Mesopotamian journal of Cybersecurity
Vol.5, No.3, **pp.** 1109–1121
DOI: https://doi.org/10.58496/MJCS/2025/059; ISSN: 2958-6542
https://mesopotamian.press/journals/index.php/cybersecurity

Research Article

A New Framework Enhancing, Evaluation, and Benchmarking for Cybersecurity DDoS Attack Detection Models Through the Integration of BWM and VIKOR Methods

Alaa Mohammed Mahmood[1,*], ,İsa AVCI[1], , Sahar Yousif Mohammed[2],

[1]*Department of Computer Engineering, Karabuk University, Karabuk, 78000, Turkey*

[2]*Department of Translation, Arts College, Anbar University, 31001, Iraq*

ARTICLEINFO

Article History

Received 8 Apr 2025
Revised 24 May 2025
Accepted 29 Aug 2025
Published 8 Oct 2025

Keywords
Cybersecurity
Application layer Attack
DDoS Attack detection
MCDM
BWM
VIKOR

ABSTRACT

The proposed work aims to develop a DDoS attack detection model that targets web servers by selecting the appropriate classifier based on several criteria and enhancing the accuracy. The detection system has been constructed using machine-learning algorithms to train and test the CIC-DDoS2019 dataset, and it is integrated with a stacked classifier to enhance accuracy. The classifiers depend on multiple criteria. For that, we employed the BWM to calculate the criteria weights and VIKOR to rank the classifiers, which are part of the MCDM methods. We consulted three experts to establish weights for the categories. Also, we utilized two methods to verify the results: objective validation and sensitivity analysis. BWM and VIKOR have effectively selected the most suitable classifier based on multiple criteria, making them one of the most appropriate choices. The BWM method achieved the highest weight of 0.2436 for the time criterion and 0.2302 for the accuracy criterion. VIKOR methods demonstrated that the SVM classifier proved more efficient and superior to other classifiers. It achieved 99.32% accuracy and a processing time of 9 seconds, making it suitable for use on high-priority websites, such as e-commerce platforms or state security websites. The stacked classifier can be applied to other projects where the time criterion is less critical, achieving a 99.57% accuracy enhancement in 143 seconds. Objective validation and sensitivity analysis confirmed the validity of the results, demonstrating that all scenarios consistently ranked the SVM as the highest among all classifiers, highlighting this classifier's remarkable ability to balance the criteria.

1. INTRODUCTION

Distributed Denial of Service (DDoS) attacks have become a pervasive and disruptive threat in the digital landscape. DDoS attacks aim to overwhelm and incapacitate targeted systems, rendering them inaccessible to legitimate users. By flooding a network, website, or online service with overwhelming traffic or malicious requests, DDoS attacks disrupt normal operations, causing significant downtime, financial losses, and damage to an organization's reputation [1,2]. DDoS attacks come in multiple forms. Flood attacks, such as TCP SYN floods, overwhelm servers by forcing them to allocate significant resources to respond to many connection requests. Similarly, HTTP floods involve sending large volumes of HTTP requests, overloading servers, and preventing them from processing legitimate traffic. Amplification attacks exploit vulnerable network protocols to magnify the impact of an attack. For example, DNS amplification exploits DNS servers to direct excessive traffic toward a target, while NTP amplification leverages Network Time Protocol servers to amplify the attack. Lastly, application-layer attacks focus on disrupting specific services. HTTP/HTTPS Layer 7 attacks generate massive, forged requests to overwhelm the application layer, rendering the service unavailable to legitimate users.

The decision-making process is considered one of the most complex challenges. Sometimes, the available alternatives are numerous, intertwined, and disparate, and the manager must carefully study them to make the best decision. The manager's success is measured by their ability to make the appropriate decision at the right time and eliminate alternatives that do not yield benefits. For the project, multicriteria decision analysis (MCDA) is one of the most effective tools for aiding decision-making [3-5]. The decision-making process involves considering multiple criteria to reach an informed conclusion. It begins with fixing criteria, where all significant impact factors are identified. Next, weights are assigned to each criterion based on their importance. After that, the influence of each factor on the criteria is analyzed. Data analysis follows if the decisions are made based on the collected information. Once all aspects are considered, a final decision is reached. The

*Corresponding author Email: alaamahmood526@gmail.com.

process does not end there; continuous assessment is necessary to monitor results and evaluate the validity of the decision over time.
Additionally, a decision matrix is often used, where rows represent alternatives and columns indicate criteria, aiding in a structured comparison of options. There are many types of MCDM: TOPSIS, AHP, and ANP, each with a different application. The applications are business, management, healthcare, and policy.

The best-worst method (BWM) is a decision analysis approach used in performance improvement management and multi-criteria decision-making (MCDM). We employ this method to evaluate and rank various alternatives based on a predefined set of criteria [6,7]. The implementation of the BWM follows a structured approach. It begins with establishing the criteria used to evaluate the alternatives. Next, alternatives are assessed based on these criteria using a predefined scale, which can be numerical or descriptive (e.g., “best,” “good,” “average,” “bad,” “worst”). After evaluation, the best and worst alternatives are identified, where the highest-rated alternative is considered the “best” and the lowest-rated one as the “worst.” The next step involves determining the weight of each criterion by calculating the evaluation difference between the best and worst alternatives, often using specific formulas. Once the weights are assigned, the final scores for each alternative are calculated by multiplying the corresponding criterion values by their respective weights. Finally, the alternatives are ranked based on their final scores, with the alternative scoring the highest being the most favourable choice.

The VIKOR approach (ViseKriterijumska Optimizacija I Kompromisno Revenge) is one of the decision-making tools included. The development of this method specifically addressed the task of categorizing and rating alternatives, assessing them according to a set of criteria [8,9]. The VIKOR approach follows a systematic decision-making procedure. It begins with identifying the available alternatives and the criteria used for evaluation. Next, weights are assigned to each criterion based on their relative importance, which can be determined through expert judgment or opinion surveys. Once the weights are established, an assessment table is constructed, where each alternative is represented as a column, and the corresponding criteria values are placed in rows. The next step involves calculating the ideal distance by applying weights and preference scores to measure the difference between the values of each option. The total and negative values are computed, representing the overall evaluation of alternatives and their differences. The alternatives are then ranked based on these values, with the highest option indicating the most suitable choice. Finally, the results are analyzed to determine the most suitable intermediate solution among the alternatives [10]. We use the VIKOR method to identify optimal solutions and balance performance, cost, and risk. VIKOR enables balanced decision-making that considers the decision-maker's personal preferences and excels at handling advanced and fuzzy data.

The problem of research is that Application-layer attacks are considered one of the types of distributed denial-of-service attacks (DDoS attacks), also known as Layer 7 attacks. They target the highest layer in the OSI model, the application layer, where communication occurs between software and end-users. The infected devices act as soldiers, executing commands from an attacker simultaneously. This attack exhausts the server's resources, preventing legitimate users from accessing it or making it invisible. Therefore, the danger of this type of attack remains significant, and eliminating it is a challenging task.

The research gap and aim are that much research has appeared in this field. However, it has not directly shed light on choosing the appropriate classifier that relies on several criteria to detect DDoS attacks, and this is one of the priorities of this paper. In addition to improving the detection accuracy for the classifiers, this work aims to develop an effective method for detecting the impact of DDoS attacks targeting web servers.

This work is motivated by many institutions, companies, and websites affiliated with state security that continue to suffer from DDoS attacks. Until now, artificial intelligence algorithms have not been classified based on several criteria. Algorithm X, which has high accuracy, cannot be relied upon entirely, as its effectiveness depends on multiple factors, including accuracy and time, rather than just accuracy itself.

2. Related Work

Akinwale et al. (2024) presented "A Regenerative Model for Mitigating Attacks on HTTP Servers for Mobile Wireless Networks", which focuses on the strength of the HTTP protocol. The CICIDS2017 dataset and techniques such as SMOTE, random sampling, random dropout, and principal component analysis were used. Akinwale et al. (HReg) demonstrated a robust definition against SQL injection and DoS attacks, enhancing mobile network security. However, researchers have highlighted the need for real-world data to evaluate model performance. They also recommend firewalls and continuous monitoring to ensure long-term reliability in network environments [11].

Dogra and Taqdir (2024), in their work "Enhancing Detection of Distributed Denial of Service Attacks and Network Elasticity through Packet Processing and Frequency Range Optimization," employed random forest algorithms to analyze network traffic and optimize frequency ranges. Their group-based approach significantly reduces packet rates, improving network elasticity and resistance to DDoS attacks. However, the effectiveness of their model decreased with more complex attack patterns, which remain underexplored. The authors recommend further testing in diverse network settings to increase adaptability and reliability [12].

Tedyyana et al. (2024) developed "Automated Learning for Network Defense: Real-Time Detection of DDoS Attacks with Telegram Notifications," which achieved 99.77% accuracy and an F1 score of 98.70% when DT, SVM, and neural networks were trained on the CICIDS2018 dataset. Integrating a Telegram-based notification system for real-time alerts enhances its practical application; however, reliance on Telegram limits integration with other notification protocols. The study recommends retraining the model with new attack data to adapt to evolving environments [13].

Bindu et al. (2024), in their study "Detection of DDoS Attacks in SDN Networks Using Machine Learning," utilized machine learning algorithms such as random forests, k-nearest neighbours, decision trees (DT), and logistic regression (LR) to analyze network traffic. The authors demonstrated that combining software-defined networking (SDN) with machine learning provides an effective method for detecting and mitigating DDoS attacks. Although the study emphasizes the significance of cooperative cybersecurity frameworks, it lacks real-time application, potentially limiting its utility during active attacks. The authors recommend further research into advanced machine-learning techniques to enhance detection capabilities in cooperative security networks [14].

Layeq et al. 2024 examined the application of Edge-IIoT networks and SMOTE for training ensemble learning models. They utilized hard voting, soft voting, and stacking techniques to improve detection rates for DDoS attacks in Edge-IIoT environments. However, class imbalance may affect model accuracy in real-world environments. Layeq et al. recommend addressing class imbalance issues and exploring broader IoT security challenges in future work [15].

Khedr et al. (2023). “A multi-layer DDoS Attack Detection and Mitigation Framework Using Machine Learning for Stateful SDN-Based IoT Networks” Dataset: Cloud traffic dataset collected from a primary cloud service provider. Algorithms: Support Vector Machines (SVM), Long Short-Term Memory (LSTM). Conclusion: The authors concluded that their machine learning-based approach effectively detected and mitigated DDoS attacks in cloud networks. SVM models were more effective at differentiating between regular and attack traffic; however, LSTM models could detect temporal patterns in network traffic more quickly, enabling earlier detection of attacks. Recommendations: For further research, the authors recommend exploring ensemble learning algorithms to improve detection. They also suggest adding real-time threat intelligence feeds and building adaptive defence mechanisms. Future Work: The authors aim to investigate explainable AI models to better understand the features that lead to attacks. They also emphasized the need for continuous research, as DDoS attack methods constantly evolve [16].

Wang & Li. (2024). “Overview of DDoS Attack Detection in Software-Defined Networks”. Dataset: Custom dataset generated through software-defined network (SDN) testbed. Algorithms: Clustering-based anomaly detection, Random Forest (RF) classification. Conclusion: The authors assert that machine learning and behavioural analysis effectively detect DDoS attacks with SDN environments. The clustering-based anomaly detection method helped them identify when networks were not behaving normally, while RF classification accurately categorized attack traffic. Recommendations: The authors aim to experiment with deep learning models, such as recurrent neural networks (RNNs) or graph neural networks (GNNs), to capture more complex relationships in network data. Future Work: Conduct experiments with real-world SDNs; develop defences that can adjust on the fly as new attacks hit [17].

3. Methodology

Figure 1 illustrates the key components of this work.

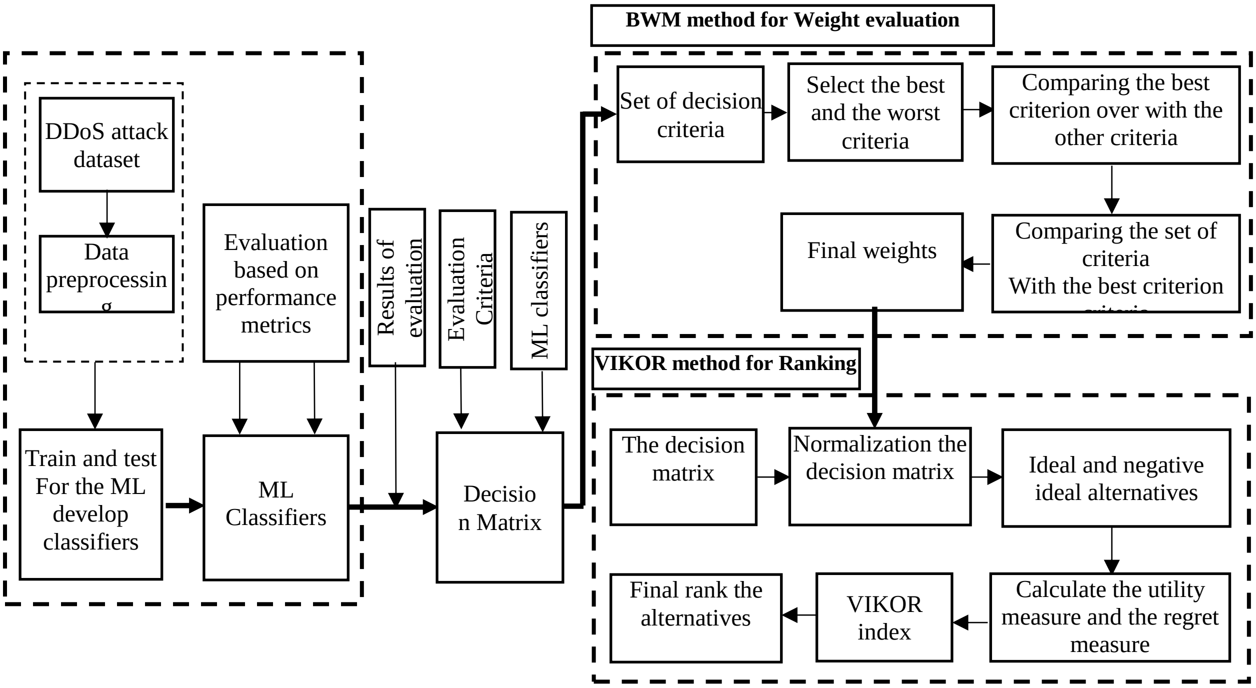
BWM method for Weight evaluation
DDoS attack dataset
Data preprocessin g
Evaluation based on performance metrics
Results of evaluation
Evaluation Criteria
ML classifiers
Set of decision criteria
Select the best and the worst criteria
Comparing the best criterion over with the other criteria
Final weights
Comparing the set of criteria With the best criterion criteria
VIKOR method for Ranking
Train and test For the ML develop classifiers
ML Classifiers
Decisio n Matrix
The decision matrix
Normalization the decision matrix
Ideal and negative ideal alternatives
Final rank the alternatives
VIKOR index
Calculate the utility measure and the regret measure

Fig. 1. Proposed Model block diagram

Fig. 1. This work used the CiC-DDoS2019 dataset for the detection phase. The dataset is available at https://www.unb.ca/cic/datasets/ddos-2019.html. See Fig. 1. Regarding data preprocessing, CIC-DDoS2019 may contain errors, missing values, outliers, and other issues that need to be addressed before data analysis or modeling can be conducted successfully. Five machine learning (ML) algorithms- decision tree (DT), support vector machine (SVM), logistic regression (LR), naive Bayes (NB), and k-nearest neighbour (KNN) with the stacked classifier (SC) were used to train and test the CiC-DDoS2019 dataset. Stacked classification (SC) is an ensemble technique that utilizes the results of multiple classifiers as input for a meta-classifier to make the final classification decision. The evaluation matrix was used to rate the detection work based on several key measurements: accuracy (ACC), precision (PREC), recall (REC), F1-score (F1), and execution time (T). We trained and examined the datasets using the Python classifiers and determined the corresponding criteria values [18, 19].

The decision matrix comprises two primary components: criteria and alternatives. The alternatives are the evaluation algorithms (DT, SVM, LR, NB, KNN, and SC), and the criteria are ACC, PREC, REC, F1, and T. Table 1 provides a simple description of the decision matrix [20].

TABLE.I :THE DECISION MATRIX

Alternatives	**Criteria**				
	ACC	**PREC**	**REC**	**Fl**	**T**
DT	ACC (1)	PREC (1)	REC (1)	F1 (1)	T (1)
SVM	ACC (2)	PREC (2)	REC (2)	F1 (2)	T (2)
LR	----	----	----	----	----
NB	----	----	----	----	----
KNN	----	----	----	----	----
SC	ACC (n)	PREC (n)	REC (n)	F1 (n)	T (n)

3.1 Benchmarking and Evaluation by Integrating BWM with VIKOR

The evaluation and benchmarking methodologies are based on MCDM techniques. This study formulates its strategy by combining BWM for assigning criteria weights and VIKOR for model ranking. Through this integration, the best option can be selected from several models. The literature analysis on MCDM techniques highlights BWM and VIKOR as suitable methods for benchmarking and ranking multiclass classification models. We suggest using the VIKOR mathematical model to address specific issues, such as managing multiple evaluation criteria within the proposed decision matrix. We further utilize BWM to assign weights to the criteria, thereby addressing the significance of each criterion in the proposed decision matrix. Therefore, the practical integration of BWM and VIKOR methods is justified for benchmarking multiclass classification models and determining their rankings.

3.1.1 BWM Method (Calculate Criteria Weights)

The Best-Worst Method (BWM) is a multi-criteria decision-making technique that involves determining the best and worst criteria (see Fig.1) within a set and then calculating the weights for each criterion based on these comparisons [21,22]. We asked three experts with a long history in cybersecurity to assign weights to the criteria, and they did so by completing a form as in Table 2.

TABLE II: SAMPLE FORM FOR WEIGHT EVALUATION (BWM METHOD)

Expert 1	**Criterion 1**	**Criterion 2**	**Criterion 3**	**Criterion 4**	**Criterion 5**
Names of Criteria	ACC	PREC	REC	F1	Time
Select the Best	Criterion name				
Select the Worst	Criterion name				
Best to Others	ACC	PREC	REC	F1	Time
Criterion name	Expert 1 (Value 1)	Expert 1 (Value 2)	Expert 1 (Value 3)	Expert 1 (Value 4)	Expert 1 (Value 5)
Others to the Worst	Criterion name				
ACC	Expert 1 (Value 1)				
PREC	Expert 1 (Value 2)				
REC	Expert 1 (Value 3)				
F1	Expert 1 (Value 4)				
Time	Expert 1 (Value 5)				
Weights	ACC weight	PREC weight	REC weight	F1 weight	Time weight

3.1.2 VIKOR Method (Model Ranking)

The VIKOR method is beneficial when dealing with conflicting criteria and the need for compromise solutions. It, in turn, provides a systematic approach to classifying alternatives during decision-making, encompassing several key criteria. Based on the average weights obtained from the experts using the BWM method (see Fig.1) and the machine learning results that extracted the criterion values, the rank of each classifier, as determined by the VIKOR method, is presented in Table 3.

TABLE III: VIKOR RANKING

Classifier Name	Rank Value
DT	X
SVM	X
LR	X
NB	X
KNN	X
SC	X

4. Results

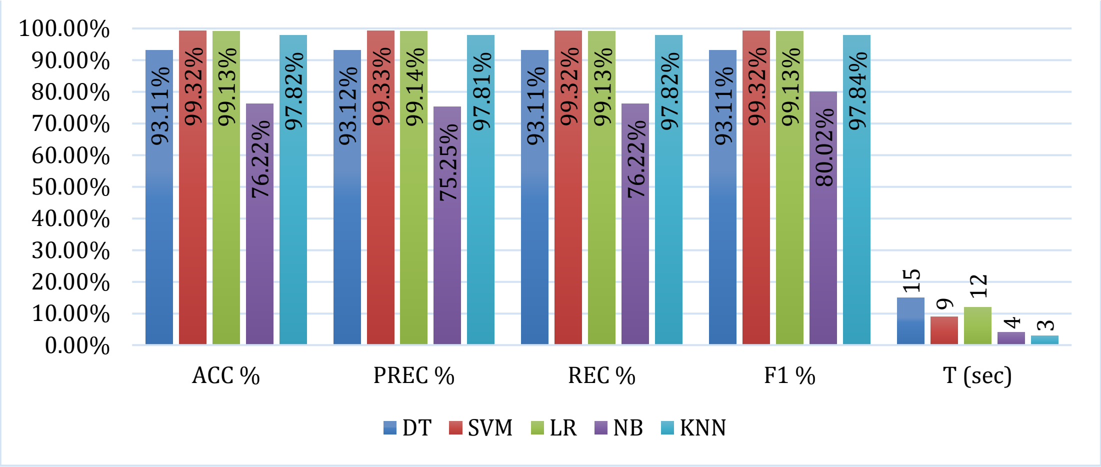

Fig. 2. ML algorithms results

In Fig. 2, the SVM classifier performed superior to the other five individual classifiers. Its results were notable, with an ACC of 99.32%. Compared to the different classifiers, the SVM classifier's supremacy highlights the dataset's underlying linear nature. It is concluded that the linear support vector classifier can define clear boundaries between different classes using linear methodologies, making it an optimal choice for addressing this challenge. Conversely, the NB classifier displayed the least robust performance of 76.22% ACC. The suboptimal performance of the NB classifier, which originated from probabilistic foundations, accentuates its incompatibility with the prevalent problem context.

We utilized the ensemble stacked classifier to enhance the overall detection performance, as demonstrated in Table 4 and Fig.3. Combining multiple classifiers yields better results than using them individually. The (DT+SVM+LR+KNN) stacked classifier performed remarkably well, outperforming single-based classifiers and other ensemble configurations with a remarkable 99.57% ACC. Fig.3 displays the stacked classifier's ACC, PREC, REC, F1, and T for the combination (DT+SVM+LR+KNN).

TABLE IV.:Stacked classifiers' accuracy

Stacked Classifiers	ACC
DT+SVM+LR+KNN	99.57%
KNN+SVM	99.31%
SVM+KNN	99.29%
DT+SVM	99.22%
LR+KNN	99.22%
DT+SVM+NB	99.20%
SVM + NB	99.17%
LR+NB	99.17%
SVM+LR	99.12%
DT+LR	98.73%
DT+KNN	98.34%
DT+NB	92.68%

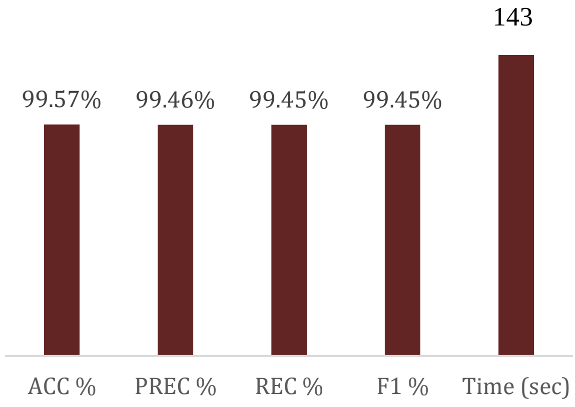

Fig3.SC (DT+SVM+LR+KNN) results

The implementation time for the combination of (DT + SVM + LR + KNN) was 143sec, which is considered relatively long compared with the rest of the classifier implementation times (Fig.3), and this conflicts with one of the interests of this work, in which the time factor is important for eliminating the attack as quickly as possible. In Table 4, all the mentioned values except the value of "DT+SVM+LR+KNN" are less than the accuracy value of the SVM classifier (99.32%), so the work relies only on the value of the stacked classifier "DT+SVM+LR+KNN", and there is no need for the remaining classifiers because their accuracy is less than the accuracy of the SVM; of course, their implementation time is larger than the SVM implementation time 9sec (Fig.2), which is due to the integration of more than one classifier.

4.1 Qualitative and Quantitative Analysis of The Algorithms Results

Through qualitative analysis, the Stacking Classifier (SC) algorithm achieved the highest accuracy but suffered from a very slow execution, making it suitable for non-time-sensitive applications. The SVM algorithm demonstrated high performance and an excellent balance between accuracy and speed, making it ideal for sensitive applications such as cybersecurity. Logistic Regression (LR) provides good performance with easy interpretation and is suitable for cases requiring clear statistical analysis. KNN is a fast and efficient algorithm with good accuracy and is suitable for real-time systems. Decision Tree (DT) is easy to understand but average in performance and is used when transparency is required. In contrast, Naive Bayes (NB) is the fastest in execution but the weakest in accuracy and is suitable for applications that prioritize speed over performance, such as initial classification Table 5 illustrates this analysis.

TABLE V: QUALITATIVE ANALYSIS OF THE ALGORITHMS

Algorithm	Advantages	Disadvantages	Most Suitable Use Case
SC (Stacking Classifier)	Highest accuracy among all algorithms	Very slow (long execution time)	Applications requiring maximum accuracy without time constraints
SVM (Support Vector Machine)	Very high accuracy, stable performance	Requires moderate computational resources	Accuracy-critical systems such as cybersecurity
LR (Logistic Regression)	Strong performance, mathematically interpretable	Less effective with non-linear data	Applications requiring interpretable models
KNN (K-Nearest Neighbors)	Fast, good accuracy, easy to implement	Sensitive to the number of neighbors, slows with large data	Real-time systems and medium-sized applications
DT (Decision Tree)	Easy to understand and interpret, transparent	Moderate performance, prone to overfitting	Educational use and applications needing decision transparency
NB (Naive Bayes)	Fastest in execution, very simple	Weakest accuracy, based on unrealistic assumptions	Preliminary classification or speed-prioritized scenarios

The summary of the quantitative analysis can be shown in Table 6.

TABLE VI: QUANTITATIVE EVALUATION TO THE ALGORITHMS RESULTS

Algorithm	Performance (Accuracy)	Time Efficiency	Overall Balance
SC	Best performance	Very slow	Excellent but time-inefficient
SVM	Very high performance	Acceptable time (9 sec)	Best balance between accuracy and speed
LR	Very close to SVM	Slightly slower (12 sec)	Very good
KNN	Very good	Fastest	Suitable for real-time systems
DT	Moderate performance	Slow (15 sec)	Below average
NB	Weakest performance	Very fast	Not recommended due to poor accuracy

4.2 Building the Decision Matrix

The classifiers represent the alternatives based on the detection results, and ACC, PREC, REC, F1, and T represent the criteria (Table 7). We will use this matrix later to find the results of the BWM and VIKOR methods.

TABLE VII: THE DECISION MATRIX

Alternatives	Criteria				
	ACC	PREC	REC	Fl	T (sec)
DT	93.11%	93.12%	93.11%	93.11%	15
SVM	99.32%	99.33%	99.32%	99.32%	9
LR	99.13%	99.14%	99.13%	99.13%	12
NB	76.22%	75.25%	76.22%	80.02%	4
KNN	97.82%	97.81%	97.82%	97.84%	3
SC	99.57%	99.46%	99.45%	99.45%	143

4.3 BWM Results

We asked three experts to employ their ideas and years of experience in benchmarking and establish the weights for the criteria. Table 8 shows the BWM result of expert 1. We note that this expert took T as the best to others and gave it the highest importance (1), and F1 is considered the worst to the others.

TABLE VIII: WEIGHT EVALUATION FOR EXPERT 1 (BWM METHOD)

Expert 1					
Names of Criteria	ACC	PREC	REC	F1	TIME
Best Criterion	T				
Worst Criterion	F1				
Best to Others	ACC	PREC	REC	F1	TIME
T	2	2	3	5	1
Others to the Worst	F1				
ACC	4				
PREC	4				
REC	3				
F1	1				
TIME	5				
Weights	0.2307	0.2001	0.1554	0.15385	0.2599

Table 9 shows how to obtain the average weight by adding the criteria weights for each expert and dividing the result by the number of experts.

TABLE IX: BWM AVERAGE WEIGHTS

	ACC	PREC	REC	F1	Time
Expert1	0.2307	0.2001	0.1554	0.15385	0.2599
Expert2	0.2400	0.2201	0.1701	0.1389	0.2309
Expert3	0.2200	0.2147	0.1650	0.1602	0.2401
Average Weight	0.2302	0.2116	0.1635	0.1509	0.2436

4.4 VIKOR Results

The VIKOR method ranks classification algorithms for DDoS detection based on various criteria. The VIKOR results are presented in Table 10, indicating that the SVM algorithm achieved the best rank among the other algorithms. This algorithm is the most suitable for our work (DDoS attack detection) because it offers high accuracy and a short execution time compared to other algorithms. The SC was ranked fifth, although its detection accuracy was high. Increasing accuracy does not necessarily mean that the concept of high accuracy alone is not correct in such work, as this SC with high accuracy can be used in work in which the time factor is not important.

TABLE X: ALGORITHM RANKING RESULTS

Algorithm	Rank
SVM	1
LR	2
KNN	3
DT	4
SC	5
NB	6

4.5 Qualitative and Quantitative Analysis of VIKOR Results

Quantitative Analysis: VIKOR results showed that the SVM algorithm ranked first due to its high balance of accuracy, performance, and acceptable execution time, followed by Logistic Regression, which came in second due to its strong performance and ease of implementation. KNN came in third due to its good accuracy and speed, although it was not the best in terms of overall performance. Decision Tree came in fourth due to its clarity and relative speed compared to more complex algorithms. In contrast, SC came in fifth, despite its highest accuracy, due to its very long execution time, which negatively impacted its ranking. Finally, Naive Bayes came in sixth and last due to its poor overall performance, despite its speed.

Qualitative Analysis: VIKOR's ranking reflects a practical evaluation that reinforces the previous qualitative results. The SVM algorithm came in first place due to its ability to provide high accuracy with good speed, making it the most suitable for real-world applications. Logistic Regression came in second place due to its ease of interpretation and efficiency, while KNN was a suitable choice for real-time systems that require rapid response. Decision Tree, despite its simplicity, came in the middle due to its modest performance, but it remains suitable for environments that require transparency. SC lagged behind in the ranking despite its superior accuracy, indicating that execution time significantly impacts its practical usability. Naive Bayes remained last because it does not achieve an acceptable balance between speed and accuracy, limiting its use in certain scenarios.

4.6 Interpreting Algorithmic Performance

Data interpretation reveals that while accuracy is often the primary metric in model evaluation, relying on it alone can be misleading in real-world distributed denial-of-service (DDoS) detection scenarios where response time is critical. The study results were interpreted by combining quantitative and qualitative analysis using the BWM and VIKOR methods, providing a deeper understanding of the performance of DDoS detection algorithms from multiple perspectives. The results showed that the SVM algorithm offered the best balance between high accuracy (99.32%) and low execution time (9 seconds), making it an ideal choice for systems that require a fast and reliable response, such as e-commerce platforms or security-sensitive websites. In contrast, the compact classifier (SC) achieved the highest accuracy (99.57%) but suffered from a long execution time (143 seconds), making it only suitable for non-time-sensitive environments. When analyzing the algorithm rankings using VIKOR, SVM was found to be at the top of all scenarios, enhancing the reliability of its results under varying weights and criteria. Both sensitive analysis and objective validation confirmed these results, with SVM maintaining the lead while NB ranked last in all cases, demonstrating its poor overall performance despite its speed. This multidimensional evaluation reflects the practical balance between efficiency, accuracy, and speed, and enhances the credibility of selecting the most appropriate algorithm based on the requirements of the target system.

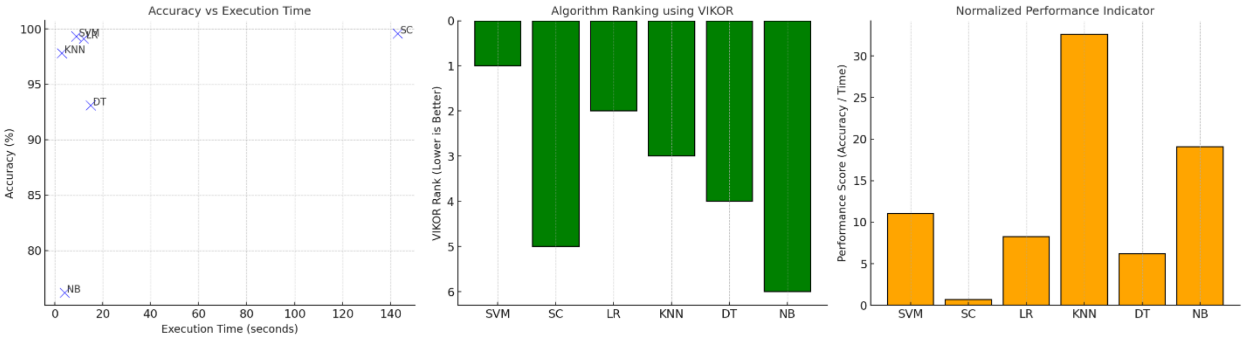

Figure 4: Interpretation of the algorithmic performance

The analytical Figure 4 illustrates three key aspects for interpreting algorithmic performance:

Graph 1 (Accuracy vs. Execution Time): It shows that the SVM algorithm achieves an ideal balance between high accuracy (99.32%) and short execution time (9 seconds). In contrast, although the SC algorithm achieves the highest accuracy (99.57%), its execution time is very long (143 seconds), limiting its use in time-sensitive systems.

Graph 2 (VIKOR Ranking): SVM ranks first in the evaluation using the VIKOR method, strengthening its credibility as the best choice. NB ranks last despite its faster execution, due to its poor overall performance.

Graph 3 (Combined Performance Index: Accuracy divided by Time): This graph shows that SVM has the highest relative performance value (accuracy/time), highlighting its effective balance. In contrast, SC, despite its high accuracy, lags behind SVM due to its slowness.

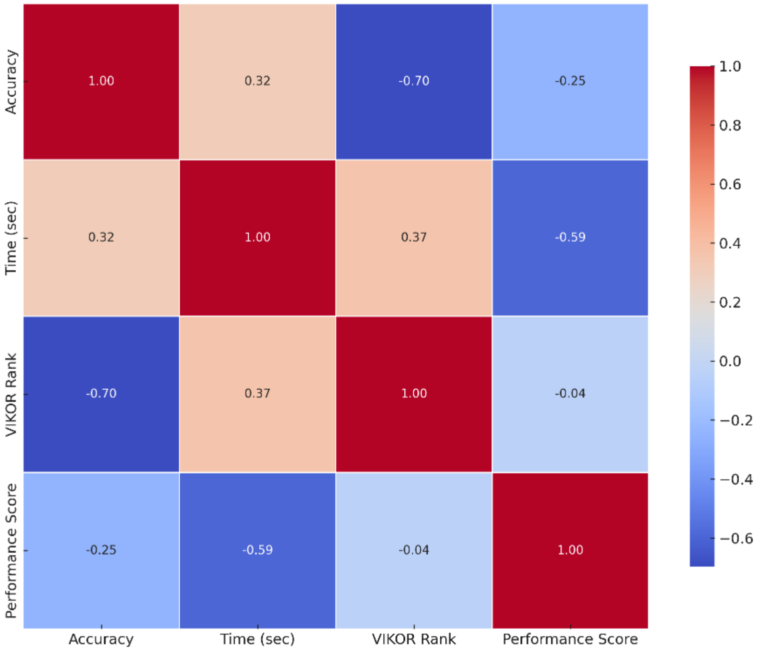

Figure 5: The correlation matrix between accuracy, execution time, VIKOR, and performance index

Figure 5 shows the relationship between the basic variables of this work or the correlation matrix between the key variables used in evaluating the performance of DDoS detection algorithms: accuracy, execution time, VIKOR ranking, and performance index (accuracy divided by time). The matrix reveals a strong inverse relationship between performance index and VIKOR ranking, meaning that as performance increases (i.e., higher accuracy and lower time), the VIKOR ranking decreases, which is a positive indicator. A negative relationship also exists between execution time and performance index, indicating that increasing time negatively impacts the algorithm's efficiency. Furthermore, there is a clear direct relationship between execution time and VIKOR ranking, indicating that slower algorithms often receive lower rankings in the evaluation, reinforcing the importance of time when selecting the most appropriate algorithm for time-sensitive systems.

4.7 Validation

Below, we describe two ways to confirm the validity of the obtained results.

4.7.1 Objective Validation

This step validates our work. We divided the classifiers into two groups: Group 1 (DT, SVM, and LR) and Group 2 (NB, KNN, and SC). We calculated the average and variance of each group. We verified that the average and variance of Group 1 were less than those of Group 2, which gives validity to the obtained results. This result is shown in Table 11.

TABLE XI: OBJECTIVE VALIDATION RESULTS

Algorithms		Average	Variance
Group 1	DT, SVM, LR	0.018222	0.008112
Group 2	SC, KNN, NB	0.070314	0.037497

4.7.2 Sensitivity Analysis

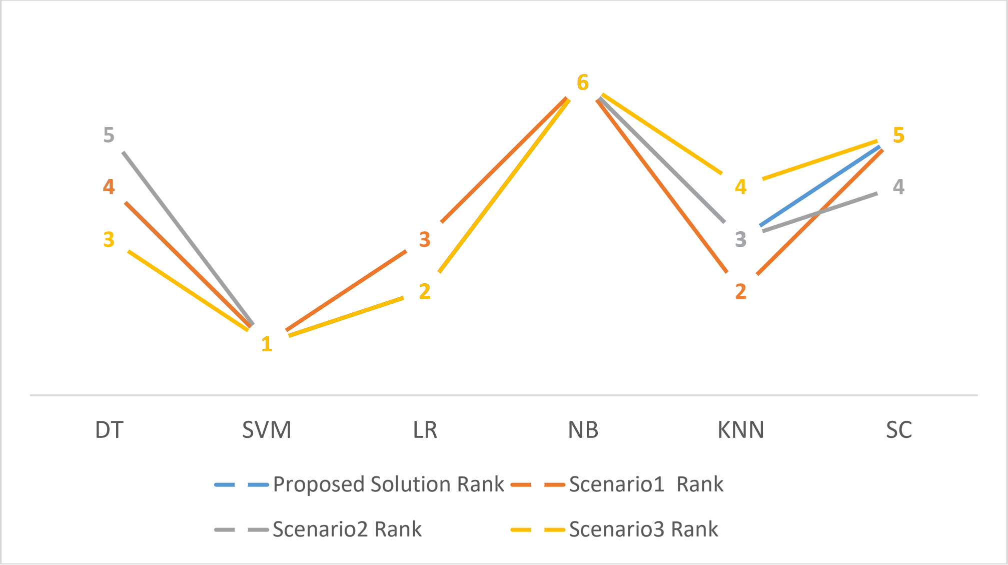

Figure 6. Sensitivity Analysis

The rank for each scenario and the rank of the proposed solution were calculated, and all of them gave rank 1 to the SVM classifier, as in Fig.6. This indicates that it is the most suitable for repelling this type of attack with high accuracy and record time. Also, everyone again agreed to give the NB classifier a rank of 6, which is the lowest, indicating that it is unsuitable for this work.

BWM and VIKOR have effectively selected the most suitable classifiers based on multiple criteria, making them the most appropriate options in this field. Stacked classifiers have the highest accuracy due to their ability to integrate or utilize various models from each classifier. The SVM classifier is more efficient and superior to the rest due to its short execution time and high accuracy. It is beneficial for high-priority websites, such as those of significant e-commerce and financial companies, or pages related to state security. For other pages where time is not a critical factor, we developed the stacked classifier, which can achieve high accuracy at the expense of time.

Conclusion

Much research has been conducted in detecting DDoS attacks; however, selecting the appropriate classifier for the current work is not straightforward, particularly when multiple criteria are involved. It cannot be said that this algorithm has high accuracy so that it will be chosen, and the current work requires fast results. Therefore, the contribution of this research was to enhance, benchmark, and evaluate (choose the best among a group of classifiers). We achieved this by utilizing multiple classifiers that identify a distributed denial-of-service attack based on various criteria, including accuracy, F1 score, precision, recall, and time. The results obtained from the BWM and VIKOR methods demonstrated that the SVM algorithm was superior and more suitable for the current work because it has an outstanding balance between its criteria.

Objective validation divided the classifiers into two groups. We calculated the average and variance for each group and found that the SVM classifier, with the lowest variance and average, outperformed the others. Sensitivity Analysis also yielded the lowest rank for the SVM classifier in all ranking scenarios, indicating that it is superior to the rest. This ranking represents the benchmarking aspect of this work, which aims to select the most suitable classifier for the current study.

Another notable achievement of this work was the development of a stacked classifier, which has demonstrated that combining these algorithms (DT, SVM, LR, and KNN) yields the best results with high accuracy, achieving a rate of 99.57%. Therefore, this represents the enhancement of accuracy in this work.

Acknowledgment: To anyone who supported this work.

Funding Statement: The author(s) received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Alaa Mahmood; data collection: Alaa Mahmood; analysis and interpretation of results: Alaa Mahmood, İsa Avcı; draft manuscript preparation: Sahar Yousif Mohammed. All authors reviewed the results and approved the final version of the manuscript.

Conflict of interest: The researchers of this paper declare that there are no conflicts of interest.

References

[1] S. Wani, M. Imthiyas, H. Almohamedh, K. M. Alhamed, S. Almotairi, and Y. Gulzar, “Distributed denial of service (DDoS) mitigation using blockchain—A comprehensive insight,” Symmetry, vol. 13, no. 2, p. 227, 2021.

[2] M. Roopak, G. Y. Tian, and J. Chambers, “Multi-objective-based feature selection for DDoS attack detection in IoT networks,” IET Networks, vol. 9, no. 3, pp. 120–127, 2020.

[3] İ. Özçelik and R. Brooks, Distributed Denial of Service Attacks: Real-world Detection and Mitigation. CRC Press, 2020.

[4] C. Fachkha, E. Bou-Harb, and M. Debbabi, “Inferring distributed reflection denial of service attacks from the darknet,” Computer Communications, vol. 62, pp. 59–71, 2015.

[5] M. Karami, Y. Park, and D. McCoy, “Stress testing the booters: Understanding and undermining the business of DDoS services,” in Proc. 25th Int. Conf. World Wide Web, 2016, pp. 1033–1043.

[6] J. Scott Sr and W. Summit, Rise of the Machines: The Dyn Attack Was Just a Practice Run, Inst. for Critical Infrastructure Technology, Washington, DC, USA, Dec. 2016.

[7] A. R. A. Yusof, N. I. Udzir, and A. Selamat, “Systematic literature review and taxonomy for DDoS attack detection and prediction,” Int. J. Digit. Enterprise Technol., vol. 1, no. 3, pp. 292–315, 2019.

[8] Z. Gavric and D. Simic, “Overview of DOS attacks on wireless sensor networks and experimental results for simulation of interference attacks,” Ingeniería e Investigación, vol. 38, no. 1, pp. 130–138, 2018.

[9] C. S. Kalutharage, X. Liu, C. Chrysoulas, N. Pitropakis, and P. Papadopoulos, “Explainable AI-based DDOS attack identification method for IoT networks,” Computers, vol. 12, no. 2, p. 32, 2023.

[10] S. Wicaksono, “Prioritizing tasks in information system projects: A novel approach using VIKOR method,” TEKNOKOM, vol. 6, no. 2, 2023. [Online]. Available: https://doi.org/10.31943/teknokom.v6i2.149.

[11] S. Y. Mohammed, M. Aljanabi, M. M. Mijwil, A. J. Ramadhan, M. Abotaleb, H. Alkattan, and Z. Albadran, “A Two-Stage Hybrid Approach for Phishing Attack Detection Using URL and Content Analysis in IoT,” in *BIO Web of Conferences*, vol. 97, art. no. 00059, 2024. [Online]. Available: https://doi.org/10.1051/bioconf/20249700059

[12] A. Akinwale, E. Olajubu, and A. Aderounmu, “A regeneration model for mitigation against attacks on HTTP servers for mobile wireless networks,” Int. J. Electr. Comput. Eng. Syst., vol. 15, no. 5, pp. 395–406, 2024.

[13] N. A. Dogra and N. Taqdir, “Enhancing DDoS attack detection and network resilience through ensemble-based packet processing and bandwidth optimization,” Deleted Journal, vol. 2, no. 4, pp. 930–937, 2024. [Online]. Available: https://doi.org/10.47392/irjaeh.2024.0130

[14] A. Tedyyana, O. Ghazali, and O. W. Purbo, “Machine learning for network defense: Automated DDoS detection with telegram notification integration,” Indones. J. Electr. Eng. Comput. Sci., vol. 34, no. 2, p. 1102, 2024.

[15] A. Bindu, A. V. S. Harika, D. Swetha, and M. Sahithi, “SDN network DDOS detection using ML,” Int. J. Innov. Sci. Res. Technol., pp. 811–817, 2024.

[16] F. Laiq, F. Al-Obeidat, A. Amin, and F. Moreira, "DDoS attack detection in Edge-IIoT network using ensemble learning," J. Phys.: Complexity, 2024.

[17] W. I. Khedr, A. E. Gouda, and E. R. Mohamed, "FMDADM: A multi-layer DDoS attack detection and mitigation framework using machine learning for stateful SDN-based IoT networks," IEEE Access, vol. 11, pp. 28934–28954, 2023.

[18] R. Singh, S. Tanwar, and T. P. Sharma, "Utilization of blockchain for mitigating the distributed denial of service attacks," Security Privacy, vol. 3, no. 3, p. e96, 2020.

[19] Cybersecurity Defence Mechanism Against DDoS Attack with Explainability (A. M. Mahmood & İsa Avcı , Trans.). (2024). Mesopotamian Journal of CyberSecurity, 4(3), 278-290. https://doi.org/10.58496/MJCS/2024/027.

[20] Development of real-time threat detection systems with AI-driven cybersecurity in critical infrastructure (N. Z. . Khalaf, I. I. . Al Barazanchi, I. I. . Al Barazanchi, A. D. . Radhi, S. . Parihar, P. . Shah, & R. . Sekhar , Trans.). (2025). Mesopotamian Journal of CyberSecurity, 5(2), 501-513. https://doi.org/10.58496/MJCS/2025/031.

[21] H. Wang and Y. Li, "Overview of DDoS attack detection in software-defined networks," IEEE Access, vol. 12, pp. 38351–38381, 2024.

[22] J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," ACM SIGCOMM Comput. Commun. Rev., vol. 34, no. 2, pp. 39–53, 2004.