Received: February 19, 2019

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Cluster Analysis-Based Approach Features Selection on Machine Learning for Detecting Intrusion

Mohammad Nasrul Aziz[1]* **Tohari Ahmad[1]**

Department of Informatics, Institut Teknologi Sepuluh Nopember,
Kampus ITS Surabaya, 60111, Indonesia
* Corresponding author's Email: nasrul.17051@mhs.its.ac.id

Abstract: Various machine learning technology approaches have been applied to intrusion detection system (IDS). To get optimal results, it needs to take several stages for processing the traffics. Among them is the feature selection method, where irrelevant and redundant features are removed. In the previous research, the system is developed based on feature grouping that used a clustering approach as the evaluation criteria. In this research, we propose a method for improving the performance of machine learning with the feature selection approach based on feature clustering. We propose cluster based feature selection derived from the value of mutual information and Pearson correlation. The cluster hierarchy is used in forming filters that are used to create selected and reduced clusters. In developing the cluster hierarchy, single, complete, and average linkage method are used to determine the formation of the best feature clusters. The classification method with Support Vector Machine (SVM), Naïve Bayes, and J48 decision tree are applied to observe the performance of the proposed feature selection. Based on the experimental results, we find that the highest accuracy (i.e., 99.842%) is obtained when a single linkage in the J48 classification is implemented in the Kyoto 2006 dataset.

Keywords: Intrusion detection, Data mining, Network security, Machine learning, Classification.

1. Introduction

In supporting the industrial revolution 4.0, network technology has been one of the most important components to support the creation of industrial automation and as information exchanging tools. However, the development of computer network technology also suffers from many threats relating to the network security. Various threats resulting from black hackers or malware can lead to serious attacks to the whole computer systems and information technology.

Intrusion Detection System (IDS) is a model which comprises both software and hardware to monitor activities on a computer network that produces a warning when there is an abnormal traffic and considered dangerous; then, it is sent to the management [1]. Based on its process, intrusion detection model is classified into two types, namely misuse/signature- and anomaly-based detection systems [2]. Misuse detection system can recognize a penetration by tapping a data packet and then compare it with the IDS rule stored in the database that includes a package of attack patterns. The principle of misuse-based IDS is as follows. If an incoming packet contains a pattern same as that stored in the database, then it is marked as an abnormal traffic or an attack. But, if it does not, the packet is considered as a normal traffic. On the other hand, the anomaly-based detection system evaluates an attack by observing unusual pattern or condition in the system. For example, a sharp increase of computing, memory usage, and a large number of parallel connections from specific IP sources. In this condition, the system can assume that an abnormal situation has occurred in the system. Therefore, it can be marked as an attack. Further action can be taken by the administrator based on the receiving alerts.

Anomaly detection system has an advantage in its application. That is, it is able to detect attacks that cannot be found in the database [3]. Based on the method to use in detecting the anomaly, IDS can be categorized into three groups: statistical, knowledge and machine learning methods. The implementation of this machine learning method can be done using a data mining approach to recognize the patterns of attacks. Some research on data mining for IDS have been carried out, including research conducted by [4]. It is a flow-based IDS using two machine learning methods: J48 Decision Tree and Multi-layer Perceptron (MLP). Another approach [5], which explores the use of feature selection, clustering, and feature transformation, is performed in both datasets: the 2006 NSLKDD and Kyoto 2006.

The successfulness of a data mining method for IDS is also determined by the quality of the training data. Good training data can be generated by performing pre-processing, for example selecting the relevant features. The method learns how to choose a collection of attributes or features for creating a model that describes the whole data. The purpose of feature selection itself is to reduce dimensions and the amount of data required for learning, remove irrelevant and redundant features, improve predictive accuracy of algorithms, and improve understanding of the built model [6].

In the research conducted by [7], feature selection can be categorized as follows. Based on the training data, feature selection is grouped into: supervised, unsupervised, and semi-supervised. Based on its relationship with the learning method, feature selection can be grouped into: filters, wrappers, and embedded models. Furthermore, according to the evaluation criteria, the selection of features can be determined by correlation, Euclidean distance, consistency, dependence, and size of information. Next, based on the search strategy, feature selection can be classified into forward increases, backward deletions, random, and hybrid models. Finally, the selection can be categorized based on the type of output. In this classification, feature selection can be categorized as rank (weighting) and subset selection models.

From those categories of feature selection, there have been many studies that discuss each of experiment and the development of feature selection methods. Among them are feature selection methods based on feature grouping that uses the cluster approach as a method of evaluation criteria [7]. Its goal is to create a group to select candidates of feature and choose one or more features from certain groups to represent them by calculating the value of mutual information from each feature pair. In addition, Chormunge and Jena [8] also use a feature selection approach based on cluster analysis. It is done by forming clusters with the k -means clustering. They combine clustering features and filter methods to solve dimensional problems and provide better performance than that of filter evaluation methods. The proposed grouping method finds the nature of features and removes irrelevant ones. Then, the process just chooses the relevant and not excessive features which are ranked based on their priorities. To test the accuracy of the method, they employ a percentage criterion. They also measure the correlation evaluation to eliminate the redundancy feature. Finally, the features are ranked in descending order.

The feature selection is done by using cluster analysis which observes some values according to the evaluation criteria proposed by [9], including the Pearson correlation coefficient and mutual information. In the previous research, the features of a cluster are found by selecting the most optimal features in each cluster. Determination of clusters tends to put optimal features in the same cluster. This can be a problem because, actually, only one of them will be taken. As the result, the representative features are only reduced features.

In this paper, we work on that problem by generating only two clusters. The first cluster is a collection of optimal and representative features; and the second one contains only non-optimal features. This second type cluster is to be reduced to get the better cluster. For this purpose, hierarchical clustering is used to build clusters that will separate the selected and reduced feature candidates. The formation of hierarchical clustering which uses several linkage methods can also influence cluster results. Some popular linkage methods are single, complete, and average linkage [9].

Based on the experiment, the proposed method which forms a new cluster model containing selected features increases the performance of the classification. This improvement is measured by evaluating the accuracy, sensitivity, and specificity for both the classifications with and without using feature selection. This has been significant to the overall performance over the existing feature selection methods.

This paper is divided into five sections. The first section is the background of the research. The next section is a review for the related works. In the third section, we explain the proposed method. The results of the experiment along with its comparisons to the existing methods is provided in section four.

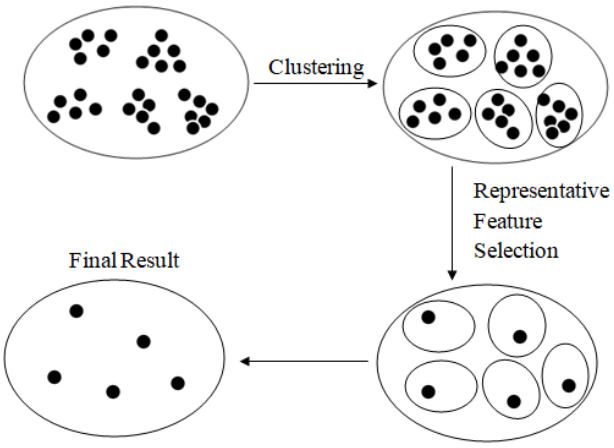


Figure. 1 Clustering-based feature selection (adapted from [13])

The last section is the conclusion and a possible future work based on this research.

2. Related works

There has been a lot of works doing research with a data mining approach in IDS. Their main focus is to improve the performance, including the false alarms. Moreover, feature selection has been one of the concerns. Several studies in anomaly detection with the IDS machine learning approach have been carried out, including [4]. In that research, a flow-based IDS uses two machine learning methods: J48 Decision Tree and Multi-layer perceptron (MLP). For the evaluation, this research uses the UNSWNB15 dataset. The experimental results show that the use of J48 has a better level of accuracy than MLP; that is, 91% and 98.5% respectively. Concerning the implementation of machine learning, its performance can still be improved by designing a pre-processing step. This additional step has been our concerns to be implemented in this research.

In the previous research conducted by [10], the performance of IDS is improved by implementing pre-processing to the data, including selecting relevant features. They do data normalization with the min-max method to get a new range of data. The next step is selecting features by using the correlation based feature selection method optimized with Particle Swarm Optimizer (CFS-PSO) to remove some features that are considered redundant. Its purpose is to increase the performance of classification from that without doing normalization and feature selection. The best results are obtained in the KDD Cup 99 dataset whose accuracy rate is 99.9291%.

Furthermore, pre-processing with data normalization and feature selection has been used in this study, including the implementation of CFS to find the optimum features of correlation evaluation. In our research, we compare the effectiveness between CFS-PSO and cluster-based optimization before selecting the suitable method.

Cluster analysis-based feature selection is done by finding the features after maximizing their diversity. In general, the grouping-based method classifies the selecting features by taking three main stages [11, 12] as described in Fig. 1 [13]. Those are: designing the structure of a feature space obtained from the domain of the distance of each feature; grouping features based on a clustering method; and the representing features of each cluster, which are selected to produce the selection results. Representation of features is often the most relevant in the class labels. Previously, clustering-based feature selection only takes one optimum feature in each cluster as shown in Fig. 1. So it is possible that the optimum features locate in the same cluster. Manipulating clusters need to be done to separate features within clusters that have high correlation values from those with low values. Therefore, at the end there are only 2 clusters available.

In other research [9], Fouedjio uses the k-means clustering method for feature selection. The first step is to find the value of the correlation evaluation from each feature. From this value, a cluster is formed based on the k-means algorithm. In the next step, the method eliminates features in each cluster that are redundant with the correlation filter. The problem of selecting features in the optimum cluster is a concern for this method. Further experiments with more features should be done to find the details of the performance of the feature selection approach. Another work that uses clustering for feature selection is presented in [7]. The research applies a clustering hierarchy based on the value of mutual information for each feature. From some generated clusters, the method recalculates mutual information from each of them. It aims at finding the best features to select. Based on this study, we combine it with the previous research in the proposed method. In this system, the cluster-based feature selection model employs the value of mutual information and Pearson correlations in the cluster hierarchy.

3. The proposed method

In this section, we present the proposed method, which consists of some stages: data collection, feature selection, classification, evaluation, and results analysis. Here, the feature selection has a cluster analysis approach.

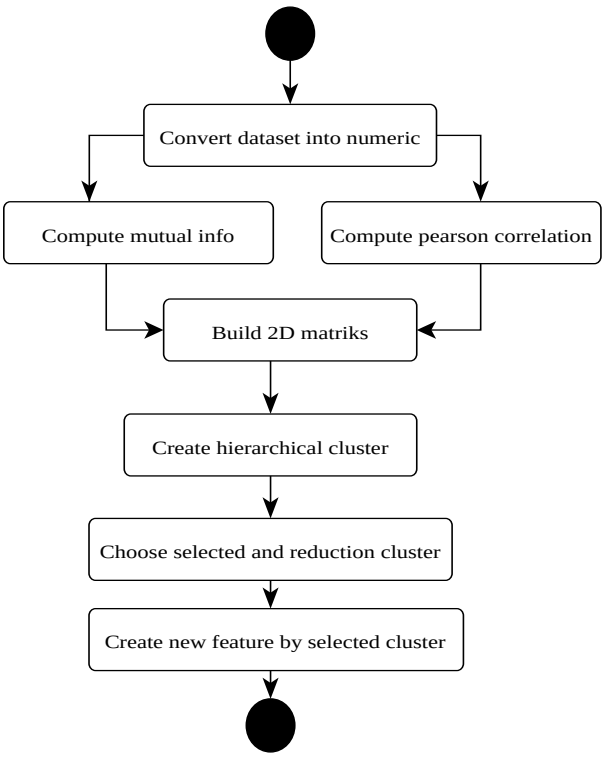


Figure. 2 Flow of feature selection

3.1 Feature selection using cluster analysis

Feature selection is one of the stages in the pre-processing of classification. It is done by selecting relevant features that affect the classification results. To get these relevant features, this research proposes a new approach by measuring evaluation and clustering analysis. This measuring evaluation is done by calculating the value of mutual information and also the Pearson correlation. Then, we form a tree with the Hierarchical Cluster approach to group selected features and those that will be reduced. For more details, Fig. 2 describes the flow of the proposed method. From this figure, it can be seen that the initial stage of the selection is to change non-numeric features to the numeric ones in the dataset.

The next step is to calculate the feature evaluation value by using that of Pearson correlation and mutual information. The Pearson correlation is used to calculate the relationship between two variables: independent and dependent variables. The relationship between those two variables occurs when there is a change in one of them that can trigger changes to the other. Calculation of Pearson correlation can be done by using Eq. (1) [13], where the range of correlation (r) has a value between -1 and 1. Here, $r\ (a, b)$ is a correlation between feature a and feature b, i is the loop containing the number of features. This range of i depends on the total number of features being used in the dataset.

Mutual information is a feature correlation discrimination method. In its application for selecting the features, mutual information is used to calculate the level of correlation between feature and label of the class [14]. Mutual information can be calculated by using Eq. (2)[15].

$$r\ (a,b) = \frac{N\ \sum a_i b_i - \sum a_i\ \sum b_i}{\sqrt{N\sum a_i^2 - (\sum a_i)^2} - \sqrt{N\sum b_i^2 - (\sum b_i)^2}} \quad (1)$$

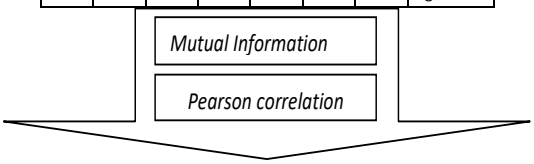
$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \quad (2)$$

In that previous formula, $p(x, y)$ is a joint probability distribution function X and Y, $p(x)$ and $p(y)$ is a marginal probability distribution function for X and Y, respectively.

After calculating the value of both mutual information and Pearson correlation of each feature, the next step is to form a two-dimensional dataset by using the mutual information (MI) value and Pearson correlation (PC) feature as shown in Fig. 2. The value of the calculation of MI and PC is then used to form a new dataset as provided in Fig. 3 where each row comprises the initial features to select. The new features that will be used as cluster attributes are the value of MI and PC.

Once the new dataset is generated, the next step is to build a grouping dataset with those two new features. In this grouping stage, the first step to do is to determine the formation of an $n \times n$ matrix as given in Fig. 4. It is developed according to the

F_1	F_2	F_3	F_4	F_5	F_6	F_n	Class
-	-	-	-	-	-	-	0
-	-	-	-	-	-	-	1
-	-	-	-	-	-	-	1
-	-	-	-	-	-	-	0



Feature	MI Value	PC Value
F_1	MI_1	PC_1
F_2	MI_2	PC_2
F_3	MI_3	PC_3
F_4	MI_4	PC_4
F_5	MI5	PC_5
F_6	MI_6	PC_6
F_n	MI_n	PC_n

Figure. 3 The formation of two-dimensional data

Euclidean distance (d) between MI_i and PC_i (see Eq. (3)) where p and q are respectively (MI_i, PC_i) and (MI_j, PC_j). It is shown that a matrix of $n \times n$ can be formed from the Euclidean distance value where we find the smallest value.

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2} \quad (3)$$

After getting the minimum value d, then it needs to calculate the linkage by comparing three linkage methods (i.e., single, complete, and average linkages). The linkage calculation is applied to that between d_{min} and other d values both vertically and horizontally. So, we get a new matrix whose size is $(n-1) \times (n-1)$. By assuming that $d_{min} = d_{12}$, the new matrix can be described in Fig. 5 that its size is $(n-1) \times (n-1)$. From the formation of this new matrix, we need to look for the minimum linkage and update the existing matrix until the cluster is too far to be merged or when there are only a small number of cluster numbers.

In this research, we use three linkage approaches. Firstly, the single linkage approach which basically means that this method measures the distance between two clusters. This is done by finding the minimum distance (d) obtained from the closest distance between all vectors in cluster U to all vectors in cluster V. So, the minimum distance in the matrix D with single linkage can be calculated by the Eq. (4).

Figure. 3 The formation of two-dimensional data

$$d(U,V) = \min(d(U,V)); d(U,V) \in D \quad (4)$$

The complete linkage measures the distance between two clusters by using the maximum distance formula. In this method, the closeness between the two clusters is determined according to the farthest distance between cluster U and V. So,

$$\begin{bmatrix} 0 & d_{12} & d_{13} & d_{1n} \\ d_{12} & 0 & d_{23} & d_{2n} \\ d_{13} & d_{23} & 0 & d_{3n} \\ d_{1n} & d_{2n} & d_{3n} & 0 \end{bmatrix}$$

Figure. 4 The matrix $\boldsymbol{n} \times \boldsymbol{n}$ from the Euclidean distance value

$$\begin{bmatrix} 0 & d_{12-3} & d_{12-n} \\ d_{12-3} & 0 & d_{3-n} \\ d_{12-n} & d_{3-n} & 0 \end{bmatrix}$$

Figure. 5 The $n \times n$ matrix formation

Eq. (5) can be designed to calculate the maximum distance (d) in the complete linkage.

$$d(U,V) = \max d(U,V); d(U,V) \in D \quad (5)$$

The third approach is the average linkage, which the closeness between the two clusters is calculated based on the average distance between clusters U and V. For this purpose, we can write Eq. (6) to find the distance between two clusters in the average linkage method. In this method, n_U and n_V are the sums of data in cluster U and V, respectively.

$$d(U,V) = \frac{1}{n_U \times n_V} \sum d(U,V); d(U,V) \in D \quad (6)$$

The last step in this feature selection is to calculate the average value of mutual information and correlation of each cluster formed. Clusters that have the smallest average value are those whose features are reduced. The whole process can be drawn in Fig. 6 which illustrates the flow of the analysis of hierarchical cluster.

3.2 Classification

This stage is to test the data for the classification. Some scenarios are developed to test the capability of the method to select the features. The first classification uses Support Vector Machine (SVM) with linear kernels, which finds the best hyperplane to divide it into two classes and maximize the margin between those two classes.

Another classification testing is performed by using the Naive Bayes classification. This method has advantages in the classification process. That is, the method does not require big training data to form parameter estimation. This is because its independent variable is the variance of the variables in the class in the classification process. Furthermore, it is not the entire covariance matrix.

The testing model and classification comparison are divided into two scenarios. The first model uses the selection of features and the second model is without the selection. This scenario is to see feature selection performance and compare it to the existing classification. Overall, the presentation of this scenario can be found in Fig. 7.

4. Experimental results

To evaluate the method, we measure the accuracy, sensitivity, and specificity to get the classification performance. Firstly, the step taken is to form the confusion matrix as shown in Table 1.

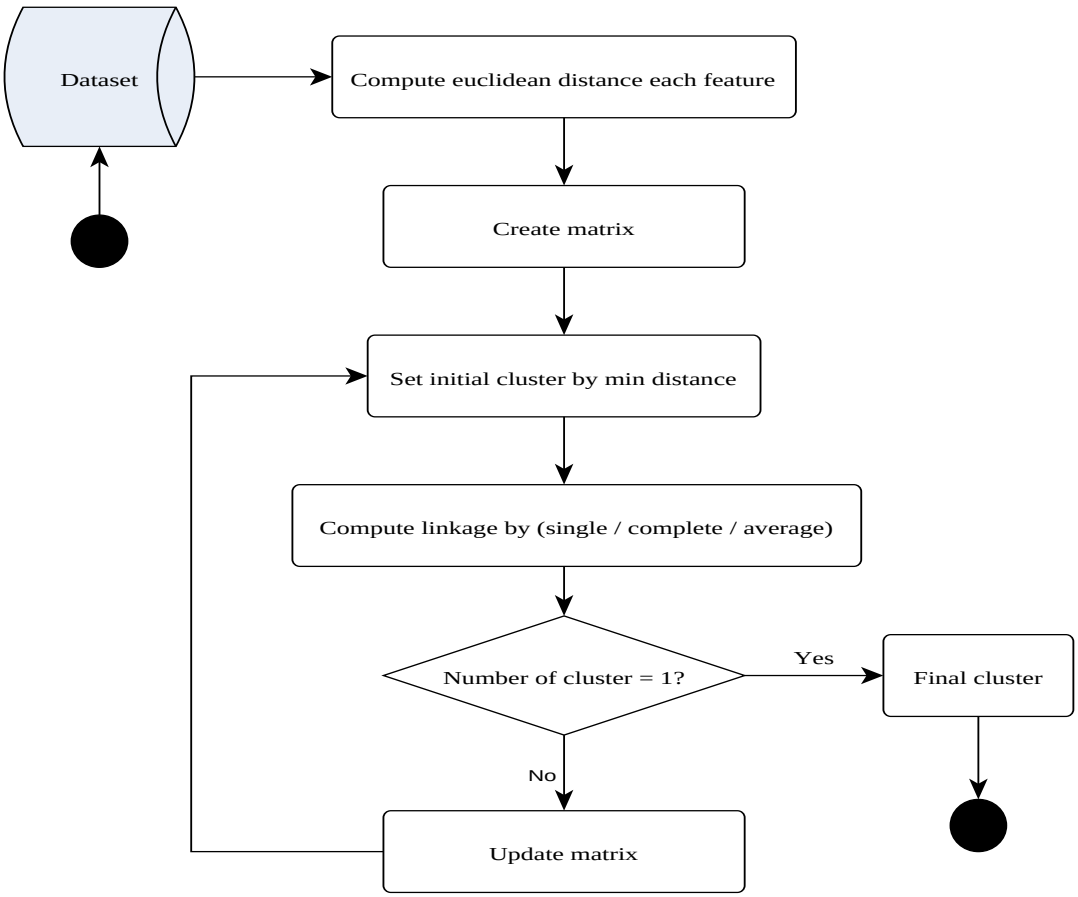


Figure. 6 Hierarchical cluster

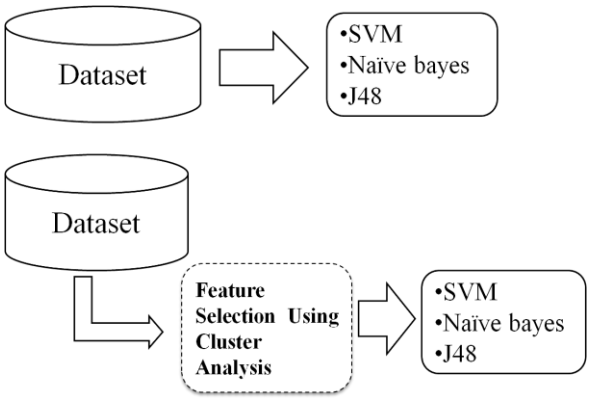


Figure. 7 Classification scenario

Table 1. Confusion matrix

		Prediction Class Results	
		Positive	Negative
Original Class	Positive	*True Positive (TP)*	*False Negative (FN)*
	Negative	*False Positive (FP)*	*True Negative (TN)*

In the classification testing, TP is a total of attack that is classified correctly as an attack. TN is the total of normal traffic that is successfully classified correctly as normal. FP is the total of normal traffic but the system detects it as an attack. FN is an attack activity, but the system marks it as a normal activity. From the result of that matrix, we can then calculate the value of the accuracy, sensitivity and specificity by using the Eqs. (7), (8), and (9) [16].

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

$$sen = \frac{TP}{TP+FN} \quad (8)$$

$$Spec = \frac{TN}{FP+TN} \quad (9)$$

The first dataset used is KDD Cup 99, which has 41 features and 2 class labels: normal and attack. In more details, this attack can be further classified into more specific ones. As provided in [17], it can be: Denial of Service Attack (DoS), User to Root Attack (U2R), Remote to Local Attack (R2L), and Probing Attack. Next, the features of KDD Cup 99 can be categorized into three groups [17]: basic, traffic, and content features.

Table 2. Experimental result with all features in KDD Cup 99

Classification	Acc (%)	Sen (%)	Spec (%)
SVM	99.595	99.6	99.9
Naïve Bayes	96.113	96.1	99.8
J48	99.311	99.3	99.9

Table 3. Experimental result with all features in Kyoto 2006

Classification	Acc (%)	Sen (%)	Spec (%)
SVM	99.547	99.5	99.8
Naïve Bayes	92.921	92.9	96.7
J48	99.842	99.8	93.7

Table 4. Experimental result using the proposed feature selection in KDD Cup 99 with single linkage

Linkage Method	Classification	Acc (%)	Sen (%)	Spec (%)
Single	SVM	99.676	99.7	99.9
Single	Naïve Bayes	97.570	97.6	99.9
Single	J48	99.433	99.4	99.9

Table 5. Experimental result using the proposed feature selection in Kyoto 2006 with single linkage

Linkage Method	Classification	Acc (%)	Sen (%)	Spec (%)
Single	SVM	99.552	99.6	99.8
Single	Naïve Bayes	94.816	94.8	95.3
Single	J48	99.842	99.8	93.6

Table 6. Selected features of KDD Cup 99 with single linkage

No.	Feature	No.	Feature
1	protocol_type	11	same_srv_rate
2	service	12	diff_srv_rate
3	flag	13	dst_host_count
4	src_bytes	14	dst_host_srv_count
5	dst_bytes	15	dst_host_same_ srv_rate
6	logged_in	16	dst_host_diff_srv_rate
7	count	17	dst_host_same_src_ port_rate
8	srv_count	18	dst_host_serror_rate
9	serror_rate	19	dst_host_srv_ serror_rate
10	srv_serror_rate		

The second dataset, which is used as the comparison, is Kyoto 2006. It has 14 features originated from the KDD cup 99 dataset. In addition, there are 10 additional features that can be used for analysis and evaluation of IDS networks.

The first experiment conducted in this research is carried out by implementing IDS applications using three classification methods: SVM, Naïve Bayes, and J48. The first experiment is performed the classification in KDD Cup 99 and Kyoto 2006 without implementing the proposed feature selection, whose results are provided in Tables 2 and 3, respectively.

From the results of this first experiment of KDD Cup 99, we see that the use of the classification method with SVM linear kernels has the highest level of accuracy that is 99.5951%. It is much higher than that of Naïve Bayes, but just slightly better than J48.

In the initial classification experiment on the Kyoto 2006 dataset, it is found that the J48 has a better degree of accuracy than both SVM and Naive Bayes, which is 99.842% with 99.8% and 93.7% of sensitivity and specificity, respectively. This is different from the experimental results obtained from KDD Cup 99.

4.1 Single linkage approach

The next experiment is by applying the proposed feature selection method using cluster analysis with Single Linkage. In the KDD Cup 99, we have 19 selected features which are then classified, whose results are provided in Table 4.

In that Table 4, we find that the use of the only selected features is able to increase the performance. In SVM and J48, there is a slight increase of accuracy; while in Naïve Bayes, the improvement is much higher, which is more than 1%, from 96.113% to 97.570%. Sensitivity also experiences the same improvement, and the specificity is stable, except for Naïve Bayes where it increases 0.1%.

In Table 5, we present the experimental results using the Kyoto 2006 dataset with a single linkage. In this data set, we are able to generate 9 out of 24 features. It is shown that overall, reducing features increases the performance, except for specificity generated by the Naïve Bayes and J48 algorithms, where it is slightly lower. Nevertheless, the accuracy and sensitivity significantly increase about 2%. It is considered to be a big improvement, considering that the previous value has been more than 90%.

In addition, the list of features for this single linkage method is provided in Tables 6 and 7 for KDD Cup 99 and Kyoto 2006, respectively. These tables comprise 19 and 9 selected features.

4.2 Complete linkage approach

In the next experiment, we use cluster formation

Table 7. Selected features of Kyoto 2006 with single, complete and average linkages

No.	Feature	No.	Feature
1	Service	6	Source_bytes
2	Destination_bytes	7	Count
3	Same_srv_rate	8	Dst_host_count
4	Dst_host_srv_count	9	Flag
5	Destination_Port_Number		

Table 8. Experimental result using the proposed feature selection in KDD Cup 99 with complete linkage

Linkage Method	Classification	Acc (%)	Sen (%)	Spec (%)
Complete	SVM	99.676	99.7	99.9
Complete	Naïve Bayes	97.733	97.7	99.9
Complete	J48	99.433	99.4	99.9

Table 9. Experimental result using the proposed feature selection in Kyoto 2006 with complete linkage

Linkage Method	Classification	Acc (%)	Sen (%)	Spec (%)
Complete	SVM	99.552	99.6	99.8
Complete	Naïve bayes	94.816	94.8	95.3
Complete	J48	99.842	99.8	93.6

Table 10. Selected features of KDD Cup 99 with complete linkage and average

No.	Feature	No.	Feature
1	protocol_type	10	srv_serror_rate
2	service	11	same_srv_rate
3	flag	12	diff_srv_rate
4	src_bytes	13	dst_host_srv_count
5	dst_bytes	14	dst_host_same_srv_rate
6	logged_in	15	dst_host_diff_srv_rate
7	count	16	dst_host_same_src_port_rate
8	srv_count	17	dst_host_serror_rate
9	serror_rate	18	dst_host_srv_serror_rate

based on the complete linkage. In the KDD Cup 99, we obtain 18 features, and the results of the experiments can be found in Table 8.

From Tables 4 and 8, we find that there is an increase of the accuracy and sensitivity which are generated by using the Naïve Bayes algorithm, from 97.570% to 97.733%, and from 97.6% to 97.7%, respectively. On the other hands, the other parameter values are stable. Also, compared to the results presented in Table 2 when all features are employed, this proposed method is superior.

In the Kyoto 2006 dataset, we see that the use of different linkage methods does not affect the performance, where the experimental results shown in Table 9 are exactly same with those in Table 5. Moreover, as previously described, the feature selection stage produces same number and type of features, as provided in Table 7. This characteristic is different from the KDD Cup 99 dataset, where the selected features of single and complete linkages are not same, as presented respectively in Tables 6 and 10. It is shown that there are 19 features in single, and 18 features in complete linkages.

4.3 Average linkage approach

The third experiment is to use the cluster formation with the average linkage. In the first case using the KDD Cup 99 dataset, the average linkage approach produces 18 selected features (see Table 10). The results of the experiment are presented in Table 11. It is shown that the performance is same with that of complete linkage provided in Table 8, which is slightly better than the performance of single linkage.

In the case of the Kyoto 2006 dataset, similar to the previous implementation of the complete linkage, the experimental results of the average linkage is also same with that of single linkage, as in Table 12. Also, the resulted selected features are same, as in Table 7. So, these features lead to the same performance.

We can summarize those experimental results in Figs. 8 and 9, which are developed based on the respective datasets: KDD Cup 99 and Kyoto 2006. From those two tables, in general we find that by using the selected features, the accuracy increases. The drastic improvement happens when the features are applied to Naïve Bayes; while on the other two methods, there is only slight rise. Nevertheless, the overall accuracy of those with Naïve Bayes are still

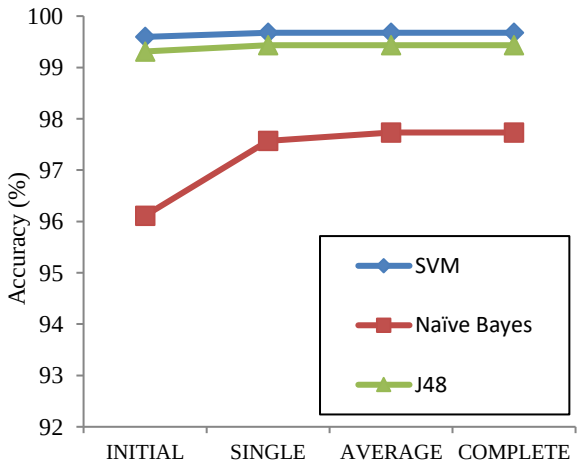


Figure. 8 Experimental result of the proposed selected features using various classifiers in KDD Cup 99

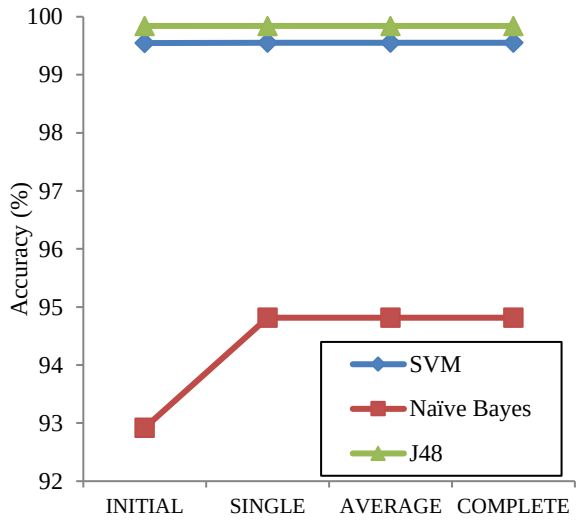


Figure. 9 Experimental result of the proposed selected features using various classifiers in Kyoto 2006

Table 11. Experimental result using the proposed feature selection in KDD Cup 99 with average linkage

Linkage Method	Classification	Acc (%)	Sen (%)	Spec (%)
Average	SVM	99.676	99.7	99.9
Average	Naïve bayes	97.733	97.7	99.9
Average	J48	99.433	99.4	99.9

Table 12. Experimental result using the proposed feature selection in Kyoto 2006 with average linkage

Linkage Method	Classification	Acc (%)	Sen (%)	Spec (%)
Average	SVM	99.552	99.6	78.7
Average	Naïve bayes	94.816	94.8	95.3
Average	J48	99.842	99.8	93.6

Table 13. Result of comparison existing method in KDD Cup 99

Method	Acc (%)	Sen (%)	Spec (%)
Info Gain [18]	99.433	99.4	99.9
CFS + Best First [19]	99.352	99.4	99.9
CFS + PSO [10]	99.514	99.5	99.9
K-Means + correlation [8]	98.706	98.7	99.6
Proposed method	99.676	99.7	99.9

Table 14. Result of comparison existing method in Kyoto 2006

Method	Acc (%)	Sen (%)	Spec (%)
Info Gain [18]	99.552	99.6	99.79
CFS + Best First [19]	99.524	99.5	99.79
CFS + PSO [10]	99.524	99.5	99.79
K-Means + correlation [8]	99.547	99.5	99.79
Proposed method	99.552	99.6	99.79

lower than the others. The selected features are more appropriate to use in KDD Cup 99 dataset; while J48 is in Kyoto 2006. The difference of the accuracy level between those two methods is actually insignificant, i.e. less than 1%. However, this low value may have a great impact on detecting the intrusion.

Overall, we find that the highest accuracy is obtained by the J48 classifier which is implemented in the Kyoto 2006 dataset by using either single, complete or average linkage. It can achieve 99.842% of accuracy, with 99.8% of sensitivity, and 93.6% of specificity. Whereas in the KDD 99 dataset, the highest accuracy is obtained by SVM, which is 99.676% with 99.7% of sensitivity and 99.9% of specificity by using either single, complete or average linkage.

The next step in our experiment is to compare the position of our proposed method with the existing feature selection method. For this purpose, we compare it with the Info Gain feature selection method [18], Corelation based selection feature with Best First (CFS-Best First) [19], Corelation based selection feature with Particle Swarm Optimization (CFS-PSO) [10], and the method proposed in [8]. We apply various feature selection and SVM classification to the two datasets whose experimental results are in Table 13 for the KDD Cup 99 dataset and Table 14 for the Kyoto 2006 dataset.

Overall, from Tables 13 and 14, we find that the proposed method has a better accuracy rate. The sensitivity and specificity of the proposed method also do not experience a decline or at least they have the same level of others. Specifically to the Kyoto 2006 dataset, our feature selection performance has the same accuracy, sensitivity, and specificity as those of info gain. The best accuracy of our method is at 99,676% with sensitivity 99.7% and specificity 99.9% in the KDD Cup 99 dataset. At the Kyoto 2006 rate, the accuracy is at 99,552% with sensitivity 99.6% and specificity 99.79%.

For more details, we present all comparisons of the accuracy by using two data sets in Fig. 10. We compare the feature selection method by applying SVM classifications to the linear kernel to see further accuracy. The results show that the best performance in the KDD Cup 99 dataset as shown in Fig. 10. In addition, our proposed method also shows superior performance in the two overall datasets.

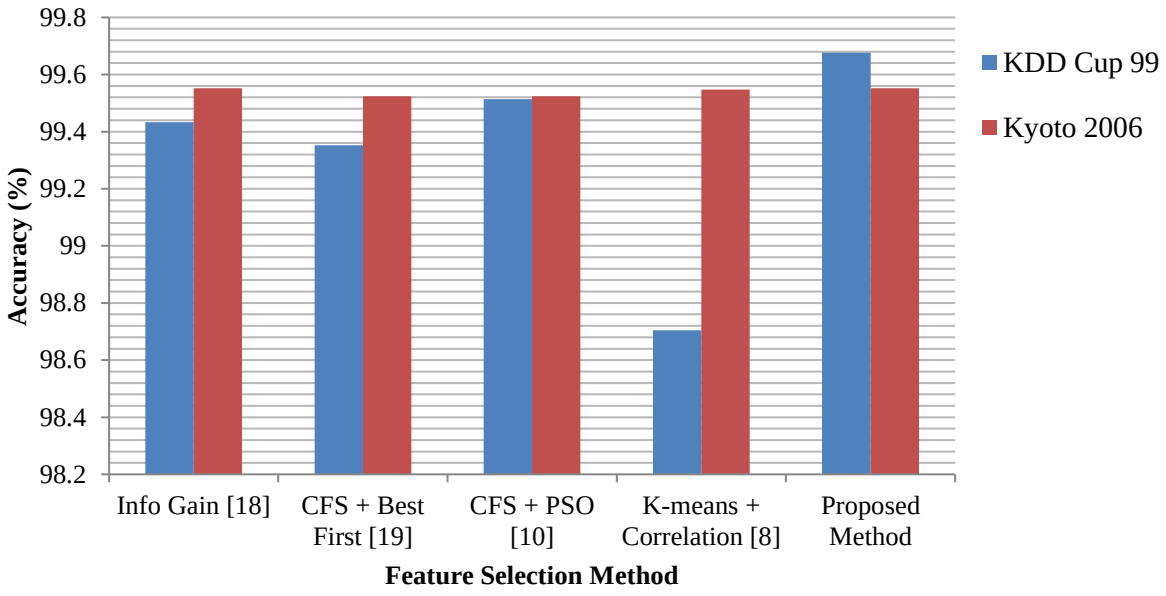


Figure. 10 Comparison of accuracy with existing methods

5. Conclusions

In its implementation, the performance of IDS is influenced by the quality of the incoming data which can be extracted. Low quality data may reduce the capability of IDS to correctly detect an attack. This condition can happen if there are many irrelevant data to process.

In this research, we do select the features of data which really have a good impact on the performance of IDS. This is done by employing a machine learning approach which is evaluated in the KDD Cup 99 and Kyoto 2006 datasets. In the cluster formation, single, complete, and average approaches are used. The selected features are then processed by machine learning-based classifiers.

The experimental results show that this feature reduction is able to improve the performance concerning the accuracy, sensitivity and specificity, regardless the dataset being used. Furthermore, different linkage approaches have an effect on the performance. The formation of clusters by forming two large clusters between clusters that have optimum and less optimum representation has an impact on the selected features. So, the better the features produced, the better the training data used will have an impact on the machine learning performance.

The increase in accuracy is shown by each experiment. The highest accuracy rate is obtained by using KDD Cup 99 obtained an accuracy value of 99.676% from 99.595 using the SVM classification. The results of experiments using the Kyoto 2006 dataset obtained an accuracy rate of 99.842% using the J48 classification. When comparing with existing methods, the proposed algorithm also has a better degree of accuracy with an increase between 0.5% and 1%.

In the next research, we will focus on cluster formation methods that are more optimal for producing the best features. It also needs to define an appropriate linkage approach for this purpose.

Acknowledgments

This work was supported by PasTi 2017 scholarship Kemenristekdikti and Department of Informatics Institut Teknologi Sepuluh Nopember.

References

[1] J. Jabez and B. Muthukumar, “Intrusion Detection System (IDS): Anomaly Detection Using Outlier Detection Approach”, *Procedia Computer Science*, Vol. 48, pp. 338–346, 2015.

[2] Akashdeep, I. Manzoor, and N. Kumar, “A Feature Reduced Intrusion Detection System Using ANN Classifier”, *Expert Systems with Applications*, Vol. 88, pp. 249–257, 2017.

[3] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, “Anomaly-Based Network Intrusion Detection: Techniques, Systems and Challenges”, *Computers & Security*, Vol. 28, No. 1, pp. 18–28, 2009.

[4] L. Van Efferen and A. M. T. Ali-Eldin, “A Multi-Layer Perceptron Approach For Flow-

Based Anomaly Detection", In: *Proc. of 2017 International Symposium on Networks, Computers and Communications*, pp. 1–6, 2017.

[5] I. Z. Muttaqien and T. Ahmad, "Increasing Performance of IDS by Selecting and Transforming Features", In: *Proc. of 2016 IEEE International Conference on Communication, Networks and Satellite*, pp. 85–90, 2016.

[6] H. Liu, E. R. Dougherty, J. G. Dy, K. Torkkola, E. Tuv, H. Peng, C. Ding, F. Long, M. Berens, H. Liu, L. Parsons, Z. Zhao, L. Yu, and G. Forman, "Evolving Feature Selection", *IEEE Intelligent Systems,* Vol. 20, No. 6, pp. 64–76, 2005.

[7] J. Song, Z. Zhu, and C. Price, "Feature Grouping for Intrusion Detection System Based on Hierarchical Clustering", In: Teufel S., Min T.A., You I., Weippl E. (eds) Availability, Reliability, and Security in Information Systems. CD-ARES 2014. Lecture Notes in Computer Science, Vol 8708, pp. 270–280, Springer, Cham, 2014.

[8] S. Chormunge and S. Jena, "Correlation Based Feature Selection with Clustering for High Dimensional Data", *Journal of Electrical Systems and Information Technology*, Vol. 5, No. 3, pp. 542–549, 2018.

[9] F. Fouedjio, "A Hierarchical Clustering Method for Multivariate Geostatistical Data", *Spatial Statistics*, Vol. 18, pp. 333–351, 2016.

[10] T. Ahmad and M. N. Aziz, "Data Preprocessing and Feature Selection for Machine Learning Intrusion Detection Systems", *ICIC Express Letter*, Vol. 13, No. 2, pp. 93–101, 2019.

[11] H. Liu, X. Wu, and S. Zhang, "Feature Selection Using Hierarchical Feature Clustering", In: *Proc. of the 20th ACM International Conference on Information and Knowledge Management*, pp. 979–984, 2011.

[12] D. M. Witten and R. Tibshirani, "A Framework for Feature Selection in Clustering", *Journal of the American Statistical Association*, Vol. 105, No. 490, pp. 713–726, 2010.

[13] J. Cai, J. Luo, S. Wang, and S. Yang, "Feature Selection in Machine Learning: A New Perspective", *Neurocomputing*, Vol. 300, pp. 70–79, 2018.

[14] L. Ting and Y. Qingsong, "Spam Feature Selection Based on The Improved Mutual Information Algorithm", In: *Proc. of 2012 Fourth International Conference on Multimedia Information Networking and Security*, Nanjing, China, pp. 67–70, 2012.

[15] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "MIFS-ND: A Mutual Information-Based Feature Selection Method", *Expert Systems with Application*s, Vol. 41, No. 14, pp. 6371–6385, 2014.

[16] C. R. Ramesh, "Fuzzy Clustering Algorithm Efficient Implementation Using Centre of Centres", *International Journal of Intelligent Engineering and Systems*, Vol. 11, No. 5, pp. 1–10, 2018.

[17] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "A Detailed Analysis of The KDD CUP 99 Data Set", In: *Proc. of 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, Canada, pp. 1–6, 2009.

[18] F. He, H. Yang, Y. Miao, and R. Louis, "A Hybrid Feature Selection Method Based on Genetic Algorithm and Information Gain", In: *Proc. of 2016 5th International Conference on Computer Science and Network Technology*, pp. 320–323, 2016.

[19] Z. Zhang, D. Wang, and J. Hu, "Cardiovascular Risk Prediction Method Based on CFS Subset Evaluation and Random Forest Classification Framework", In: *Proc. of 2017 IEEE 2nd International Conference on Big Data Analysis*, pp. 228–232, 2017.