Received: July 6, 2023. Revised: August 4, 2023.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Feature Selection for Intrusion Detection Using Independence Level Test with an Exhaustive Approach

Aulia Teaku Nururrahmah[1] Tohari Ahmad[1]*

[1]*Department of Informatics, Institut Teknologi Sepuluh Nopember,*
Kampus ITS Surabaya, 60111, Indonesia
* Corresponding author's Email: tohari@if.its.ac.id

Abstract: The intrusion detection system (IDS) has been developed to detect attacks or suspicious activity on a network. IDS are generally classified into two types: signature-based and anomaly detection. Many studies widely use anomaly-based detection because it can detect new types of attacks on the internet network, but it has several shortcomings. Handling data with high dimensionality directly to the classification process could lead to low accuracy and increased false alarm rates. Selecting relevant features and removing irrelevant features from classification results could be the solution to overcome this issue. In this research, we propose an intrusion detection model using a combination of the Chi-square independence test and an exhaustive search. Firstly, this proposed method employs the independence levels of the Chi-square test to calculate the statistical scores for each feature. The feature list obtained from the first process is continued to the optimisation stage using an exhaustive search. This process aims to calculate the accuracy values of all possible feature combinations from the early-stage feature list and check each feature combination to see if that combination has the best accuracy. This method was tested on four datasets: KDD Cup 99, NSL-KDD, Kyoto 2006+, and UNSW-NB15 using three classifiers: Support vector machine, decision tree, and Naive Bayes. This method achieved the highest accuracy when tested on the UNSW-NB15 dataset using the SVM. Accuracy, precision, recall, and F-score reached values above 95%. Likewise, the FPR value reached the lowest rate of 1.56%.

Keywords: Chi-square, Exhaustive search, Feature selection, Intrusion detection system, Network Infrastructure, Network security.

1. Introduction

In this decade, technology has advanced at an astounding rate. Access to information becomes more rapid, as if the data were in our hands. The advancement of this technology is like two sides of a coin, with a positive value on one side and a negative value on the other. The positive aspect of technological advancement is that it can make human labour easier. However, this technology is vulnerable to cybercrime. The internet world poses a significant threat to data security and user privacy.

It is critical to distinguish between normal Internet network activity and suspicious activity. So, there is a computer system called intrusion detection system (IDS) that can help us detect internet network attacks. Dorothy denning and Peter Neuman invented IDS between 1984 and 1986. IDS was developed on the assumption that suspicious activity patterns differ significantly from normal activity in an internet network [1]. As a result, this system model is critical in recognising specific patterns. Detecting attacks in a computer network does not have to be done manually; instead, machine learning is employed.

IDS are classified into two types: signature-based detection and anomaly-based detection [2]. Signature-based detection identifies attack patterns on the internet based on data from previous attacks [2, 3]. Attack data is stored in a dataset used as a rule in the IDS model. This method has a high degree of accuracy in recognising patterns of attacks that have occurred. However, if this IDS model encounters a previously unknown attack pattern, its accuracy may drop significantly [4]. This new type of attack was considered normal rather than attack activity because

the pattern was not found in the dataset that included the attack. The second approach, anomaly-based detection, can address this flaw in the signature-based IDS model. Activity will be correctly identified as an attack even if a new attack pattern has never been seen before [5]. The mechanism for detecting an anomaly method attack involves studying the profile of a sample of internet activity and determining a set of rules or restrictions based on certain parameters specific attack is defined as any activity that crosses these boundaries or violates these rules [6]. This method has a relatively high accuracy value because new attacks can be detected precisely as an attack. The most significant disadvantage is the high false positive and negative rates [4].

Various studies have been conducted so far to overcome the drawbacks of this method. Feature selection is frequently used as the main topic in multiple studies. The next major challenge in IDS is to devise a method for detecting internet attacks that is both effective and efficient. This is necessary because large-scale data can cause the IDS model to overfit [4]. Overfitting occurs when training data contains much irrelevant information, also known as meaningless data. Datasets typically have many features, some of which are relevant while others are not. These irrelevant features have no or little impact on the classification results. In IDS models, removing irrelevant features improves algorithm performance and accuracy [7]. However, feature selection can be detrimental but only beneficial if the number of features eliminated is sufficient that directly impacts the accuracy level. Otherwise, removing too few features can result in overfitting, leading to increased misleading or inaccurate results and longer training times [8]. The primary goal of this research is to identify the best solution to this problem.

In general, the three types of feature selection methods are filters, wrappers, and embedded selectors. The filter method utilises a statistical measurement-based assessment technique for each feature [9]. This feature determines which features are discarded or kept based on the ranking or value threshold. Our previous research [10] developed a filter-based method called the Chi-square test. The outcomes of the study improved performance, as evidenced by increased accuracy. However, the previous filter method has a flaw that must be addressed. When used alone, the Chi-square test determines that a feature is irrelevant to the classification result, but it has become relevant when combined with other features [11].

This is our motivation for developing research as a solution to conventional methods. The contribution of this study is to increase the performance of IDS using the wrapper method to compensate for the filter method's shortcomings. Exhaustive-search methods select features by looking for the highest accuracy among all possible feature combinations. The feature combination is searched to produce the best combination while removing features that are not included in the combination. The method is proper but inefficient due to its high algorithm complexity and long train extended time. The shortcomings of the wrapper and filter methods can be addressed by combining the two. The main principle in this research is an IDS model that can eliminate irrelevant features and generate the best feature combination.

The rest of this paper is structured as follows: Section 2 presents related works. The proposed method is described in section 3. Section 4 discusses the experiment scenario and results in detail, while section 5 draws the conclusion and describes future research.

2. Related works

Many novel studies in the field of IDS have recently been published. Recent studies have focused on improving performance, as evidenced by increased accuracy and a lower false alarm rate. One of several methods for improving performance is the feature selection. Several studies use the anomaly-based method to select and discard features that are relevant to the classification process in their machine-learning models. Several IDS studies have produced a machine-learning approach for improving intrusion detection performance on a computer network. Dealing with data with high dimensionalities, such as Kyoto 2006 [12], KDD Cup99 [13], NSL-KDD [14], and UNSW-NB15 [15], is a significant challenge in detecting computer network attacks. As a result, several studies, such as the one conducted by Sunyoto and Hanafi [16], reduce the data dimension. This study proposes an approach based on a stack-denoising autoencoder with a mechanism for reducing and selecting features to improve the effectiveness of IDS. Choosing features using SDAE reduced the number of features, but not all wasted features are irrelevant. It chose suboptimal features set. This resulted in a high number of false positives and negatives in each experiment. Almazini and Ku-Mahamud [17] used the binary grey wolf optimisation (EBGWO) method to overcome this weakness, controlling the balancing parameter. The evaluation was performed with an accuracy of 87.46% using the same dataset. This method has been shown to improve attack detection accuracy. Nevertheless, detecting all types of attacks remains challenging because the method only considers the

detection between normal and attack, which is not specific to the type of attacks. Setiawan et al. [7] focused their research on increasing the score while ensuring completeness in detecting all types of attacks. Their primary approach employed the log normalization, min-max, and z-score normalization methods. There is a process of changing decimals based on rounding techniques. Based on the results, log normalization and z-score have the highest score. It can be inferred that they choose the log normalization method because it is on the verge of being safe when converting decimals. However, log normalization is irrelevant for this case because the rounding error is too high, leading to high FPR. The research from [18] proposed a pigeon-inspired optimizer approach to choose essential features. The features of NSL-KDD were reduced a lot. This proposed method successfully maintained the FPR stay low with a high accuracy rate. The accuracy of this proposed method reaches 86.90%. The study uses cosine similarity for discretization and works best on continuous attributes, while the symbolic or categorical attributes tend to be ignored. This approach is too risky because it can eliminate the important categorical features. Alwan et al. [19] provided a solution for improving IDS accuracy while shortening the training time of machine learning algorithms. A modified version of the firefly algorithm with mutation operations on binary and multi-class classification was proposed in that study. This method shows a good result in eliminating irrelevant features to the target class, but this method does not consider the possibility of a relationship between one feature and another. Removing relevant features when combined with others will lead to detection errors and bias. To overcome this issue, the wrapper-based method was proposed [20]. The wrapper effectively increases IDS accuracy, but training takes a long time. The study proposes a new approach based on particle swarms called restoration particle swarms optimization (RPSO). The experiment was conducted on the NSL-KDD dataset with up to 85% accuracy. This value has increased by approximately 1.14% over the standard PSO. This method has significant drawbacks. It did help in increasing the accuracy. However, it consumes much time in the training and testing process when this method is used alone. A brute-force method is used to search all possibilities for a solution. It can lead to effective methods but could be more efficient.

Mahboob et al. [21] reduced misdiagnosis and improve the accuracy of IDS. They offer a hybrid approach that employs the arithmetic optimizer algorithm to select the best feature subset and the majority vote classifier (MVC) to perform classification. The UNSW-NB15 dataset was used for the evaluation. This experiment had an accuracy value of 98.36%. Disha and Waheed [22] developed a backward elimination-based feature selection method to enhance attack detection accuracy on computer networks. The Chi-square value of each feature is utilised to implement this technique. The proposed method addresses the problem of precision in several classification approaches. However, this study has drawbacks. This method shows decreasing precision and F1-Score results. In addition, it was only tested on one dataset, so the performance improvement shown in this study cannot prove the method's reliability on other datasets.

Gharaee and Hosseinvand [23] had the idea to use genetic algorithms to improve the performance of attack detection in computer networks as addressing methods with a filter approach. This experiment's performance was evaluated using the KDD Cup99 and UNSW-NB15 datasets. The results of this study show that the method can improve IDS performance in both datasets. The average accuracy value of each type of attack in the KDD Cup 99 dataset is around 99%, while it is about 90% in the UNSW-NB15 dataset. Nevertheless, the purpose of this method is to reduce the dimensionality using a stochastic genetic algorithm, i.e., there is no guarantee of the optimality or quality of the solution.

Another study that employs a filter is proposed in [24]. They solve the problem of feature correlation analysis in the classification process. The procedure uses the pearson correlation algorithm to determine the threshold for measuring each feature. The feature with the highest correlation has been designated to use in the classification process. The filter-based method is excellent in determining the relevancy of each feature with the target feature (class/label) using mathematical/statistical calculations. However, this method needs to determine the relationship between one feature and another, so no essential features are wasted. Knowing the relationship between features can affect the performance of the IDS model. Two or more features that have a significant effect when working together in detecting intrusions must be considered for selection as a feature set so that the proposed method does not remove essential features.

Aside from filters, another method for detecting IDS in anomaly detection is a wrapper, such as that proposed by Al-Yaseen et al [6]. This study employs a hybrid method in which differential evolution (DE) selects the most optimal features. Extreme learning machine (ELM) is used to generate the chosen features. For the evaluation, they employ NSL-KDD to evaluate the impact of their proposed method in the IDS model. The study shows that it can eliminate

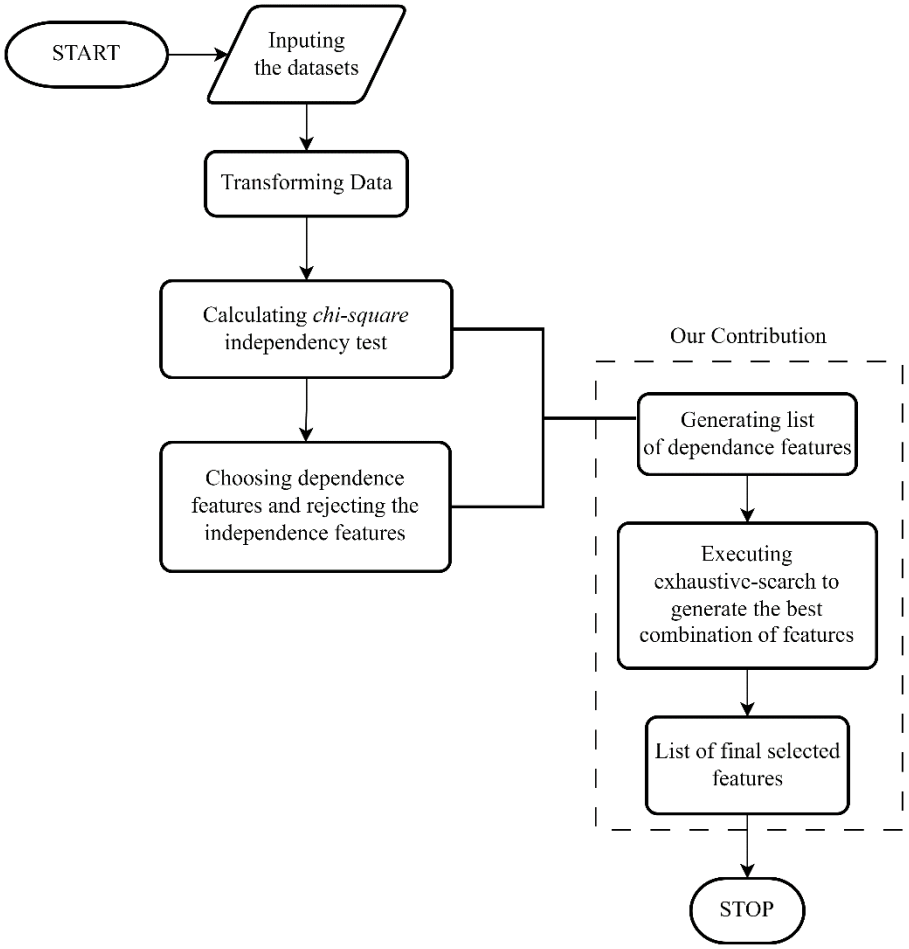

Figure. 1 Feature selection model

irrelevant features, optimise effectiveness, and optimise accuracy while dropping false alarm rates. It succeeds in improving performance. However, those studies did not consider the state of an irrelevant feature that becomes relevant when combined with other features. This necessitates the development of a method for determining the most optimal subset of relevant features in the IDS model.

Those flaws became our motivation for conducting this research. Our contribution to this study is to improve attack detection performance in the IDS model while determining the most optimal feature subset by combining filter and wrapper approaches. The purpose is to generate a feature set that works well with other features in improving the performance of intrusion detection on computer networks, regardless it is continuous or categorical features. The Chi-square independence test is used in this hybrid method to eliminate features with a very high level of independence from the target class, and features that are not eliminated will enter the second selection stage using an exhaustive approach. In the testing process, the combination subset features with the highest accuracy in the training process will be chosen to become a set of features, which will then proceed to the classification process. We also put the method through its paces with four publicly available datasets (Kyoto 2006 [12], KDD Cup99 [13], NSL-KDD [14], and UNSW-NB15 [15]) and three commonly used classification methods: Support vector machine (SVM), decision tree (DT), and Naïve Bayes (NB).

3. The proposed method

In this section, we describe the proposed hybrid model for intrusion detection using a Chi-square dependency test and an exhaustive search algorithm. First, we collect data from publicly available datasets widely used in other studies. The Chi-square independence test is used to calculate the level of independence of two categorical features, which results in a set of dependent features on the target class and rejects independent features of the that class. We use an exhaustive approach to find the best combination of the subset feature. Fig. 1 and Algorithm 1 depict the flow and pseudocode of the proposed method, respectively. The following section provides a more detailed explanation.

Algorithm 1. Chi-square Independence test and exhaustive search

Input:
data = csv dataset

//Algorithm:
Applying 10-vold-cross-validation
Tr = Training set of dataset
Ts = Testing set of dataset
Initialize generateFeatures = {f1, f2…, fn}
For each feature {f} in training set,
State null hypothesis and alternative hypothesis
Apply chi-square independence test (γ^2)
Compute *p-value*
Accept or reject {f}
Input to SubsetFeatures
End of foreach

//exhaustive search
Initialize combFeatures = {f1, f1 f2, ….)
Initialize accuracy1 = 0
For each {f} in combFeatures
Compute accuracy
Input to accuracy2
If (accuracy2 > accuracy1) **then**
accuracy1 = accuracy2
selectedFeatures = obtain combination of features with highest accuracy
end if
end foreach
display the subset of selectedFeatures

For every training set,
Train the data using SVM/DT/NB
compute confusion matrix
End For

3.1 Pre-processing

Data pre-processing is the first step before beginning the classification process. This stage is critical in this study because the dataset has a relatively sizeable dimensional scale, repetitive features, missing values, and attributes irrelevant to the detection process. So, before the classification process, we clean the data to remove any missing values from the dataset. Following the description of each feature, it is found that the data must be normalised before the classification process can begin. The data was then normalised using the Z-score normalisation method.

3.2 Feature selection

This study proposes a feature-based selection method for selecting relevant features, with criteria for optimal features. The first stage is the Chi-square independence test, which Karl Pearson first issued. Here, Chi-square is to select features using statistical theory to test the independence of a term with its target category or class. This procedure eliminates the most likely independence and irrelevant attributes for classification. From this method, we generate only the dependence features. The Chi-square statistical test is calculated using Eq. (1). Here, N is the total number of datasets, A is the number of times the observed feature and the target class label appear together, B is the number of times the observed feature appears without a target class, C is the number of times the target class appears without features, D is the amount of time the target class and features did not appear.

$$CHI^2(f,c) = \left[\frac{N\times(AD+CB)^2}{(A+C)(B+D)(A+B)(C+D)}\right] \quad (1)$$

Step 1: Determining hypothesis 0 and alternative hypotheses. Null hypothesis (H_0) means two attributes are unrelated. Another hypothesis (H_1) suggests two variables are related. We need to decide whether null hypothesis is accepted or rejected. Accepted means two variables are independent, and the feature is eliminated.

Step 2: After calculating each feature's dependency level, we created a contingency table displaying the distribution of one parameter in rows and another in columns. First, we need to calculate the degree of freedom using Eq. (2).

$$df = (r-1)(c-1) \quad (2)$$

Table 1 shows the form of a contingency table, where T stands for Total, TA for the total value of specific a column or attribute, and TR for the total value of a specific row.

Step 3: The following process determines the expected value using Eqs. (3) and (4). From these equations, we get the value of A from the total value in Table 1, marked with the variable TA. At the same time, B is the total value marked with TR, and O is the observed value. Simply, Eq. (4) is transformed into Eq. (5). We build contingency tables of expected values, whose results are shown in Table 2.

$$P(A \cap B) = P(A) \times P(B) \quad (3)$$

$$E_{ij} = n \times p \quad (4)$$

Table 1. Contingency table of observed values

	Col. 1	Col. 2	…	Col. *j*	Total
Row 1	O_{11}	O_{12}	…	O_{1j}	*TR*1
Row 2	O_{21}	O_{22}	…	O_{2j}	*TR*2
……	….	….	…	….	….
Row *i*	O_{i1}	O_{i2}	…	O_{ij}	*TRi*
Total	*TA1*	*TA2*	...	*TAj*	*T*

Table 2. Contingency table of expected values

	Attribute 1	…..	Attribute *j*
Row 1	E_{11}	….	E_{1j}
…..	….	….	….
Row *i*	E_{i1}	….	E_{ij}

$$E_{ij} = \frac{TA_j \times TB_i}{T^2} \quad (5)$$

The E represents the expected value of each feature calculated by multiplying the total row and column. Finally, the result is divided by the overall total.

Step 4: After all expected values have been obtained, we can build the required table using Eq. (6). The χ^2_{df} is the Chi-square score of each feature, while O is the observed value from Table 1, and E is the expected value obtained from Table 2.

$$\chi^2_{df} = \sum \frac{(O_i - E_i)^2}{E_i} \quad (6)$$

In choosing features, we intend to select those relying on the outcome. The considered count is close to the one expected if those features do not relate to each other (independence); this causes their Chi-square value to be relatively low. Consequently, a high value indicates that the predicted independence is incorrect. It can be inferred that features with higher dependency on both response and Chi-square values will be the model training.

Step 5: Finding and comparing the critical value level to our $chi - squared$ test statistic value from the distribution table. We compare the value of *df* and the α (significance value) using the Chi-square distribution table. In this study, we use a significance level of 0.05 because, according to the study [4], this level could achieve better performance.

Step 6: Accepting or rejecting the null hypothesis. Based on the Chi-square distribution table, if the χ value of the feature has a score above the statistical value, it falls into the null hypothesis rejecting area. That means H0 is rejected, which indicates H1 is accepted. Then the feature will be accepted as a list of selected features. Conversely, if the feature score falls in the null hypothesis accept area. The feature is not selected. Suppose the Chi-square value of certain features is higher than the Chi-square statistic value from the distribution table. In that case, the feature is accepted or rejected. We get the list of features dependent on the target class from the previous process, which generates the list of relevant features based on Chi-square. It is worth noting that they are not the final list. The following stage combines one feature with another and search for the best optimal features among all possibility combinations. An exhaustive search algorithm directly processes the list of those features.

3.3 Exhaustive approach

The feature selection process uses exhaustive aims to find relationships between features. This method is used to avoid the possibility of deleting features that are irrelevant when alone but is becoming important when combined with other features. The exhaustive process is done by comparing the accuracy values of 1 to n-feature combinations. The feature subset with the best accuracy will be selected as a feature subset and proceed to the classification process. Fig. 2 shows the feature selection process using exhaustive. It illustrates the exhaustive approach model. For a better understanding, Algorithm 1 shows the pseudocode of the proposed method.

Table 3. List of selected features

Dataset	**Number**	**Number of Features**
Kyoto 2006+	10	2, 4, 5, 6, 9, 10, 14, 16, 17, 18
KDD Cup99	11	3, 4, 6, 11, 13, 18, 22, 23, 36, 37, 39
NSL-KDD	14	2, 3, 6, 8, 10, 11, 13, 23, 27, 30, 32, 35, 36, 39
UNSW-NB15	11	6, 10, 11, 19, 20, 27, 34, 37, 42, 44, 46

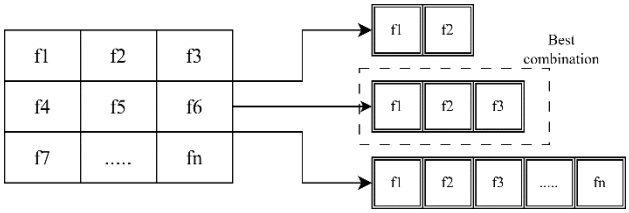

Figure. 2 Exhaustive approaches to select the best combinations of subset features

4. Experimental results

This section describes the experiment scenario and the discussion of the outcomes. The proposed method is implemented in Jupyter Notebook using the *sci-kit-learn* library. All work is completed on a computer with an Intel(R) Core i5-7200U processor, 12GB of RAM, and 1TB of storage. Python 3 is an environment for implementing the proposed IDS model using a variety of available libraries such as *NumPy*, *pandas*, and *sci-kit-learn.*

4.1 Datasets

This study aims to analyse the significant effect of the proposed method on various conditions. That is why this research uses four datasets. We employ 25,000 records from KDDCup-10% in the KDD Cup99 dataset, which includes 41 features and one label class. Tavallaee et al. [13] released NSL-KDD, the most recent version of KDD Cup99. This dataset contains 43 features and 41 connection records; the other two are attack/normal labels and scores. The records used in this study are 25,192 lines from KDDTrain 20%.

Kyoto 2006+ is the following dataset, which has 24 features. In this study, approximately 25,000 instances are used. The UNSW-NB15 [15] was the final dataset we used. The training set contains 25,000 records out of a total of 2,218,761 data records. The UNSW-NB15 has 49 characteristics. The 10-fold cross-validation is employed to divide the four datasets used in this study, 70% for the training set and 30% for the testing set.

4.2 Classification method

The classification process in this study employs three common classifiers: Support vector machine (SVM), decision tree (DT), and Naïve Bayes (NB). We tested three classifications to examine the proposed method's impact on various classification techniques.

4.3 Metrics

The IDS model can be evaluated using one of two criteria: efficiency or effectiveness [25]. Efficiency refers to the best use of resources, such as RAM or processing time. Energy is measured by performance-related measures such as accuracy, precision, recall, and misdiagnosis. The level of effectiveness of a method was used to evaluate studies in the present research. This is demonstrated by changes in the IDS's detection performance. As a result, we employ a confusion matrix to assess the method proposed in this study.

The confusion matrix is determined by counting the number of correct and false detections. The attack activity successfully identified as an attack is recorded as a true positive (TP). The number of normal occurrences the IDS model successfully identified as normal is true negative (TN). The number of normal events mistakenly labelled attacks is known as the false positive (FP). False negative (FN) is an attack that is incorrectly classified as normal activity.

Based on the explanation above, we evaluate the IDS model using four main criteria:

- Accuracy: the proportion of correct detections among all detections made. The accuracy formula is represented in Eq. (7).
- Precision: the number of positive detections out of all optimistic predictions as in Eq. (8).
- Recall: the proportion of positive detections in the whole data set, described in Eq. (9).
- F-score: a weighted average precision and recall comparison, provided in Eq. (10).

$$Accuracy(\%) = \frac{TP+TN}{TP+FP+TN+FN} \quad (7)$$

$$Precision(\%) = \frac{TP}{TP+FP} \quad (8)$$

$$Recall(\%) = \frac{TP}{TP+FN} \quad (9)$$

$$F-Score(\%) = \frac{2\times(Recall\times Precision)}{Recall+Precision} \quad (10)$$

4.4 Results

The outcomes of this study are divided into four sections. Table 3 describes the features chosen after

Table 4. Experiment result using SVM

Dataset	Kyoto 2006+ (%)	KDD Cup99 (%)	NSL-KDD (%)	UNSW-NB15 (%)
Method	Acc | Prec | Rec | F-Sc | FPR	Acc | Prec | Rec | F-Sc | FPR	Acc | Prec | Rec | F-Sc | FPR	Acc | Prec | Rec | F-Sc | FPR
No Feature Selection	92.92 | 92.91 | 93.41 | 93.13 | 10.17	73.81 | 73.01 | 69.08 | 70.99 | 25.12	91.4 | 90.02 | 88.98 | 89.46 | 9.75	91.02 | 91.9 | 90.03 | 90.96 | 8.41
Using Chi-Square	96.32 | 95.22 | 96.13 | 95.67 | 3.78	86.12 | 88.73 | 87.23 | 87.97 | 4.34	93.56 | 90.12 | 91.65 | 90.88 | 6.23	96.12 | 96.01 | 96.88 | 96.44 | 3.93
Proposed Method	96.67 | 96.12 | 96.24 | 96.18 | 1.23	91.02 | 89.12 | 87.01 | 88.05 | 1.67	96.78 | 94.12 | 94.72 | 94.42 | 2.43	98.12 | 96.02 | 97.23 | 96.62 | 1.56

Note: Acc: Accuracy, Prec: Precision, Rec: Recall, F-Sc: F-Score, FPR: False Positive Rate

Table 5. Experiment result using DT

Dataset	Kyoto 2006+ (%)	KDD Cup99 (%)	NSL-KDD (%)	UNSW-NB15 (%)
Method	Acc | Prec | Rec | F-Sc | FPR	Acc | Prec | Rec | F-Sc | FPR	Acc | Prec | Rec | F-Sc | FPR	Acc | Prec | Rec | F-Sc | FPR
No Feature Selection	95.51 | 91.3 | 95.91 | 93.56 | 2.45	73.98 | 74.32 | 71.42 | 72.84 | 15.12	92.37 | 91.43 | 91.71 | 91.57 | 6.71	83.17 | 82.66 | 85.22 | 83.92 | 13.21
Using Chi-Square	95.12 | 92.12 | 95.91 | 93.98 | 4.78	80.21 | 81.37 | 78.32 | 79.81 | 10.34	93.61 | 91.81 | 92.18 | 92.14 | 5.66	90.09 | 90.44 | 91.78 | 91.11 | 8.32
Proposed Method	94.33 | 95.26 | 95.13 | 95.21 | 5.32	89.22 | 89.76 | 87.25 | 88.49 | 10.09	92.56 | 91.21 | 92.57 | 91.88 | 5.71	96.12 | 97.13 | 96.91 | 97.02 | 3.12

Note: Acc: Accuracy, Prec: Precision, Rec: Recall, F-Sc: F-Score, FPR: False Positive Rate

each dataset's feature selection process. Tables 4, 5, and 6 show the effect of the method on the SVM, DT, and NB classification methods, respectively. Table 7 compares our current study to our previous study published in [10] and the state-of-the-art methods.

Table 3 shows the number of selected features and the list of selected features. According to that table, the proposed method successfully removed irrelevant features and set a specific number of features. Kyoto 2006+ includes ten features: service, *Destination bytes*, *Count*, *Samesrvrate*, *Dsthostcount*, *Dsthostsrvcount*, *Flag*, *Malware detection*, *Label*, and *SourceIPAddress*. They could generate sequentially 11, 14, and 11 features for other datasets such as KDD Cup99, NSL-KDD, and UNSW-NB15. The proposed method also eliminates repeated features, such as those found in Kyoto 2006+; duration and start time.

Table 4 displays the SVM classification test results for the four datasets. The accuracy, precision, recall, and F-score of the IDS model after using the feature selection method proposed for each dataset are the highest. The proposed feature selection method increased the accuracy of Kyoto 2006+ by 3.75%, KDD Cup99 by 4.92%, NSL-KDD by 5.38%, and UNSW-NB15 by 7.10%. The FPR rate also fell significantly by 11.4%, 23.45%, 7.32%, and 6.85% in Kyoto 2006+, KDD Cup99, NSL-KDD, and UNSW-NB15, respectively. This occurs because these datasets have significant data redundancy and missing values at the outset, affecting network detection performance. The proposed feature selection method has proven effective in reducing data redundancy, eliminating missing values, removing irrelevant features, and, most importantly, retaining elements that have played an essential role in the classification process. In the SVM classification method, the proposed method significantly improves the performance of the IDS model on all datasets.

Table 5 shows the test results using the DT classification. The IDS model's performance has generally improved, though not significantly. The accuracy increases by 15.24% in the KDD Cup99 dataset. The accuracy of the NSL-KDD and UNSW-NB15 datasets was enhanced by 0.13% and 12.85%, respectively. In contrast, there was a decrease in performance in the Kyoto 2006+ dataset, with values for accuracy, precision, recall, and F-score decreasing by 1.18%, 2.29%, 0.77%, and 3.28%, respectively. This reduction occurred as the value of the false alarm rate increased. The Kyoto 2006+ dataset may have experienced a decline because of relatively large data outliers [26]. Meanwhile, we still

Table 6. Experiment result using Naïve Bayes

Dataset	Kyoto 2006+ (%)	KDD Cup99 (%)	NSL-KDD (%)	UNSW-NB15 (%)
Method	**Acc | Prec | Rec | F-Sc | FPR**	**Acc | Prec | Rec | F-Sc | FPR**	**Acc | Prec | Rec | F-Sc | FPR**	**Acc | Prec | Rec | F-Sc | FPR**
No Feature Selection	80.12 | 84.32 | 82.32 | 83.31 | 9.71	71.12 | 74.01 | 75.23 | 74.62 | 12.32	92.23 | 90.12 | 90.02 | 90.07 | 6.21	66.81 | 67.86 | 71.02 | 69.49 | 27.12
Using Chi-Square	91.55 | 96.41 | 96.01 | 97.96 | 5.42	90.21 | 90.25 | 88.19 | 89.21 | 3.92	96.31 | 90.12 | 94.51 | 92.26 | 5.35	94.12 | 95.01 | 94.02 | 94.51 | 4.62
Proposed Method	92.43 | 94.12 | 95.24 | 94.68 | 1.23	90.23 | 89.12 | 90.01 | 89.56 | 1.98	96.41 | 90.23 | 94.45 | 92.38 | 5.35	98.21 | 96.02 | 97.23 | 96.62 | 1.56

Note: Acc: Accuracy, Prec: Precision, Rec: Recall, F-Sc: F-Score, FPR: False Positive Rate

Table 7. Comparison with other methods

Method	Features	Accuracy
CHI2CV [10]	29	96.70%
Pigeon Inspired optimizer [18]	18	86.90%
Wrapper based on extreme learning [6]	9	87.70%
Binary Grey Wolf Optimisation [17]	19	87.46%
Hybrid Firefly with Mutation Operator [19]	9	96.51%
The Proposed Method	14	96.78%

need to remove data outliers in the previous pre-processing step. This demonstrates that the feature selection method we propose can improve performance on several datasets when using the DT classification method; however, data must still be normalised and outliers removed so that training and testing data are more prepared and effective when entering the classification process.

The results of the Naïve Bayes classification method are shown in Table 6. The proposed method can significantly improve the IDS model's performance. This is demonstrated by increasing each metric in a variety of datasets. The accuracy increased by 12.31% in the Kyoto 2006+ dataset, while the false positive rate decreased by 8.48%. The same thing happened with the remaining datasets. The performance improvement when using the KDD Cup99 dataset is insignificant, but it does increase accuracy by 0.02%. The NSL-KDD dataset consistently performs whether all features are included without or after being selected.

Table 7 compares our proposed method with our previous research and the state-of-the-art methods, such as the PIO or optimizer inspired by the pigeon [18], the approach of extreme learning as part of the wrapper method [6], the novel approach of the grey wolf [17], and the last one is firefly optimization from [19]. The experimental results show that the method proposed in our most recent study improved the IDS model's performance. We can see in Table 7 that each method successfully reduced the number of features. Nevertheless, it does not mean that the lesser the number of features, the accuracy is more significant. Compared to other methods, the advantage of the proposed method is that it eliminates the most irrelevant feature regardless of whether it is a numeric or categorical feature. In contrast, other methods only focus on reducing dimensions, and the categorical features are mostly eliminated.

The proposed method reduces the features from 41 to 14. The selected features tend to be relevant and dependent on the class target. The accuracy result has proved it. Compared with the previous research, accuracy, precision, recall, and F-Score values of the proposed method increase by 1.5%, 0.87%, 2.08%, and 2.33%, respectively. The increase in accuracy occurs due to differences in the proposed methods. Our previous study in [10] only used the Chi-square method. Thus, the last method only eliminated features based on the mathematical score of the Chi-square test. This method has the disadvantage of not considering the relationships between characteristics. So, in our latest method, the feature selection is combined with an exhaustive search. That is where the performance results in accuracy can increase. Thus, the approach we propose is successful in improving accuracy performance.

This proposed method has the highest accuracy among all state-of-the-art methods, which is 97.78%. It differs significantly from the bio-inspired method, such as pigeon-inspired (PIO) [18] and grey wolf optimisation (GWO) [17]. The difference in accuracy between the proposed method and the PIO method is 9.88%. The deviation in accuracy is not much

different compared to the GWO method, which is 9.32%. This considerable difference can occur due to differentiations in the data normalization and transformation methods used and the method's accuracy in selecting the relevant features. Methods [17, 18] did not use a scoring system for each feature, so important features may be deleted because they are not included in the selected cluster. These results indicate that the proposed method can surpass the performance of the [17, 18] methods in terms of accuracy. It happens because the proposed method succeeds in selecting relevant features with the highest accuracy considerations from every possible combination. Meanwhile, the method [18] has succeeded in reducing the number of features relevant to the class, but there has been no consideration of the relationships between features that can affect accuracy.

Compared with [6], the proposed method has a different accuracy value of 9.08%. The features selected when using this extreme learning-based method are nine features. The next state-of-the-art method is a firefly-inspired [19] method with nine selected features. The accuracy results are not too far from the proposed method, with a difference of 0.27. The firefly method has advantages in the process. Features are selected by considering each solution to obtain the best solution. This process resembles the exhaustive method, but the Chi-square with exhaustive search has better accuracy because it combines with the mathematical calculation of the Chi-square test.

Based on the evaluation results, the proposed method can improve the IDS model's performance in distinguishing normal activities from attacks on computer network connection data. It also outperforms other methods in terms of accuracy.

5. Conclusion

Cyber-attacks are common in computer networks and on the internet. To alleviate cyber-attacks and be cautious of an attack, a machine learning model called IDS was developed that can be used to detect an attack on a computer network. Because of the large number of features in this IDS area, meaning the dataset has an immense dimensional scale, we propose a feature selection method in this study. This study aims to select only relevant features because not all available features are related to detecting attacks or normal.

This study proposes a hybrid approach involving Chi-square for eliminating irrelevant features and continuing to select the most optimal and effective subset of features to improve IDS detection performance using an exhaustive approach. We tested the method using four datasets and three classifiers. According to the test results, the proposed method can improve the performance of attack detection on computer networks. This study is a continuation and resolution to the problems identified in our previous study [10]. A comparison of these two methods reveals that our approach has a 0.08% increase in accuracy.

Even though we use several different datasets and classification methods, the proposed method has a positive impact on increasing detection performance. The proposed method also outperforms other state-of-the-art methods that we compared. This fact leads us to believe that it has improved computer network detection performance under various conditions and variables.

However, other issues have arisen. It indicates the need for an effective method of removing outliers or redundancy in some cases, such as using the Naive Bayes classification on the NSL-KDD dataset and the Decision Tree classification on the Kyoto 2006+ dataset. This will be our future contribution.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization, ATN and TA; methodology, ATN; software, ATN; validation, ATN and TA; formal analysis, ATN; writing—original draft preparation, ATN; writing—review and editing, TA; supervision, TA; project administration, TA; funding acquisition, TA

References

[1] P. C. Nguyen, Q. T. Nguyen, and K. H. Le, “An Ensemble Feature Selection Algorithm for Machine Learning based Intrusion Detection System”, In: *Proc. of the 2021 8th NAFOSTED Conference on Information and Computer Science*, pp. 50–54, 2021.

[2] S. Mohammadi, H. Mirvaziri, M. G. Ahsaee, and H. Karimipour, “Cyber Intrusion Detection by Combined Feature Selection Algorithm”, *Journal of Information Security and Applications*, Vol. 44, pp. 80–88, 2019.

[3] A. Heryanto, D. Stiawan, M. Y. B. Idris, M. R. Bahari, A. A. Hafizin, and R. Budiarto, “Cyberattack Feature Selection using Correlation-Based Feature Selection Method in an Intrusion Detection System”, In: *Proc. of the International Conference on Electrical*

Engineering, Computer Science and Informatics (EECSI, pp. 79–85, 2022.

[4] L. A. C. Ahakonye, C. I. Nwakanma, J. M. Lee, and D. S. Kim, "SCADA Intrusion Detection Scheme Exploiting The Fusion of Modified Decision Tree and Chi-square Feature Selection", *Internet of Things (Netherlands)*, Vol. 21, p. 100676, 2023.

[5] J. Maldonado, M. C. Riff, and B. Neveu, "A Review of Recent Approaches on Wrapper Feature Selection for Intrusion Detection", *Expert Systems with Applications*, Vol. 198. p. 116822, 2022.

[6] W. L. A. Yaseen, A. K. Idrees, and F. H. Almasoudy, "Wrapper Feature Selection Method based Differential Evolution and Extreme Learning Machine for Intrusion Detection System", *Pattern Recognition*, Vol. 132, p. 108912, 2022.

[7] B. Setiawan, S. Djanali, and T. Ahmad, "Increasing accuracy and completeness of intrusion detection model using fusion of normalization, feature selection method and support vector machine", *International Journal of Intelligent Engineering and Systems*, Vol. 12, No. 4, pp. 378–389, 2019, doi: 10.22266/ijies2019.0831.35.

[8] N. F. Syed, M. Ge, and Z. Baig, "Fog-cloud based Intrusion Detection System using Recurrent Neural Networks and Feature Selection for IoT Networks", *Computer Networks*, Vol. 225, p. 109662, 2023.

[9] P. Shunmugapriya and S. Kanmani, "A Hybrid Algorithm using Ant and Bee Colony Optimization for Feature Selection and Classification (AC-ABC Hybrid)", *Swarm and Evolutionary Computation*, Vol. 36, pp. 27–36, 2017.

[10] A. T. Nururrahmah and T. Ahmad, "CHI2CV : Feature Selection using Chi-Square with Cross-Validation for Intrusion Detection System", In: *Proc. of the ISDFS 2023 - 11th International Symposium on Digital Forensics and Security*, pp. 1–6, May 2023.

[11] K. Mnich and W. R. Rudnicki, "All-relevant feature selection using multidimensional filters with exhaustive search", *Information Sciences*, Vol. 524, pp. 277–297, 2020.

[12] Kyoto University, "Traffic Data from Kyoto University's Honeypots", *Homepage on The Internet*, 2006. https://www.takakura.com/Kyoto_data/ (accessed Jun. 10, 2023).

[13] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set", In: *Proc. of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada*, pp. 1–6, Dec. 2009.

[14] Canadian Institute for Cybersecurity & University of New Brunswick, "NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB", 2009. Accessed: Jun. 11, 2023. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html

[15] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)", In: *Proc. of the 2015 Military Communications and Information Systems Conference*, pp. 1–6, Dec. 2015.

[16] A. Sunyoto and Hanafi, "Enhance Intrusion Detection (IDS) System Using Deep SDAE to Increase Effectiveness of Dimensional Reduction in Machine Learning and Deep Learning", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 4, pp. 125–141, 2022, doi: 10.22266/ijies2022.0831.13.

[17] H. Almazini and K. R. K. Mahamud, "Grey Wolf Optimization Parameter Control for Feature Selection in Anomaly Detection", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 2, pp. 474–483, 2021, doi: 10.22266/ijies2021.0430.43.

[18] 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, p. 113249, 2020.

[19] K. M. Alwan, A. H. A. E. Atta, and H. H. Zayed, "Feature Selection Models Based on Hybrid Firefly Algorithm with Mutation Operator for Network Intrusion Detection", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 1, pp. 192–202, 2021, doi: 10.22266/ijies2021.0228.19.

[20] M. R. Aziz and A. S. Alfoudi, "Feature Selection of The Anomaly Network Intrusion Detection Based on Restoration Particle Swarm Optimization", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 5, pp. 592–600, 2022, doi: 10.22266/ijies2022.1031.51.

[21] A. S. Mahboob, M. R. O. Moghaddam, and S. Yousefi, "AOV-IDS: Arithmetic Optimizer with Voting classifier for Intrusion Detection System", In: *Proc. of the 2021 12th International Conference on Information and Knowledge Technology*, pp. 124–129, 2021.

[22] R. A. Disha and S. Waheed, "A Comparative study of machine learning models for Network Intrusion Detection System using UNSW-NB 15 dataset", In: *Proc. of the International Conference on Electronics, Communications and Information Technology*, pp. 1–5, 2021.

[23] H. Gharaee and H. Hosseinvand, "A new feature selection IDS based on genetic algorithm and SVM", In: *Proc. of the 2016 8th International Symposium on Telecommunications*, pp. 139–144, Mar. 2017.

[24] D. P. Hostiadi, Y. P. Atmojo, R. R. Huizen, I. M. D. Susila, G. A. Pradipta, and I. M. Liandana, "A New Approach Feature Selection for Intrusion Detection System Using Correlation Analysis", In: *Proc. of the 2022 4th International Conference on Cybernetics and Intelligent System (ICORIS)*, pp. 1–6, 2022.

[25] G. Kumar, "Evaluation metrics for intrusion detection systems-a study", *International Journal of Computer Science and Mobile Applications*, Vol. 2, No. 11, pp. 11–17, 2014.

[26] D. Protić, "Review of KDD Cup '99, NSL-KDD and Kyoto 2006+ datasets", *Vojnotehnicki Glasnik*, Vol. 66, No. 3, pp. 580–596, 2018.