JCBI
Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 10 Issue 02
2026

Research Article
https://doi.org/10.56979/1002/2026/1342

A Machine Learning-Based Firewall Model for Effective Attack Detection Using Dragonfly and Bat Algorithms

Hani Al-Mimi[1*], Ali Al Dahoud[1], and Ahmad Al Dahoud[2]

[1]Faculty of Science and Information Technology, Al Zaytoonah University of Jordan, Amman, Jordan.
[2]Al-Zaytoonah University of Jordan, Faculty of Architecture and Design, Jordan.
[*]Corresponding Author: Hani Al-Mimi. Email: hani.mimi@zuj.edu.jo

Received: December 23, 2025 **Accepted:** February 28, 2026

Abstract: This work proposes a new machine learning (ML)-based firewall model for attack detection in modern networks. In this regard, the proposed ML-based firewall model is integrated with an advanced feature selection method to optimize the significant features that will improve the accuracy of detection by using the Dragonfly Algorithm (DA) and Bat Algorithm (BA). Logistic Regression (LR) and Gradient Boosting Trees (GBT) were used for the classification in this model. The model was tested and validated on the UNSW-NB15 dataset for the experimentation process because this dataset represents a comprehensive modern network activity. The GBT classifier achieved an accuracy of 100%, demonstrating its great capability in handling selected features and finding the attacks. The LR also attained a very high of accuracy of 99.84%. These outcomes highlight the efficiency of the proposed model in detecting attacks with minimal false positives. The integration of DA and BA for feature selection and the use of robust classifiers make the proposed ML-based firewall a promising solution for safeguarding modern networks.

Keywords: Feature Selection; Dragonfly Algorithm; Bat Algorithm; UNSW-NB15 Dataset

1. Introduction

Artificial intelligence (AI) has emerged as a transformative technology across multiple domains, driving innovation in infrastructure, finance, healthcare, and organizational management. Recent studies highlight its role in enhancing smart transportation systems through intelligent traffic sign recognition, optimizing recruitment and employee engagement via data-driven approaches, and advancing electronic financial services [1-4]. Moreover, AI contributes significantly to decision-making processes, software project management, and emerging paradigms such as quantum-enhanced intelligent systems, while also raising important ethical considerations regarding its deployment and societal impact [5-8]. In cybersecurity, AI is integrated into various applications, including smart home security systems and advanced security tools such as firewalls [9, 10].

A firewall is a system that is used to secure cyber realm hosts and network infrastructure from hackers. Conventional firewalls, such as stateful firewalls, create a set of rules that work like the if-else clause to alleviate attacks on the cyber realm. If the incoming traffic matches one of the rules, then the traffic is passed or stopped [11-13]. Figure 1 shows a symbolic representation of the network firewall. Nevertheless, conventional firewalls need to keep up with modern utilities and mechanisms that hackers use. Thus, the conventional firewall ought to be upgraded to contain innovative methods that can handle the new attack types. The modern firewall and security tools incorporates modified AI mechanisms that can process and analyze traffic to prevent new attacks [14-17].

Deep Learning (DL) and Machine Learning (ML) techniques, as fundamental components of AI, are increasingly incorporated into modern firewall systems. DL is a subset of artificial intelligence that employs multi-layered neural networks to automatically learn complex patterns from large-scale data, and its performance can be significantly enhanced through advanced optimization strategies such as particle

swarm optimization, reinforcement learning-based methods, and distributed computing frameworks for scalable model training [18-20]. ML enables systems to learn patterns from data and make predictions or decisions without being explicitly programmed [21-24]. ML models are trained on the historical normal and abnormal network data to avoid forthcoming hacks [25-27]. However, the network data is large, which makes the training process very difficult for ML models. Furthermore, many attributes of the network data are nonessential to hacks. Thus, the ML models will be inaccurate in distinguishing the normal and abnormal network data [28-30]. Consequently, the large size of the normal and abnormal network data has to be condensed by preserving only the significant attributes that can be utilized efficiently to avoid attacks on the network and system. The ML models frequently utilize the feature selection algorithms to complete this task. One of the main categories that is employed for feature selection is metaheuristic algorithms [31, 32].

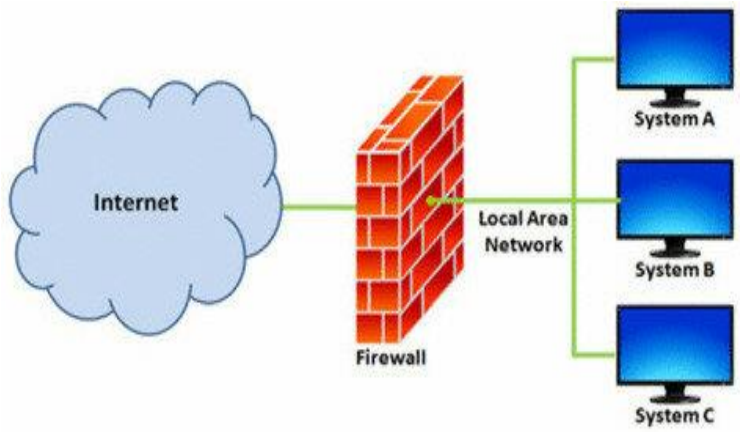

Figure 1. Network firewall

Examples of such algorithms include the Dragonfly Algorithm (DA), Bat Algorithm, Numbat Optimization Algorithm (NOA), Tuatara Optimization Algorithm (TOA), Whale Optimization Algorithm (WOA), and marine predators optimization [33-37]. In this research, the DA and BA algorithms have been employed to boost the efficiency of the AI-embedded firewalls. In particular, the DA and BA algorithms have been employed as feature selection algorithms to identify the crucial attributes in the network traffic that contribute to preventing hacking in the cyber realm. Moreover, the designed AI-embedded firewall employs Logistic Regression (LR) and Gradient Boosting Trees (GBT) classifiers.

The designed AI-embedded firewall should be assessed real-world attack scenarios or realistic simulations that closely mirror them. Therefore, one of the datasets that is appropriate for assessing the designed AI-embedded firewall is the UNSW-NB15 dataset. The UNSW-NB15 dataset comprises nine diverse and real-world attack scenarios. These attacks are Fuzzers, Analysis, Backdoors, DoS (Denial of Service), Exploits, Generic, Reconnaissance, Shellcode, and Worms. Moreover, the UNSW-NB15 dataset has 257,673 diverse samples and 42 features. Table 1 shows the features of the UNSW-NB15 dataset [38, 39].

Table 1. Features of the UNSW-NB15 dataset

#	Feature	Type	Description
1	dur	Numeric	Duration of the flow in seconds.
2	proto	Categorical	Protocol used (e.g., TCP, UDP, ICMP).
3	service	Categorical	Network service on the destination (e.g., HTTP, FTP, DNS).
4	state	Categorical	State of the connection (e.g., FIN, CON, INT).
5	spkts	Numeric	Number of packets sent by the source.
6	dpkts	Numeric	Number of packets sent by the destination.
7	sbytes	Numeric	Number of bytes sent by the source.
8	dbytes	Numeric	Number of bytes sent by the destination.
9	rate	Numeric	Data transfer rate in bytes/second.

10	sttl	Numeric	Time-to-live value set by the source.
11	dttl	Numeric	Time-to-live value set by the destination.
12	sload	Numeric	Source's bits per second (bps) rate.
13	dload	Numeric	Destination's bits per second (bps) rate.
14	sloss	Numeric	Number of packets lost by the source.
15	dloss	Numeric	Number of packets lost by the destination.
16	sinpkt	Numeric	Average time between source packets.
17	dinpkt	Numeric	Average time between destination packets.
18	sjit	Numeric	Source jitter (variation in packet delay).
19	djit	Numeric	Destination jitter (variation in packet delay).
20	swin	Numeric	Source TCP window advertisement value.
21	stcpb	Numeric	Source TCP base sequence number.
22	dtcpb	Numeric	Destination TCP base sequence number.
23	dwin	Numeric	Destination TCP window advertisement value.
24	tcprtt	Numeric	TCP connection setup round trip time.
25	synack	Numeric	Time between SYN and ACK packets.
26	ackdat	Numeric	Time between ACK and data packets.
27	smean	Numeric	Mean size of packets sent by the source.
28	dmean	Numeric	Mean size of packets sent by the destination.
29	trans_depth	Numeric	Depth of the connection in terms of transaction count.
30	response_body_len	Numeric	Size of the response body in bytes.
31	ct_srv_src	Numeric	Number of connections to the same service by the source.
32	Ct_state_ttl	Numeric	Number of connections with the same state and TTL.
33	ct_dst_ltm	Numeric	Number of connections to the same destination in the last 100 connections.
34	ct_src_dport_ltm	Numeric	Number of connections from the source to the same destination port.
35	ct_dst_sport_ltm	Numeric	Number of connections to the destination from the same source port.
36	ct_dst_src_ltm	Numeric	Number of connections between the same source and destination addresses.
37	is_ftp_login	Binary	Indicates if an FTP login attempt was successful (1: Yes, 0: No).
38	ct_ftp_cmd	Numeric	Number of FTP commands in the connection.
39	ct_flw_http_mthd	Numeric	Number of HTTP request methods in the connection.
40	ct_src_ltm	Numeric	Number of connections from the source IP in the last 100 connections.
41	ct_srv_dst	Numeric	Number of connections to the same service by the destination.
42	is_sm_ips_ports	Binary	Indicates if source and destination IPs and ports are the same (1: Yes, 0: No).

2. Related Works

S. Bagui et al. [40] have proposed a hybrid feature selection method to improve intrusion detection rate. The hybrid method combines the k-means clustering and a correlation-based feature selection to find the optimal subset of features. Overall, the key features are dur, service, sttl, and dttl, and ct_srv_src. The NB and J48 have been used in the classification process. A portion of 8000 samples from the UNSW-NB15 dataset has been used to evaluate the efficiency of the selected features. The NB has achieved better accuracy when feature selection is used with all attack types. For example, in case of Worms attack, the NB achieved 84% accuracy without feature selection and achieved 99% with feature selection. On the other hand, the J48 has achieved less accuracy when feature selection is used with all attack types. For example,

in the case of the Worms attack, the J48 achieved 99.59% accuracy without feature selection and achieved 99.94% with feature selection.

Zhou et al. [41] introduce an IDS model featuring an enhanced Harris Hawks Optimization (HHO)algorithm. The internal operations of the original HHO are adjusted with a different mechanism aimed at mitigating the impact of redundant and noisy features on the IDS model's performance. The classification task is carried out using a deep neural network (DNN). Through simulation experiments on the NSL-KDD dataset, the model achieves an accuracy of 93.12% for binary classification and 86.79% for multi-classification.

Walling S. and Sibesh L. [42] have proposed an IDS system to enhance intrusion detection in IoT networks. The proposed IDS uses a novel feature selection method that combines two filter-based algorithms. These two algorithms are Analysis of Variance (ANOVA) and Pearson Correlation Coefficient (PCC). The ANOVA is used to identify significant differences between features, while the PCC algorithm is used to evaluate the linear correlation between the features. The operations performed by the PCC and ANOVA algorithms allow the selection of the key features that help in intrusion detection operations. The proposed IDS system was evaluated on the UNSW-NB15 intrusion detection dataset. The experiments that were conducted show that the system achieved an accuracy of 97.7%.

3. Methodology

3.1. UNSW-NB15 dataset preprocessing

Figure 2 shows the proposed model. The data within the UNSW-NB15 dataset is not ready to train and test the classifiers. Therefore, first, the non-numeric data (e.g.,) within the UNSW-NB15 dataset must be converted to numeric. A well-known and robust ML algorithm that is used to numerize the data is called label-encoding. The label-ncoding algorithm numerizes the text data to numbers that start from 0 to the number of values minus 1. For example, the "protocol" feature contains text values including idrp, ifmp, igmp, igp, il, i-nslp, ip,ipcomp, ipcv, ipip, iplt, etc. The label encoding algorithm will numerize these values into 0, 1, 2, 3, 4, 5, 6, 7, 8, etc., respectively. The values in the "service" and "state" features will be numerized, using the label encoding algorithm, as well. Second, a well-known and robust ML algorithm that is used to smallerize the data is called the min-max scaler algorithm [43-47].

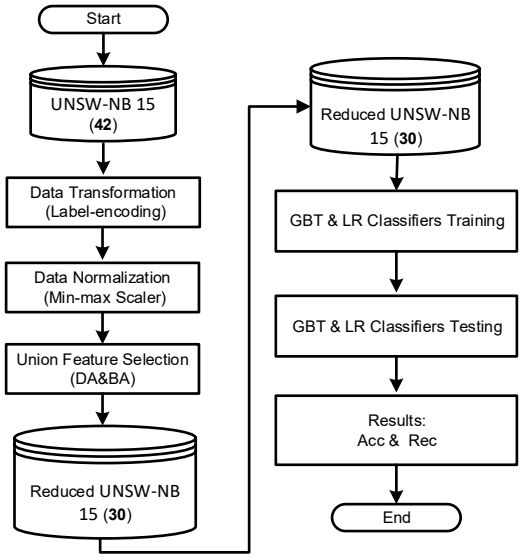

Figure 2. The proposed ML-based firewall.

3.2. DA Union BA for feature selection

The DA is a swarm-based metaheuristic optimization algorithm inspired by the static and dynamic swarming behaviors of dragonflies. The DA algorithm effectively balances exploration and exploitation using separation, alignment, cohesion, attraction, and distraction. The dynamic adaptability and strong global search make the DA robust in identifying the most relevant attacks. On the other hand, the BA is a nature-inspired metaheuristic optimization algorithm that mimics the echolocation behavior of bats. The

BA algorithm adjusts the loudness and pulse rate dynamically, which balances exploration and exploitation and leads to escape local optima. Table 2 compares and contrast the DA and BA algorithms in terms of feature selection [48-51].

Table 2. Comparison of the DA and BA algorithms

Aspect	Dragonfly Algorithm (DA)	Bat Algorithm (BA)
Inspiration	Mimics the swarming behavior of dragonflies in nature.	Mimics the echolocation behavior of bats.
Search Mechanism	Uses five swarm behaviors (separation, alignment, cohesion, attraction, and distraction) to explore the search space.	Adjusts loudness and pulse rate dynamically for exploration and exploitation.
Exploration vs. Exploitation	Stronger exploration due to its adaptive swarming mechanism.	Good balance between exploration and exploitation using frequency tuning.
Feature Selection Efficiency	Efficient in avoiding local optima and selecting minimal but highly relevant features.	Identifies optimal feature subsets effectively but may need fine-tuning to prevent premature convergence.
Computational Complexity	Requires more computational resources due to swarm behavior calculations.	Generally faster with simpler update equations.
Suitability for Attack Detection	Well-suited for handling complex attack patterns and high-dimensional datasets.	Performs well in attack detection but may struggle with highly dynamic attack patterns.
Handling of High-Dimensional Data	Better at handling high-dimensional data due to its adaptive nature.	Effective but may require additional tuning for very large feature spaces.

In this paper, a combination of DA and BA algorithms is utilized to select features from the UNSW-NB15 dataset. Figure 3 illustrates the procedures followed by the DA and BA algorithms for feature selection. The DA has selected 28 features from the 42 features in the UNSW-NB15 dataset. In parallel, the BA has selected 8 features from the 42 features in the UNSW-NB15 dataset. Then, the selected features from the BA and DA have been combined using union set theory. Therefore, the resulting set obtained by combining BA and DA contains 28 features. Table 3 shows the features selected by DA, BA, and DA union BA.

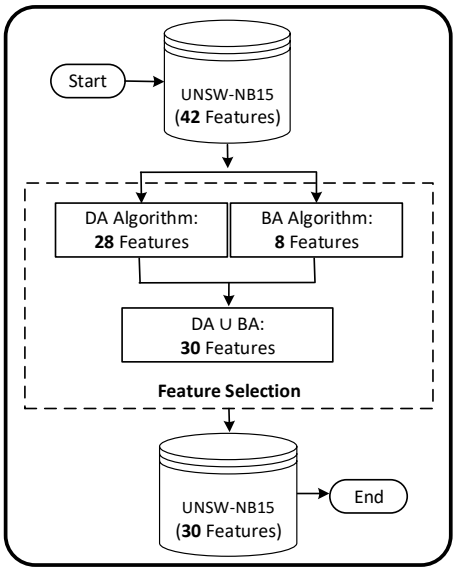

Figure 3. The proposed DA U BA features selection method.

Table 3. Features Selected by DA, BA, and DA–BA Union

Method	Selected Features
DA	1, 3, 5, 6, 8, 9, 10, 13, 14, 15, 17, 18, 19, 20, 22, 24, 25, 26, 27, 29, 30, 31, 34, 35, 37, 38, 39, 40
BA	4, 10, 13, 14, 28, 29, 38, 39
DA Union BA	1, 3, 4, 5, 6, 8, 9, 10, 13, 14, 15, 17, 18, 19, 20, 22, 24, 25, 26, 27, 28, 29, 30, 31, 34, 35, 37, 38, 39, 40

3.3. GBT and LR for classification

The GBT classifier is an iterative boosting technique that combines weak learners, typically decision trees, to minimize errors and enhance predictive accuracy by optimizing a loss function. The LR classifier is a discriminative model that uses a logistic function to estimate class probabilities efficiently and interpretably. Table 4 presents a comparative discussion of the GBT and LR classifiers. Table 5 outlines the main steps involved in the classification process for both the GBT and LR classifiers [52-54].

Table 4. Comparison the GBT and LR classifiers

Aspect	GBT	LR
Learning Approach	Ensemble learning (boosting)	Linear discriminative learning
Working Mechanism	Builds trees sequentially, correcting errors of previous ones using gradient descent.	Models the probability of class membership using a linear combination of features and a logistic (sigmoid) function.
Strengths	Handles complex relationships, reduces bias, and improves accuracy.	Simple, interpretable, computationally efficient, and effective on high-dimensional feature spaces.
Performance on Attack Detection	High accuracy due to iterative boosting but may require tuning.	Performs well for linearly separable attack patterns but limited in capturing nonlinear relationships.
Default Hyperparameters	n_estimators=100, learning_rate=0.1, max_depth=3, subsample=1.0, min_samples_split=2	penalty='l2', C=1.0, solver='lbfgs', max_iter=100

Table 5. The main steps of performing the classification process

Aspect	GBT	LR
Feature Sampling	Randomly selects features for each decision tree split, reducing correlation between trees.	Considers all input features simultaneously and assigns a learned coefficient to each feature during optimization.
Tree/Model Initialization	Creates multiple fully randomized decision trees.	Initializes model weights (coefficients) and bias, typically starting from zero or small random values.
Splitting Criteria	Randomly selects split points instead of using traditional criteria (e.g., Gini Index, entropy).	Optimizes a logistic loss function using gradient-based methods to find the best linear decision boundary.

Training Process	Trains each tree independently on a randomly sampled subset of the data.	Iteratively updates feature weights to maximize the likelihood (or minimize loss) over the training data.
Decision Making	Uses majority voting from multiple randomized trees to classify samples as attack or normal.	Applies a decision threshold (e.g., 0.5) on the predicted probability to classify samples as attack or normal.
Final Prediction Output	The sample is classified as attack or normal based on the consensus of all decision trees.	The sample is labeled as attack or normal based on the logistic model's predicted probability.

4. Results and Discussion

The effectiveness of the designed AI-based firewall will be investigated on Lenovo Legion Pro 7 Laptop with the following hardware specs: Intel Core i9-14900HX processor (4.10 GHz up to 5.8 GHz, 36MB cache, 32 threads, and 24 efficient and performance cores), NVIDIA RTX 4090 (16GB), RAM DDR5-5600Mhz (32GB), and 1TB SSD M.2. The Linux OS was used as environment to test the proposed AI-based firewall. In addition, several packages from Python 3.13 were utilized to implement the firewall operations. Using Python and Linux for deploying ML models offers a powerful combination of flexibility and efficiency, as Python provides extensive libraries and frameworks for machine learning, while Linux ensures optimal resource management and scalability.

The designed AI-based firewall will be investigated by exploiting two different measures. These measures are accuracy of the firewall (FA) and recall of the firewall (FR). Table 6 discusses these two measures. These measures are computed based on the values of the confusion matrix. The confusion matrix contains four values. These values represent the firewall true positive (FTPo), true negative (FTNe), false positive (FFPo), and false negative (FFNe). FA can be calculated based on the values of confusion matrix using Equation (1). FR can be calculated based on the values of confusion matrix using Equation (2) [55-59].

$$\text{Accuracy} = \frac{(F_{tpo}+F_{tne})}{(F_{tpo}+F_{tne}+F_{fpo}+F_{fne})} \quad (1)$$

$$\text{Recall} = \frac{F_{tpo}}{(F_{tpo}+F_{fne})} \quad (2)$$

Table 6. Evaluation measures

Metric	Definition	Purpose	Strengths
FA	Calculates the proportion of correct predictions (FTPo + FTNe) to the total predictions (FTPo + FTNe + FFPo + FFNe).	Used to evaluate overall performance and determine how well the model works in general.	Straightforward and provides a single performance measure; effective for balanced datasets.
FR	Evaluates the proportion of actual positives correctly identified (FTPo / (FTPo + FFNe)).	Assesses the model's ability to capture all relevant positive cases.	Essential for scenarios where missing positive cases (false negatives) is costly, like medical testing.

Figure 4 demonstrates the FA of the designed AI-based firewall. The GBT and LR classifiers have been exploited to present the FA of the designed AI-based firewall. The GBT accomplished the optimal FA of 100%, while the LR accomplished the near-optimal FA of 99.94%. Figure 5 demonstrates the FR of the designed AI-based firewall. The GBT and LR classifiers have been exploited to present the FR of the designed AI-based firewall. The GBT accomplished the optimal FR of 100%, while the LR accomplished the near-optimal FR of 99.98%.

Figure 6 highlights the superior performance of the proposed AI-based firewall compared to related works, demonstrating notable improvements in accuracy. By employing a hybrid feature selection approach that combines DA and BA, the firewall effectively identifies the most relevant features, enhancing

classification performance across multiple algorithms. The system achieved perfect accuracy (100%) using GBT, while LR reached a high accuracy of 99.84%. These results significantly outperform existing methods, such as those in Ref [40] with NB (99.00%) and J48 (99.94%), Ref [41] (93.31%), and Ref [42] (97.70%), all fall short of the accuracy attained by the proposed system. The dramatic improvement, especially when compared to Ref [41], illustrates the critical role of advanced feature selection in boosting classifier performance. Furthermore, the results suggest that, with effective feature engineering, traditional machine learning models like GBT can outperform more complex deep learning approaches, offering a highly efficient and accurate solution for attack detection.

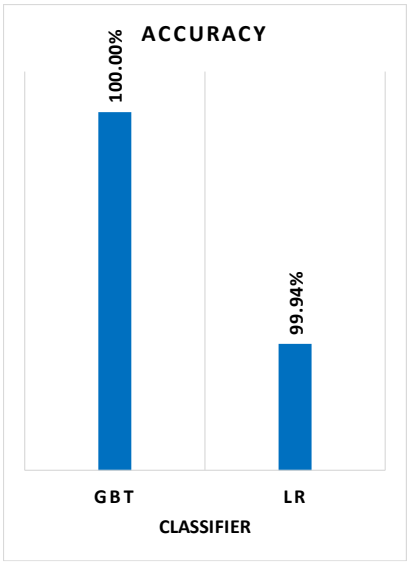

Figure 4. FA of the designed AI-based firewall.

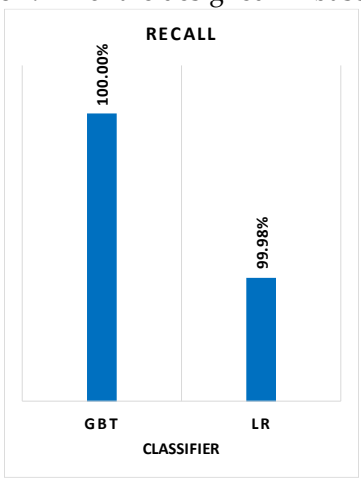

Figure 5. FR of the designed AI-based firewall.

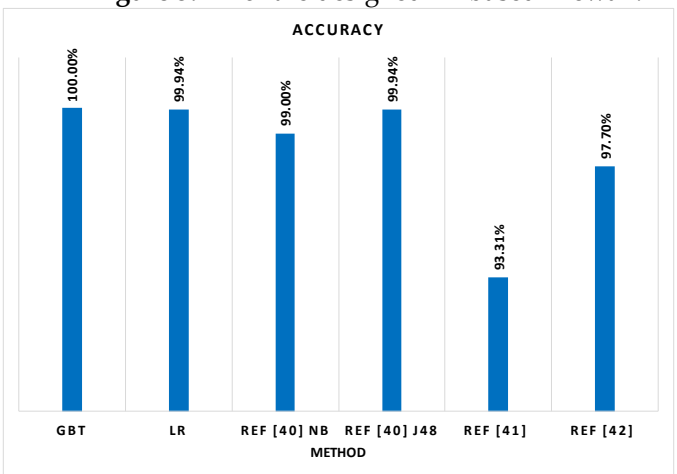

Figure 6. FA of the designed AI-based firewall vs previous works.

5. Conclusion

This research proposed an effective ML-based firewall model for attack detection in network environments. It uses advanced metaheuristic feature selection algorithms (DA and BA) to figure out which parts of the UNSWNB15 dataset are the most important. DA and BA were used together to choose the best features for the model, which improved its performance and let it find attacks more accurately with fewer false positives. We classified features using three well-known ML classifiers: GBT and LR. Among them, GBT recorded the best accuracy with 100%, while LR reached 99.94%, proving the strength of the model. The above results demonstrate that feature selection is one of the key issues in improving detection accuracy and reducing computational overhead. In the future, researchers will focus on making the model more powerful so that it can handle bigger and more varied datasets, work in real-time, and adapt to changing network environments and new attack patterns.

References

1. Ghazal, T.M.; et al. Artificial intelligence-powered smart roads: leveraging orange3 for traffic signs recognition. Bull. Electr. Eng. Informatics 2025, 14, 3. doi: 10.11591/eei.v14i3.8241
2. Khan, M.A.; et al. AI and big data-driven social media recruitment: the mediating role of talent acquisition and employee engagement in bank performance. J. Bus. Res. 2025, 186. doi: 10.1016/j.jbusres.2024.114983
3. Alzubi, E.; et al. The impact of artificial intelligence on the development of electronic financial services. Int. J. Data Netw. Sci. 2025, 9, 1. doi: 10.5267/j.ijdns.2024.9.021
4. Alshraideh, M.; et al. Analyzing the Impact of Chatbots Using Artificial Intelligence in Treating Psychological Disorders: A Bibliometric Study. Healthcare 2024, 12, 14. doi: 10.3390/healthcare12141421
5. Nabot, A., Alnaimat, F., Jebreen, I., Al-Qerem, A., & Ali, A. M. (2025, May). AI for Software Project Management: Predicting Delivery Times & Resource Allocation. In 2025 12th International Conference on Information Technology (ICIT) (pp. 277-284). IEEE.
6. Hanandeh, Essam, Shengxiang Zang, Jiajin Kang, Gang Hu, Samila Sighm, Aseel Smerat, Absalom E. Ezugwu, Vaclav Snasel, and Laith Abualigah. "Quantum Computing with Artificial Intelligence: A Paradigm Shift in Intelligent Systems." In Mastering the Minds of Machines, pp. 156-163. CRC Press, 2025.
7. Hanandeh, Essam, Faisal Al-Saqqar, Mohammad Said El-Bashir, Raed Abu Zitar, Hung Vo Thanh, Magd Abdel Wahab, Mohammad H. Nadimi-Shahraki et al. "Ethical Frontiers in Artificial Intelligence: Addressing the Challenges of Machine Intelligence." In Mastering the Minds of Machines, pp. 108-116. CRC Press, 2025.
8. Awaisheh, Sadam Mohammad, Mohammad Abdalhafid Alkhamaiseh, Mohammed Mufadi AL-Maagbeh, Lana Al Khalaileh, Mohammad Kamal Khreisat, and Mustafa AlAtiyat. "Artificial intelligence and its impact on administrative decision-making." Journal of Human Security 20, no. 1 (2024): 99-103.
9. Al Sharah, Ashraf, Tareq A. Alawneh, Hamza Abu Owida, Jawdat S. Alkasassbeh, and Zahid Iqbal. "Artificial intelligence in smart home security: balancing innovation with ethics." Bulletin of Electrical Engineering and Informatics 14, no. 6 (2025): 4601-4613.
10. Almomani, O., Arabiat, A. M., Hind, A. A., & Alsariera, E. (2026). Hybridization of Deep Learning Models for Multiclass Attack Detection in Wireless Sensor Networks. Journal of Communications, 21(1).
11. Bagheri, S.; Shameli-Sendi, A. Dynamic firewall decomposition and composition in the cloud. IEEE Trans. Inf. Forensics Secur. 2020, 15, 3526-3539. doi: 10.1109/TIFS.2020.2990786
12. Abdi, H.; et al. Security Control and Data Planes of SDN: A Comprehensive Review of Traditional, AI, and MTD Approaches to Security Solutions. IEEE Access 2024, 12, 69941–69980. doi: 10.1109/ACCESS.2024.3393548
13. Modi, N.; Acha, K. Virtualization layer security challenges and intrusion detection/prevention systems in cloud computing: A comprehensive review. J. Supercomput. 2016, 73, 1192–1234. doi: 10.1007/s11227-016-1805-9
14. Abualhaj, M.M.; Al-Khatib, S.N.; Abu-Shareha, A.A.; Hyassat, A.; Daoud, M.S. Smart Firewall for Phishing Detection Powered by Bio-Inspired Algorithms. J. Adv. Inf. Technol. 2025, 16, 1529-1539. doi: 10.12720/jait.16.11.1529-1539
15. Al Mosuli, A.; Abualhaj, M.; Abu-Shareha, A.; Yousif, M.; Daoud, M. Enhancing Intrusion Detection System Performance Using a Modified Grey Wolf Optimizer. Ann. Emerg. Technol. Comput. 2026, 10, 21-44. doi: 10.33166/AETiC.2026.01.002
16. Abualhaj, M.M.; Hiari, M.O.; Al-Khatib, S.N.; Abu-Shareha, A.A.; Daoud, M.S.; Faheem, M.R.; Al-Allawee, A.; Anbar, M. A Robust IDS System for Intelligent Phishing Website Detection. Int. J. Electr. Electron. Eng. Telecommun. 2026, 15, 1-12.
17. Alaskar, N.M.; Hussain, M.; Almheiri, S.J.; Atta-ur-Rahman; Khan, A.; Adnan, K.M. Big Data-Driven Federated Learning Model for Scalable and Privacy-Preserving Cyber Threat Detection in IoT-Enabled Healthcare Systems. Comput. Mater. Contin. 2025, early access. doi: 10.32604/cmc.2025.074041
18. Kraidia, I., Kassoul, K., Cheikhrouhou, N., Hassan, S., & Belhaouari, S. B. (2025). ExPSO-DL: An Exponential Particle Swarm Optimization Package for Deep Learning Model Optimization. Journal of Open Research Software, 13(1).
19. Shehab, M., Smerat, A., & Abualigah, L. (2025). Reinforcement Learning-based Optimization Algorithms: A Survey. Mastering the Minds of Machines, 172-177.
20. N. N. El Emam and K. Qaddoum, "Enhanced data embedding using adaptive neural networks with modified whale optimization algorithm," International Journal of Advanced Soft Computing Applications, vol. 18, no. 1, pp. 1–35, Mar. 2026.

21. Wang, Yan, Mohammed A. El-Meligy, Haytham F. Isleem, Asmaa Y. Hamed, Diyar N. Qader, Mohamed Sharaf, Pradeep Jangir, Arpita, and Ghanshyam G. Tejani. "Correction: Modeling of concrete-filled PVC tube columns confined with CFRP strips under uniaxial eccentric compression: machine learning and finite element approaches." J. Big Data 12, no. 1 (2025): 76.
22. Kausar, M. A., Ghazal, T. M., Al Batahari, M., Nasar, M., Al-Dmour, N. A., & Saeed, A. Q. (2025, May). Kidney Disease Prediction Using Machine Learning. In 2025 3rd International Conference on Business Analytics for Technology and Security (ICBATS) (pp. 1-8). IEEE.
23. Alabdallat, R.; Abualhaj, M.; Abu-Shareha, A. Enhanced Multiclass Android Malware Detection Using a Modified Dwarf Mongoose Algorithm. Int. J. Anal. Appl. 2025, 23, 1–23. doi: 10.28924/2291-8639-23-2025-0
24. M. M. Abualhaj, Q. Y. Shambour, A. A. Abu-Shareha, S. N. Al-Khatib, and A. Amer, "Enhancing malware detection through self-union feature selection using gray wolf optimizer," Indonesian Journal of Electrical Engineering and Computer Science, vol. 37, no. 1, p. 197, Jan. 2025, doi: https://doi.org/10.11591/ijeecs.v37.i1.pp197-205.
25. M. M. Abualhaj, M. O. Hiari, A. Alsaaidah, and M. M. Al-Zyoud, "Comparative analysis of whale and Harris Hawks optimization for feature selection in intrusion detection," Indonesian Journal of Electrical Engineering and Computer Science, vol. 37, no. 1, p. 179, Jan. 2025, doi: https://doi.org/10.11591/ijeecs.v37.i1.pp179-185.
26. Abualhaj, M.M.; Al-Khatib, S.N.; Abu-Shareha, A.A.; Almomani, O.; Al-Mimi, H.; Al-Allawee, A.; Daoud, M.Sh.; Anbar, M. Spam Detection Boosted by Firefly-Based Feature Selection and Optimized Classifiers. Int. J. Adv. Soft Comput. Appl. 2025, 17, 1–19. doi: 10.15849/IJASCA.251130.01
27. Abualhaj, M.M.; Al-Khatib, S.N.; Shalaldeh, A.; Al-Zyoud, M.; Daoud, M.Sh.; Al-Mimi, H.; Anbar, M. Intelligent Malware Detection through Bio-Inspired Optimization and Gradient Boosting. J. Adv. Inf. Technol. 2026, 17, 1-13.
28. Shambour, Q., Al-Zyoud, M., & Almomani, O. (2025). Quantum-inspired hybrid metaheuristic feature selection with SHAP for optimized and explainable spam detection. Symmetry, 17(10), 1716.
29. Sanjalawe, Y.; Fraihat, S.; Al-E'mari, S.; Abualhaj, M.M.; Makhadmeh, S.; Alzubi, E. Smart load balancing in cloud computing: Integrating feature selection with advanced deep learning models. PLOS ONE 2025, 20, e0329765. doi: 10.1371/journal.pone.0329765
30. Abu-Shareha, A.A.; Abualhaj, M.M.; Hussein, A.; Almomani, O.; Amer, A.; Achuthan, A.; Halin, A.A. A comparative study of the diabetes progression prediction techniques. Discov. Artif. Intell. 2025. doi: 10.1007/s44163-025-00123-x
31. Alabdallat, R.; Abualhaj, M.; Abu-Shareha, A. Android Malware Detection Using a Modified Dwarf Mongoose Algorithm. Int. J. Intell. Eng. Syst. 2025, 18. doi: 10.22266/ijies2025.0930.21
32. Abualhaj, M.M.; Al-Khatib, S.; Al-Zyoud, M.; Hiari, M.O.; Al-Allawee, A.; Alsharaiah, M.A. A Customized Machine Learning Model for Improving Malware Detection. Int. J. Comput. Netw. Inf. Secur. 2026, 18, 1–17. doi: 10.5815/ijcnis.2026.01.01
33. Al-Ou'n, A.; Alomari, S.A.; Zitar, R.A.; Ismoilov, M.; Smerat, A.; Airbow, W.; Montazeri, Z.; Dehghani, M.; Malik, O.P.; Eguchi, K. Numbat Optimization Algorithm (NOA): A Bio-inspired Metaheuristic for Solving Optimization Problems. Int. J. Intell. Eng. Syst. 2026, 19, 996-1015. doi: 10.22266/ijies2026.0228.60
34. Migdady, H.; Zitar, R.A.; Smerat, A.; Montazeri, Z.; Dehghani, M.; Malik, O.P.; Eguchi, K. Tuatara Optimization Algorithm (TOA): A Novel Bio-inspired Metaheuristic for Constrained Engineering Design Optimization. Int. J. Intell. Eng. Syst. 2026, 19, 421-435. doi: 10.22266/ijies2026.0228.26
35. Mahfouz, Khaled Houssam, Mohammed Azmi Al-Betar, Sharif Naser Makhadmeh, and Qusai Yousef Shambour. "Mitigating the task scheduling Problem in fog computing environments using marine predators optimization algorithm." Cluster Computing 28, no. 15 (2025): 973.
36. Abualhaj, M.M.; Al-Khatib, S.N.; Al-Zyoud, M.; Qaddara, I.; Hiari, M.O.; Aldossary, S.M.A. Enhanced Network Communication Security Through Hybrid Dragonfly-Bat Feature Selection for Intrusion Detection. J. Commun. 2025, 20, 607-618. doi: 10.12720/jcm.20.5.607-618
37. A. H. Alsheyab and N. N. El-Emam, "Enhanced intrusion detection systems dataset synthesis using conditional generative adversarial networks with the adaptive whale optimization algorithm," International Journal of Advanced Soft Computing Applications, vol. 18, no. 1, pp. 1–23, Mar. 2026.
38. Xie, J.; Wang, H.; Garibaldi, J.M.; Wu, D. Network Intrusion Detection Based on Dynamic Intuitionistic Fuzzy Sets. IEEE Trans. Fuzzy Syst. 2022, 30, 3460–3472. doi: 10.1109/TFUZZ.2021.3117441
39. Disha, R.A.; Waheed, S. Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. Cybersecurity 2022, 5. doi: 10.1186/s42400-021-00103-8

40. Bagui, S.; Kalaimannan, E.; Bagui, S.; Nandi, D.; Pinto, A. Using machine learning techniques to identify rare cyber-attacks on the UNSW-NB15 dataset. Secur. Priv. 2019, 2, e91. doi: 10.1002/spy2.91
41. Zhou, P.; Zhang, H.; Liang, W. Research on hybrid intrusion detection based on improved Harris Hawk optimization algorithm. Connect. Sci. 2023, 35, 2195595. doi: 10.1080/09540091.2023.2195595
42. Walling, S.; Lodh, S. Network intrusion detection system for IoT security using machine learning and statistical based hybrid feature selection. Secur. Priv. 2024. doi: 10.1002/spy2.429
43. Maz, Y.A.; Anbar, M.; Manickam, S.; Abualhaj, M.M.; Almalki, S.A.; Alabsi, B.A. Transfer Learning-Based Approach with an Ensemble Classifier for Detecting Keylogging Attack on the Internet of Things. Comput. Mater. Contin. 2025, 82, 1–10. doi: 10.32604/cmc.2025.068257
44. Abu-Shareha, A.A.; Abualhaj, M.; Hussein, A.H.; Amer, A.; Achuthan, A.; Halin, A.A. Diabetes Prediction Using Hybrid Supervised and Unsupervised Techniques Based on PIMA Dataset. J. Artif. Intell. Technol. 2025.
45. Abualhaj, M.M.; Al-Khatib, S.N.; Hiari, M.O.; Shambour, Q.Y.; Al-Allawee, A.; Almomani, O.; Daoud, M.S.; Anbar, M. An Efficient Feature Selection Technique to Enhance Spam Email Detection. TEM J. 2026, 15, 91-101. doi: 10.18421/TEM151-09
46. Kumar, M.; Bhardwaj, V. Evaluating Label Encoding and Preprocessing Techniques for Breast Cancer Prediction Using Machine Learning Algorithms. Int. J. Comput. Intell. Syst. 2025, 18, 218. doi: 10.1007/s44196-025-00957-7
47. Abualhaj, M.M.; et al. A Bio-Inspired Hybrid Optimization Framework for Efficient and Accurate Real-Time Malware Detection. Sci. Rep. 2026.
48. Mirjalili, S. Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Comput. Appl. 2016, 27, 1053–1073. doi: 10.1007/s00521-015-1920-1
49. Tsai, P.W.; Pan, J.S.; Liao, B.Y.; Tsai, M.J.; Istanda, V. Bat Algorithm Inspired Algorithm for Solving Numerical Optimization Problems. Appl. Mech. Mater. 2011, 148–149, 134–137. doi: 10.4028/www.scientific.net/amm.148-149.134
50. Taji, K.; Sohail, A.; Shahzad, T.; Khan, B.S.; Khan, M.A.; Ouahada, K. An Ensemble Hybrid Framework: A Comparative Analysis of Metaheuristic Algorithms for Ensemble Hybrid CNN Features for Plants Disease Classification. IEEE Access 2024, 12, 61886–61906. doi: 10.1109/ACCESS.2024.3389648
51. Ghanem, W.A.H.M.; et al. Cyber Intrusion Detection System Based on a Multiobjective Binary Bat Algorithm for Feature Selection. IEEE Access 2022, 10, 76318–76339. doi: 10.1109/ACCESS.2022.3192472
52. Wijaya, A.K.; Prastuti, W. Gradient Boosted Tree Based Feature Selection and Parkinson's Disease Classification. In Proc. 2019 5th Int. Conf. Sci. Technol. (ICST), 2019, 1-5. doi: 10.1109/ICST47872.2019.9166264
53. Disha, R.A.; Waheed, S. A Comparative study of machine learning models for Network Intrusion Detection System using UNSW-NB 15 dataset. In Proc. 2021 Int. Conf. Electron. Commun. Inf. Technol. (ICECIT), 2021, 1-5. doi: 10.1109/ICECIT54077.2021.9641471
54. Gaur, V.; Kumar, R. ET-RF based Model for Detection of Distributed Denial of Service Attacks. In Proc. 2022 Int. Conf. Sustain. Comput. Data Commun. Syst. (ICSCDS), 2022, 1205-1212. doi: 10.1109/ICSCDS53736.2022.9760938
55. Zubair, M.; Owais, M.; Hassan, T.; Bendechache, M.; Hussain, M.; Hussain, I.; Werghi, N. An interpretable framework for gastric cancer classification using multi-channel attention mechanisms and transfer learning approach on histopathology images. Sci. Rep. 2025, 15, 13087. doi: 10.1038/s41598-025-97256-0
56. Hussain, M.; Chen, C.; Hussain, M.; Anwar, M.; Abaker, M.; Abdelmaboud, A.; Yamin, I. Optimised knowledge distillation for efficient social media emotion recognition using DistilBERT and ALBERT. Sci. Rep. 2025, 15, 30104. doi: 10.1038/s41598-025-16001-9
57. Almomani, O., Arabiat, A., Al Tayeb, M., Almaiah, M.A., Obeidat, M., Aldhyani, T.H., Shehab, R. and Rowad, M., 2025. A robust model for android malware detection via ML and DL classifiers. Mesopotamian Journal of Big Data, 2025, pp.261-277.
58. F. Abdel-Fattah, K. A. Farhan and S. Fayyad, "Detecting double-spending attack in Blockchain Networks using k-nearest neighbor algorithm," *2025 12th International Conference on Information Technology (ICIT)*, Amman, Jordan, 2025, pp. 304-308, doi: 10.1109/ICIT64950.2025.11049288.
59. A. Chopra, S. Behal and V. Sharma, "Evaluating Machine Learning Algorithms to Detect and Classify DDoS Attacks in IoT," *2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, 2021, pp. 517-521.