Alkadhim Journal for Computer Science, Vol. 3, No.1 (2025)

Open Access ISSN: 3007-1437 (Online), ISSN: 3007-1429 (Print)

IRAQI Academic Scientific Journals

Alkadhim Journal for Computer Science (KJCS)

Journal Homepage: https://alkadhum-col.edu.iq/JKCEAS

KJCS ALKADHIM JOURNAL FOR COMPUTER SCIENCE

Hybrid PSO-Bagging Approach for Efficient and Accurate Network Anomaly Detection

[1]Hayder Adnan Mohammed*, [1]Ahmed Sadeq Jaafar

[1]Ministry of Education Iraq, General Direction of Al-Basrah Education, Al-Basrah – Iraq

Article information

Article history:
Received Nov, 11, 2024
Accepted: Dec, 23, 2024
Available online: Mar, 25, 2025

Keywords:
Anomaly Detection
IOT
Deep learning
Machine learning

Corresponding Author:
Hayder Adnan Mohammed
Haideraljasim2024@gmail.com

DOI:
https://doi.org/10.53523/ijoirVolxIxIDxx

Abstract

The exponential growth of internet usage has led to a significant rise in network attacks, posing critical cybersecurity challenges. Fog computing, an extension of cloud computing, offers low-latency services but is highly susceptible to such attacks due to its decentralized architecture and resource constraints. Traditional Intrusion Detection Systems (IDS) designed for centralized networks are often ineffective in fog environments, necessitating the development of specialized detection methods. This paper proposes a novel hybrid approach for network anomaly detection tailored for fog computing environments. The method integrates Particle Swarm Optimization (PSO)-based wrapper feature selection with the Bagging technique to address computational and accuracy challenges. Using the NSL-KDD dataset, the proposed system achieves an impressive accuracy of 98.3% while maintaining a low false positive rate of 1.5%. These results demonstrate the effectiveness of the PSO-Bagging framework in enhancing the security of fog computing systems, making it a robust solution to the growing problem of network intrusions in distributed computing environments

1. INTRODUCTION

The Internet, one of the most significant inventions in modern history, has seen exponential growth that profoundly impacts sectors such as travel, business, research, and education. A key development in this digital expansion is the Internet of Things (IOT), which, through the deployment of affordable and efficient devices like sensors and actuators, has revolutionized industries over the past decade. According to Cisco, by 2020, it is anticipated that over fifty billion IOT devices will be connected [1-3]. This proliferation has facilitated the emergence of smart environments such as cities, grids, and homes. That enhance human comfort and well-being, exemplified by cities like Padua in Italy [4].

However, IOT's expansive connectivity and constant data transmission make these devices particularly vulnerable to cyber threats. Cyber-attacks can disrupt normal operations and lead to severe consequences, as demonstrated by the widespread impact of the 2016 Mirai virus attack on Dyn, a major DNS provider. Such vulnerabilities highlight the necessity for specialized and robust intrusion detection systems (IDSs) tailored for IOT environments, which differ significantly from traditional networks in scale and capabilities [5].

Traditional IDSs are often inadequate for IOT due to devices' limited memory, network bandwidth,

computational power, and battery life. To enhance security measures within these constraints, fog computing plays a crucial role by bringing computational resources closer to where data is generated in IOT networks [6]. This approach reduces latency and bandwidth use, which is crucial for real-time processing and quick response actions. However, fog computing's decentralized and distributed nature also multiplies the potential security vulnerabilities, needing more nuanced and effective security mechanisms [7].

This research introduces a novel network anomaly detection methodology specifically designed for fog computing environments. This methodology integrates Particle Swarm Optimization (PSO)-based wrapper feature selection with the Bagging technique. This approach significantly reduces computational overhead while enhancing the accuracy and scalability of IDS tailored for IOT's unique requirements. Utilizing the NSL-KDD dataset, we demonstrate our methodology's efficacy, achieving 98.3% accuracy with a remarkably low false positive rate of 1.5%, thereby addressing critical security challenges in IOT networks facilitated by fog computing [8,9]. This paper fills the research gap by proposing an optimized IDS that is effective in detecting anomalies and scalable and efficient. This ensures robust security in distributed computing environments like fog computing, which is increasingly prevalent in IOT implementations.

This paper is structured as follows: Section 2 reviews related work in network security for IOT and fog computing. Section 3 discusses the foundational theories behind the methodologies used. Section 4 explains the methods and optimization techniques, focusing on Particle Swarm Optimization and Bagging. Section 5 details the experimental setup and results, demonstrating the model’s effectiveness. The paper concludes in Section 6, summarizing the findings and outlining future research directions.

2 RELATED WORK

Intrusion Detection Systems (IDS) have been extensively studied, with research focusing on improving detection accuracy, reducing computational complexity, and enhancing scalability for real-world applications. Below, we group related work into machine, deep, and hybrid learning approaches.

1. Machine Learning Approaches

Hashem [15] introduced a Naïve Bayes-based IDS tailored for detecting Denial of Service (DOS) attacks using the NSL-KDD dataset. By applying gain ratio feature selection, the system achieved accuracy levels of 86%, 87%, and 88% across different test datasets. Bong et al. [16] explored the Gaussian Naïve Bayes (NB) model for zero-day attack detection. The study analyzed the impact of smoothing factors on detection performance, with accuracy ranging from 38.80% (no smoothing) to 94.53% (optimal smoothing factor). This research highlights the significance of hyperparameter tuning in enhancing IDS performance.

2. Deep Learning Approaches

Su et al. [10] introduced a deep learning approach combining Bidirectional Long Short-Term Memory (BLSTM) and an attention mechanism. The BAT model achieved 94.7% accuracy on the NSL-KDD dataset, demonstrating the potential of attention mechanisms for anomaly detection. Xu et al. [11] proposed a 5-layer Autoencoder (AE)-based model with advanced pre-processing methods, achieving 90.61% accuracy. The two-sigma outlier removal and Mean Absolute Error (MAE) metric contributed to its performance. Türk [13] explored Multilayer Perceptron (MLP) and Long Short-Term Memory (LSTM) models, achieving 97.8% accuracy for binary classification and 93.4% for multi-class classification on the NSL-KDD dataset. Mohammed [14] developed an IDS using Deep Neural Networks (DNN) and Recurrent Neural Networks (RNN) with Recursive Feature Elimination (RFE). The model achieved an accuracy of 94% on the NSL-KDD dataset.

3. Hybrid Learning Approaches

Al-Yaseen et al. [12] introduced a hybrid model integrating Support Vector Machines (SVM) with Extreme Learning Machines (ELM). This method achieved 97.85% accuracy, demonstrating the effectiveness of combining different machine-learning techniques. Pakanzad et al. [17] proposed a hybrid IDS combining Convolutional Neural Networks (CNN) with Long Short-Term Memory (LSTM). Validated on the NSL-KDD and CICIDS2017 datasets, this method achieved classification accuracies of 98.1% and 96.7%, respectively, addressing multi-class classification challenges. Sarvari et al. [18] introduced an anomaly-based IDS using Mutation Cuckoo Fuzzy (MCF) for feature selection and Multi-Verse Optimizer Artificial Neural Network (MVO-ANN) for classification. This approach achieved an average accuracy of 98.13% by selecting 22 key features from the NSL-KDD dataset, significantly outperforming MVO-ANN without feature selection. Table 1 show the related work summary.

Table 1. Related work summary

Authors	Techniques	Dataset	Accuracy	Key Features
Hashem [15]	Naïve Bayes with gain ratio feature selection	NSL-KDD	86%-88%	Focused on detecting DoS attacks with minimal computational requirements.
Bong et al. [16]	Gaussian Naïve Bayes with varying smoothing factors	NSL-KDD	38.80%-94.53%	Highlighted the importance of hyperparameter tuning for optimal detection.
Su et al. [10]	BLSTM with attention mechanism	NSL-KDD	94.7%	Attention mechanism enhances anomaly detection in network traffic.
Xu et al. [11]	5-layer Autoencoder with advanced pre-processing (e.g., two-sigma outlier removal)	NSL-KDD	90.61%	Focused on dimensionality reduction and anomaly reconstruction.
Türk [13]	MLP and LSTM	NSL-KDD	97.8% (binary) 93.4% (multi-class)	Demonstrated deep learning's potential for IoT network security.
Mohammed [14]	DNN and RNN with Recursive Feature Elimination (RFE)	NSL-KDD	94%	Combined feature selection with deep learning to enhance accuracy.
Al-Yaseen et al. [12]	SVM integrated with ELM	NSL-KDD	97.85%	Hybrid model combining machine learning techniques for higher accuracy.
Pakanzad et al. [17]	CNN-LSTM hybrid	NSL-KDD, CICIDS2017	98.1% (NSL-KDD) 96.7% (CICIDS2017)	Addressed multi-class classification challenges in IDS.
Sarvari et al. [18]	MCF-based feature selection and MVO-ANN	NSL-KDD	98.13%	Combined evolutionary algorithms with neural networks to select optimal features.

While machine learning approaches provide computational simplicity, they often struggle with complex data patterns. Deep learning methods offer better accuracy but at the cost of increased computational complexity.

Hybrid approaches effectively balance accuracy and computational efficiency, making them more suitable for resource-constrained environments like fog computing. However, there is still a need for methods addressing high false positive rates and real-time scalability. In addition to the NSL-KDD dataset, researchers have used other datasets like UNSW-NB15, CICIDS-2017, KDDCup-99, and private datasets for intrusion detection.

3 BACKGROUND

This section briefly covers the essential techniques utilized in this research: PSO for selecting key features and Bagging for enhancing classification accuracy. These methods are integral to the proposed effective network anomaly detection approach in fog computing environments.

3.1. PSO/wrappers based selecting features

The PSO algorithm initializes a swarm of optimal solutions by generating random particles, with the data type used to represent each particle (e.g., bit, character, or integer) depending on the NSL-KKD database properties and features. Each particle is evaluated using the fitness function, and if the current fitness value is greater than the particle's previous best objective fitness value, it becomes the new best. The overall most optimal particle among all particles is also determined. The particle locations and velocities are then used to emend the outcomes, as described in algorithm one, until the maximum iteration number is reached. The NSL-features database achieved 8 features after the dimensionality reduction process [19, 20].

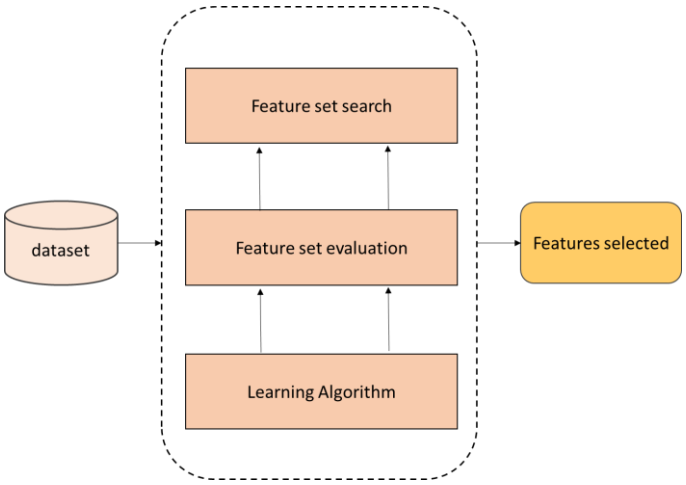

Figure. 1. Choosing features using a wrapper strategy.

3.2. Bagging technique

In the bagging method used in this work to classify emotions, a subset of the original data is sent to each classifier. This means that each classifier observes a part of the data set and builds its model according to the subset it has. The selection of this subset is associated with substitution. Based on this, each sample can be selected several times. The research has shown that the classification method can increase the ability to learn and recognize all data types with higher accuracy [21]. Figure 2 shows the general operation of this method.

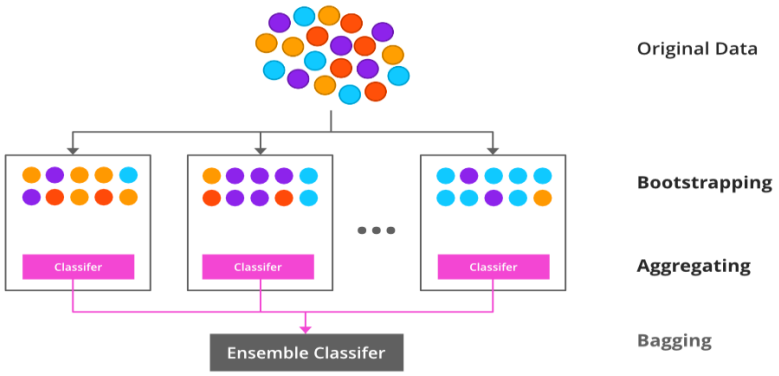

Figure 2: Classification diagram using Bagging algorithm

4. THE PROPOSED METHOD

The proposed method addresses the challenge of network anomaly detection in fog computing environments, where traditional Intrusion Detection Systems (IDS) often fail due to limited computational resources, high latency, and the distributed nature of fog nodes. Our approach integrates Particle Swarm Optimization (PSO)-based feature selection with the Bagging technique to ensure high accuracy while minimizing computational overhead. This hybrid model is specifically designed to balance real-time performance and scalability, making it suitable for fog computing systems where efficient data processing is critical. The proposed method integrates advanced techniques, including comprehensive data preprocessing, feature extraction using Principal Component Analysis (PCA), Particle Swarm Optimization (PSO) for feature selection, and the Bagging technique. This integrated approach aims to enhance network anomaly detection within fog computing environments by efficiently managing large-scale datasets, improving classification accuracy, and addressing the challenges that fog computing poses, such as limited computational resources and the need for real-time processing. The steps involved in the proposed method are illustrated in Figure 3.

4.1 Data Preprocessing

The first step involves rigorous data preprocessing to prepare the NSL-KDD dataset for analysis. This process includes:

- Label Encoding and Normalization: Label encoding converts Categorical variables into numerical values. Data normalization is then applied to scale the features, ensuring that attributes with larger ranges do not dominate the learning process [22].
- Handling Class Imbalance: The NSL-KDD dataset is significantly imbalanced between normal and attack instances. The Synthetic Minority Oversampling Technique (SMOTE) generates synthetic samples for the minority classes, addressing this imbalance and improving the classifier's performance on underrepresented attack types [23].

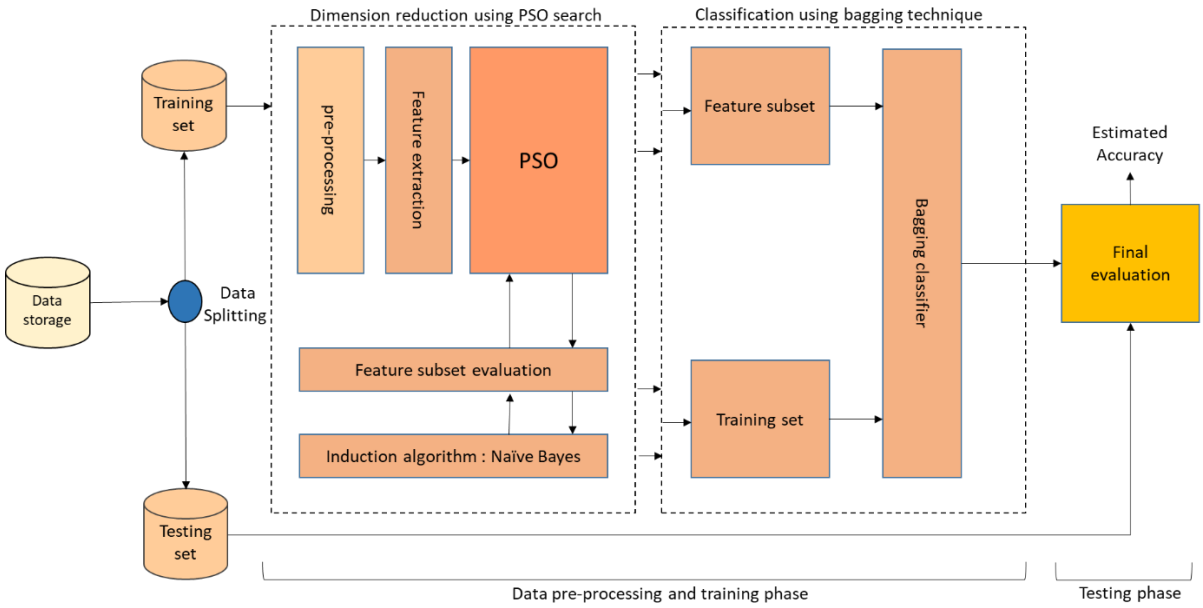

Figure 3. Proposed method architecture

4.2 Feature Extraction

Following preprocessing, the method applies Principal Component Analysis (PCA) for feature extraction. PCA reduces the dimensionality of the dataset by transforming the original features into a smaller set of uncorrelated components that retain most of the data's variance. This step is crucial for simplifying the dataset and enhancing the relevance of the features, facilitating more effective feature selection and classification [24].

4.3 PSO and Wrapper-Based Feature Selection

After feature extraction, the PSO algorithm is used for feature selection. PSO simulates the social behavior of birds flocking to find the optimal solution, which in this context is the best subset of features. The particles in the swarm represent different feature subsets, and their positions are updated based on individual and group experiences. The fitness of each particle is evaluated using performance metrics like accuracy, AUC, or F1 score. The Wrapper-based approach retains only the most relevant features, reducing the dataset's dimensionality and computational complexity.

4.4 Bagging Technique for Enhanced Classification

Following feature selection, the Bagging (Bootstrap Aggregating) technique enhances the classification process. Bagging generates multiple subsets of the training data through random sampling with replacement. Each subset is used to train a different instance of the classifier, and the final prediction is obtained by aggregating the predictions of all classifiers. This approach reduces variance and mitigates over fitting, particularly in imbalanced datasets while ensuring the model's stability across different data subsets [25].

4.5 Integrated Approach and Evaluation

The integrated methodology, data preprocessing, PCA-driven feature extraction, PSO-based feature selection, and Bagging. Form a comprehensive framework optimized for network anomaly detection in fog computing environments. The diverse attack types and normal instances in the NSL-KDD dataset provide a robust tested for evaluating the model's performance [26]. The results demonstrate significant improvements in detection accuracy, reduced computational overhead, and enhanced resilience to data imbalance and noise, making this method highly suitable for real-time deployment in fog computing scenarios.

5. EVALUATION

5.1 The dataset's preparation and description

This study utilized the NSL-KDD database [27] to evaluate the proposed model. For simulation, a subset of

25,192 samples was chosen. Any instance included forty-one continuous and categorical features, with the 42nd column indicating the attack type or normal class. The whole number of attacks and normal instances for every fold of training and test data are provided in Tables 2 and 3. The pre-processing stages were conducted to prevent the model from being complicated and minimize errors. The first stage involved converting string features into numerical values. The second stage involved data normalization to decrease the range of attribute values. This was done to ensure that those with higher values did not overshadow features with lower values. The proposed model employs one standard deviation normalization technique and zero mean. In the third stage of our pre-processing, we aimed to tackle the class imbalance issue in the NSL-KDD database, which is evident in Table 2. As a result of this imbalance, the model may incorrectly categorize U2R and R2L groups. To overcome the raised challenge, we utilized Synthetic Minority Oversampling (SMOTE) to level the dataset, as shown in Tables 3 and 4. Our model detected uncommon threats in the training data while identifying known threats in the testing data. DOS, Probe, U2R, and R2L attacks were among the important types of attacks identified in two training and testing datasets.

Table 2. Type of Attacks in the KDD-Train Dataset

Types of Attack	Samples number
Normal	67,343
Probe	11,656
U2R	52
R2L	995
DOS	45,927

Table 3. Type of Attacks in the KDD-Test Dataset

Types of Attack	Samples number
Normal	9,711
Probe	2,421
U2R	200
R2L	2,885
DOS	7,458

4.2 Assessment Criteria

The effectiveness of the proposed method is evaluated through key performance metrics, including Recall, F1 score, accuracy, and precision. These metrics are essential for comprehensively assessing the model's performance and ability to detect network anomalies accurately and efficiently. Firstly, we should define these terms [28].

- True Positive (TP) refers to instances where the model correctly identifies a normal event as normal.
- True Negative (TN): is when the model accurately classifies an abnormal event as abnormal.
- False Positive (FP): occurs when the model incorrectly classifies an abnormal event as normal, leading to a false alarm.
- False Negative (FN): happens when the model mistakenly classifies a normal event as abnormal, missing the correct detection.

1. **Accuracy:** The proportion of correctly classified instances (both positive and negative) out of the total instances [29].

$$A = \frac{(TP + TN)}{(TP + FP + TN + FN)}$$

2. **Precision:** The proportion of true positive instances among all instances classified as positive.

$$P = \frac{TP}{(TP\ +\ FP)}$$

3. **Recall:** The proportion of true positive instances among all positive ones.

$$R = \frac{TP}{(TP + FN)}$$

4. **F1 Score:** The harmonic mean of precision and recall, providing a balanced measure of the model's accuracy, especially in cases of imbalanced class distribution [30].

$$F1\ score = \frac{2PR}{(P + R)}$$

These metrics are critical for comprehensively assessing the model's performance and effectiveness in detecting network anomalies.

4.3 Configuration of the experimental procedures

The feature selection and classification tasks were carried out using the MATLAB Library tools, leveraging the 2021 version of the software. The MATLAB environment provided the necessary computational tools to implement the Particle Swarm Optimization (PSO) algorithm for feature selection and the Bagging technique for classification, ensuring robust and accurate results.

For the evaluation, the NSL-KDD dataset, which contains a variety of network traffic instances, was used. The dataset was categorized into four main types of attacks [31]:

- **Denial of Service (DoS):** Includes attack methods such as Back, Smurf, Neptune, Land, Teardrop, Pod, Mail bomb, Edstrom, Apache2, Processable, and Worm.
- **Remote to Local (R2L):** Involves attacks like Ftp_write, Multichip, Guess password, Xlock, Imap, Xsnoop, Snmpguess, Phf, Httptunnel, Snmpgetattack, Sendmail, Warezmaster, and Named.
- **User to Root (U2R):** Comprises attack vectors such as Loadmodule, Perl, Rootkit, Buffer_overflow, and Sqlattack.

Probe: Includes techniques like Ipsweep, Nmap, Portsweep, and Satan.

These categories were used to classify the various instances within the dataset, allowing for a thorough evaluation of the proposed intrusion detection model across different attack types. The experimental setup ensured that the model's performance could be assessed comprehensively, considering the dataset's diverse nature of network attacks [32].

Table 4. Overview of Attack Types and Instances in the NSL-KDD Dataset

Type of Attack	Details
DoS	*Back, Smurf, Neptune, Land, Teardrop, Pod, Mailbomb, Udpstorm, Apache2, Processtable, Worm.*
R2L	*Ftp_write, Multihop, Guess_password, Xlock, Imap, Xsnoop, Snmpguess, Phf, Httptunnel, Snmpgetattack, Sendmail, Warezmaster, Named.*
U2R	*Loadmodule, Perl, Rootkit, Buffer_overflow, Sqlattack*
Probe	*Ipsweep, Nmap, Portsweep, Satan*

5. RESULT AND DISCUSSION

This study used The NSL-KDD dataset, containing 25,192 instances with 41 attributes, was used to evaluate the proposed method. The dataset encompasses four major types of attacks: Probing, DoS, R2L, and U2R, in addition to a basic class label for normal traffic. To mitigate the risk of overfitting, a 10-fold cross-validation technique widely recognized for its robustness was applied throughout all experiments.

5.1. Overall Efficacy of the Proposed Method

The proposed model was evaluated in two experimental phases: using the original dataset and the rebalanced dataset. The results, shown in Table 6, indicate that the proposed method achieved a false positive rate (FPR) of 1.5% and a true positive rate (TPR) of 98.5%. When specifically evaluating against U2R attacks, the method achieved a TPR of 98%, while the TPR for R2L attacks was slightly lower at 95%. The detailed performance is illustrated in Figure 4.

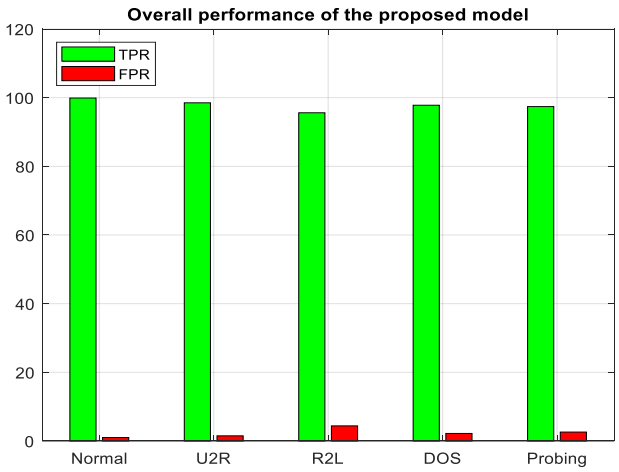

Figure 4. The overall performance of the suggested method

5.2. Comparative Analysis with Other Models

A comparative analysis evaluated the proposed model against established models, including those by Su et al. [10] (94.7%), Xu et al. [11] (90.61%), Al-Yaseen et al. [12] (97.85%), Türk [13] (97.8%), Mohammed [14] (94%), Hashem [15] (86-88%), Bong et al. [16] (38.80%-94.53%), Pakanzad et al. [17] (98.1% and 96.7%), and Sarvari [18] (98.13%). As shown in Table 6, the proposed model outperformed all, achieving an accuracy of 98.3%.

Table 5. Performance parameters

	Actual: Yes	**Actual: No**
Predicted: Yes	True Positive (TP)	False Positive (FP)
Predicted: No	False Negative (FN)	True Negative (TN)

5.3. Detailed Evaluation of Attack Detection

The model's effectiveness in detecting different types of attacks was also analyzed. Table 8 presents the True Positive Rate (TPR) and False Positive Rate (FPR) for Normal, R2L, Probing, and DoS attacks. The model achieved an overall average TPR of 97.36% and an FPR of 2.3%, indicating its reliability in identifying various attack types while maintaining a low false alarm rate.

Table 6. The proposed model performance

Class	**True Positive Rate (TPR) (%)**	**False Positive Rate (FPR) (%)**
Normal	99.89	0.11
R2L	95.7	4.3
Probing	97.5	2.5
DoS	98.3	1.7
Probing	97.5	2.5
Average Weight	97.36	2.3

4.3. Proposed methods for picking characteristics include the wrapper approach and others

Table 8 compares the wrapper approach's performance with different methods of selecting features. The wrapper approach selects 8 out of 41 available features and achieves an accuracy rate of 98.30%, outperforming CFS and consistency-based methods.

Using rank search, the consistency-based method also achieves an accuracy rate of 98.3%. However, the suggested wrapper strategy significantly outperforms the CFS filter method, which achieved 91.13% accuracy, illustrated in Tables 8 and 9.

Table 7. Results from a comparison of different methods

Metrics	Bayesian network	J48	SMO	Proposed model
Accuracy	0.86	0.97	0.96	0.98
Precision	0.85	0.95	0.94	0.97

A complete database and a subset of 8 features selected using the presented technique were used to assess the presented method regarding accuracy. As depicted in Figs. 6 and 7, the subset of 8 features achieved 98.30% accuracy compared to the method performance.

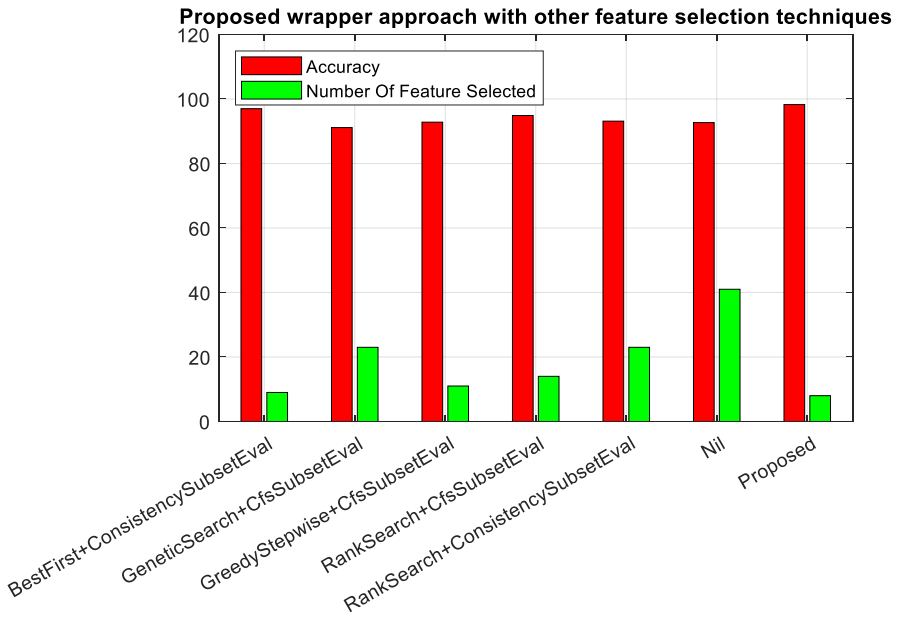

Figure. 5 Results comparison between the presented model and various feature-selection techniques

Table. 8 suggested wrapper method is contrasted with earlier techniques for feature selection.

Feature Selection techniques	Number of Selected Feature	Accuracy
BestFirst+ConsistencySubsetEval	9	97.0
GreedyStepwise+CfsSubsetEval	11	92.8
RankSearch+ConsistencySubsetEval	26	93.15
Presented Method	8	98.3
BestFirst+ConsistencySubsetEval	9	97.0
GreedyStepwise+CfsSubsetEval	11	92.8
RankSearch+ConsistencySubsetEval	26	93.15
Presented Method	**8**	**98.3**

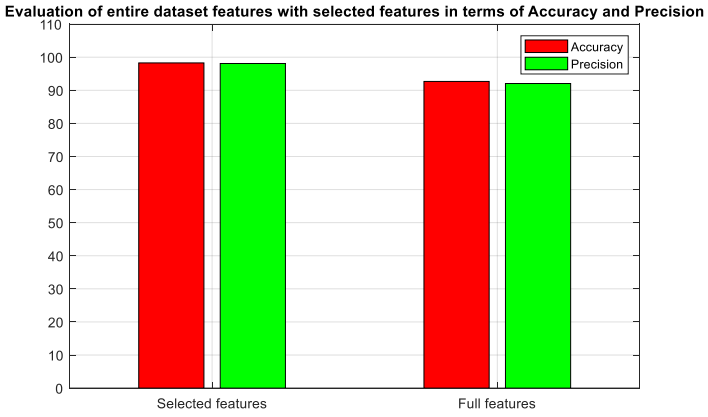

Figure 6. Comparing total dataset features and the Accuracy/Precision-based features

Table 9. Comparison of Full Dataset Features with Selected Features Based on Accuracy and Precision

NSL-KDD Database	Accuracy	Precision
Total Features	92.68	92.05
8 Selected Features	98.30	98.10

4.5. Comparing the obtained results by the presented approach with related research

According to Table 10, this research project outperforms other relevant investigations regarding accuracy. Moreover, Table 10 shows that the suggested technique performs better than various techniques in this field based on the F-score metric. Table 11 shows the proposed method's efficiency with different types of attacks.

Table 9. Comparison of the stated and alternative techniques using the F-score standards

Reference	Model	Accuracy
Su et al. [10]	BAT model	94.7%
Xu et al. [11]	5-layer Autoencoder	90.61%
Al-Yaseen et al. [12]	Hybrid SVM + ELM	97.85%
Türk [13]	MLP & LSTM	97.8%(MLP)
Mohammed [14]	DNN + RNN	94%
Hashem [15]	Naïve Bayes IDS	86% - 88%
Bong et al. [16]	Gaussian Naïve Bayes	38.80% - 94.53%
Pakanzad et al. [17]	CNN + LSTM	98.1%
Sarvari [18]	MCF + MVO-ANN	98.13%
Proposed method	PSO-Bagging	98.3%

Table 11. Comparison of SVM, Random Forest, Decision Tree, and the Proposed Model for Various Attack Types

Attacks	SVM	Random Forest	Decision Tree	Proposed Model
Normal	0.93	0.99	0.99	0.99
DOS	0.96	0.99	0.99	0.99
R2L	0.40	0.94	0.94	0.98
probe	0.88	0.99	0.99	0.96
U2R	0.61	0.85	0.79	0.79

The proposed PSO-Bagging approach, while effective in enhancing accuracy and reducing the feature set, still involves computational complexity due to the iterative nature of the PSO algorithm and the ensemble-based Bagging technique. These processes can be resource-intensive, which may pose challenges for real-time deployment in environments with limited computational resources, such as fog computing systems.

5. Conclusion and future works

Using PSO and Bagging techniques in fog computing environments offers a novel and effective approach for detecting network intrusions. The proposed method combines a Bagging-based classification technique with a wrapper feature selection method, reducing the original 41 features to a subset of 8 optimized features. This streamlined feature set significantly enhances performance, achieving an accuracy of 98.30% and a false positive rate (FPR) of 1.6%, outperforming existing classifiers. Additionally, the method demonstrates superior F-scores for Decision Tree and Random Forest algorithms compared to SVM, highlighting its robustness in detecting anomalies. Future work will address dataset limitations by validating the model using more diverse and modern datasets, such as UNSW-NB15 and CICIDS17. Moreover, extending the evaluation to include additional attack types and exploring real-time detection capabilities will further enhance its applicability in dynamic fog computing environments.

Reference

1. S. Krishnamoorthy, S. Amit, and G. Shashank, "Role of emerging technologies in future IoT-driven Healthcare 4.0 technologies: A survey, current challenges and future directions," J. Ambient Intell. Humaniz. Comput., vol. 14, no. 1, pp. 361-407, 2023.
2. K. Ashok and S. Gopikrishnan, "Statistical Analysis of Remote Health Monitoring Based IoT Security Models & Deployments From a Pragmatic Perspective," IEEE Access, vol. 11, pp. 2621-2651, 2023.
3. M. Douiba, et al., "An improved anomaly detection model for IoT security using decision tree and gradient boosting," J. Supercomput., vol. 79, no. 3, pp. 3392-3411, 2023.
4. M. Casillo, et al., "An IoT-based system for expert user supporting to monitor, manage and protect cultural heritage buildings," in Robotics and AI for Cybersecurity and Critical Infrastructure in Smart Cities, Cham: Springer International Publishing, 2022, pp. 143-154.
5. S. Kumar and B. R. Chandavarkar, "Analysis of Mirai Malware and Its Components," in Machine Learning, Image Processing, Network Security and Data Sciences: Select Proceedings of 3rd International Conference on MIND 2021, Singapore: Springer Nature Singapore, 2023.
6. M. Mohy-eddine, et al., "An efficient network intrusion detection model for IoT security using K-NN classifier and feature selection," Multimedia Tools Appl., vol. 1-19, 2023.
7. D.-M. Ngo, et al., "HH-NIDS: Heterogeneous Hardware-Based Network Intrusion Detection Framework for IoT Security," Future Internet, vol. 15, no. 1, p. 9, 2023.
8. M. Mahamat, G. Jaber, and A. Bouabdallah, "Achieving efficient energy-aware security in IoT networks: a survey of recent solutions and research challenges," Wireless Netw., vol. 29, no. 2, pp. 787-808, 2023.
9. D. K. Reddy, et al., "Deep neural network based anomaly detection in Internet of Things network traffic tracking for the applications of future smart cities," Trans. Emerg. Telecommun. Technol., vol. 32, no. 7, p. e4121, 2021.
10. T. Su, H. Sun, J. Zhu, S. Wang, and Y. Li, "BAT: Deep learning methods on network intrusion detection using NSL-KDD dataset," IEEE Access, vol. 8, pp. 29575-29585, 2020.
11. W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei, and F. Sabrina, "Improving performance of autoencoder-based network anomaly detection on NSL-KDD dataset," IEEE Access, vol. 9, pp. 140136-140146, 2021.
12. W. L. Al-Yaseen, Z. A. Othman, and M. Z. A. Nazri, "Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system," Expert Syst. Appl., vol. 67, pp. 296-303, 2017.
13. F. Türk, "Analysis of intrusion detection systems in UNSW-NB15 and NSL-KDD datasets with machine learning algorithms," Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, vol. 12, no. 2, pp. 465-

477, 2023.
14. B. Mohammed and E. K. Gbashi, "Intrusion detection system for NSL-KDD dataset based on deep learning and recursive feature elimination," Eng. Technol. J., vol. 39, no. 7, pp. 1069-1079, 2021.
15. S. Hashem and H. Adil, "Denial of service intrusion detection system (IDS) based on Naïve Bayes classifier using NSL KDD and KDD cup 99 datasets," J. Al-Rafidain Univ. Coll. Sci., no. 2, pp. 206-231, 2017.
16. K. Bong and J. Kim, "Analysis of intrusion detection performance by smoothing factor of Gaussian NB model using modified NSL-KDD dataset," in 2022 13th International Conference on Information and Communication Technology Convergence (ICTC), 2022, pp. 1471-1476.
17. A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: hybrid deep neural network for network intrusion detection system," IEEE Access, vol. 10, pp. 99837-99849, 2022.
18. S. Sarvari, N. F. M. Sani, Z. M. Hanapi, and M. T. Abdullah, "An efficient anomaly intrusion detection method with feature selection and evolutionary neural network," IEEE Access, vol. 8, pp. 70651-70663, 2020.
19. R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 1995, pp. 39-43.
20. A. P. Engelbrecht, *Computational Intelligence: An Introduction*, 2nd ed. Hoboken, NJ, USA: Wiley, 2007.
21. L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.
22. Z. Liang, D. Schwartz, G. Ditzler, and O. O. Koyluoglu, "The impact of encoding–decoding schemes and weight normalization in spiking neural networks," Neural Networks, vol. 108, pp. 365-378, 2018.
23. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321-357, 2002
24. H. Al-Kabbi, M.-R. Feizi-Derakhshi, and S. Pashazadeh, "Multi-type feature extraction and early fusion framework for SMS spam detection," IEEE Access, 2023.
25. Y. Zhang, X. Li, and Y. Sun, "An improved bagging ensemble method for intrusion detection," IEEE Access, vol. 10, pp. 10229-10241, 2022.
26. S. Rani and V. Sharma, "A comprehensive review of anomaly detection methods in fog computing," Future Generation Computer Systems, vol. 116, pp. 65-83, 2021.
27. R. Bala and R. Nagpal, "A review on KDD Cup 99 and NSL KDD dataset," International Journal of Advanced Research in Computer Science, vol. 10, no. 2, 2019.
28. H. Al-Kabbi, M.-R. Feizi-Derakhshi, and S. Pashazadeh, "A hierarchical two-level feature fusion approach for SMS spam filtering," Intelligent Automation & Soft Computing, vol. 39, no. 4, 2024.
29. F. A. Bida, "Medical image improvement using a proposed algorithm," *AlKadhim Journal for Computer Science*, vol. 2, no. 1, pp. 1-1, Mar. 14, 2024.
30. M. R. F. Derakhshi, E. Zafarani-Moattar, H. A. Al-Kabi, and A. H. J. Almarashy, "PCLF: Parallel CNN-LSTM fusion model for SMS spam filtering," *BIO Web of Conferences*, vol. 97, p. 00136, 2024.
31. D. K. Reddy et al., "Deep learning for intrusion detection in cloud computing: A survey," Journal of Cloud Computing: Advances, Systems and Applications, vol. 9, no. 1, pp. 1-15, 2020.
32. KHALAF, ALI D. "A Robust Privacy Preserving Authentication Scheme for IOT Environment by 5G Technology." *Alkadhim Journal for Computer Science* 2, no. 1 (2024).