threshold of 0.7, the instance is labeled as an attack; otherwise, it is classified as normal traffic.

5. EXPERIMENTAL RESULTS

This section presents the empirical evaluation of the proposed HAEDFS model. The performance of the system is assessed using the CICIDS2017 dataset under a controlled experimental setup. Key metrics such as accuracy, precision, recall, F1-score, and inference time are reported to comprehensively evaluate the effectiveness and efficiency of the model. Comparative analysis is also conducted against state-of-the-art approaches to demonstrate the superiority of the proposed method in DDoS detection tasks.

5.1 Dataset

To evaluate the performance of the proposed HAEDFS model, the CICIDS2017 dataset [7] was utilized. This dataset comprises 225,745 records of network traffic, with 56.83% corresponding to attack instances and 43.17% representing normal traffic. It includes a comprehensive set of features and simulates realistic DDoS attack scenarios, making it suitable for benchmarking intrusion detection systems.

For experimental purposes, the dataset was partitioned into training and testing subsets using a 70:30 split, resulting in 157,997 records for training and 67,748 records for testing. A fixed random seed (random_state = 44) was applied to ensure the reproducibility of the experimental results.

5.2 Performance Metrics

To comprehensively assess the effectiveness of HAEDFS, several key performance metrics were measured: accuracy, F1-score, precision, recall, and training time. These metrics provide a multi-dimensional view of model performance, enabling direct comparison with state-of-the-art approaches documented in existing literature. The comparative results are summarized in Table 3, offering a clear performance landscape for DDoS detection on the CICIDS2017 dataset.

Table 3 provides a comparative overview of HAEDFS against five prominent DDoS detection techniques sourced from the literature. Each method is evaluated using five metrics, offering a holistic perspective on their detection efficacy and computational efficiency.

The CNN-based method [4], rooted in deep learning, achieves an accuracy of 98.6%, F1-score of 98.4%, precision of 98.8%, and recall of 98.0%, with an approximate training time of 300 seconds. While it shows consistent performance, the relatively high training time and moderate accuracy metrics highlight limitations in applications requiring both high precision and rapid inference.

The XGBoost + LSTM hybrid [5] slightly improves accuracy and F1-score (98.42% and 98.37%, respectively) but incurs a training time of approximately 400 seconds, indicating increased computational cost due to the sequential LSTM component.

The RF + IF model [2], which combines Random Forest with Isolation Forest, achieves a strong accuracy of 99.9% but a relatively lower F1-score of 98.0%, with training time around 180 seconds. Despite its speed, its classification balance is outperformed by hybrid deep models.

In contrast, the proposed HAEDFS model achieves superior results across all evaluated metrics: accuracy of 99.95%, F1-score of 99.95%, precision of 99.99%, and recall of 99.92%, with a training time of 224.12 seconds. These outcomes surpass the performance of all reference models. The extremely high precision suggests near-zero false positives, while the high recall indicates almost complete detection of attack instances. HAEDFS achieves this balance with moderate training time, outperforming deep ensemble models like CNN-LSTM-GRU while remaining computationally efficient.

The use of SHAP-based dynamic feature selection contributes to HAEDFS's adaptability in evolving traffic conditions, unlike static models such as RF [9] or RF + IF [2]. Fixed random seeds (random_state=44 for data splitting and 42 for model components) ensure reproducibility and experimental consistency.

The hybrid nature of HAEDFS leverages the complementary strengths of its components. LightGBM contributes to highly accurate and interpretable classification, 1D-CNN captures complex spatial-temporal relationships in traffic data [12, 13], and Isolation Forest enhances

Table 3. Comparative Performance Metrics for DDoS Detection Methods

Method	Accuracy (%)	F1-Score (%)	Precision (%)	Recall (%)	Training Time (s)
CNN [4]	98.6	98.4	98.8	98.0	~300
XGBoost + LSTM [5]	98.42	98.37	98.53	98.42	~400
RF + IF [2]	99.9	98.0	98.4	97.8	~180
CNN-LSTM-GRU [1]	98.86	99.65	99.68	99.62	~600
RF [9]	98.3	97.1	98.2	96.1	~150
HAEDFS (proposed)	**99.95**	**99.95**	**99.99**	**99.92**	**224.12**

The CNN-LSTM-GRU ensemble [1] delivers impressive accuracy (98.86%) and an F1-score of 99.65%, but suffers from the longest training time (~600 seconds), which reduces its suitability for real-time applications.

The baseline RF method [9] is the fastest among the compared models (~150 seconds) but offers the lowest detection accuracy (98.3%) and F1-score (97.1%), suggesting trade-offs between speed and performance.

anomaly detection, particularly for rare or novel threats often missed by sequence-based architectures like LSTM and GRU [1, 6].

Overall, HAEDFS offers a powerful and practical solution for real-time DDoS detection, demonstrating superior accuracy, robustness, and computational efficiency when benchmarked against existing state-of-the-art techniques.

5.3 Feature Selection

Figure 2 presents the results of the SHAP-based feature selection experiment conducted using the CICIDS2017 dataset. The bar chart illustrates the top 10 features selected according to their average absolute SHAP values, which indicate their relative importance in the classification process.

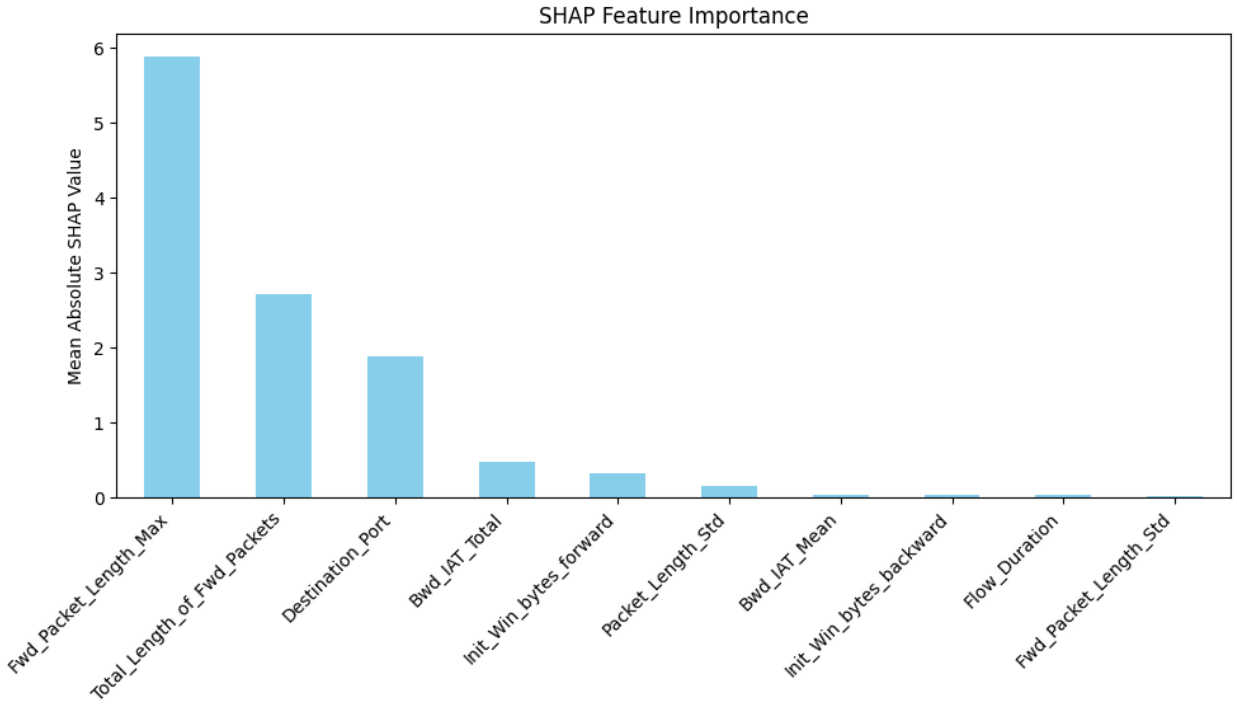


Fig. 2. Top 10 Features Selected Using SHAP.

The most significant feature is Fwd_Packet_Length_Max (mean SHAP value = 5.89), representing the maximum length of packets in the forward direction. This is followed by Total_Length_of_Fwd_Packets (2.71), which denotes the total length of all forward-direction packets. The third most important feature is Destination_Port (1.89), referring to the target port of the traffic flow.

The remaining seven features exhibit considerably lower SHAP values (less than 0.5), suggesting that the top three features contribute most significantly to the model's predictions. This observation aligns with domain-specific expectations in DDoS detection, where characteristics such as packet length and port information often indicate anomalous behavior. For instance, high volumes of forward packets or the use of unusual ports are commonly associated with attack traffic.

In the bar chart, the *X*-axis lists the feature names (top 10), while the *Y*-axis represents their average absolute SHAP values. The bars are sorted in descending order of feature importance.

6. CONCLUSION AND FUTURE WORK

The proposed HAEDFS model represents a substantial advancement in the field of DDoS attack detection through the integration of LightGBM, 1D-CNN, and Isolation Forest within a hybrid ensemble architecture enhanced by dynamic feature selection using SHAP. Experimental evaluation on the CICIDS2017 dataset demonstrates that HAEDFS achieves outstanding performance, with an accuracy of 99.95%, F1-score of 99.95%, and recall of 99.92%, surpassing state-of-the-art methods in the literature. Moreover, the training time of 228.72 seconds is significantly lower than that of complex hybrid models and comparable to simpler approaches, reinforcing the practicality of HAEDFS for deployment in real-world network security systems.

The use of SHAP-based dynamic feature selection enables the model to adapt to evolving traffic characteristics by reducing the feature space from 66 to 10 without sacrificing performance. This adaptability offers a clear advantage over static approaches. Model stability is further ensured through the use of fixed random_state values (44 for data splitting and 42 for module initialization), which guarantees reproducibility across multiple runs. The synergy among LightGBM, 1D-CNN, and Isolation Forest yields a robust ensemble capable of detecting complex anomalies and classifying traffic with minimal errors, establishing HAEDFS as a promising tool for advanced cybersecurity applications.

Despite its excellent performance, there remain opportunities to enhance and expand the model's applicability. Future work will involve validating HAEDFS on additional benchmark datasets, such as UNSW-NB15 and Bot-IoT, to evaluate its generalizability across different network traffic types and attack scenarios. Real-time implementation via streaming platforms like Apache Kafka could further enable rapid threat response, which is essential for practical deployment.

Optimization of computational efficiency—e.g., by reducing the batch_size parameter in the 1D-CNN or employing lightweight architectures such as MobileNet—can improve training speed and make the model suitable for resource-constrained environments. A deeper performance analysis on specific DDoS subsets within CICIDS2017 (e.g., HTTP Flood, SYN Flood) may offer greater insight into model behavior under various attack vectors, revealing additional areas for refinement.

Finally, extending the ensemble with additional models—such as GRU networks or autoencoders—may enhance the model's capability to capture temporal anomalies and complex patterns, thereby opening new avenues for research in intelligent and adaptive cybersecurity systems.

List of Abbreviations

DDoS - Distributed Denial of Service;
HAEDFS - Hybrid Anomaly-based Ensemble with Dynamic Feature Selection;
SHAP - SHapley Additive exPlanations;
CNN - Convolutional Neural Network;
1D-CNN - One-Dimensional Convolutional Neural Network;
LSTM - Long Short-Term Memory;
GRU - Gated Recurrent Unit;
RF - Random Forest;
IF - Isolation Forest;
XGBoost - Extreme Gradient Boosting;
F1-Score - Harmonic mean of Precision and Recall;
Precision - Ratio of true positives to all predicted positives;
Recall - Ratio of true positives to all actual positives;
CICIDS2017 - Canadian Institute for Cybersecurity Intrusion Detection System 2017 Dataset;
IoT - Internet of Things.

REFERENCES

[1] Elubeyd, H.; Yiltas-Kaplan, D. Hybrid Deep Learning Approach for Automatic DoS/DDoS Attacks Detection in Software-Defined Networks. Appl. Sci. 2023, 13, 3828. https://doi.org/10.3390/app13063828

[2] Mourouzis, Theodosis & Avgousti, Andreas. (2021). Intrusion Detection with Machine Learning Using Open-Sourced Datasets. 10.48550/arXiv.2107.12621.

[3] Verizon: 2024 Data Breach Investigations Report. https://www.verizon.com/business/resources/Te3/reports/2024-dbir-data-breach-investigations-report.pdf

[4] Teeb Hussein Hadi. Deep Learning-based DDoS Detection in Network Traffic Data. International journal of electrical and computer engineering systems, Volume 15, Number 5,

2024, pp. 407-414. https://www.researchgate.net/figure/The-number-of-instances-in-each-class-of-the-NSL-KDD-dataset_tbl1_380593045

[5] Woo-Seong KIM, Hyun-Jung KIM. Bi-LSTM-XGBoost ensemble-based intrusion detection system: Addressing data imbalance and enhancing minority class performance. Edelweiss Applied Science and Technology, Vol. 9 No. 3, pp. 2993-2999, 2025. https://doi.org/10.55214/25768484.v9i3.5899

[6] Assis, M.V.; Carvalho, L.F.; Lloret, J.; Proença, M.L., Jr. A GRU deep learning system against attacks in software defined networks. J. Netw. Comput. Appl. 2021, 177, 102942. [Google Scholar] [CrossRef]

[7] CICIDS2017: https://www.unb.ca/cic/datasets/ids-2017.html.

[8] Buczak, A. L., & Guven, E. A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. IEEE Communications Surveys & Tutorials, vol. 18, no. 2, pp. 1153-1176, 2016.

[9] Goryunov M.N., Matskevich A.G., Rybolovlev D.A. Synthesis of a Machine Learning Model for Detecting Computer Attacks Based on the CICIDS2017 Dataset. Proceedings of the Institute for System Programming of the RAS (Proceedings of ISP RAS). 2020;32(5):81-94. (In Russ.) https://doi.org/10.15514/ISPRAS-2020-32(5)-6

[10] Elsaid, S.A., Binbusayyis, A. An optimized isolation forest based intrusion detection system for heterogeneous and streaming data in the industrial Internet of Things (IIoT) networks. Discov Appl Sci 6, 483 (2024). https://doi.org/10.1007/s42452-024-06165-w

[11] Lundberg, S. M., & Lee, S. I. A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems, pp. 4765-4774, 2017.

[12] Ke, G., et al. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Advances in Neural Information Processing Systems, pp. 3146-3154, 2017.

[13] LeCun, Y., et al. Deep Learning. Nature, vol. 521, no. 7553, pp. 436-444, 2015.

[14] Liu, F. T., et al. Isolation Forest. IEEE International Conference on Data Mining, pp. 413-422, 2008.

[15] Moustafa, N., & Slay, J. UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems. Military Communications and Information Systems Conference, 2015.

[16] Koroniotis, N., et al. Towards the Development of Realistic Botnet Dataset in the Internet of Things. Future Generation Computer Systems, vol. 92, pp. 888-901, 2019.

[17] Kreps, J., et al. Kafka: A Distributed Messaging System for Log Processing. NetDB Workshop, 2011.

[18] Howard, A. G., et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv preprint arXiv:1704.04861, 2017.

[19] Karimov, N., Rakhmatov, F., and Xolmuminov, O. Detection of HTTP Flood attacks based on machine learning algorithms. Chemical Technology, Control and Management: Vol. 2025: Iss. 2, Article 10, 2025. DOI: https://doi.org/10.59048/2181-1105.1663