INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

A Hybrid Feature Selection Approach Incorporating Mutual Information and Genetics Algorithm for Web Server Attack Detection

Sainath Patil[1]*, **Rajesh Bansode**[2]

1 Research Scholar, Information Technology, Thakur College of Engineering and Technology, Mumbai, Maharashtra, India
2 Professor, Information Technology, Thakur College of Engineering and Technology, Mumbai, Maharashtra, India

OPEN ACCESS

Received: 07-11-2023

Accepted: 28-12-2023

Published: 20-01-2024

Citation: Patil S, Bansode R (2024) A Hybrid Feature Selection Approach Incorporating Mutual Information and Genetics Algorithm for Web Server Attack Detection. Indian Journal of Science and Technology 17(4): 325-332. https://doi.org/10.17485/IJST/v17i4.2820

* **Corresponding author**.

psai17@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: To improve accuracy and reduce the computational overheads of Machine Learning (ML) classifiers to identify web server threats, develop a feature selection strategy that extracts pertinent and important features from the network dataset. **Methods**: This research work progressed in three phases i) Mutual Information (MI) was used first for the feature ranking and selection to reduce the dimension of feature space; ii) Genetic Algorithms (GA) were used to pick significant features for boosting the accuracy of ML classifiers; and iii) evaluates the performance of four ML classifiers; Naive Bayes (NB), k-Nearest Neighbor (k-NN), Random Forest (RF), and Support Vector Machine (SVM), using the selected features. The evaluation is conducted on the UNSW-NB 15 dataset, measuring accuracy, False Positive Rate (FPR), and computational time. **Findings**: The results indicate that the proposed feature selection method remarkably improves the accuracy of ML classifiers, reducing the number of features to just four. The accuracy of ML classifiers improved by 13.11%, resulting in a reduction of about 99% in computational time compared to the results reported in the literature. **Novelty**: A novel hybrid feature selection method is proposed, which combines feature reduction by MI, a filter-based method and further feature extraction by GA, a wrapper-based method. This approach effectively identifies essential features for enhancing the accuracy of ML classifiers.

Keywords: Genetic Algorithm; Feature Selections; Machine Learning; Mutual Information; Webserver Security

1 Introduction

Web applications have become indispensable tools for numerous organizations, offering convenient access to a wide range of services and information. However, these web applications are not immune to vulnerabilities that can be exploited by malicious users. These vulnerabilities encompass common issues such as injection flaws, unprotected

authentication, web server/ application misconfiguration, etc. The motives behind the attack on the web server are business contretemps, extortion or any sort of gain and sometimes in excitement. There need arises for techniques which will detect and prevent such attacks on web applications and make them secure. A multi-layered approach that includes securing both web applications and the underlying web server, along with proactive monitoring and user education, is crucial to maintaining a strong cybersecurity posture. Numerous studies have been conducted to detect malicious/intrusive attacks on web applications or web servers. Extensive research has been conducted to identify and thwart malicious or intrusive attacks targeting web applications and web servers. Tekerek [1] introduced an innovative web attack detection architecture that leverages deep learning models to discern anomalies within web traffic. This system relies on web request log file data to identify irregularities in web requests, employing deep learning algorithms for this purpose. In various research studies, analyzed attacks on web server and ML algorithms like k-NN, Decision Tree (DT), SVM, etc. have been applied to construct classification models aimed at detecting and mitigating these attacks [2,3].

The efficacy of attack detection models hinges on two key factors: the careful selection of relevant features from attack datasets and the performance of the classification algorithms employed. With the rapid advancements in cybersecurity technologies, there has been a notable surge in the size of databases in recent years. This surge, coupled with the inclusion of attributes that are either irrelevant or redundant, has rendered the process of feature selection challenging in terms of both efficiency and effectiveness. Interestingly, many research efforts focused on web application attack detection models have often overlooked the incorporation of feature selection techniques.

Dong and Sarem presented an improved k-NN attack detection method wherein the authors used only four features of the network traffic and these features were selected manually by study and observations [4]. Ahuja et al. analyzed network traffic and identified prospective features that could discover attack characteristics. They extracted sixteen different flow and packet-based features from the network traffic [5]. Su. J. et al. analyzed ML models for prediction of attacks on Internet of Things (IoT) devices. The importance of features was measured though their correlation and precisely selected 10 features from feature importance ranking [6].

Azmi et al. used ANN, Naïve Bayes (NB), and DT algorithms to detect DDoS attacks. 10 features are selected out of 45 from the UNSW-NB 15 dataset using the filtered-based feature selection method, Information Gain (IG). The features were identified as important and less important based on the IG. The highest accuracy observed by DT was 88.43%, followed by NB 87.74%, and Artificial Neural Network (ANN) 84.66% [7].

Kshirsagar and Kumar [8] put forward an average weight-based feature extraction technique to pick important network attributes for DoS attack detection. The technique employed filtered-based feature selection techniques like Correlation (CR), Information Gain Ratio (IGR), and ReliefF (ReF). The average weight of each feature was calculated based on statistical measures for each algorithm. Subsets for each algorithm were formed by selecting features whose weights were greater than thresholds. By recognizing the occurrence of a feature in all three subsets from each algorithm, a new subset was created. CICIDS 2017 and KDD Cup 99 datasets were used for experiment purposes and achieved accuracy up to 99% and an average computational time of about 50 seconds. Filtered-based methods select features based on their statistical metrics, not on the usefulness of the features for ML models.

Alanazi H. et al. present study on feature selection for detection of network probing attack. In this study authors introduced novel threefold feature selection module, wherein first they removed highly correlated features. Secondly mutual information with feature importance ranking for detecting network probing attack prepare feature set of relevant features. Lastly applied fine-grained refinement done by Long Short-Term Memory (LSTM) to improve detection accuracy. KDD CUP 99 dataset used for experimental analysis and selected 13 important features out of 41 features on which they applied RF, NB, XGBoost and Ada Boost models and observed the accuracy of about 96 – 99% with significantly less time [9]. The accuracy of the NB method was approximately 96.48%, which might be improved further.

As a result, when constructing an ML model, it is critical to understand the significance of feature selection in order to improve the model's overall accuracy and efficiency. The study presents a novel hybrid feature selection method to select appropriate and relevant features in web application and web server attack detection. The proposed study coalesces filter-based and wrapper-based feature selection techniques together. This approach effectively identifies essential features for enhancing the accuracy of machine learning classifiers. GA wrapper-based is a common and appealing feature selection technique used in various research dimensions to extract relevant features like text mining, disease diagnosis in medical research, fault diagnosis in the automation industry and cyber security applications to detect attacks [10]. GA wrapper-based technique has a high computational cost. To reduce computational cost, this study employs a filtered-based technique to pull out a subset of features from an uncondensed one before applying GA.

Three types of feature selection techniques are filter-based, wrapper-based and embedded. Filter-based feature selection techniques select features by filtering out irrelevant ones using specific metric of data attributes. Wrapper-based techniques

use specific ML models for searching the space of all potential feature subsets by learning and evaluating with that feature subset. Embedded techniques combine the benefits of both the filter and wrapper methods by adding feature relationship while maintaining appropriate computing costs.

These studies embolden us to coalesce filtered-based and wrapper-based techniques to select relevant features in web server attack detection with improvements in accuracy and reductions in computational cost.

2 Methodology

The study proposed a novel approach for feature selection to extract essential features from the network traffic in order to detect attacks on a web server. The proposed hybrid technique is a combination of mutual information metric, a filter-based approach and genetic algorithm, a wrapper-based approach. Figure 1 shows the process of the proposed methodology. In the pre-processing and cleaning phase, remove null values and any other special characters if present in the dataset. Then it converts strings or labels to numerical values using LableEncoder.

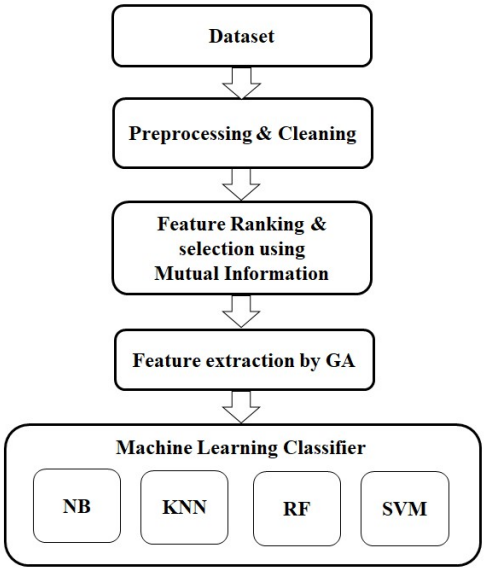

Fig 1. Proposed Methodology for Feature Selection

Mutual information is computed for each of the attributes in the network dataset. MI is the statistical metric representing the dependency relationship between each feature and target variable[11]. Features with an MI value above the threshold are selected for the next step. The impetus of this is to reduce the dimension of feature space. Reduced feature space helps to lessen the computational cost of GA and speeds up processing. GA was applied to the feature subset obtained from the feature ranking and selection using mutual information step for further reduction. Resultant feature subsets have only relevant and important features useful for attack detection using ML models.

This study evaluated the performance metrics of NB, SVM, k-NN and RF ML algorithms using extracted features from GA. Algorithm 1, represents the process of proposed novel approach for feature selection.

Algorithm 1:

Input: FS number of features in the Feature set

Output: FS" number of features in the extracted Feature set

1. **procedure** FEATURE_SELECTION (FS)
2. Estimate mutual information (MI) between independent variable (S) and target variable (T);
   $MI(S,T) = H(T) - H(T|S)$
3. Select the features with MI value greater than threshold (average)
   $FS' = MI(S, T) >= AVG [MI(S, T)]$
4. Apply genetic algorithm on selected features FS'
   $FS'' = GA(FS')$
5. FS" are relevant and important features for attack detection on which ML model applied.

6. **end procedure**

2.1 Mutual Information

Information gain calculates the reduction in entropy resulting from a particular data transformation. MI is a measure of the connectedness between independent (feature) and dependent (target) attributes. When used for feature selection, information gain is referred to as mutual information [12]. It identifies the dependency statistics of the datasets. If there is no relation between the feature set and the target variable, then the MI is zero. A higher value of MI signifies a higher dependency of the target variable on the respective feature in the dataset. MI between feature (S) and target variable (T) is represented by $MI(S,\ T\ )$is given by Equation (1).

$$MI(S,T\ ) = H(T\ ) - H(T|S) \tag{1}$$

where $H(S)$ is entropy of feature S as in Equation (2) and H(T |S) entropy of target variable T given feature S given by Equation (3).

$$H(S) = -\sum P_S(S)\, log(S) \tag{2}$$

$$H(T|S) = -\sum\sum P_{T|S}(T|S)\, log(P_{T|S}(T|S)) \tag{3}$$

2.2 Genetic Algorithm

Genetic algorithms are a sort of optimization and search algorithm that is inspired by the ideas of natural selection and genetics. They are frequently used to find approximate solutions to difficult optimization and search issues. Genetic algorithms are a subset of the larger class of algorithms known as evolutionary algorithms. Genetic algorithm works as follows.

1. Initialization: A collection of probable solutions to the optimization problem is generated randomly. Individual solution is referred to as chromosome and represented as a set of features or genes (Figure 2). A stream of binary digits is used to represent chromosomes. Each bit in the chromosome is a feature and 1/0 signifies its presence or absence in the set, i.e., if i^{th} bit 1 in the chromosome means respective feature is selected otherwise 0.

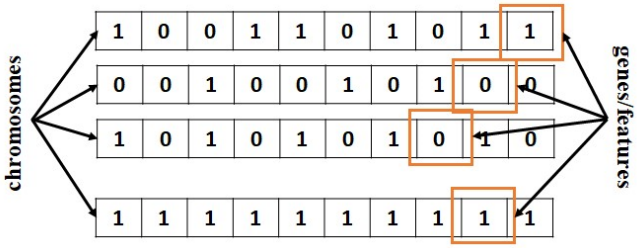

Fig 2. Chromosomes Population from Feature Set

2. Fitness Evaluation: The fitness of each solution in the chromosomes is evaluated by a fitness function. The fitness function quantifies how well each solution solves the problem. Solutions that perform better receive higher fitness scores.

3. Selection: A subset of solutions is selected to serve as parents for the next generation. Solutions with higher fitness scores are more likely to be selected, but some degree of randomness is often introduced to maintain diversity in the population.

4. Crossover: Pairs of selected solutions are combined to create new solutions, known as offspring. This process mimics genetic recombination in biology. Crossover points are chosen in the solution representations, and genes are swapped between parents to produce offspring (Figure 3).

5. Mutation: Random changes are introduced into some of the offspring's genes. This introduces diversity into the population and prevents the algorithm from getting stuck in local optima (Figure 4)

6. Replacement: The new offspring, along with some of the best-performing solutions from the previous iteration, form the chromosomes for the next iteration. The worst-performing solutions may be discarded.

7. Termination: The algorithm iterates through generations until a termination condition is reached. A high fitness score fixed computational costing or certain number of generations are common termination conditional parameters.

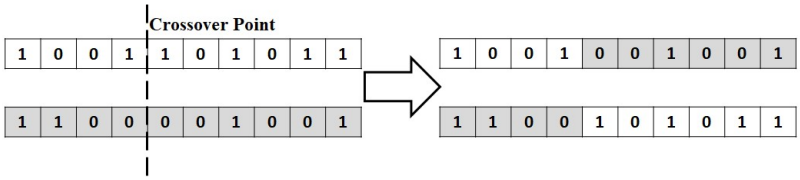

Fig 3. Crossover in Chromosomes to Check Feature Importance

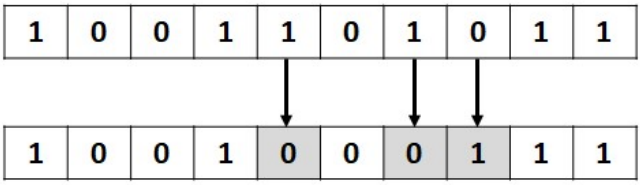

Fig 4. Mutation in Chromosomes for Feature Selection

3 Results and Discussion

3.1 Experimental Setup

This research work was carried out using python with pandas and sklearn libraries on Microsoft Windows 64-bit platform within VMware Virtual Machine. The virtual machine is configured with 2 processors and 32GB memory deployed on powerful hardware of Dell Poweregde T440 having Intel Xeon 4210 CPU @ 2.2GHz, 10C/20T, 64 GB RAM.

3.1.1 Dataset

UNSW-NB 15 dataset [13] developed in the UNSW security lab using IXIA PerfectStorm tool. The network traffic packets are represented by 49 features. The traffic packets are labelled as normal and abnormal (attack), represented by two dependent (target) variables.

3.2 Results

This research work progressed in three phases: i) Initial feature ranking and selection using MI ii) feature extraction using GA and iii) performance evaluation using machine learning algorithms.

3.2.1 Feature Ranking and selection using Mutual Information

In the UNSW-NB15 dataset, 47 are independent features and 2 are target features listed in [13]. The mutual information of each feature was calculated and ranked as per their MI metric. The feature selected for the next stage has MI value greater than the threshold. Here, the average of MI values was considered as threshold. Table 1 represents the 23 features selected after MI ranking and selection.

Table 1. Features Selected after MI Ranking

Sr. No.	Feature Name	Sr. no.	Feature Name
1	*srcip*	13	*Swin*
2	*dstip*	14	*Dwin*
3	*proto*	15	*smeansz*
4	*state*	16	*dmeansz*
5	*dur*	17	*dintpkt*
6	*sbytes*	18	*tcprtt*
7	*dbytes*	19	*synack*
8	*sttl*	20	*ackdat*
9	*dttl*	21	*ct_state_ttl*

Continued on next page

Table 1 continued

10	*sload*	22	*ct_src_dport_ltm*
11	*dload*	23	*ct_dst_dport_ltm*
12	*dpkts*		

3.2.2 Feature extraction by GA

The suitable parameter set of GA is usually chosen by conducting a number of trials with different combinations, and the combination that delivers a good result is selected. We set GA parameters as: initial population size of 100; number of generations to run the evolutionary algorithm of 50; parent selection by tournament process with tournament size of 3; and scoring strategy to evaluate performance, which is considered accuracy. The feature set obtained from MI ranking and selection passed through this phase. Four important features are extracted after this step (Table 2).

Table 2. Features Extracted by GA out of Selected after MI Ranking

Sr. No.	Feature Name
1	*srcip*
2	*Proto*
3	*dpts*
4	*ct_state_ttl*

3.2.3 Machine Learning Classifier

Using features extracted from GA in the previous phase, the performance metrics of NB, SVM, k-NN, and RF ML algorithms are evaluated for UNSW-NB15 dataset. Table 3 shows the comparison of metrics accuracy, FPR and computation time of ML algorithms with 23, 5 and 4 features selected by only MI, only GA and the proposed approach, respectively.

Table 3. Metrics of Classifiers Evaluated for UNSW-NB dataset

	MI (# features =23)			GA (# features = 5)			Proposed (#features = 4)		
	Accuracy in %	FPR in %	Time in Sec	Accuracy in %	FPR in %	Time in Sec	Accuracy in %	FPR in %	Time in Sec
NB	96.34	1.49	20	98.12	1.23	4.82	99.25	07.70	0.391
KNN	96.57	1.43	126	98.33	0.94	80	99.37	0.41	48
RF	97.24	1.14	78	98.39	0.91	52	99.46	0.52	15
SVM	98.52	1.09	2280	98.44	0.89	1844	99.42	0.56	1560

Figure 5 shows the comparison of accuracy evaluated by using features extracted by MI ranking only, GA only, and the proposed approach. It clearly reveals that the proposed approach to feature selection improves the output of ML classifiers for attack detection.

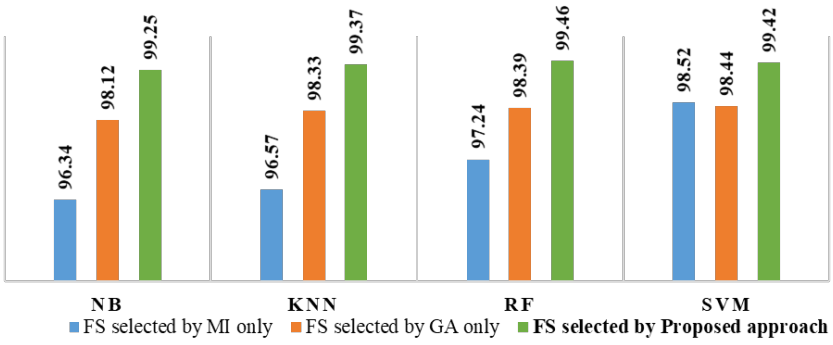

Fig 5. Accuracy of ML Algorithms using Feature Sets Selected by MI, GA and Proposed approach

Table 4 shows the comparative analysis of the results obtained using the proposed feature selection method with the results reported in the literature. Dhanya et al.[2] used all features of the UNSW NB15 dataset; hence, computational time is longer. The study presented by Azmi et al.[7] selected 10 features from the UNSW-NB15 dataset using the filter-based feature extraction method. The comparatively proposed method helps in accuracy improvement of about 13.11% within fractions of a second. Similarly, the comparative accuracy improvement of the proposed feature selection method with respect to Kshirsagar and Kumar[8] and Alanazi et al.[9] is about 0.40% and 3%, with a reduction in computational time of about 99% and 32.5%, respectively. Figure 6 depicts a comparative analysis of the improvement in number of features, accuracy, and time complexity of the proposed feature selection method with the state-of-the-art algorithms.

Table 4. Comparative Analysis of Proposed Results with The Results Reported

	# of features	Accuracy in %	Time in Sec
Dhanya et al.[2]	47	98.96	253.48
Azmi et al.[7]	10	87.74	72
Kshirsagar and Kumar[8]	12	98.86	50.42
Alanazi et al.[9]	13	96.36	0.58
Proposed	**4**	**99.25**	**0.391**

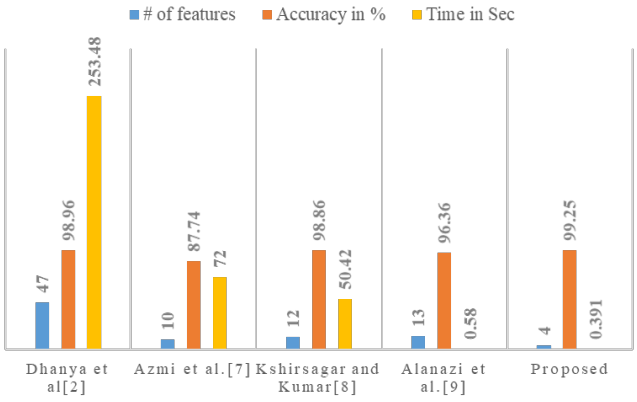

Fig 6. Results of Proposed Method and Comparison with Existing Studies

4 Conclusion

Selecting the right set of features can significantly impact the model's performance. A genetic algorithm is applied to a feature set selected by MI ranking to reduce the dimensionality further and derive a subset of the most pertinent features. This hybrid approach, which is a combination of MI and GA, incorporates the advantages of both filter-based and wrapper-based feature selection techniques. The study evaluates the performance of ML classifiers (NB, KNN, SVM and RF) using the selected features. The UNSW-NB15 dataset was used for evaluation. The evaluation includes accuracy, false positive rate, and computational time.

The proposed novel hybrid feature selection method outperforms other approaches, significantly improving the accuracy of ML classifiers. It reduces the number of features to only four as compared to features selected in existing studies, thus reducing computational cost while enhancing attack detection accuracy. The accuracy of machine learning algorithms is improved by 0.40% - 13.11%, and by about a 99% reduction in computational time compared to state-of-the-art feature selection algorithms. The UNSW-NB15 dataset used is an imbalanced one; hence, the performance could be evaluated with a balanced dataset. And the future work will be to analyze the proposed feature selection method with real-time network datasets.

References

1) Tekerek A. A novel architecture for web-based attack detection using convolutional neural network. *Computers & Security*. 2021;100:102096. Available from: https://doi.org/10.1016/j.cose.2020.102096.

2) Dhanya KA, Vajipayajula S, Srinivasan K, Tibrewal A, Kumar TS, Kumar TG. Detection of Network Attacks using Machine Learning and Deep Learning Models. *Procedia Computer Science*. 2023;218:57–66. Available from: https://doi.org/10.1016/j.procs.2022.12.401.
3) Patil S, Vanmali AV, Bansode R. Cyber Security Concerns for IoB. In: Internet of Behaviors (IoB). CRC Press. 2023;p. 141–155. Available from: https://www.taylorfrancis.com/chapters/edit/10.1201/9781003305170-9/cyber-security-concerns-iob-sainath-patil-ashish-vanmali-rajesh-bansode.
4) Dong S, Sarem M. DDoS Attack Detection Method Based on Improved KNN With the Degree of DDoS Attack in Software-Defined Networks. *IEEE Access*. 2019;8:5039–5048. Available from: https://doi.org/10.1109/ACCESS.2019.2963077.
5) Ahuja N, Singal G, Mukhopadhyay D, Kumar N. Automated DDOS attack detection in software defined networking. *Journal of Network and Computer Applications*. 2021;187:103108. Available from: https://doi.org/10.1016/j.jnca.2021.103108.
6) Su J, He S, Wu Y. Features selection and prediction for IoT attacks. *High-Confidence Computing*. 2022;2(2):1–6. Available from: https://doi.org/10.1016/j.hcc.2021.100047.
7) Azmi MAH, Foozy CFM, Sukri KAM, Abdullah NA, Hamid IRA, Amnur H. Feature Selection Approach to Detect DDoS Attack Using Machine Learning Algorithms. *JOIV : International Journal on Informatics Visualization*. 2021;5(4):395–401. Available from: http://dx.doi.org/10.30630/joiv.5.4.734.
8) Kshirsagar D, Kumar S. An efficient feature reduction method for the detection of DoS attack. *ICT Express*. 2021;7(3):371–375. Available from: https://doi.org/10.1016/j.icte.2020.12.006.
9) Alanazi HO, Bi S, Wang T, Hou T. Exquisite Feature Selection for Machine Learning Powered Probing Attack Detection. In: ICC 2023 - IEEE International Conference on Communications. IEEE. 2023;p. 783–789. Available from: https://doi.org/10.1109/ICC45041.2023.10278886.
10) Chanu US, Singh KJ, Chanu YJ. A dynamic feature selection technique to detect DDoS attack. *Journal of Information Security and Applications*. 2023;74:103445. Available from: https://doi.org/10.1016/j.jisa.2023.103445.
11) Alalhareth M, Hong SC. An Improved Mutual Information Feature Selection Technique for Intrusion Detection Systems in the Internet of Medical Things. *Sensors*. 2023;23(10):1–21. Available from: https://doi.org/10.3390/s23104971.
12) Qu K, Xu J, Hou Q, Qu K, Sun Y. Feature selection using Information Gain and decision information in neighborhood decision system. *Applied Soft Computing*. 2023;136:110100. Available from: https://doi.org/10.1016/j.asoc.2023.110100.
13) Moustafa N, Slay J. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS). IEEE. 2015. Available from: https://doi.org/10.1109/MilCIS.2015.7348942.