A significant features vector for internet traffic classification based on multi-features selection techniques and ranker, voting filters

Alhamza Munther[1], Mosleh M. Abualhaj[2], Alabass Alalousi[3], Hilal A. Fadhil[4]
[1]Information Technology Department, College of Computing and Information Sciences, University Technology and Applied Sciences, Sur, Oman
[2]Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan
[3]Information Technology Department, Faculty of Informatics and Computers, Thamar University, Thamar, Yemen
[4]Electrical and Computer Engineering Program, Engineering Faculty, Sohar University, Sohar, Oman

Article Info

Article history:

Received Oct 30, 2023
Revised Jul 16, 2024
Accepted Aug 6, 2024

Keywords:

Data mining
Data reduction
Features selection
Filter feature selection
Machine learning
Supervised learning

ABSTRACT

The pursuit of effective models with high detection accuracy has sparked great interest in anomaly detection of internet traffic. The issue still lies in creating a trustworthy and effective anomaly detection system that can handle massive data volumes and patterns that change in real-time. The detection techniques used, especially the feature selection methods and machine learning algorithms, are crucial to the design of such a system. The fundamental difficulty in feature selection is selecting a smaller subset of features that are more related to the class but are less numerous. To reduce the dimensionality of the dataset, this research offered a multi-feature selection technique (MFST) using four filter techniques: fast correlation-based filter, significance feature evaluator, chi-square, and gain ratio. Each technique's output vector is put via ranker and Borda voting filters. The feature with the highest number of votes and rank values will be selected from the dataset. The performance of the given MFST framework was the best when compared to the four strategies listed above functioning alone; three different classifiers were employed to test the accuracy. C4.5, nave Bayes, and support vector machine. The experiment outcomes employed ten datasets of different sizes with 10,000-300,000 instances. Only 8 out of 248 characteristics were chosen, with classifiers percentages averaging 65%, 93.8%, and 95.5%.

Corresponding Author:

Alhamza Munther
IT Department, College of Computing and Information Sciences, University Technology and Applied Sciences
411 Alhusainya Street, Sur, Oman
Email: alhamza.wardi@utas.edu.om

1. INTRODUCTION

A common technique for preparing data in machine learning is feature selection or variable selection. It is a dimensionality reduction technique primarily used to remove unnecessary and undesired features from any dataset [1]. To diagnose anomalies in Internet traffic, the feature selection process removes redundant and pointless attributes from the dataset. In network security, it can pinpoint the most significant factors connected to a certain attack. Anomaly detections greatly benefit the network security sector as decision-making has become easier and faster with the rapid accumulation of high-throughput technology and cutting-edge machine learning methodologies. Machine learning is essential in helping network operators analyze the risk variables associated with a specific assault quickly [2]. Additionally, these solutions support security administrators in making wise decisions. Researchers are being pushed today to use feature reduction and classification algorithms for efficient attack diagnosis. Machine learning is essential in helping network operators analyze the risk features associated with a specific assault quickly. Additionally, these solutions support security analysts in making wise decisions. Researchers are being pushed today to use feature reduction and classification algorithms for efficient attack diagnosis. To effectively classify Internet traffic, classification is a data mining task. From training data, it first creates a model. Next, it is applied to anticipate new instances for which the class values are unknown. Before classifying, unnecessary and redundant attributes can be removed from the dataset using the proper feature selection technique. As a result, classifier accuracy is improved because it just needs to examine the dataset's key features. The classifier model performs better when properly chosen features, decreasing computational complexity and execution time. It also simplifies data visualization and boosts the data's readability.

Generally, three basic categories of feature selection strategies fall under the filter, wrapper, and hybrid models. The filter technique relies on the common traits of the training data to choose features independently of any classifier. The advantage of this type is high processing speed. However, it has low accuracy [1], [3], [4]. Examples of this technique are Relief [5] and correlation features selection correlation features selection (CFS) [6], fast correlation based features selection fast correlation-based feature selection (FCBF) and chi-square (Chi^2) [7], and information gain [8]. Wrappers feature selection techniques. This method uses a specific classifier to measure the significance of the feature set. Hence, this type depends on the classification method. This method introduces a better performance than the filter method regarding accuracy classification. However, wrapper methods cause an expensive computation cost [3], [4]. These techniques include genetic algorithm [9] and ants colony [10]. A hybrid methodology that combines the benefits of the two preceding approaches is suggested. We must list some of these methods for classifying network traffic as [11], [12]. However, this approach will carry over some drawbacks along with the benefits of both. This paper adopted the filter technique due to the filter method's quick processing and the fact that time processing is the most important aspect of Internet traffic classification, particularly in online and real-time settings.

This study employs four filtering methods: chi-square, gain ratio, significance feature evaluator, and fast correlation-based filter. Ranker and Borda's voting filters are applied to the output vector of each method. Only the features with the most votes and rank values will be chosen. The next section will present the studies related to the work, and then the methodology will be presented. Subsequently, the dataset details will be explored, the results and analysis will be discussed, and lastly, the conclusion of this research will be introduced. The studies that are relevant to the work will be presented in the following section. After that, the methodology of the technique will be described. The dataset specifics will then be explored. The results and analysis will then be covered. Finally, the research's conclusion will be introduced.

2. RELATED WORKS

The pre-processing stage of feature selection is increasingly important in developing various systems for monitoring internet traffic and spotting anomalies and attacks because of the ongoing development of data dimensionality. Therefore, one of the difficulties most researchers encounter is choosing the most important feature set. The goal of the selection process should be to lessen the data dimensionality while maintaining high accuracy.

Based on the best features of the network transaction data that were made available for training, Aljawarneh *et al.* [13] developed a hybrid model that can be utilized to determine the intrusion scope threshold degree. The vote algorithm with information gain was used to filter the data, combining the base learners' probability distributions to select the important features that improve the accuracy of the suggested model. The authors used the KDD dataset, which has 41 features. Different classifiers based on decision trees were employed for classification. Ambusaidi *et al.* [14] suggested a mutual information-based approach for choosing the best feature for classification through analytical selection. This mutual information-based feature selection algorithm can handle features with linear and nonlinear dependencies in the data. In the instances of network intrusion detection, its effectiveness is assessed. With the help of the features chosen by their suggested feature selection technique, an intrusion detection system (IDS) known as least square support vector machine-based IDS (LSSVM-IDS) is constructed. Utilizing three intrusion detection assessment datasets—KDD Cup 99, NSL-KDD, and Kyoto 2006+dataset—the performance of LSSVM-IDS is assessed. Each dataset contained 41, 41, and 24 features, respectively. The authors successfully reduced the number of features in each dataset above to 19, 18, and 4 features. Zhou *et al.* [15] proposed an intrusion detection framework based on feature selection and ensemble learning approaches. The authors utilized a heuristic technique for dimensionality reduction, which selects the best subset based on feature correlation. Then, they presented a hybrid strategy incorporating the C4.5, random forest (RF), and forest by penalizing attributes (Forest PA) algorithms. Finally, for attack recognition, the voting mechanism is utilized to aggregate the probability distributions of the base learners. For the experimental results, they used three separate datasets. A newly created anomaly detection system built on the natural fusion of various deep learning methods was presented by Zhong *et al.* [16]. The first step was to extract features from network traffic using the damped incremental statistics algorithm. The second step involved training the autoencoder with a small amount of label data. The third step involved using the autoencoder to mark network traffic with an abnormal score. Finally, the abnormal score label data was used to train the long-short-term memory. A type of recurrent neural network called LSTM outperforms conventional recurrent neural networks in terms of memory. Salo *et al.* [17] proposed an ensemble classifier based on support vector machine (SVM), instance-based learning algorithms (IBK), and multilayer perceptron (MLP), with an information gain (IG) and principal component analysis (PCA)-based dimensionality reduction technique for intrusion detection. ISCX 2012, NSL-KDD, and Kyoto 2006+ were used as test datasets to evaluate the performance of this IG-PCA-Ensemble approach. The experimental findings show that the suggested hybrid dimensionality reduction strategy with the ensemble of base learners offers more key qualities and performs individual strategies regarding accuracy and false alarm rates. A comparison of the proposed technique to comparable work is undertaken, and they find that the suggested IG-PCA-Ensemble method performs better in terms of classification accuracy. Abdullah *et al.* [18] developed an approach to partition the input dataset into various subgroups based on each attack. Then, for each subset, they used a feature selection technique with an information gain filter. The best feature set is then constructed by combining the list of feature sets collected for each attack. The findings of experiments on the NSL-KDD dataset show that the proposed strategy for feature selection with fewer characteristics improves system accuracy while minimizing complexity. Moustafa *et al.* [19] presented an ensemble intrusion detection technique to reduce harmful events. In internet of things (IoT) network traffic, specifically botnet assaults on the domain name system (DNS), hypertext transfer protocol (HTTP), and message queuing telemetry transport (MQTT) protocols used in IoT networks. The methods develop new statistical flow features based on examining their prospective attributes. Then, to effectively detect detrimental occurrences, an AdaBoost ensemble learning system is developed using three machine learning techniques: decision trees, naive Bayes (NB), and artificial neural networks. The proposed characteristics are extracted, and the ensemble technique is evaluated using the UNSW-NB15 and network information management and security (NIMS) botnets datasets, including simulated IoT sensor data. The experimental findings show that the suggested attributes have the potential to exhibit both normal and malicious behavior traits using the correlation coefficient and correntropy metrics. The suggested ensemble technique also has a higher detection rate.

Tang *et al.* [20] offered an upgraded Adaboost algorithm (MF-Adaboost) and several network traffic features as the foundation for their LDoS attack detection approach. After analyzing the traffic, they created a network feature set that is utilized for feature selection and feature calculation of network traffic data. By calculating features, one can minimize the amount of network data while extracting the most valuable information from network traffic data. Feature selection is utilized to choose the best classification features to guarantee that the detection algorithm can be trained successfully. Tests are run on a test-bed platform and the NS2 simulation platform to assess the effectiveness of this approach. The experiment's findings show that the approach can efficiently detect LDoS attacks.

By combining temporal, byte, and statistical data aspects, Lin *et al.* [21] created a multi-level feature fusion model (MFFusion) that extracts reliable information from many viewpoints and creates a more robust and efficient model. Trials demonstrate that models can cut training time, stabilize the training process, and enhance model performance. Attention loss adaptively modifies sample weights to increase the detection rate of weird samples. With several real network datasets, MFFusion has demonstrated excellent performance in terms of detection rate and false alarm rate. Using the newest IoT malicious traffic dataset, IoT23, they also use MFFusion for IoT network anomaly detection. Multifunctionality and suitability for network anomaly detection in the IoT are demonstrated by experiments conducted using MFFusion.

Using feature fusion and machine learning, Li *et al.* [22] suggested a method for detecting malicious mining codes. They begin by extracting multi-dimensional information through static and statistical analytic techniques. Next, using the n-gram model and TF-IDF to extract feature vectors for multi-dimensional text features, the classifier selects the best feature vectors fused with additional statistical features to train our detection model. At last, the machine learning framework is utilized to conduct automatic detection. According to the experimental findings, our technique can achieve 98.0% identification accuracy, an F1-score of 0.969, and an AUC of 0.973 for the ROC.

3. METHOD

This research proposed a multi-features selection method MFST that uses four techniques: the fast correlation-based filter (FCBF), significance feature evaluator (SFE), chi-square (Chi²), and gain ratio (GR).

Each technique produces a features vector exposed to the ranker and Borda voting filters. From the dataset, the characteristics with the most votes will be chosen. Four feature selection strategies are used instead of one for the following reasons: Use four techniques to benefit from each technique's advantages and try to overcome the disadvantages of one technique by adopting other techniques. Adopting four techniques, each introduces a different selection schema, producing a robustness features vector. Using one feature selection may affect the classifier's performance in terms of accuracy. It allows selecting the most significant features subset from the important features set. As illustrated in Figure 1, the MFST framework consists of three major phases. The first step, known as the dataset phase, comprises of the raw dataset's entry. Second, it is the feature selection step, which is divided into two parts.

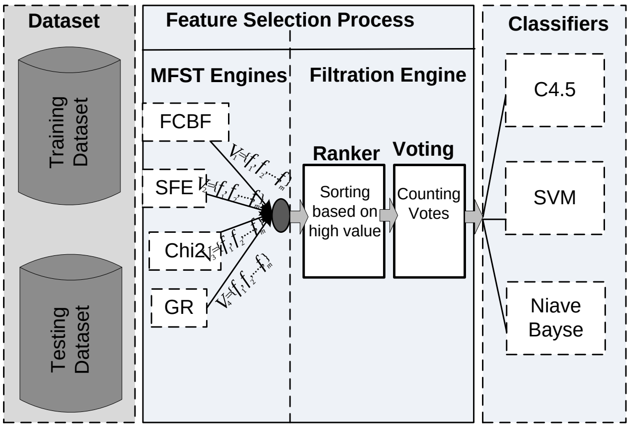
Dataset
Training Dataset
Testing Dataset
Feature Selection Process
MFST Engines
Filtration Engine
FCBF
SFE
Chi2
GR
$V_1=\{f_1,f_2,\ldots,f_m\}$
$V_2=\{f_1,f_2,\ldots,f_m\}$
$V_3=\{f_1,f_2,\ldots,f_m\}$
$V_4=\{f_1,f_2,\ldots,f_m\}$
Ranker
Voting
Sorting based on high value
Counting Votes
Classifiers
C4.5
SVM
Niave Bayse

Figure 1. The proposed MFST framework

These vectors are transmitted to the ranker filter within the filtration engine, which is divided into two portions (ranking and voting). The features are ranked according to the degree of importance measured by the ranking filter algorithm. Following that, the vectors of ranked features go through the second filtration step, which counts the votes for each feature. In other words, choose the first M highly ranked features from each preceding feature selection and then count the votes for each feature. The characteristics with the most votes and the highest rank value (value) will be chosen as the most discernment features. As a result, we can assure that this collection of attributes is significant when we select four selection approaches. The next section will discuss each unit of the proposed MFST framework.

3.1. Filter features techniques

Feature selection with filtering techniques is a method of ranking relevant characteristics according to specific statistical metrics in order to choose a subset of them. These techniques just consider the intrinsic qualities of the data and operate independently of machine learning algorithms. This paper will adopt four different filter techniques to extract the significant features vector in Internet traffic.

3.1.1. Significance feature evaluator

Significance feature evaluator (SFE) is one of the important filter techniques used to select the significant features for certain datasets. SFE is built based on the conditional probability technique where the features are selected based on hypothesis state features with different values and classes, while this may not be for insignificant features. Two major reasons to select SFE are that it increases computation speed. It enhances the quality of knowledge in terms of classification by using the likelihood-based method for classificatory knowledge. Therefore, SFE is integrated with the classification method based on likelihood, and the selected significant features are used to make the classification decision.

3.1.2. Fast correlation-based filter

This approach uses correlation measurements to find redundant and relevant features to the class. The general guideline for correlation measurement is that a feature is good if it is relevant to the class but not redundant with other time-related features. A feature is considered good if it is substantially linked with the class rather than with other features. There are often two methods. To overcome the limitations of linear correlation, which can be summed up as not always assuming a linear correlation between features in the real world, being unable to capture correlations that are not linear, and the calculation requiring all features to be congruent, FCBF adopted a second approach based on information theory. The correlation measure is based on the entropy notion from information theory to assess the random object's level of uncertainty [7], [23], [24].

3.1.3. Chi-squared feature selection

Chi-squared or Chi² is based on the x2 statistic and consists of two main phases. Phase 1, it begins with a high significance level called sigLevel for all numeric attributes. Each attribute is sorted according to its values. Then the following is performed:

– For each pair of neighboring intervals, calculate the x2 using (3).
– The adjacent intervals with the smallest x2 value should be combined.
– Merging keeps on until all interval pairs have x2 values greater than the parameter set by sigLevel.
The procedure is repeated with a lower sigLevel up until an inconsistency rate in the discretized data is exceeded, at which point Chi² automatically chooses an appropriate x2 threshold that preserves the accuracy of the original data. Phase 2 is a more refined procedure of Phase 1, beginning with the sigLevel established in Phase 1. Each attribute i is connected to a certain sigLevel $[i]$, and each attribute is merged one at a time [25].
– Verifying consistency following each attribute merge.
– The sigLevel for attribute i is lowered for the subsequent round of merging if the inconsistency rate is not exceeded. This process continued until no attribute's values could be merged.
– If an attribute has only one value at the end of Phase 2, it merely signifies that the attribute does not accurately represent the original data set.
– At the conclusion of discretization, feature selection is completed.

3.1.4. Gain ratio feature selection

The gain ratio improves information gain by providing a normalized score of a feature's contribution to the best classification decision based on information gain. The gain ratio is used in an iterative process where we choose increasingly smaller groups of features [26]. These iterations end when just a certain number of features are left. The gain ratio is one of the disparity measurements employed, and a feature's high gain ratio suggests that it will be helpful for categorization. Gain ratio applied normalization to the information gain score using a split information value, which was initially utilized in the decision tree (C4.5) [27].

3.2. Features filter

Concerning the proposed model shown in Figure 1, each feature selection technique will choose $Features\ vector = [f1, f2, f3, f4, \dots fn]$. This is what the MFST engine will experience. Each feature vector in the filtering engine will be sorted using a ranker filter depending on its ranking value to find the most important characteristics. The most important aspect is its high value. After that, a vote will be conducted for each feature to determine how many of the four techniques are selected for that feature. The feature that receives the most votes and rank value will be chosen. The two filters will be explained in the next subsections.

3.2.1. Fast feature ranking

MFST framework is adopting two filters to select the most significant features subset among the important features set by selecting the highly ranked features (based on the ranker filter) as well as selecting the feature that has the highest number of votes from the four feature selection techniques (based on voting filter). Feature ranking determines the most significant feature from the extracted features. One of the most effective algorithms used in feature ranking is based on information theory, such as mutual information, which is considered one of the fastest ranking techniques, which is calculated based on algorithms built in the proposed method, which was adapted via different data mining programs MATLAB, Weka, and Rapid. Ranking process consists of four steps:

- Initialize set F_s to the whole set of F_p features. E is an empty set.
- For all features $f \in F_s$ compute mutual information (MI).
- Find feature f that maximizes (MI) and move it to E.
- Repeat unit the Cardinal of E is F_p.

3.2.2. Features voting based on Borda count

After ranking the features as the first step in the filtration zone, the features undergo the second step of filtration, which is voting, which works tightly with the ranking unit—The proposed MFST system adopted Borda-count as a type of voting. Generally, there are three major voting methods: Majority, Plural, and Borda. The main reason that motivated us to choose Borda-count rather than others is that Borda considers rank values for features besides the decision of voters (i.e., for each voter (Techniques) ranks the candidates (features)) that made work fit with ranking unit [28].

3.3. Internet traffic datasets

Benchmark datasets were employed in this investigation which can be find in [29]. To generate data traffic, a worldwide source depends on research undertaken at Queen Mary University of London's Department of Computer Science and mentioned in Caida databases. This information is gathered from a network's edge. It permits access to all transmission control protocol (TCP), user datagram protocol (UDP), and internet protocol (IP) connection-related packets traveling in both directions (from sender to recipient and vice versa). Therefore, it can get additional features for each packet as a result. Different dataset sizes are used to prevent the proposed notion from being overfitted and to track behavior using various datasets. The dataset consists of 248 features based on various traffic behaviors extracted from packet headers of the three data transfer methods listed above as in Figure 2: traffic duration, TCP, UDP port, and Payload size statistics. Each feature's specifics are described in [30]. As indicated in Table 1, the classes present in the datasets are divided into ten groups. Each data entry will allocate 80% for training and building the classifier model and 20% for testing. Ten subsets were all gathered using different protocol connections; the variety of instances allows for testing the suggested solution in a heterogeneous environment. The instance subset size for each dataset is displayed in Table 2.

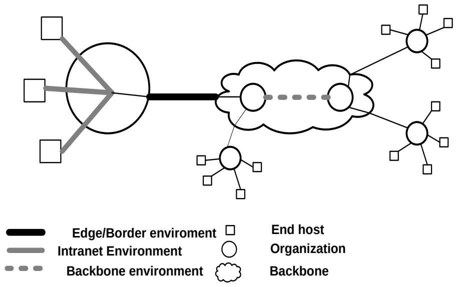

Figure 2. Dataset environment

Table 1. Datasets categorization

#	Category	Applications description
1	Web-browsing	http, https
2	Mail	Imap, POP2, POP3, SMTP
3	Bulk	FTP
4	Attack	Portscan, worms, viruses, email attacks
5	P2P	Napster, KAZAA, eMule, Gnutella, eDonkey
6	Database	MySQL, dbase, Oracle SQLNet
7	Multimedia	Windows media player, realmedia
8	Service	X11, DNS, iDent, Idap, NTP
9	Interactive	SSH, Telnet, Klogin, rlogin
10	Games	Microsoft direct play

Table 2. Number of instances in each dataset

Dataset Name	# Instances
Subset 01	24863
Subset 02	23801
Subset 03	22932
Subset 04	22285
Subset 05	21648
Subset 06	19384
Subset 07	55835
Subset 08	55494
Subset 09	66248
Subset 10	65036

3.4. Implementation of MFST

This section explores the implementation steps for MFST. As mentioned, MFST consists of four feature selection techniques and two filters. Firstly, we need to run all the above ten datasets individually in each of the four feature selection techniques and get the rank value to get the most significant feature vector. MFST considers only the first log_2^{m+1} based on information gain [31], where m is the total number of $features = 248$. So, the number of considered features is only the first eight highly ranked dataset features. The next subsection shows the generated feature vectors for the above datasets. Tables 3 and 4 list the feature vectors selected by each technique according to the proposed technique SFE, FCBF, Chi², and GR. All the selected features are ordered based on the rank value for each feature.

Table 3. Selected features vector by SFE, FCBF

Dataset	Selected features vector by each Technique	
	SFE	FCBF
Subset 01	1, 90, 170, 96, 95, 94, 187, 180	1, 60, 95, 96, 86, 84, 82, 186
Subset 02	1, 45, 170, 187, 90, 177, 113, 59	1, 95, 137, 60, 45, 113, 125, 59
Subset 03	1, 90, 96, 95, 86, 45, 94, 177	1, 95, 60, 96, 45, 113, 84, 59
Subset 04	71, 1, 96, 90, 180, 187, 94, 184	1, 60, 137, 95, 45, 96, 187, 180
Subset 05	1, 90, 65, 71, 94, 88, 187, 180	1, 60, 45, 95, 88, 113, 59, 137
Subset 06	70, 1, 187, 180, 90, 96, 95, 177	1, 83, 95, 60, 187, 180, 96, 88
Subset 07	1, 95, 96, 187, 162, 125, 180, 173	1, 125, 137, 95, 113, 59, 45, 133
Subset 08	1, 71, 96, 95, 162, 180, 45, 187	1, 125, 95, 113, 59, 45, 137, 162
Subset 09	1, 95, 180, 187, 184, 165, 158, 96	1, 95, 162, 83, 59, 45, 113, 47
Subset 10	1, 162, 187, 180, 184, 170, 96, 95	1, 162, 59, 45, 113, 95, 137, 60

Table 4. Selected features vector by Chi², GR

Data Name	Selected vector by MFST arranged based on rank value	Votes
Subset 01	1, 96, 95, 187, 180, 82, 86, 90	(4,3,3,3,3,3,2,2)
Subset 02	1, 95, 180, 45, 113, 59, 162, 137	(4,4,3,2,2,2,2,2)
Subset 03	1, 95, 96, 45, 186, 179, 113, 180	(4,4,4,3,3,2,2,2)
Subset 04	1, 95, 96, 187, 180, 94, 184, 90	(4,3,3,3,3,2,2,2)
Subset 05	1, 95, 71, 88, 96, 187, 184, 180	(4,3,2,2,2,2,2,2)
Subset 06	1, 95, 96, 187, 180, 88, 83, 186	(4,3,3,3,3,2,2,2)
Subset 07	1, 125, 95, 96, 83, 45, 162, 187	(4,3,3,3,3,2,2,2)
Subset 08	1, 95, 96, 71, 162, 45, 187, 180	(4,3,3,2,2,2,2,2)
Subset 09	1, 95, 162, 43, 95, 96, 184, 187	(4,3,2,2,2,2,2,2)
Subset 10	1, 95, 96, 83, 162, 137, 187, 180	(4,3,3,3,2,2,2,2)
The highly ranked and votes features vector		1, 95, 96, 180, 187, 162, 45, 83

3.4.1. MFST based on ranking and voting

This section presents the selected vectors of MFST, which are arranged based on rank value. In addition, the votes for each feature are calculated. Next, the highly ranked and votes vector of features is concluded for the datasets as in Table 4. The table displays the top 8 features for each data input, which were chosen using four techniques in the proposed MFST framework. In voting, the number 4 denotes a feature chosen by four techniques, the number 3 indicates a feature chosen by three techniques, and the number 2 shows a feature chosen by just two techniques.

Based on Table 5 reveals the description of features vector that were selected based on proposed MFST for the datasets. Table 5 also displays the final feature vector chosen based on MFST. The Moore dataset which carries the numbers 1, 95, 96, 180, 187, 162, 45, and 83, which represent the feature, server port, initial window-bytes client-server, initial window-bytes server-client, var data wire b a, var data ip b a, med data IP a b, actual data pkts a b, min seg m size a b.

Table 5. Highly ranked and popular features vectors chosen by MFST

Dataset	Selected features vector by each technique	
	Chi-Squared	Gain Ratio
Subset 01	1, 95, 187, 96, 93, 90, 180, 186	1, 24, 151, 143, 133, 147, 101, 137
Subset 02	1, 180, 95, 96, 85, 93, 187, 186	1, 137, 133, 125, 151, 24, 143, 147
Subset 03	1, 95, 187, 184, 180, 177, 93, 96	1, 133, 71, 137, 125, 60, 95, 188
Subset 04	1, 95, 96, 94, 90, 83, 186, 187	1, 167, 24, 133, 137, 125, 71, 92
Subset 05	1, 187, 184, 95, 100, 180, 44, 96	1, 167, 24, 101, 188, 125, 137, 133
Subset 06	1, 95, 83, 96, 187, 186, 184, 180	1, 24, 167, 125, 133, 137, 61, 63
Subset 07	1, 95, 96, 83, 90, 187, 180, 184	1, 167, 133, 125, 57, 151, 24, 135
Subset 08	1, 95, 83, 96, 187, 184, 186, 179	1, 167, 151, 71, 133, 135, 125, 57
Subset 09	1, 96, 95, 93, 184, 177, 83, 187	1, 167, 24, 151, 57, 92, 133, 101
Subset 10	1, 93, 187, 95, 96, 159, 180, 184	1, 77, 75, 24, 137, 133, 125, 145

4 RESULTS AND DISCUSSION

Classification accuracy is one of the most important metrics used to measure performance and assess the techniques and methods in the data mining area. Accuracy is evaluated using many factors. But the standard factor combining all these factors is called classification accuracy, which is defined and formulated in (1):

$$Accuracy = \frac{(TN+TP)}{(TN+TP+FN+FP)}\ 100\% \qquad (1)$$

MFST is evaluated and compared with the four most famous filter feature selection techniques (FCBF, GR, Chi², and SFE). MFST and other feature selection techniques are tested over three classifiers, namely (naïve Bayes, support vector machine, and C4.5). Those classifiers achieved high performance in terms of accuracy, processing time, and memory consumption. Therefore, the experiments were repeated five times for each classifier, resulting in 15 values for each chart.

Generally, in this section, the discussion of proposed methods will be divided into two groups, the first referring to the first six datasets of the dataset (Moore datasets) and the second referring to the last five datasets. In all experiments, we allocated 70% of each dataset for the training phase, while 30% was allocated to the testing phase. Normally, the data for training is more than that of the testing phase because training needs extra data to build the classifier's model.

Figure 3 shows the classification accuracy results for the first six different datasets whose sizes are convergent, ranging between (24,836 and 19,384) instances/input. Those datasets gave similar accuracy, presenting approximately naïve Bayes (NB), which resulted in the lowest value over the six datasets and using the four techniques (FCBF, GR, Chi², and SFE), which range between (59%-78%). While the proposed MFST ranges between (81%-85%). Notably, the MFST optimized NB's performance, resulting in more accuracy obtained. SVM and C4.5 show high performance over the six datasets whether using MFST or any of the four competing techniques, which reflected the powerfulness of those classifiers ranging between (96.4%-99.7%); still, one can see a stable performance with MFST where the percentage ranges between (99.2%-99.7%). Overall, C4.5 was slightly better than SVM.

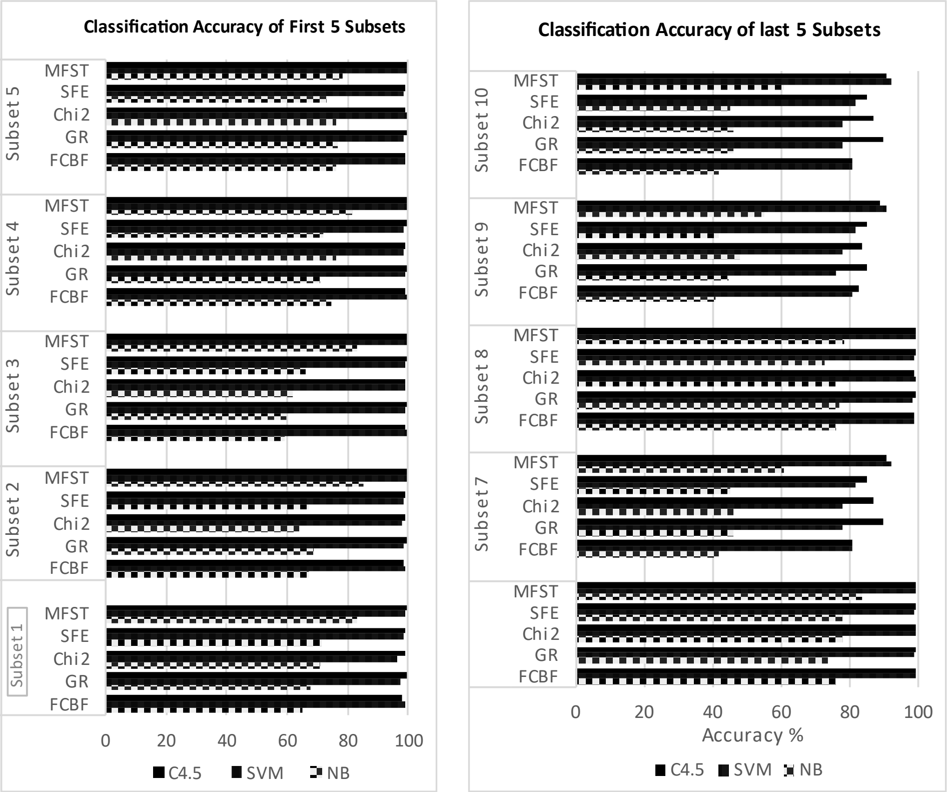

Figure 3. The classification accuracy using 3 classifiers and 4 feature selection strategies was compared to the suggested MFST

Also, Figure 3 highlights the classification accuracy of the last five Moore datasets. The size of this group of datasets ranges between (55,494 and 66,248) instances. Generally, the performance of classifiers based on the feature selection techniques is dropped back. Naïve Bayes was the worst with a percentage ranging between (40%-52%) based on the four techniques, while NB with MFST raises the accuracy ranging between (55%-66%). SVM and C4.5 offer higher accuracy than NB. Still, the range is decayed compared with the first and abovementioned group, where the percentage lies between (76-90) using the four techniques, while the proposed MFST achieved a higher percentage ranging between (89%-93%).

Also, it is noted that the GR technique always introduced better results in contrast with the four techniques with the C4.5 classifier. At the same time, the SFE and Chi² are competitive with a slight difference in favor of the SFE by ranging the percentage between (82%-85%) while Chi² achieved (77%-87%). FCBF had given less percentage than other techniques with the NB classifier. In contrast, the percentage was increased using other classifiers such as SVM and C4.5, where the value ranged between (41%-43%) over this group of datasets.

4.1. Results analysis of classification accuracy

This section initiates a critical discussion of the proposed features section technique MFST over ten benchmark datasets to highlight its differences from other techniques in the context of classification accuracy. The results of the first six datasets are similar due to the concurrent data size. NB reported the lowest accuracy due to its strong assumption (i.e., any two features are independent, given the output class). However, the accuracy of the MFST can be increased by introducing feature sets that are closely correlated to the class. The other classifiers presented a high percentage with convergent performance for the features technique. The second group of datasets showed a slight difference in the result of the SVM and C4.5 due to increased data size and the classifiers' yield lower percentages. The MFST reported the highest percentage due to its dependence on high relative discriminates sets, which helps optimize the performance of the classifiers and their corresponding classification accuracy, which matches with the suggested objectives.

5 CONCLUSION

Features play a significant role in classification; that's why many studies have been proposed to find the most important feature vector, which helps classify the traffic accurately and without cost. This paper presents hybrid feature selection techniques, which adopt four techniques and two filter methods. The selection features vector is performed only once at the beginning of classification. Then, the selected features undergo ranking and voting filters to identify the most significant features among the important features set. The experiments used ten datasets with varying instances, proving that the proposed technique increases the accuracy, especially for high-traffic datasets.

REFERENCES

[1] H. Alazzam, A. Sharieh, and K. E. Sabri, "A feature selection algorithm for intrusion detection system based on Pigeon inspired optimizer," *Expert Systems with Applications*, vol. 148, Jun. 2020, doi: 10.1016/j.eswa.2020.113249.

[2] M. M. Abualhaj, A. A. Abu-Shareha, M. O. Hiari, Y. Alrabanah, M. Al-Zyoud, and M. A. Alsharaiah, "A paradigm for DoS attack disclosure using machine learning techniques," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 3, pp. 192–200, 2022, doi: 10.14569/IJACSA.2022.0130325.

[3] C. W. Chen, Y. H. Tsai, F. R. Chang, and W. C. Lin, "Ensemble feature selection in medical datasets: combining filter, wrapper, and embedded feature selection results," *Expert Systems*, vol. 37, no. 5, Apr. 2020, doi: 10.1111/exsy.12553.

[4] M. Hammami, S. Bechikh, C. C. Hung, and L. Ben Said, "A multi-objective hybrid filter-wrapper evolutionary approach for feature selection," *Memetic Computing*, vol. 11, no. 2, pp. 193–208, Jul. 2019, doi: 10.1007/s12293-018-0269-2.

[5] R. J. Urbanowicz, M. Meeker, W. La Cava, R. S. Olson, and J. H. Moore, "Relief-based feature selection: introduction and review," *Journal of Biomedical Informatics*, vol. 85, pp. 189–203, Sep. 2018, doi: 10.1016/j.jbi.2018.07.014.

[6] K. R. Pushpalatha and A. G. Karegowda, "CFS based feature subset selection for enhancing classification of similar looking food grains-a filter approach," Dec. 2018, doi: 10.1109/ICECIT.2017.8453403.

[7] N. Gopika and A. M. kowshalaya M.E., "Correlation based feature selection algorithm for machine learning," in *2018 3rd International Conference on Communication and Electronics Systems (ICCES)*, Oct. 2018, pp. 692–695, doi: 10.1109/CESYS.2018.8723980.

[8] B. Venkatesh and J. Anuradha, "A review of feature selection and its methods," *Cybernetics and Information Technologies*, vol. 19, no. 1, pp. 3–26, Mar. 2019, doi: 10.2478/CAIT-2019-0001.

[9] S. Jadhav, H. He, and K. Jenkins, "Information gain directed genetic algorithm wrapper feature selection for credit rating," *Applied Soft Computing Journal*, vol. 69, pp. 541–553, Aug. 2018, doi: 10.1016/j.asoc.2018.04.033.

[10] M. Ghosh, R. Guha, R. Sarkar, and A. Abraham, "A wrapper-filter feature selection technique based on ant colony optimization," *Neural Computing and Applications*, vol. 32, no. 12, pp. 7839–7857, Apr. 2020, doi: 10.1007/s00521-019-04171-3.

[11] A. Munther, I. J. Mohammed, M. Anbar, and A. M. Hilal, "Performance evaluation for four supervised classifiers in internet traffic classification," in *Communications in Computer and Information Science*, vol. 1132 CCIS, Springer Singapore, 2020, pp. 168–181.

[12] A. Munther, R. R. Othman, A. S. Alsaadi, and M. Anbar, "A performance study of hidden markov model and random forest in

internet traffic classification," in *Lecture Notes in Electrical Engineering*, vol. 376, Springer Singapore, 2016, pp. 319–329.

[13] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *Journal of Computational Science*, vol. 25, pp. 152–160, Mar. 2018, doi: 10.1016/j.jocs.2017.03.006.

[14] M. A. Ambusaidi, X. He, P. Nanda, and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," *IEEE Transactions on Computers*, vol. 65, no. 10, pp. 2986–2998, Oct. 2016, doi: 10.1109/TC.2016.2519914.

[15] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, Jun. 2020, doi: 10.1016/j.comnet.2020.107247.

[16] Y. Zhong *et al.*, "HELAD: A novel network anomaly detection model based on heterogeneous ensemble learning," *Computer Networks*, vol. 169, Mar. 2020, doi: 10.1016/j.comnet.2019.107049.

[17] F. Salo, A. B. Nassif, and A. Essex, "Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection," *Computer Networks*, vol. 148, pp. 164–175, Jan. 2019, doi: 10.1016/j.comnet.2018.11.010.

[18] M. Abdullah, A. S. Al-Shannaq, S. Almabdy, A. Balamash, and A. Alshannaq, "Enhanced intrusion detection system using feature selection method and ensemble learning algorithms," *International Journal of Computer Science and Information Security*, vol. 16, no. 2, pp. 48–55, 2018.

[19] N. Moustafa, B. Turnbull, and K. K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4815–4830, Jun. 2019, doi: 10.1109/JIOT.2018.2871719.

[20] D. Tang, L. Tang, R. Dai, J. Chen, X. Li, and J. J. P. C. Rodrigues, "MF-Adaboost: LDoS attack detection based on multi-features and improved Adaboost," *Future Generation Computer Systems*, vol. 106, pp. 347–359, 2020, doi: 10.1016/j.future.2019.12.034.

[21] K. Lin, X. Xu, and F. Xiao, "MFFusion: a multi-level features fusion model for malicious traffic detection based on deep learning," *Computer Networks*, vol. 202, Jan. 2022, doi: 10.1016/j.comnet.2021.108658.

[22] S. Li, L. Jiang, Q. Zhang, Z. Wang, Z. Tian, and M. Guizani, "A malicious mining code detection method based on multi-features fusion," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 5, pp. 2731–2739, Sep. 2023, doi: 10.1109/TNSE.2022.3155187.

[23] L. Yu and H. Liu, "Feature selection for high-dimensional data: a fast correlation-based filter solution," *Proceedings, Twentieth International Conference on Machine Learning*, vol. 2, pp. 856–863, 2003.

[24] Q. Song, J. Ni, and G. Wang, "A fast clustering-based feature subset selection algorithm for high-dimensional data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 1, pp. 1–14, Jan. 2013, doi: 10.1109/TKDE.2011.181.

[25] A. Thakkar and R. Lohiya, "Attack classification using feature selection techniques: a comparative study," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 1, pp. 1249–1266, Jun. 2021, doi: 10.1007/s12652-020-02167-9.

[26] A. H. Mohammad, "Comparing two feature selections methods (Information gain and gain ratio) on three different classification algorithms using arabic dataset," *Journal of Theoretical and Applied Information Technology*, vol. 96, no. 6, pp. 1561–1569, 2018.

[27] A. Sharma and S. Dey, "Performance investigation of feature selection methods and sentiment lexicons for sentiment analysis," *International Journal of Computer Applications*, no. June, pp. 15–20, 2012.

[28] P. Emerson, "The original Borda count and partial voting," *Social Choice and Welfare*, vol. 40, no. 2, pp. 353–358, Oct. 2013, doi: 10.1007/s00355-011-0603-9.

[29] A. W. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in *Performance Evaluation Review*, 2005, vol. 33, no. 1, pp. 50–60, doi: 10.1145/1071690.1064220.

[30] A. W. Moore and K. Papagiannaki, "Toward the accurate identification of network applications," in *Lecture Notes in Computer Science*, vol. 3431, Springer Berlin Heidelberg, 2005, pp. 41–54.

[31] H. Liu and H. Motoda, "Feature selection aspects," in *Feature Selection for Knowledge Discovery and Data Mining*, Springer US, 1998, pp. 43–72.

BIOGRAPHIES OF AUTHORS

Alhamza Munther is an assistant professor in university technology and applied sciences in Sur, Sultanate of Oman. He received his B.Sc. in computer and software engineering from the University of Technology in Baghdad in 2003. He received a master's degree in advanced computer networks from Universiti Sains Malaysia (USM) in 2012, and in March 2017, he was awarded a Ph.D. in computer engineering from Universiti Malaysia Perlis, Malaysia. His research focuses on overlay networks, multimedia distribution, traffic engineering, data mining, and machine learning. He can be contacted at email: alhamza.wardi@utas.edu.om.

Mosleh M. Abualhaj is a professor in Al-Ahliyya Amman University. He received his first degree in computer science from Philadelphia University, Jordan, in July 2004, master's degree in computer information system from the Arab Academy for Banking and Financial Sciences, Jordan in July 2007, and doctorate degree in multimedia networks protocols from University Sains Malaysia in 2011. His research area of interest includes VoIP, multimedia networking, and congestion control. Apart from research, Dr. Abu-Alhaj also does consultancy services in the above research areas and directs the Cisco academy team at Al-Ahliyya Amman University. He can be contacted at email m.abualhaj@ammanu.edu.jo.

Alabass Alalousi is an assistant professor in Thamar university. He is currently head of the Information Technology Department. He received his first degree in computer sciences from the University of Al Mustansiriyah, Baghdad, Iraq in 2001, and master's degree in computer science from Iraqi Commission for Computers and Informatic, Institute for Post Graduate Studies in Informatic, Baghdad/Iraq in 2003 and doctorate degree in educational administration and planning, University of Sana'a in 2019. His research area of interest includes decision making, machine learning, computer networks and information hiding. He can be contacted at email: alabassmunther79@gmail.com.

Hilal A. Fadhil holds a Ph.D. in telecommunication engineering. He is currently employed as an assistant professor at Sohar University in Oman. His research interests include Optical CDMA, LiFi technologies, and wavelength division multiplexing for optical access networks. He holds several patents in the UK and Malaysia and has published more than 100 indexed journal articles and reviewed conference papers. Owing to his many research and product achievements and contributions, Hilal A. Fadhil was awarded many awards and medals in the UK, Germany, South Korea, and Malaysia. He is a senior member of IEEE communication (MIEEEUSA), a member of the Institution of Engineering and Technology (MIET-UK), and a fellow of the Optical Society of America (SPIE). He is also an editor for IEEE Communications Magazine, the Optical Engineering Society, IET Optoelectronics, and the founding reviewer of the IEEE Communication Letter. He can be contacted at email: hfadhil@su.edu.om.