Received: June 29, 2020. Revised: August 11, 2020.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Data Reduction for Optimizing Feature Selection in Modeling Intrusion Detection System

Alif Nur Iman[1] **Tohari Ahmad[1]***

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

Abstract: With the development and ease of access to internet networks, the potential for attacks and intrusions have increased. The intrusion detection system (IDS), an approach to overcome this problem, is grouped into two models: signature-based and anomaly-based. An anomaly-based IDS can be implemented by machine learning; one of the schemes in machine learning is data reduction. IDS datasets are usually obtained through a real-time process that has undefined proportional data. The purpose of data reduction is to speed up and optimize the process, improving accuracy, precision, and specifications. There are several methods to perform data reduction, one of which uses outlier detection techniques. Proper outlier detection has a positive impact on improving the classification results of machine learning. In this research, the outlier detection is done by a circle generated from the *k* -means clustering of all selected features. Two scenarios are designed for the evaluation: a circle generated from two points of the minimum and maximum cluster and median of all clusters. The formation of clusters conducted by *k* -means clustering determines the size and direction of the outlier circle so that it dynamically adjusts the distribution of data from the feature selection results. By employing the previous feature selection algorithms, the comparison is performed to evaluate the proposed method's performance. Our empirical results show that the second scenario can significantly improve the classification results in terms of accuracy, detection rate, and precision. The first and second experiments can increase the accuracy by 0.02%, and the third experiment is by 0.1%. The detection rate in the first, second, and third experiments increases by 0.01%, 0.02%, and 0.07. At the same time, precision increases by 0.04%, 0.02%, and 0.01%, correspondingly.

Keywords: Data reduction, Intrusion detection system, K-means clustering, Machine learning, Network security.

1. Introduction

In the development of the internet, aside from providing easy access to information, it also harms data security. A proper security mechanism plays an essential role in detecting or even preventing attacks. Intrusion Detection System (IDS) is an approach that is currently popular for securing computer networks. IDS works defensively to identify whether access is a normal or an attack. Research on IDS has been developed to obtain the most optimal intrusion detection model. In general, it is grouped into two types: signature-based and anomaly-based IDS. The anomaly-based model is often constructed by machine learning, which calculates accuracy, precision, and specifications for its standard evaluation metrics.

Input data are a crucial factor in producing a decent classification. Based on research surveys conducted in [1-3], the progress of optimizing the intrusion detection models with the concept of machine learning requires a reliable dataset. Malowidzki et al. [1] argue that the lack of a proper dataset for research causes difficulty in evaluating methods and comparing the performance of research results. Ring et al. [2] identify that establishing a good IDS model requires a proportional, balanced dataset with a clear label. The NSL-KDD dataset has comparable data and labels in its evaluation, but the amount of data is not balanced. Thakkar and Lohiya [3] also reveal that NSL-KDD has a few types of attacks, and the amount of data from each attack is not equal. In the machine learning environment, input is significant for improving classification results.

In terms of pre-processing and classification, various machine learning-based IDS optimization techniques have been carried out, one of which uses feature selection [4–6] and data reduction [7]. Aburomman and Reaz [8] identify the relationship between pre-processing and classification techniques in building intrusion detection models. In their research, pattern classification issues are overcome through the use of ensemble-based algorithms.

The use of outlier detection for data reduction is developed in machine learning. The formation of dynamic outliers using the features of the dataset has become a reasonably good technique in generating classifications, such as the research in [9-11]. Research conducted by Lyutikova [12] discusses the use of outlier detection to improve the quality of classification in multidimensional data. Not only the amount of data but the importance of features also needs to be considered.

In this research, we propose an outlier detection technique for data reduction. Outliers are dynamically formed through the results of *k*-means clustering of all features. There are two techniques proposed in this paper; the first, the outlier is formed based on the minimum and maximum cluster values. The second, outlier is formed based on the median value of the cluster. This method is applied to several feature selection techniques to determine data reduction performance on a different number of features. Data from feature selection only prioritizes the score of relations between features without calculating the balance of data. The proposed method creates an outlier circle as a dynamic data reduction following the feature selection results' data distribution. Furthermore, the J48 classifier is applied for evaluating the performance. It is compared with that without data reduction and the two other proposed outlier detection methods.

The structure of this paper is as follows; Section 2 explains related works. Section 3 describes the proposed method. Section 4 contains the results and analysis of the experiment, and the conclusions are given in Section 5.

2. Related work

In recent research, many technologies are applied to create IDS systems, and machine learning is one of the popular techniques to implement. At each stage of the machine learning process, various optimization algorithms are carried out to produce the best classification results. Based on the process, there are two essential stages in machine learning that is becoming research to build an IDS model: pre-processing and classification. Pre-processing in IDS means managing network traffic records into an excellent form of data input in the machine learning process. There are many pre-processing applications, such as data reduction and feature selection. As for the application of classification, many studies compare each model.

Regarding feature selection, Khammassi and Krichen [4] apply a Genetic Algorithm (GA) technique as a search strategy and Logistic Regression (LR) as a predictor to analyze the features of a subset. Variations in crossover probability, mutation probability, and population size are performed with a maximum iteration of 1000 times in GA. The selected features are classified using decision tree classifiers. This study concludes that a combination of GA, LR, and C4.5 delivers the best results. Akashdeep et al. [5] implement feature selection based on Information Gain (IG) and Correlation (CR). The ranking is done on all features using both methods. A union and intersection are performed to select features. The ANN classifier is applied to the selected features. The results show that this method can produce optimal accuracy, even for DoS and R2L types. Both methods provide a higher accuracy value than those of previous studies. However, the evaluation of that research is done by using the old KDDCUP99 dataset.

The polished version of that dataset, NSL-KDD, is taken by Donkal and Verma [6] for implementing the selection feature method using Multimodal Fusion. Some parameters are dynamically regulated with a fixed ratio. A combination of Logistic Regression (LR), Gradient-Boosted Trees (GBT), Random Forest (RF), Decision Trees (DT), and Support Vector Machine (SVM), is implemented as a proposed method. Based on their evaluation, the feature selection can improve classification results. These three techniques only prioritize linkages between features without considering balancing data, which might improve classification results.

Aziz and Ahmad [13] also develop a clustering technique to perform feature selection. The linkage between clusters is implemented to calculate the average value of mutual information and correlations in each cluster. This value is used as a determinant of the selected feature. Clustering techniques in this study are only used to calculate the importance of each feature. In a further development, clustering techniques can be used as data reduction.

Semenets et al. [7] combine data reduction and feature selection to speed up the execution process and reduce the False Positive Rate (FPR) level. Three different methods are compared, namely Relief F (RF), Chi-Squared (CHI), and Ranking Filter Information Gain (IG). Data reduction, which is

implemented in this study, only deletes duplicate data obtained from the feature selection process. By using this method, the classification results are quite better than that of traditional classification. A comparison of results shows that IG is the best classification method. In this research, a duplicate removal technique, which is a traditional method for reducing data, is implemented, and some datasets such as NSL-KDD provide duplicated-free data.

Further research is proposed by Wang and Mao [9] about outlier detection using an adaptive *k*-nearest neighbor as an assistant. Support Vector Data Description (SVDD) is a well-known algorithm to build outlier boundaries. The proposed method helps to specify the relevance of the local measurement. In most cases, it is found that a scheme with dynamic selection outperforms both ensemble and single models when they are tested in 20 different datasets of UCI Machine Learning. However, the use of these two algorithms together causes a high computation cost.

Shi et al. [14] also implement an outlier detection technique to solve class imbalance problems. Robust geodesic-based outlier detection is proposed and implemented on ten different datasets, comprising real-world and synthetic data. Using the Global Disconnectivity and Local Degree (GDLD) algorithms, it is found that robust parameters are needed in working on outliers. Because there is no standard for determining the quality of boundaries of outliers, the value is used as an evaluation measurement.

Lastly, the classification method is an important step in the IDS model used as evaluation metrics. Aliakbarisani et al. [15] compare five methods (*k*-NN, Naïve Bayes, Random Forest (RF), Multilayer Perceptron (MLP), and Iterative Classifier Optimizer (ICO)) with the addition of metric learning implementation. They find that the RF, MLP, and ICO methods produce good classification results, but when metric learning is implemented, *k*-NN and RF are becoming the best classifier. Next, Dhannabal and Shantharajah [16] compare three classification methods (J48, SVM, and Naive Bayes) on NSL-KDD data. They find that J48 can produce the best accuracy. By applying conditional random fields, Mahendiran and Appusamy [17] develop an IDS for situational awareness. Nevertheless, they only achieve about 93% in detecting individual attacks. In further research, Semerci et al. [18] introduce an intelligent-based system for detecting an attack; however, it only focusses on specific types of attacks: Distributed Denial of Services (DDOS). Different points of view in defending a system are provided in [19-20]. This research explains how the attack occurs, which illustrates a defending model. Nevertheless, they do not discuss the detail of IDS.

3. Proposed method

Development and optimization methods are always needed to produce better IDS models than the previous ones. Fig. 1 is provided to illustrate the design and functionality of the IDS model. We propose a data reduction method using the outlier detection technique as the research focus.

The IDS scheme is built using machine learning. There are three essential steps to create machine learning with good classification results: feature selection, data reduction, and classification. So, data reduction plays a vital role in producing a reliable IDS scheme.

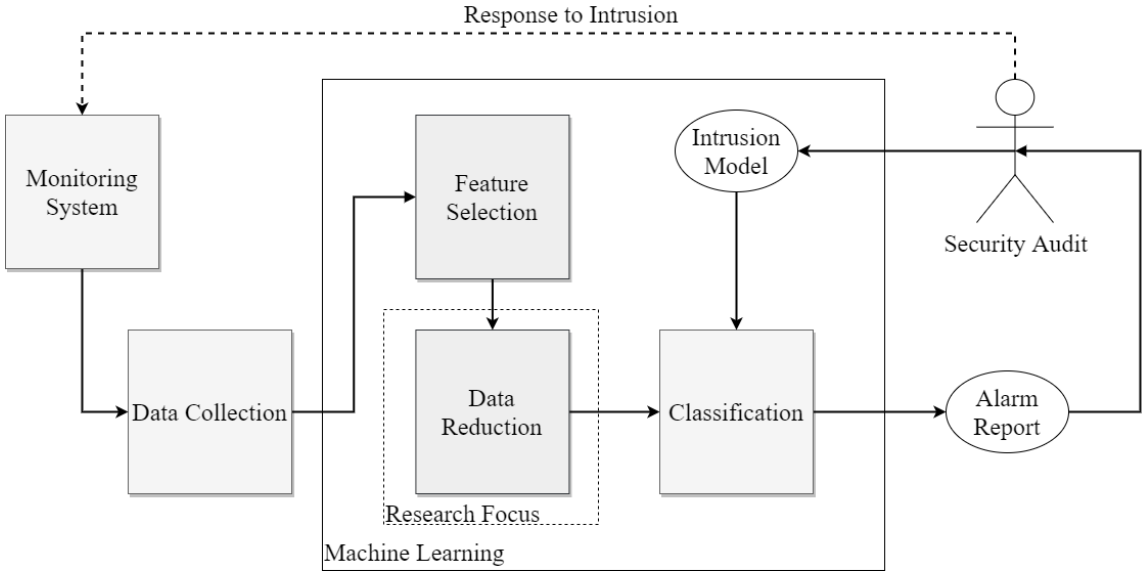

Figure. 1 Flow of intrusion detection system

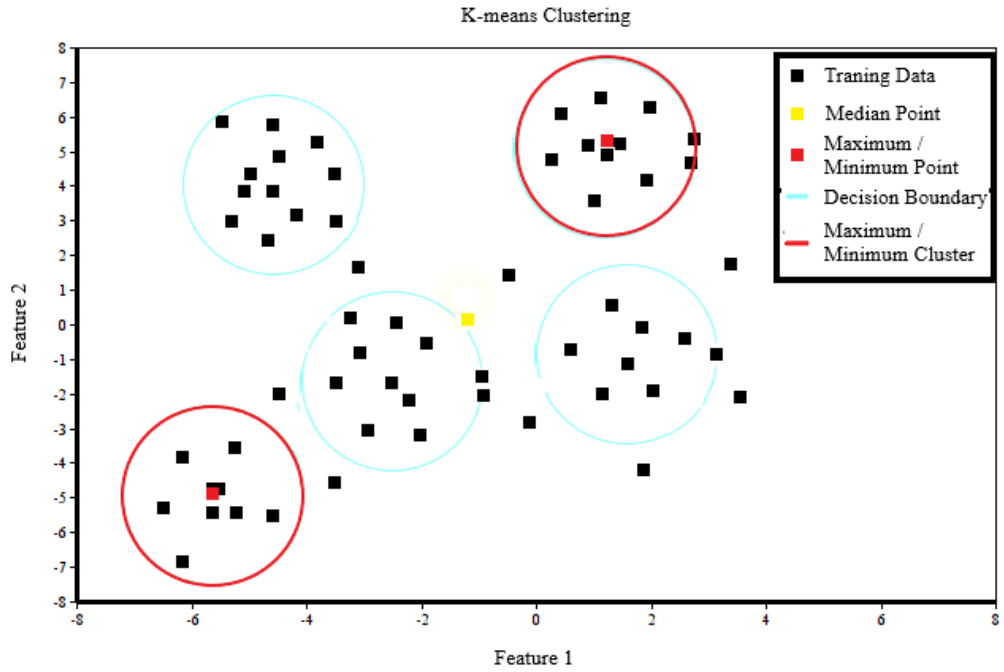

Figure. 2 An example of *k*-means clustering in a 2-dimension dataset

In this section, we define the basic theory of the stages of machine learning in general and its methods. Then, we describe the proposed method in the data reduction stage.

3.1 Feature selection

In this stage, we will implement three different methods as that conducted in [4-6] to determine whether data reduction can improve the performance of the methods. Each feature selection has the number of total selected features, used as a parameter '*k*' in *k*-means. Fig. 2 illustrates an example of $k = 5$.

3.2 Data reduction

This stage is the focus of this research. After the cluster is obtained through the feature selection process, the minimum, maximum, and median cluster values are calculated, as illustrated in Fig. 2.

There are two different scenarios to form the outlier circle depicted in Fig. 3. Each scenario is detailed in the next subsection, compared to find out which method is the most optimal. All data outside the outlier circle is omitted, while data inside the outlier circle is used in the next process.

3.3 Classification

The classification stage is carried out to determine the level of accuracy, sensitivity, and specificity. The result of the combination of feature selection and data reduction techniques will be compared.

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (1)$$

$$Detection\ Rate = \frac{TP}{(TP+FN)} \quad (2)$$

$$Precision = \frac{TP}{(TP+FP)} \quad (3)$$

J48 algorithm is applied to determine the value of TN (True Negative), TP (True Positive), FN (False Negative), and FP (False Positive). This value is used to obtain the accuracy, detection rate, and precision represented in Eqs. (1), (2), and (3), respectively.

3.4 Scenario 1

In scenario 1, distances between the clusters of selected features are calculated. The longest distance is selected, and the center point of the circle is determined. An illustration of generating an outlier circle is described in Fig. 4. Below is a detailed description of each step.

Step 1: Calculate the centroids of all selected features from the feature selection process using the *k*-means clustering method.

Step 2: Calculate the distance of all centroids using Eq. (4) and find the longest distance between two centroids.

Step 3: Calculate the center of the circle using Eq. (5) as (h, k).

Step 4: Calculate the outlier circle using the Eq. (6).

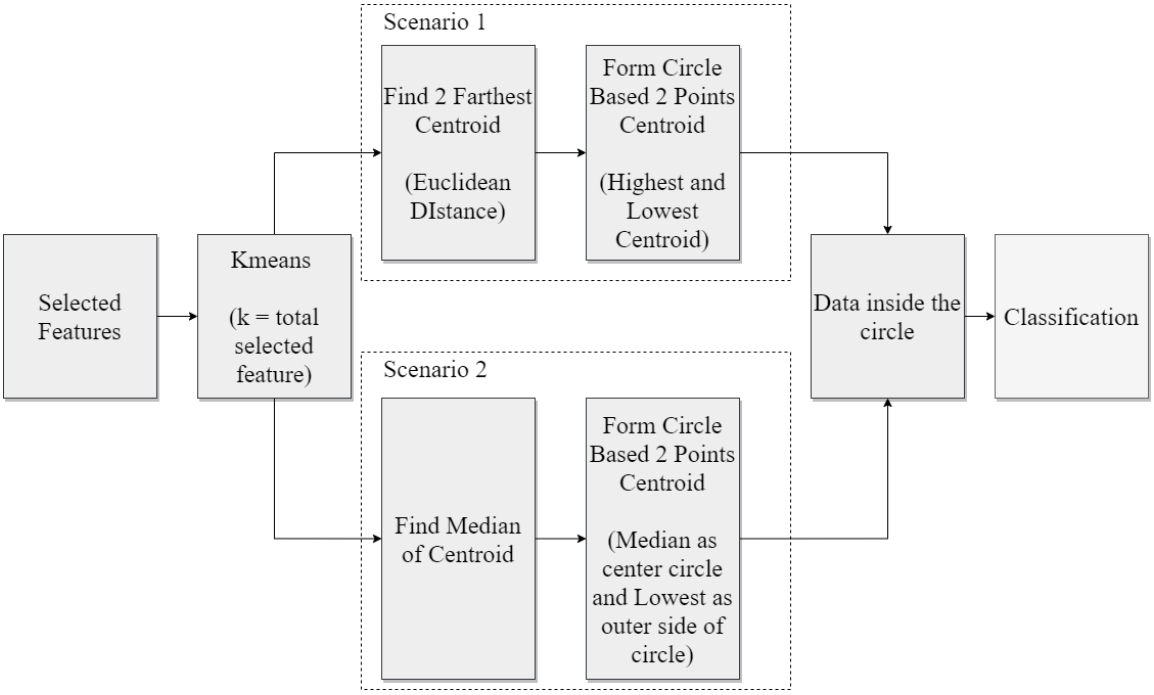

Figure. 3 Flow of the proposed method

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (4)$$

$$[h, k] = \left[\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2}\right] \quad (5)$$

$$(x - h)^2 + (y - k)^2 \leq r^2 \quad (6)$$

The symbol d in Eq. (4) is an Euclidean value to measure the distance between 2 points. Half of *d* is *r* (radius). The Eq. (5) is used to find the value (*h*, *k*), both are used in the Eq. (6) as outlier detection.

3.5 Scenario 2

In scenario 2, the median of all centroids is calculated. It is used as the center of the circle (*h*, *k*). Like scenario 1, the outlier is generated using Eq. (6) after (*h*, *k*) is obtained. The median is obtained by sorting all the centroid to get the mean value. The Eq. (7) is used when the features obtained are odd, while Eq. (8) is used if the features are even.

$$[h, k] = \left[x\left(\frac{n+1}{2}\right),\ y\left(\frac{n+1}{2}\right)\right] \quad (7)$$

$$[h, k] = \begin{bmatrix} \frac{1}{2}\left(x\left(\frac{n}{2}\right) + x\left(\frac{n}{2}+1\right)\right), \\ \frac{1}{2}\left(y\left(\frac{n}{2}\right) + y\left(\frac{n}{2}+1\right)\right) \end{bmatrix} \quad (8)$$

4. Experimental results

In this section, we present the results from the experiments. Data reduction aims to reduce the processing time of IDS modeling and improve classification performance. The method is implemented using python programming with the sklearn library. The device uses Google Collaboratory with GPU Tesla K80 and 12 GB RAM. The proposed method is evaluated by comparing it with [4-6] since the aims of this research are to improve the performance of feature selection.

In the NSL-KDD dataset, there are 125.973 records with 41 features. Before entering the selection stage, we change the string type data into numbers as a representation, so that the dataset can be applied to methods in the reduction, selection, and classification. As explained in Table 1, this process is implemented on protocol type, service, and flags.

In these experiments, information gain and correlation, multimodal fusion, and GA have 25, 30, and 18 selected features. This number is used as *k* in the *k*-means algorithm to generate clusters according to the number of features selected. Scenario 1 uses the minimum and maximum clusters, and scenario 2 uses the median of clusters to form outlier circles as described in the proposed method. The amount of reduced data is dependent on the position of the cluster from this process. It is also worth mentioning that Euclidean, with a vast range of differences, will cause less data reduction.

The J48 algorithm is used as an evaluation metric to get data classification. Data reduction results are divided into testing and training data; then, the cross-validation process is carried out using a *k*-fold with a size of 20%.

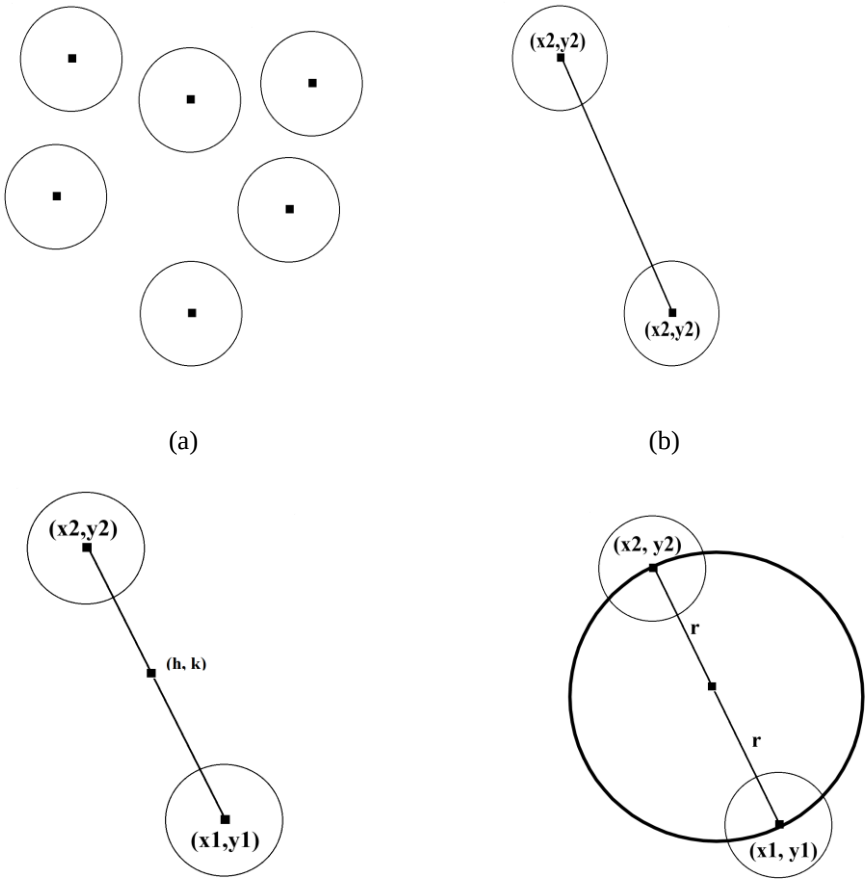

Figure. 4 Develop a circle based on two points: (a) calculate all centroid using *k*-means, (b) find two points and calculate the distance, (c) find (*h*, *k*) based on the distance, and (d) create a circle using *r* and (*h*, *k*)

Table 1. Non-numeric value to convert

Feature	Data
protocol_type	icmp, tcp, udp
service	other, link, netbiosssn, smtp, netstat, ctf, ntpu, harvest, efs, klogin, systat,exec, nntp, pop3, printer, vmnet, netbiosns, urhi, ssh, http8001, isotsap,aol, sqlnet, shell, supdup, auth, whois, discard, sunrpc, urpi, rje, ftp,daytime, domainu, pmdump, time, hostnames, name, ecri, bgp, telnet,domain, ftpdata, nnsp, courier, finger, uucppath, X11, imap4, mtp, login,tftpu, kshell, private, http2784, echo, http, idap, timi, netbiosdgm, uucp,ecoi, remotejob, irc, http443, redi, z3950, pop2, gopher, csnetns
Flag	OTH, S1, S2, RSTO, RSTSr, RSTOS0, SF, SH REJ, S0, S3

4.1 Information gain and correlation.

In experiments using a combination of information gain and correlation, there are 25 numbers of features used. Information gain is carried out using entropy to determine the value of each feature, while the correlation is done by calculating the correlation coefficient of each column pair, and the mean value is taken as the value of the column. The technique generates its ranking list, which is devised to get strong feature values from both IG and CR calculations.

By applying the proposed method to this technique, the data are reduced by 1565 in scenario 1 and 1450 in scenario 2. This application can reduce both FP and FN values shown in Table 2. The classification results increase in terms of accuracy, detection rate, and precision.

4.2 Multimodal fusion

Multimodal fusion is an improvement of NSGA-II algorithm. NSGA-II works by choosing the most prominent feature. The population size and length of binary chromosomes are 40 and 41, respectively, representing 41 features of NSL-KDD. Some other parameters, such as crowding distance sorting, Pareto front, and non-dominant rank, are set as dynamic. The features selected by NSGA-II are classified by five different methods, namely SVM, GBT, DT, RF, and LR. The multimodal fusion technique is applied by taking the majority output from the five classification results. There are 30 features selected by this algorithm.

Like the previous experiment, the proposed method has better classification results in accuracy, detection rate, and precision. The detection rate and precision are still the same as the previous experimental experiments, where scenario 1 has the best detection rate, and scenario 2 has the highest precision. However, the accuracy of scenario 1 is more visible than that of scenario two or the previous method. The amount of data reduced in scenario 1 is 1506, and in scenario two is 1978. The detailed classification data are shown in Table 3.

4.3 Genetic algorithm and logistic regression

Genetic Algorithm and Logistic Regression (GA-LR) works by comparing the subset of features, the accuracy of the subset, CPU time, fitness value, and the number of features. There are 18 features, which have been selected using this algorithm.

An unexpected result has occurred when the proposed method is implemented in this experiment. As shown in Table 4, more than half TP and some of TN are significantly reduced to lowering FP and FN. The data reduced in scenario one, and scenario 2 are 46170 and 50290, respectively. In this case, scenario 1 has the lowest classification result. Nevertheless, scenario two still has better results than that without reduction or scenario 1.

4.4 Performance evaluation

Table 5 shows the overall performance of data reduction in all experiments. In terms of accuracy, detection rate, and precision, the proposed method is always better than that without data reduction. The amount of data reduced based on the formation of a circle is dependent on the clusters formed in *k* means, which dynamically follows the distance and direction of each cluster to produce balanced data.

Table 2. Comparison between the proposed method and that with information gain and correlation [5]

Scenario	TP	TN	FP	FN
Without Reduction [5]	13425	11725	23	22
Scenario 1	13348	11495	19	20
Scenario 2	13305	11561	18	21

Table 3. Comparison between the proposed method and that with multimodal fusion [6]

Scenario	TP	TN	FP	FN
Without Reduction [6]	13410	11706	38	41
Scenario 1	13247	11576	37	34
Scenario 2	13352	11373	36	38

Table 4. Comparison between the proposed method and that with genetic algorithm [4]

Scenario	TP	TN	FP	FN
Without Reduction [4]	13419	11727	29	20
Scenario 1	6043	9885	19	14
Scenario 2	5350	9772	11	4

5. Conclusion

Several modeling IDS methods have been developed for years. Feature selection is an attempt to improve classification performance. By eliminating data that interfere with classification, it is expected that accuracy, detection rate, and precision increase. However, this technique only removes data whose value does not meet the criteria without considering data integrity. The proposed method can solve this problem by balancing data.

Furthermore, data are reduced precisely, improving the accuracy, detection rate, and precision of classification. The system is put to the test on three different feature selection, whose results are encouraging. As an implication, the proposed method can enhance the feature selection process for classification. In most cases, the method's performance with data reduction has better results than that without reduction. Another positive impact is that smaller data also cause the classification process to run faster. So, it affects modeling with far fewer time complexities.

In scenario 2, the proposed method always has better results than that without reduction. In the IG + Correlation and Multimodal Fusion experiments, accuracy increases by 0.02%, and in GA-LR, it increases by 0.1%. The detection rate also increases by 0.01%, 0.02%, and 0.07% in the first, second, and third experiments. Precision also increased by 0.04%, 0.02% and 0.01%.

Table 5. Summary of the comparison between the proposed method and existing ones

Feature selection method (number of selected features)	Data Reduction	Accuracy (%)	Detection Rate (%)	Precision (%)
IG + Correlation (25)	Without Data Reduction [5]	99.82	99.83	0.9982
	Scenario 1	99.84	99.85	0.9985
	Scenario 2	99.84	99.84	0.9986
Multimodal Fusion (30)	Without Data Reduction [6]	99.68	99.69	0.9971
	Scenario 1	99.71	99.74	0.9972
	Scenario 2	99.70	99.71	0.9973
GA + LR (18)	Without Data Reduction [4]	99.80	99.85	0.9978
	Scenario 1	99.79	99.76	0.9968
	Scenario 2	99.90	99,92	0.9979

Whereas in scenario 1, accuracy can increase by 0.03%, which is the best efficiency. The best detection rate is also obtained in scenario 1, which is in the IG + Correlation and Multimodal Fusion experiments. However, the application of scenario 1 causes a decrease in classification results when it is applied to the GA-LR method.

Although the present work seems convincing, some factors such as data range and the number of features are crucial in the performance evaluations. Proposed methods can work well only if feature selection has removed features that are not relevant. However, as explained in the experimental results, the range of biased data must be handled even more in future works.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

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

References

[1] M. Małowidzki, P. Berezi, and M. Mazur, "Network Intrusion Detection : Half a Kingdom for a Good Dataset", In: *Proc. of ECCWS 2017 16th Eur. Conf. Cyber Warf. Secur.*, pp. 1–6, 2017.

[2] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A survey of network-based intrusion detection data sets", *Comput. Secur.*, Vol. 86, pp. 147–167, 2019.

[3] A. Thakkar and R. Lohiya, "A Review of the Advancement in Intrusion Detection Datasets", *Procedia Comput. Sci.*, Vol. 167, No. 2019, pp. 636–645, 2020.

[4] C. Khammassi and S. Krichen, "A GA-LR wrapper approach for feature selection in network intrusion detection", *Comput. Secur.*, Vol. 70, pp. 255–277, 2017.

[5] Akashdeep, I. Manzoor, and N. Kumar, "A feature reduced intrusion detection system using ANN classifier", *Expert Syst. Appl.*, Vol. 88, pp. 249–257, 2017.

[6] G. Donkal and G. K. Verma, "A multimodal fusion based framework to reinforce IDS for securing Big Data environment using Spark", *J. Inf. Secur. Appl.*, Vol. 43, pp. 1–11, 2018.

[7] V. H.-Semenets, O. Andrés Pérez-García, R. Hernández-León, J. van den Berg, and C. Doerr, "A data reduction strategy and its application on scan and backscatter detection using rule-based classifiers", *Expert Syst. Appl.*, Vol. 95, pp. 272–279, 2018.

[8] A. A. Aburomman and M. B. I. Reaz, "A survey of intrusion detection systems based on ensemble and hybrid classifiers", *Computers and Security*, Vol. 65, pp. 135–152, 2017.

[9] B. Wang and Z. Mao, "A dynamic ensemble outlier detection model based on an adaptive k-nearest neighbor rule", *Inf. Fusion*, Vol. 63, pp. 30–40, 2020.

[10] G. Gan and M. K. P. Ng, "K-Means Clustering With Outlier Removal", *Pattern Recognit. Lett.*, Vol. 90, pp. 8–14, 2017.

[11] F. Angiulli, S. Basta, S. Lodi, and C. Sartori, "Reducing distance computations for distance-based outliers", *Expert Syst. Appl.*, Vol. 147, 2020.

[12] L. Lyutikova, "Logical Analysis of Data for outliers detection," *Procedia Comput. Sci.*, Vol. 169, No. 2019, pp. 330–336, 2020.

[13] M. N. Aziz and T. Ahmad, "Cluster analysis-based approach features selection on machine

learning for detecting intrusion", *Int. J. Intell. Eng. Syst.*, Vol. 12, No. 4, pp. 233–243, 2019.
[14] C. Shi, X. Li, J. Lv, J. Yin, and I. Mumtaz, "Robust geodesic based outlier detection for class imbalance problem", *Pattern Recognit. Lett.*, Vol. 131, pp. 428–434, 2020.
[15] R. Aliakbarisani, A. Ghasemi, and S. Felix Wu, "A data-driven metric learning-based scheme for unsupervised network anomaly detection", *Comput. Electr. Eng.*, Vol. 73, pp. 71–83, 2019.
[16] L. Dhanabal and D. S. P. Shantharajah, "A Study On NSL-KDD Dataset For Intrusion Detection System Based On Classification Algorithms", *Int. J. Adv. Res. Comput. Commun. Eng.*, Vol. 4, No. 6, pp. 446–452, 2015.
[17] A. Mahendiran and R. Appusamy, "An Intrusion Detection System for Network Security Situational Awareness Using Conditional Random Fields", *International Journal of Intelligent Engineering and Systems*, Vol. 11, No. 3, pp. 196–204, 2018.
[18] M. Semerci, A. T. Cemgil, and B. Sankur, "An intelligent cyber security system against DDoS attacks in SIP networks", *Computer Networks*, Vol. 136, pp. 137-154, 2018.
[19] K. C. Lalropui and V. Gupta, "Modeling cyber-physical attacks based on stochastic game and Markov processes", *Reliability Engineering and System Safety*, Vol. 181, pp. 28-37, 2019.
[20] S. Bernardi, U. Gentile, S. Marrone, J. Merseguer, and R. Nardone, "Security modelling and formal verification of survivability properties: Application to cyber-physical systems", *Journal of Systems and Software*, in press.