International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. , No. (Mon-20..)

http://dx.doi.org/10.12785/ijcds/XXXXXX

An Effective Hybrid Feature Selection and Classifier Model for Intrusion Detection Systems (IDS)

Ricky Aurelius Nurtanto Diaz[1,2], I Ketut Gede Darma Putra[3], Made Sudarma[4], I Made Sukarsa[3] and Emy Setyaningsih[5]

[1]*Department of Computer Systems, STIKOM Bali Institute of Technology and Business, Denpasar, Bali, Indonesia*
[2]*Faculty of Engineering, Udayana University, Bali, Indonesia*
[3]*Department of Information Technology, Faculty of Engineering, Udayana University, Bali, Indonesia*
[4]*Department of Electrical Engineering, Faculty of Engineering, Udayana University, Bali, Indonesia*
[5]*Department of Computer Systems Engineering, Universitas AKPRIND Indonesia, Yogyakarta, Indonesia*

Received Mon. 20, Revised Mon. 20, Accepted Mon. 20, Published Mon. 20

Abstract: Network intrusion threats can have an impact on business losses. One mechanism that can be applied is Intrusion Detection Systems (IDS). IDS model development is carried out in various categories, starting from experimenting with classifiers, combining classifier models, and carrying out optimization, including implementing various feature selections. This experiment cannot be separated from the current need for an accurate IDS model with a minimum response time. This study was conducted to find the most efficient and proper combination of classifiers and features from three feature options, namely Grey Wolf Optimizer (GWO), Gain Ratio, and Chi-Square. At the same time, the classification algorithms used are Logistic Regression, Support Vector Machine, Random Forest and Decision Tree. The combination of these models will be tested on the NSL-KDD and UKM-IDS20 datasets. The tests showed that the Random Forest classifier can be used hybrid with the GWO feature selection algorithm and produces high accuracy with low computation time. In detail, for the NSL KDD dataset, the combined GWO-RF model has the highest accuracy, with 99.99% for training and 99.89% for testing. The GWO-RF model outperformed all other feature selection and classifier alternatives on the UKM-IDS20 dataset in terms of accuracy, where the resulting accuracy value reaches 99.98% for training and 99.97% accuracy for the testing process.

Keywords: Chi-Square, Decision Tree, Gain Ratio, GWO, Logistic Regression, Random Forest, SVM

1. INTRODUCTION

Network security is one of the main focuses in the world of work, where everything is connected digitally. This makes computer networks vulnerable to various threats, such as cyberattacks and other malicious activities. In 2021, based on statistical data collected for cybersecurity, cyberattacks are predicted to reach three trillion, possibly executing zero-day exploits every day [1]. Another case in 2022 is the substantial increase in data storage capacity in private and cloud services run by Facebook, Twitter, and Amazon Web services [1], [2]. Network intrusion can have an impact on business losses, including monetary losses, reputational damage, legal liability, and the ability to eliminate confidential information [3], [4], [5]. Social engineering attacks also have a high percentage because they are included in classifying the leading causes of financial losses in the digital world [6].

Various mechanisms and strategies can be implemented to increase network security from intrusion threats, including prevention and passive protection methods such as firewalls, antivirus, and VPN. Researchers have also developed proactive prevention and protection methods, namely Intrusion Detection Systems (IDS), which can detect and respond to threats that may have managed to get through other layers of defense. As a result, IDS becomes a crucial part of an all-encompassing network security plan, enhancing the capacity to identify and neutralize attacks that have the potential to do a great deal of harm. IDS functions to monitor network activity and detect signs of attacks or security breaches. IDS monitors network traffic and looks for indicators of security breaches or irregularities. The two main categories of IDS systems are network-based IDS (NIDS) and host-based IDS (HIDS). The task of NIDS is to monitor network traffic to detect attacks, while HIDS is tasked with monitoring activity on specific devices. NIDS focuses on detecting attacks by monitoring network traffic to detect intrusions, while HIDS monitors activity on individual devices, including traces, system calls, application activity, and their parameters [7].

Research in the field of IDS has grown rapidly and has

E-mail address: ricky@stikom-bali.ac.id, ikgdarmaputra@unud.ac.id, msudarma@unud.ac.id, sukarsa@unud.ac.id, emysetyaningsih@akprind.ac.id

experienced tremendous growth over the past few decades. To increase intrusion detection's precision, efficacy, and accuracy, a number of techniques and algorithms have been developed. One significant area of research is the use of machine learning techniques to improve IDS performance [2], [7], [8]. Machine learning allows systems to learn from previous data and improve their detection capabilities over time. The diversity of applications of machine learning algorithms used for IDS is staggering, ranging from [1], [7], [9], [10]. Each algorithm for training and testing IDS datasets has different techniques, showcasing the breadth and depth of the research in this field.

In general, classification methods can be divided into two main categories: mining-based and statistical-based. Data mining-based classification methods refer to discovering patterns and knowledge from big data; one of the advantages of mining-based methods here is data flexibility and scalability, considering the large IDS dataset and its diverse data characteristics [11]. On the other hand, statistical classification methods focus more on data analysis using probability theory and statistical inference. One of its standout advantages in the IDS field is the robust mathematical model it provides, instilling confidence in its ability to identify various attacks and network scenarios [12]. This machine learning method is still widely used, considering that the computing time and resources needed are smaller than those used in deep learning [13].

Some of the classifier algorithms applied include Decision Tree [14], [15], Random Forest [9], [16], Logistic Regression [9] and Support Vector Machine [17], [18], [19], [20]. This classical machine learning algorithms, used in the formation of the IDS model is carried out based on several datasets, such as NSL-KDD [9], [17], [18] and UKM-IDS20 [21] where the selected dataset has examples of network attacks that have been grouped for training and testing. Optimization is done on features, parameters, and measurements of variability, and for testing, it can be compared from the values of precision, accuracy, and efficiency.

In addition, the proper selection of features is also a critical factor in improving IDS performance [18], [22]. A preprocessing phase called feature selection (FS) gathers the most pertinent features to construct a reliable model. This critical step directly affects how well IDS performs [23]. Bio-inspired metaheuristic algorithms are commonly used in approaches to feature selection processes in intrusion detection systems due to their better accuracy. FS is an approach that can eliminate features that are not suitable and tend to be excessive and then select the best subset of features to improve the formation of patterns and data groups [24].

One of the metaheuristic methods that can work as a technique in feature selection is the Grey Wolf Optimizer (GWO). GWO is one of the algorithms used for feature selection in IDS [25]. GWO mimics the behavior of gray wolves on the hunt to find optimal solutions. Using GWO, we can identify the most relevant and significant features for use in the IDS model, thereby improving detection accuracy and efficiency [24], [26], [27], [28].

In this study, several feature selection techniques are proposed to find out the best feature selection results, in addition to GWO, namely Chi-Square and Gain Ratio, as a comparison. The feature selection technique chosen as a comparison refers to the previous study [23], where this feature selection method can increase the classification model's accuracy, help select appropriate features, and reduce complex data [23]. In addition, another study revealed that the Gain Ratio, which is an extension of the Information Gain criterion, has shown that the Gain Ratio can improve the detection accuracy of the IDS model by selecting features that provide the most information about the target variables, such as in research on the selection of features aimed at effective disease risk prediction [29], [30]. Regarding selection features with Chi-Square, it is explained that it can determine threshold optimization, a simple algorithm. It efficiently reduces excessive data while paying attention to excellent and appropriate data edges [31], [32]. Where the results of each of these selection features will be classified by Logistic Regression, Support Vector Machine, Random Forest and Decision Tree. The datasets selected are NSL-KDD and UKM-IDS20.

Based on related research, determining the correct classifier and feature selection is crucial in implementing IDS. The proper classifier can improve intrusion detection capabilities while selecting the most accurate features, reducing model complexity, and increasing the classification algorithm's processing speed to obtain the best accuracy. Therefore, analyzing the classifier's performance for the IDS model using various variations and comparisons of different feature selection techniques is crucial to ensure an efficient IDS model and optimal network security. The main contribution of this researcher is to test method combinations with classification algorithms, such as SVM, Random Forest, Decision Tree, and Logistic Regression, and feature selection techniques such as the GWO, Gain Ratio, and Chi-Square. To ensure the model's capabilities, this research evaluates the selected datasets, namely NSLKDD and UKM-IDS20. The main contribution of the results of this study will be the comparison of the accuracy and efficiency of computing time in the classification process, reassuring the audience about the practicality of the IDS model. From the results of this study, we can find a combination of classification algorithms with the best feature selection algorithms that can be applied to specific IDS datasets, as well as the development of other IDS models with various other dataset options.

Next, our paper is organized in the following order: Section 2 presents the research method that describes the stages of our research. Section 3 presents the research

results to test each dataset, followed by a comparison of the overall results of the model with each dataset. Section 4 is the final stage in concluding our experimental results and looking at future research opportunities.

2. RESEARCH METHOD

We used four main stages in this study; where we started by preparing the dataset, applied the dataset to the selection feature algorithm, then tested the accuracy of the classifier we used, and finally, compared the performance of each classifier using each existing selection feature model for each dataset we used. An overview of our research method is shown in Fig. 1.

A. IDS Datasets

The NSL-KDD and UKM-IDS20 datasets are used in this study; the dataset details are displayed in Table I

TABLE I. Datasets Information

Datasets	Dataset Information		
	Source	Number of features	Amount of data
NSL-KDD	https://www.kaggle.com/datasets/hassan06/nslkdd	43	125.972
UKM-IDS20	https://www.kaggle.com/datasets/muatazsalam/ukm-ids20	48	10.308

To ensure fair results in the classifier model and selection features we used, we selected three types of datasets with different sizes, starting from the largest, namely the NSL-KDD dataset and the smallest data size represented by the UKM-IDS20 dataset.

B. Feature Selection Methods

The research used Grey Wolf Optimizer (GWO), Gain Ratio, and Chi-Square methods to find the best features of the dataset.

- Grey Wolf Optimizer (GWO).
  The GWO algorithm is swarm-based and was put forth by Mirjalili (2014). The social dynamics of natural grey wolf packs serve as the inspiration for the GWO. Grey wolves prefer to live in packs in the wild. Based on the wolf's standing within the pack, which helps to enhance the hunting process; Alpha, Beta, Delta, and Omega are the four categories into which the pack members are separated. [33]. In this research, we used GWO to find and classify the best feature using some classifiers. The original pseudocode of this algorithm is represented in Fig 2[34].

- Gain Ratio (GR).
  In machine learning and data analysis, one of the measures used to categorize data or choose features is the gain ratio. When discussing decision tree algorithms, the gain ratio is frequently brought up. Optimized values for a feature are refined in classification by applying gain ratios, which enhance information gain. Since the gain ratio can yield greater precision than other filter approaches, it was selected [35]. To compute the gain ratio, we must first calculate the information gain. The procedure for computing the Gain Ratio is shown in Equation (1).

$$Gain\_Ratio = \frac{Gain}{SplitInfo} \tag{1}$$

  SplitInfo is the result of splitting the entropy computation, while Gain is the information gain calculation.

- Chi-Square (CS).
  The discrepancy between an observed distribution and a theoretical (assumed) distribution is tested statistically using the chi-square test. Quantitative research uses this test, primarily qualitative research with categorized data. Each characteristic in the Chi-Square test is assigned a score for each class, and the final maximum value is determined by summing these individual scores. [31]. We can see the process of calculating Chi-Square as shown in Equation (2).

$$X_c^2 = \sum \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \tag{2}$$

  where, c is degree of freedom, O_{ij} is the observed frequency in cell ij, E_{ij} is the expected frequency in cell ij calculated as Equation **??**.

$$E_{ij} = \frac{R_i \times C_j}{N} \tag{3}$$

  where, R_i is totals of row i, C_j is totals of column j and N is total number of Observations.

C. Classifier

The research used five different classifiers to test each dataset with a combination of existing selection features.

- Support Vector Machines (SVM).
  The SVM algorithm can handle high-dimensional data and is heavily dependent on machine learning. Finding the best hyperplane to provide a better dataset generalization is the fundamental idea behind SVM. It creates a model using a hyper-plane that forecasts whether a fresh sample fits into any existing categories or not [36].

- Random Forest (RF).
  The Random Forest classifier, a form of bagging classifier, tackles the issue of decision trees' poor performance during testing. It does so by using multiple decision trees as separate models, each receiving input through row and column sampling techniques. This approach effectively resolves the issue, making Random Forest a practical and effective solution [37].

- Decision Tree (DT).
  A supervised learning approach presents a visual representation that includes a decision tree. A hier-

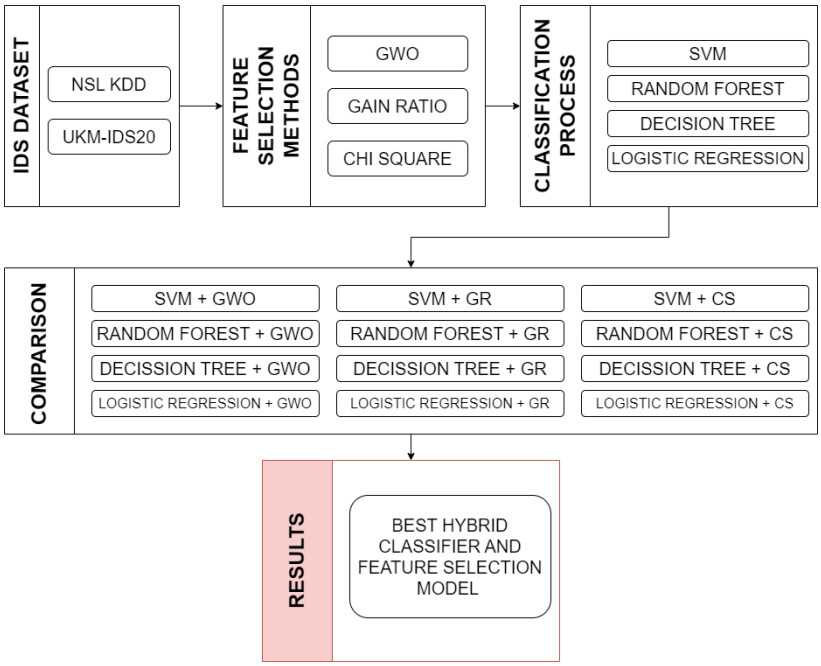

Figure 1. Our Research Method

Algorithm 1. Pseudocode of GWO

Decide the max value of iterations I
Set the population Xn (n = 1,2,...,N)
Set z, R, and P
Compute wolves fitness value
W_α = best search agent
W_β = 2nd best
W_δ = 3rd best
while (q < I) do
    For every search agent do
    Renew the current positions
    End for
    Update z, R, and P
    Count all search agents fitness
    Renew W_α, W_β, W_δ
    q = q + 1
end while
Return W_α

Figure 2. Pseudocode of GWO

archical model with many connected nodes is used by decision trees. These nodes stand in for tests of the dataset's properties, and each branch results in a distinct node or a classification conclusion for the data. The training data plays a crucial role in building the tree, ensuring a robust learning process. The prediction data is then routed through the tree's nodes until it can be classified [38]. To select attributes as roots and branches, the algorithm will calculate the highest gain value of the existing attributes with the calculations as shown in Equation (4).

$$Gain(S,A) = entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy(S_i) \quad (4)$$

where A is the attribute, S is the number of cases in S, S_i is the number of cases in partition i, n is the number of partitions of attribute A, and S is the case set. To get the Entropy value, use the calculation as shown in Equation (5).

$$Entropy(S) = \sum_{i=1}^{n} -p_i * log_2(p_i) \quad (5)$$

When S is the case set, p_i is the ratio of S_i to S, and n is the number of partitions in S.

- Logistic Regression (LR) The supervised learning algorithm known as Logistic Regression uses the logistic function, often known as the Sigmoid function. Usually, this function enables us to classify the inputs into binary valued labels [38], but we used to classify multi-class classification in our research.

D. Comparison

In this stage, we compare all classifiers' computational time and accuracy with each feature selection model for our two datasets. There are 24 experimental results involving

four classifiers, three feature selection models, and two datasets.

3. RESULT AND DISCUSSION

A. *Feature List*

In our experiment, we compared the accuracy and computing time during training and testing each classifier using a dataset where Chi-Square (CS), Gain Ratio (GR), and GWO are used in the feature selection process. From the feature selection process, each feature selection algorithm that we use produces selected features starting with the NSL-KDD dataset, as illustrated in Fig. 3.

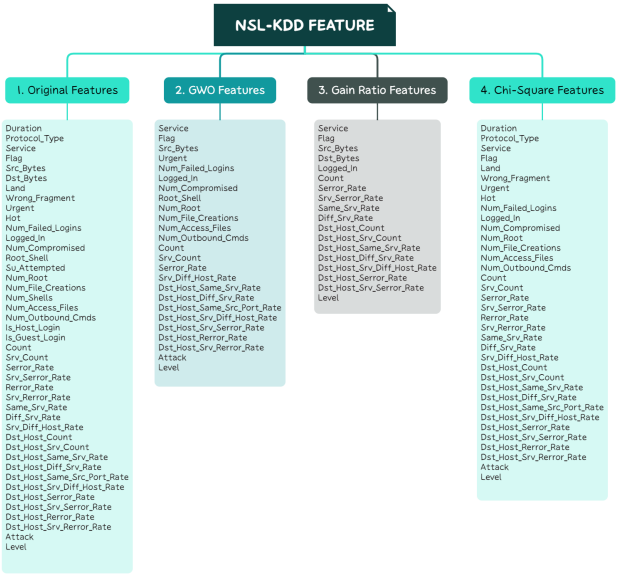

Figure 3. NSL-KDD Feature List

For NSL KDD dataset, GWO produced 25 features, 18 features obtained by Gain Ratio, and 36 features generated by Chi-Square. These selected features will then be used as the main features in forming a model with a predetermined classifier.

Furthermore, we did the same with different datasets UKM-IDS20, where the GWO algorithm produces 17 features, the Gain Ratio produces six features, and Chi-Square produces ten features. Details of the selected features for this dataset can be seen in Fig. 4.

B. *Computational Time*

To test the effectiveness of the feature selection results processed by the classifier algorithm, we first want to test the computational time for each dataset both in the training and testing processes. For the NSL-KDD dataset, we found the computation time with the highest GWO selection feature when using the SVM classification algorithm with a computation time of 22.24 seconds and Logistic Regression with a computation time of 9.48 seconds. Decision Tree produced the lowest computation time with GWO, 0.28

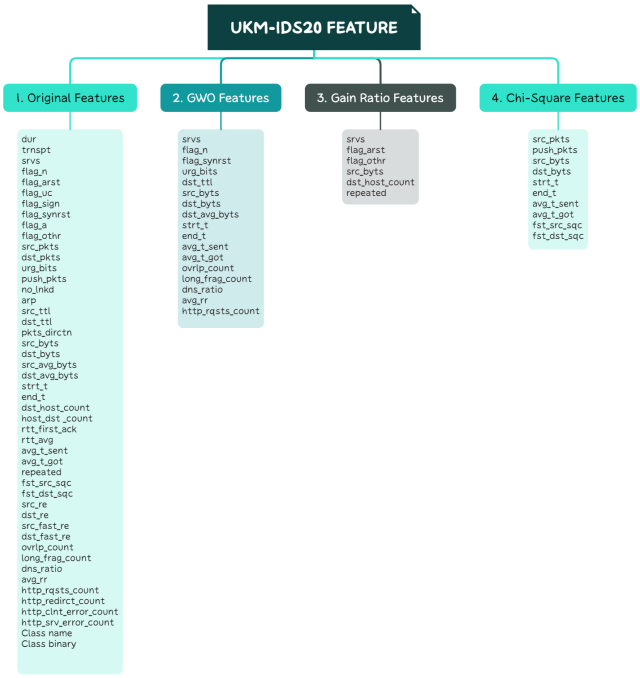

Figure 4. UKM-IDS20 Feature List

seconds. When using the Gain Ratio feature, the highest computation time was produced when using SVM, and the lowest computation time was when using the Decision Tree as the classification algorithm, which had a time of 0.73 seconds. The same results were obtained using the Chi-Square feature, with the highest computation time being SVM and the lowest computation time being the Decision Tree. The testing process computation time for the same dataset obtained results when using the GWO feature; the highest computation time was produced when combining GWO with SVM (28.71 seconds) and the lowest computation time when combining GWO with Logistic Regression(0.01 seconds). The highest computation time was obtained using SVM for the Gain Ratio feature, with a computation time of 132.66 seconds. The lowest computation time was when using Logistic Regression, with a computation time of 0.01 seconds. Finally, when combining Chi-Square features, the classifier with the highest computation time is SVM, with a computation time of 67.74 seconds, and the lowest when combining Chi-Square with Logistic Regression (0.03 seconds). Details of the computing time of the test can be seen in Table II.

TABLE II. Time Comparison for NSL-KDD Dataset

Classifier	Training Time			Testing Time		
	GWO	GR	CS	GWO	GR	CS
Decision Tree	0.28	0.73	0.81	0.01	0.03	0.03
Random Forest	1.56	1.56	4.75	0.29	0.29	0.78
Support Vector Machine	22.24	79.11	34.14	28.71	132.67	67.74
Logistic Regression	9.48	32.25	28.50	0.01	0.016	0.03

Lastly, we tested with the UKM-IDS20 dataset, which

has the smallest amount of data compared to the previous datasets, and we found that the SVM classifier always has a higher compute time than all other classifiers with any selection feature option during the training and testing process. The details of the training and testing computational time for this dataset can be seen in Table III.

TABLE III. Time Comparison for UKM-IDS20 Dataset

Classifier	Training Time			Testing Time		
	GWO	GR	CS	GWO	GR	CS
Decision Tree(DT)	0.03	0.009	0.03	0.001	0.001	0.0001
Random Forest(RF)	0.21	0.15	0.29	0.02	0.01	0.02
Support Vector Machine(SVM)	0.37	0.55	0.37	0.77	1.45	0.85
Logistic Regression(LR)	0.14	0.16	0.13	0.0001	0.0001	0.0001

C. Accuracy

We then test the accuracy of each classifier by combining each classifier with the existing feature selection algorithm on each dataset. For the first experiment using the NSL KDD dataset, we found two classifiers with the highest accuracy in training and testing: Random Forest and Support Vector Machine.

The highest accuracy for the training process when using GWO is Random Forest with an accuracy of 99.99%; when using Gain Ratio, Random Forest also shows the highest accuracy with 99.98%, and when using Chi-Square, Random Forest achieves the highest accuracy result with an accuracy of 99.98%. For the testing process, the highest accuracy when using GWO is Random Forest with an accuracy of 99.89%; when using Gain Ratio, Random Forest also shows the highest accuracy with 99.85%, and when using Chi-Square, Random Forest achieves the highest accuracy result with an accuracy of 99.87%. Details can be seen in Table IV.

TABLE IV. Accuracy for NSL-KDD Dataset

Classifier	Training Accuracy			Testing Accuracy		
	GWO	GR	CS	GWO	GR	CS
Decision Tree(DT)	95.19%	94.59%	95.17%	95.09%	94.47%	95.03%
Random Forest(RF)	99.99%	99.98%	99.98%	99.89%	99.85%	99.87%
Support Vector Machine(SVM)	99.25%	99.09%	99.58%	99.24%	99.05%	99.54%
Logistic Regression(LR)	96.38%	95.31%	97.67%	96.25%	95.31%	97.67%

Furthermore, our experiment with the UKM-IDS20 dataset, which has the smallest data, showed excellent accuracy for all classifiers. For GWO as a selection feature, Random Forest has the highest accuracy at 99.98%. When using the Gain Ratio, Random Forest produces the highest accuracy at 99.91%, and when using Chi-Square, Random Forest also produces the highest accuracy at 99.97%. For the testing process, when GWO is a selection feature, the highest accuracy is generated by Random Forest with an accuracy of 99.97%. Random Forest also produced the highest accuracy when using Chi-Square with an accuracy of 99.91%, and the same classifier also produced the highest accuracy when combined with a Gain Ratio of 96.44%. Table V shows the details result for the UKM-IDS20 dataset.

TABLE V. Accuracy for UKM-IDS20 Dataset

Classifier	Training Accuracy			Testing Accuracy		
	GWO	GR	CS	GWO	GR	CS
Decision Tree(DT)	96.63%	95.14%	96.26%	96.97%	94.83%	96.94%
Random Forest(RF)	99.98%	99.91%	99.97%	99.97%	96.44%	99.91%
Support Vector Machine(SVM)	93.85%	80.22%	93.48%	93.89%	80.89%	93.47%
Logistic Regression(LR)	93.77%	73.95%	90.86%	93.42%	73.55%	90.42%

D. Classifier Comparison

From the extensive and rigorous test results, we can observe the performance of each classifier and compare the accuracy of each dataset used. The accuracy of the testing process for all classifiers against all selection features and all datasets is meticulously compared. The test results using the NSL-KDD Dataset, presented in Table VI, show that the GWO-RF model combination has the highest accuracy, with a value of 99.99% for training and 99.89% for testing.

TABLE VI. Hybrid Model Comparison for NSL-KDD Dataset

Feature	Hybrid Model	Training Accuracy	Testing Accuracy
GWO	**GWO-RF**	**99.99%**	**99.89%**
	GWO-SVM	99.25%	99.24%
	GWO-DT	95.19%	95.09%
	GWO-LR	96.38%	96.25%
CS	CS-RF	99.98%	99.87%
	CS-SVM	99.58%	99.54%
	CS-DT	95.17%	95.03%
	CS-LR	97.67%	97.67%
GR	GR-RF	99.98%	99.86%
	GR-SVM	99.09%	99.05%
	GR-DT	94.59%	94.47%
	GR-LR	95.31%	95.31%

Fig. 5 presents a visual comparison of the accuracy of the NSL-KDD dataset. The Random Forest and SVM classifiers consistently demonstrate the highest accuracy across all feature selection models. However, the GWO-RF model, a combination of the GWO and Random Forest, achieves the best accuracy, underscoring its importance in this context.

On the UKM-IDS20 dataset, the GWO and Random Forest models have the highest accuracy compared to all other feature selection and classifier options, where the resulting accuracy value reaches 99.98% for training and 99.97% for the testing process, as shown in Table VII.

Fig. 6 shows a visual of the accuracy comparison table for the UKM-IDS20 dataset; we can see that Random Forest and Decision Tree are classifiers that consistently have the highest accuracy for the training and testing process, using the GWO, Chi-Square, and Gain Ratio selection features. As in the previous dataset, GWO and Random Forest (GWO-RF) produced the best model combination for testing this UKM-IDS20 dataset.

Table VIII showing in comparison with other studies conducted for both datasets, it was found that the best combination of models obtained in this study is superior in accuracy compared to previous studies.

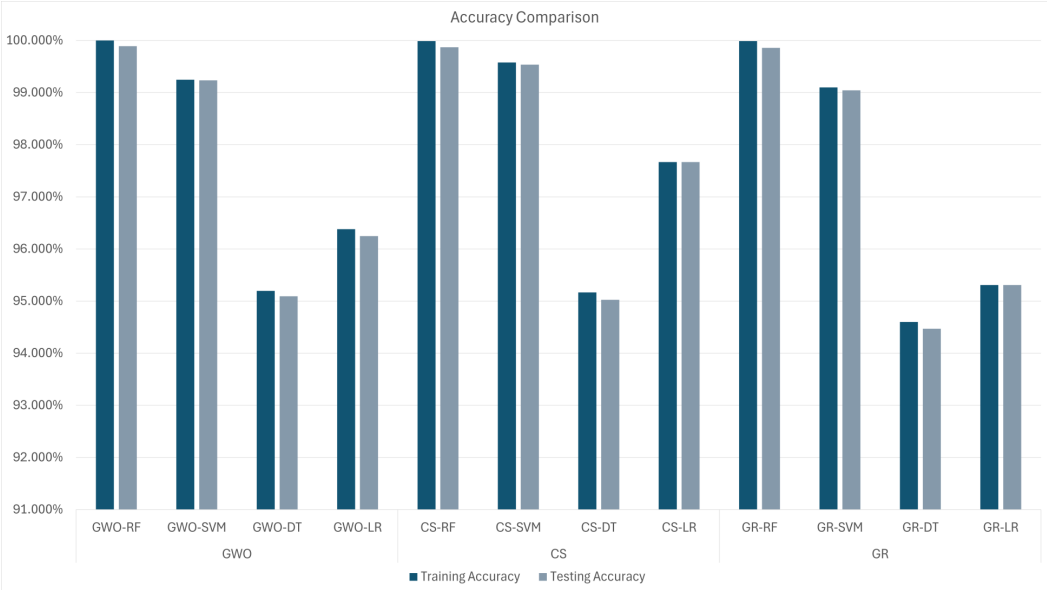

Figure 5. Graphical Accuracy Representation NSL-KDD Dataset

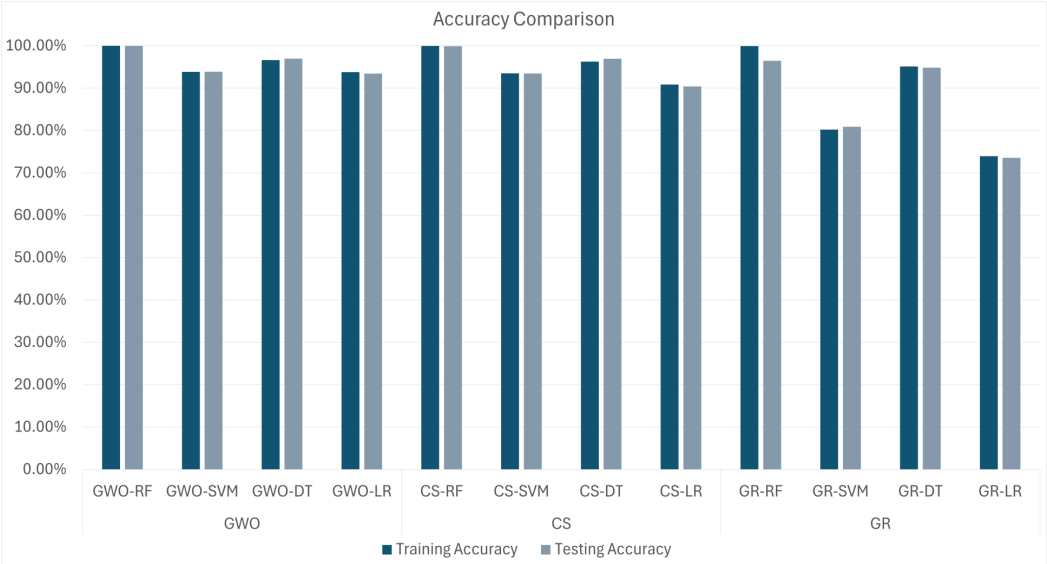

Figure 6. Graphical Accuracy Representation UKM-IDS20 Dataset

Table VIII shows that the previously developed models have good accuracy, with the most accuracy being above 90%. Most of the compared models have high accuracy by utilizing selection features obtained manually or through algorithms. Model SVM-KSE [18] and model Hybrid RF, Gradient Boosting, XGBoost-AES [21] produce high accuracy but do not use specific feature selection techniques. From the existing table and the tests carried out, it can be seen that by using the selection feature, it can produce good accuracy while maintaining average computing time, while without using the selection feature, the selection of the correct classification algorithm needs to be considered so that good accuracy can be produced, but of course at the expense of longer computing time and more significant hardware resources. By looking at these conditions and the accuracy results obtained, the best hybrid classifier and feature selection model from this study (GWO-RF) is more effective in classifying data on both datasets, namely NSL-KDD and UKM-IDS20.

TABLE VII. Hybrid Model Comparison for UKM-IDS20 Dataset

Feature	Hybrid Model	Training Accuracy	Testing Accuracy
GWO	**GWO-RF**	**99.98%**	**99.97%**
	GWO-SVM	93.85%	93.87%
	GWO-DT	96.63%	96.97%
	GWO-LR	93.77%	93.42%
CS	CS-RF	99.97%	99.91%
	CS-SVM	93.48%	93.47%
	CS-DT	96.26%	96.94%
	CS-LR	90.86%	90.42%
GR	GR-RF	99.91%	96.44%
	GR-SVM	80.22%	80.89%
	GR-DT	95.14%	94.83%
	GR-LR	73.95%	73.55%

TABLE VIII. Accuracy Comparison to Other Research

Authors	Model	Dataset	Accuracy
[9]	ML Classifiers RF	NSL-KDD	99.48%
[16]	EGA-PSO+IRF	NSL-KDD	98.97% (bc) 88.14% (mc)
[17]	Random Forest	NSL-KDD	99.1%
[18]	SVM-KSE	NSL-KDD	99.45%
[21]	Hybrid RF, GB, XGBoost-AES	UKM-IDS20	90.85%
[27]	GBDT-GWO	KDD99	96.21%
Proposed Method	**This research best model (GWO-RF)**	**NSL-KDD UKM-IDS20**	**99.89% 99.97%**

4. CONCLUSIONS AND FUTURE WORK

This research used a combination of classifiers with feature selection algorithms and two datasets with different amounts of data but similar features. The tests showed that the Random Forest classifier can be used hybrid with the GWO feature selection algorithm and produces high accuracy with low computation time. In detail, for the NSL KDD dataset, the combined GWO-RF model has the highest result, with 99.99% for training accuracy and 99.89% for testing accuracy. The GWO-RF model outperformed all other feature selection and classifier alternatives on the UKM-IDS20 dataset in terms of accuracy, where the resulting accuracy value reaches 99.98% for training and 99.97% accuracy for the testing process. Furthermore, the performance and capability of deep learning algorithms were compared with the best model in this study to provide a comprehensive understanding of their effectiveness. Other traditional data mining techniques were also examined and compared with it for future research.

REFERENCES

[1] A. Thakkar and R. Lohiya, "A review of the advancement in intrusion detection datasets," in *Procedia Computer Science*, vol. 167. Elsevier B.V., 2020, pp. 636–645.

[2] A. M. AL-Hawamleh, "Advanced spam filtering in electronic mail using hybrid the mini batch k-means normalized mutual information feature elimination with elephant herding optimization technique," *International Journal of Computing and Digital Systems*, vol. 13, pp. 1409–1422, 5 2023.

[3] M. Pallepati, "Network intrusion detection system using machine learning with data preprocessing and feature extraction," *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, pp. 2360–2365, 6 2022.

[4] S. Perera, X. Jin, A. Maurushat, and D. G. J. Opoku, "Factors affecting reputational damage to organisations due to cyberattacks," *Informatics*, vol. 9, 3 2022.

[5] T. Mazhar, H. M. Irfan, S. Khan, I. Haq, I. Ullah, M. Iqbal, and H. Hamam, "Analysis of cyber security attacks and its solutions for the smart grid using machine learning and blockchain methods," 2 2023.

[6] R. F. A. Hweidi and D. Eleyan, "Social engineering attack concepts, frameworks, and awareness: A systematic literature review," *International Journal of Computing and Digital Systems*, vol. 20, pp. 2210–142. [Online]. Available: http://dx.doi.org/10.12785/ijcds/XXXXXX

[7] S. Meftah, T. Rachidi, and N. Assem, "Network based intrusion detection using the unsw-nb15 dataset," *International Journal of Computing and Digital Systems*, vol. 8, pp. 477–487, 2019.

[8] D. M. S. Othman, R. Hicham, and M. M. Zoulikha, "An efficient spark-based network anomaly detection," *International Journal of Computing and Digital Systems*, vol. 9, pp. 1175–1185, 2020.

[9] I. Abrar, Z. Ayub, and F. Masoodi, "A machine learning approach for intrusion detection system on nsl-kdd dataset," *Proceedings of the International Conference on Smart Electronics and Communication (ICOSEC)*, 2020.

[10] N. Mohd, A. Singh, and H. S. Bhadauria, "A Novel SVM Based IDS for Distributed Denial of Sleep Strike in Wireless Sensor Networks," *Wireless Personal Communications*, vol. 111, no. 3, pp. 1999–2022, apr 2020. [Online]. Available: https://doi.org/10.1007/s11277-019-06969-9

[11] A. H. Azizan, S. A. Mostafa, A. Mustapha, C. F. M. Foozy, M. H. A. Wahab, M. A. Mohammed, and B. A. Khalaf, "A Machine Learning Approach for Improving the Performance of Network Intrusion Detection Systems," *Annals of Emerging Technologies in Computing*, vol. 5, no. 5, pp. 201–208, mar 2021.

[12] V. S. Manvith, R. V. Saraswathi, and R. Vasavi, "A performance comparison of machine learning approaches on intrusion detection dataset," in *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, 2021, pp. 782–788.

[13] M. A. Hossain and M. S. Islam, "Ensuring network security with a robust intrusion detection system using ensemble-based machine learning," *Array*, vol. 19, 9 2023.

[14] M. A. Ferrag, L. Maglaras, A. Ahmim, M. Derdour, and H. Janicke, "Rdtids: Rules and decision tree-based intrusion detection system for internet-of-things networks," *Future Internet*, vol. 12, 3 2020.

[15] I. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan, "Intrudtree: A machine learning based cyber security intrusion detection model," *Symmetry*, vol. 12, 5 2020.

[16] A. K. Balyan, S. Ahuja, U. K. Lilhore, S. K. Sharma, P. Manoharan, A. D. Algarni, H. Elmannai, and K. Raahemifar, "A hybrid intrusion detection model using ega-pso and improved random forest method," *Sensors*, vol. 22, 8 2022.

[17] A. S. Ahanger, S. M. Khan, and F. Masoodi, "An effective intrusion detection system using supervised machine learning techniques," in *Proceedings - 5th International Conference on Computing Methodologies and Communication, ICCMC 2021*. Institute of Electrical and Electronics Engineers Inc., 4 2021, pp. 1639–1644.

[18] D. Liang, Q. Liu, B. Zhao, Z. Zhu, and D. Liu, "A clustering-svm ensemble method for intrusion detection system," in *2019 8th International Symposium on Next Generation Electronics (ISNE)*, 2019, pp. 1–3.

[19] M. Mohammadi, T. A. Rashid, S. H. Karim, A. H. M. Aldalwie, Q. T. Tho, M. Bidaki, A. M. Rahmani, and M. Hosseinzadeh, "A comprehensive survey and taxonomy of the svm-based intrusion detection systems," *Journal of Network and Computer Applications*, vol. 178, p. 102983, 2021.

[20] M. Mittal, R. P. de Prado, Y. Kawai, S. Nakajima, and J. E. Muñoz-Expósito, "Machine Learning Techniques for Energy Efficiency and Anomaly Detection in Hybrid Wireless Sensor Networks," *Energies*, vol. 14, no. 11, p. 3125, may 2021.

[21] W. F. Faris and R. R. Mirajkar, "Securing the digital perimeter intrusion detection for robust data protection in cybersecurity," *Research Journal of Computer Systems and Engineering*, vol. 4, pp. 84–92, 6 2023.

[22] S. Lata and D. Singh, "Intrusion detection system in cloud environment: Literature survey future research directions," *International Journal of Information Management Data Insights*, vol. 2, no. 2, p. 100134, 2022.

[23] R. A. N. Diaz, I. K. G. D. Putra, M. Sudarma, I. M. Sukarsa, and N. Jawas, "Comparison of Gain Ratio and Chi-Square Feature Selection Methods in Improving SVM Performance on IDS," *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, vol. 15, no. 1, p. 64, mar 2024.

[24] S. S. Kareem, R. R. Mostafa, F. A. Hashim, and H. M. El-Bakry, "An Effective Feature Selection Model Using Hybrid Metaheuristic Algorithms for IoT Intrusion Detection," *Sensors*, vol. 22, no. 4, p. 1396, feb 2022.

[25] O. Almomani, "A hybrid model using bio-inspired metaheuristic algorithms for network intrusion detection system," *Computers, Materials Continua*, vol. 68, pp. 409–429, 02 2021.

[26] A. Alzaqebah, I. Aljarah, O. Al-Kadi, and R. Damaševičius, "A modified grey wolf optimization algorithm for an intrusion detection system," *Mathematics*, vol. 10, no. 6, 2022. [Online]. Available: https://www.mdpi.com/2227-7390/10/6/999

[27] M. Madhavi and D. R. Nethravathi, "Gradient boosted decision tree (gbdt) and grey wolf optimization (gwo)based intrusion detection model," *Journal of Theoretical and Applied Information Technology*, vol. 31, 2022. [Online]. Available: www.jatit.org

[28] H. Dalmaz, E. Erdal, and H. M. Ünver, "A new hybrid approach using gwo and mfo algorithms to detect network attack," *CMES - Computer Modeling in Engineering and Sciences*, vol. 136, pp. 1277–1314, 2023.

[29] N. D. Cilia, C. De Stefano, F. Fontanella, S. Raimondo, and A. Scotto di Freca, "An experimental comparison of feature-selection and classification methods for microarray datasets," *Information*, vol. 10, no. 3, 2019. [Online]. Available: https://www.mdpi.com/2078-2489/10/3/109

[30] S. J. Pasha and E. S. Mohamed, "Advanced hybrid ensemble gain ratio feature selection model using machine learning for enhanced disease risk prediction," *Informatics in Medicine Unlocked*, vol. 32, p. 101064, 2022.

[31] S. Bahassine, A. Madani, M. Al-Sarem, and M. Kissi, "Feature selection using an improved chi-square for arabic text classification," *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 2, pp. 225–231, 2020.

[32] C.-J. Zhang, X.-Y. Huang, and M.-C. Fang, "Mri denoising by neighshrink based on chi-square unbiased risk estimation," *Artificial Intelligence in Medicine*, vol. 97, pp. 131–142, 2019.

[33] T. A. Alamiedy, M. Anbar, Z. N. Alqattan, and Q. M. Alzubi, "Anomaly-based intrusion detection system using multi-objective grey wolf optimisation algorithm," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 3735–3756, 9 2020.

[34] H. Chantar, M. Mafarja, H. Alsawalqah, A. A. Heidari, I. Aljarah, and H. Faris, "Feature selection using binary grey wolf optimizer with elite-based crossover for arabic text classification," *Neural Computing and Applications*, vol. 32, pp. 12 201–12 220, 8 2020.

[35] P. Nimbalkar and D. Kshirsagar, "Feature selection for intrusion detection system in internet-of-things (iot)," *ICT Express*, vol. 7, no. 2, pp. 177–181, 2021.

[36] K. S. Sahoo, B. K. Tripathy, K. Naik, S. Ramasubbareddy, B. Balusamy, M. Khari, and D. Burgos, "An evolutionary svm model for ddos attack detection in software defined networks," *IEEE Access*, vol. 8, pp. 132 502–132 513, 2020.

[37] P. K. Keserwani, M. C. Govil, E. S. Pilli, and P. Govil, "A smart anomaly-based intrusion detection system for the internet of things (iot) network using gwo–pso–rf model," *Journal of Reliable Intelligent Environments*, vol. 7, pp. 3–21, 3 2021.

[38] R. Divya and R. S. S. Kumari, "Genetic algorithm with logistic regression feature selection for alzheimer's disease classification," *Neural Computing and Applications*, vol. 33, pp. 8435–8444, 7 2021.

RICKY AURELIUS NURTANTO DIAZ graduated with a Bachelor of Computers from the STIKOM Bali Computer Systems Study Program in 2008 and obtained a Master's degree (M.T.) after graduating from the Faculty of Engineering, Udayana University, Information Systems and Computer Management Study Program in 2015. Currently, he has the status of assistant professor in the Computer Systems Study Program at the STIKOM Bali Institute of Technology and Business and teaches Data Communication, Computer Networks and Data Mining courses. His research focus is in the fields of Data Mining, Machine Learning and Networking.

I KETUT GEDE DARMA PUTRA is a Professor at Udayana University since 2014 with a focus on Image Processing, Machine Learning and Data Science. Apart from being active as a teacher in Machine Learning, Image Processing and Artificial Intelligence courses at the Udayana University Engineering Doctoral Study Program, he is also active in various activities in professional organizations, community service, collaboration in the government sector, and the publication of scientific works in journals and conferences.

I MADE SUDARMA is a professor of information technology science at the Electrical Engineering Study Program, Faculty of Engineering, Udayana University at Udayana University since 2019. His research includes internet and web applications, cloud computing, artificial intelligence, data warehousing and data mining, computer graphics and virtual reality, as the author of books and as a reviewer in international and national journals. In addition, he also completed vocational education (IPU., ASEAN Eng) and is active in academic activities, and also active as an Information Technology consultant in local government, private sector, and tourism.

I MADE SUKARSA hold a Doctoral degree from Udayana University, Indonesia, in 2019. He also obtained his Master of Engineering degree from Gajah Mada University, Indonesia, in 2005. He received his S.T degree in informatics engineering from the Gajah Mada University, Indonesia, in 2000 and now he is a Associate Professor at the Department of Information Technology, Faculty of Engineering Udayana, Indonesia. Currently actively as a lecturer and conducting research on IT governance, dialog models on chatbot engines, datawarehouses and system integration.

EMY SETYANINGSIH is a lecturer in the Department of Computer Systems Engineering at AKPRIND University, Indonesia. She obtained a Bachelor's in Computer Science from IST AKPRIND Yogyakarta, Indonesia, in 1996. She has also earned M.Com. and Ph.D. in computer science from Universitas Gadjah Mada (UGM), Yogyakarta, Indonesia, in 2004 and 2019. Her research interests include include cryptography, digital image processing, pattern recognition, and Artificial Intelligence. Scopus ID: 55943273200, Orcid: https://orcid.org/0000-0003-3254-3520.