Enhanced Gaining-Sharing Knowledge Optimization Algorithm for 3D Compression of Intrusion Detection Dataset

Hadeel Qasem Gheni[1](✉) and Wathiq L. Al-Yaseen[2]

[1] Department of Software, Information Technology College, University of Babylon, Babylon, Iraq
wsci.hadeel.qasem@uobabylon.edu.iq

[2] Karbala Technical Institute, Al-Furat Al-Awsat Technical University, 56001 Karbala, Iraq

Abstract. The study addresses challenges associated with intrusion detection datasets in terms of high dimensionality by adopting new methods to reduce their size and improve efficiency. Firstly, the dataset's features were reduced using a selection method based on Pearson Correlation, Entropy, and Information Gain (PC-E-IG) to identify only those with a high correlation to the target class. Secondly, filtering methods were applied to reduce the dataset's records based on two situations. The Gaining–Sharing Knowledge (GSK) optimization algorithm was employed, where its fitness function was replaced via re-combination methods, and the best method was chosen using the Ackley evaluation function. The first method reduced the size of the data by 20% by filtering a group of training data with different fitness values. The second method filtered and isolated records with the Normal target from the training and testing datasets, resulting in a 19.69% and 19.48% reduction in the size of the dataset, respectively. The study used the KDDCup99 dataset and Multilayer Perceptron to test the efficiency of the system and compare its results with the original algorithm, resulting in an impressive increase in model accuracy and a clear reduction in execution time. The study's first reduction method achieved an accuracy of 92.45% and an execution time of 0.09 s. The second reduction method resulted in an accuracy of 93.8% and an execution time of 0.07 s. The main benefit of these results is that they demonstrate the effectiveness of the study's methods in improving the accuracy of intrusion detection models while also reducing execution time.

Keywords: GSK Algorithm · Three Dimension Compression · Fuzzy Recombination · Filtering Layer · Dimensionality Reduction

1 Introduction

Intrusion is accessing and entering the network or a particular computer to spy on the information on it, steal data, change the system, or obtain security holes in the operating system to sabotage and destroy it [1]. For these reasons, there was a need to use a system to detect intrusions and reduce the risks related to network security and personal computer security [2].

A. Abraham et al. (Eds.): ISDA 2023, LNNS 1048, pp. 213–228, 2024.
https://doi.org/10.1007/978-3-031-64650-8_21

An Intrusion Detection System (IDS) is a software or hardware system that monitors networks for suspicious movements and has primary duties: network monitoring, breach detection, and reporting to the administrator [3]. There are three types of ID systems: host, network, and hybrid; each one has its tactic for detecting intrusion and how it protects data, therefore each type has benefits and at the same time has limitations or weaknesses [4].

The amount of data on computer networks has increased, making it more difficult for intrusion detection systems to deal with high dimensions that contain redundant and unnecessary information, which takes time and makes it harder to accurately detect attacks [5]. Not all features are necessary for detecting attacks, where the prediction performance can be improved by reducing the number of features, and also decreasing the amount of time needed for detection and boosting detection rates [6]. Reducing the number of features for the dataset may not lead to the desired result of the system, so, reducing the total number of data is the proposed work for this research by developing an optimization algorithm called the Gaining–Sharing Knowledge algorithm (GSK).

The optimization algorithm is an iterative process that compares different solutions until the best or most acceptable one is identified [7]. The GSK algorithm is founded on the idea of gaining and sharing human knowledge over the course of a person's life [8]. In this algorithm, every person learns from others and engages in social interaction to reap the rewards of learning and impart their expertise where it is more practical to obtain knowledge from their little network, then learn new things and impart them to the most suited people so they can improve their skills [9]. The GSK algorithm contains an important characteristic, it first finds the most useful information present in each node, and then shares the most important of it [10]. This idea was exploited in the same way, but by identifying the node that generates the least percentage of useful information, as it was considered an attack, and this corresponds to the idea of intrusion. The weak point of this algorithm is that its kernel needs high complexity because of its stages. Therefore, replacing the kernel according to the type of problem is beneficial.

Despite the extensive research on IDSs, there are still many crucial issues to be resolved [11], including low detection rates, false alarm rates, response time, and unbalanced datasets [12].

In our research, we worked to handle the unbalanced data and thus increase the accuracy in detecting malicious activities and reduce the time required for this by reducing the size of the data in three dimensions, value, feature, and record.

The remaining parts of the essay are arranged as follows. Section 2 went over the optimization technique employed in this work. The proposed model is presented in Sect. 3. Section 4 displays the performance assessment. In Sect. 5, the conclusion and future work are introduced.

2 Gaining-Sharing Knowledge Optimization Algorithm

Gaining-Sharing Knowledge (GSK) is a revolutionary optimization algorithm based on human knowledge that has been created in recent years [13]. The GSK algorithm is built on how people learn and impart knowledge throughout their lives [14]. Junior and Senior are the two major stages of GSK, representing gaining knowledge and sharing it

[15]. Initially, all members of the population do not possess any knowledge (the junior level), but as soon as they begin to interact with the environment around them, they will acquire knowledge and then be able to share it with other people; thus, the interaction will increase, they will acquire more knowledge, and they can reach the senior level [16]. In the GSK algorithm, the first population is generated at random within border restrictions [17], and then the dimensions of the two stages are calculated. The junior dimension is as follows:

$$D_j = D \times (1 - (\frac{G}{G_{\max}}))^k \tag{1}$$

where: D_j is the dimension of the junior stage, D is the dimension of population, *Gmax* is the max number of generations, *G* is the current generation, and *k* is the knowledge rate that determines the experience rate [17].

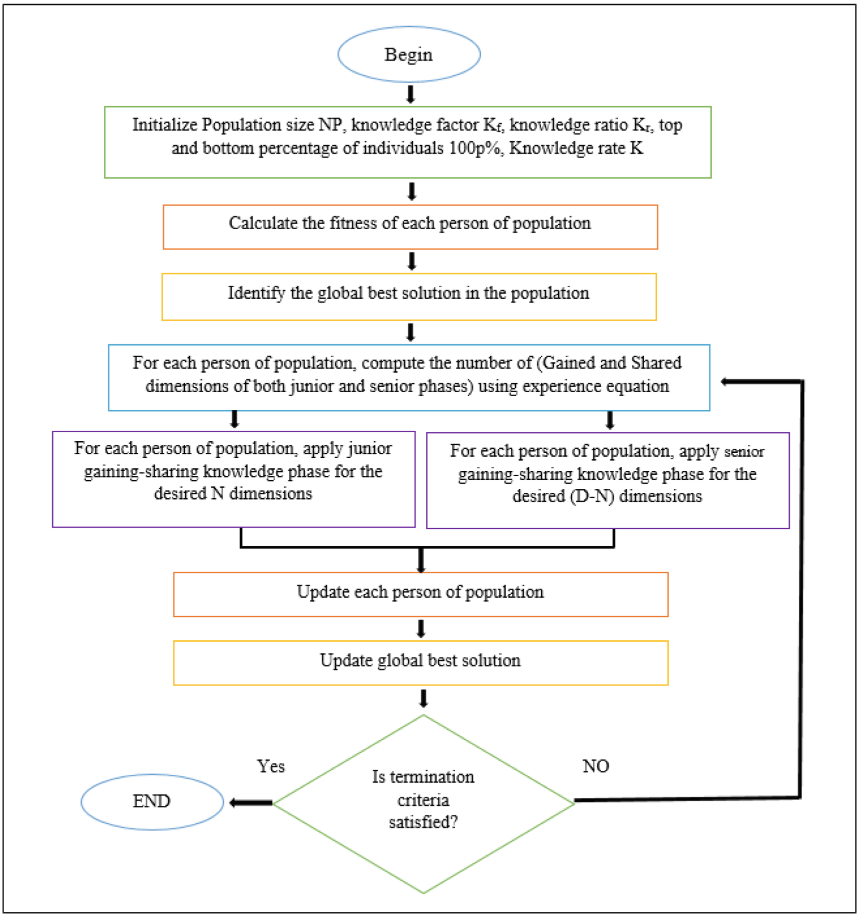
Begin
Initialize Population size NP, knowledge factor K_f, knowledge ratio K_r, top and bottom percentage of individuals 100p%, Knowledge rate K
Calculate the fitness of each person of population
Identify the global best solution in the population
For each person of population, compute the number of (Gained and Shared dimensions of both junior and senior phases) using experience equation
For each person of population, apply junior gaining-sharing knowledge phase for the desired N dimensions
For each person of population, apply senior gaining-sharing knowledge phase for the desired (D-N) dimensions
Update each person of population
Update global best solution
Is termination criteria satisfied?
Yes
NO
END

Fig. 1. Diagram of GSK Algorithm [18]

The senior dimension is as follows:

$$D_s = D - Dj \tag{2}$$

The population will enter the junior stage, where people learn from their surroundings and share what they've learned with others. Then, continue to the next stage, the senior stage, in which the impact of other people on the individual will be calculated, whether this effect is positive or negative. Figure 1 illustrates the GSK algorithm flowchart.

3 EGSK-CIDD Model

This paper presents a model called EGSK-CIDD that consists of four layers as explained in Fig. 2. The first layer is called Preprocessing and its main functions are (checking for missing values, transformation, normalization, and feature selection). The second layer is called *GSK-DK* and its main functions are (finding the best fitness function, developing GSK by using a different kernel, computing distance, and data grouping). The third layer is called *Filtering* and its main functions are (filtering distinct fitness values, passing similar fitness values to the next layer, filtering normal data, and passing normal data to post-processing). The fourth layer is called MLP and its main functions are (training the model to find types of attack, reducing model complexity, and increasing model accuracy).

3.1 Description of the KDDCup99 Dataset

One of the most popular datasets that have been used to evaluate the work of intrusion detection techniques is the KDDCup99 dataset, which has been in use since 1999 [19]. The 57 different attack types in the KDDCup99 dataset (four main attacks and their subcategories) make it a big reservoir of attack vectors and IDSs [20]. KDDCup99 contains two separate datasets, one for training with size 494,021 and the other for testing with size 311,029. Table 1 explains the size of Normal and each type of attack in both the training and testing datasets used in the work [21].

It is clear from Table 1 that the number of attack types in the training dataset is 22, while in the test dataset is 39, which means that there are 17 types of attacks present in the test dataset and not present in the training dataset.

3.2 Preprocessing the Datasets

For the KDDCup99 training and testing datasets, after reading the dataset from the file, it is subjected to several steps to prepare it for the later stages. The preprocessing stage includes:

Checking for Missing Values. It is necessary to make sure that the data is free of missing values.

Transformation from Categorical to Numerical Data. For the classification algorithms to deal with the categorical data, it must be converted into numerical data. Therefore, the symbolic values in each feature will be given a serial numerical value starting from zero.

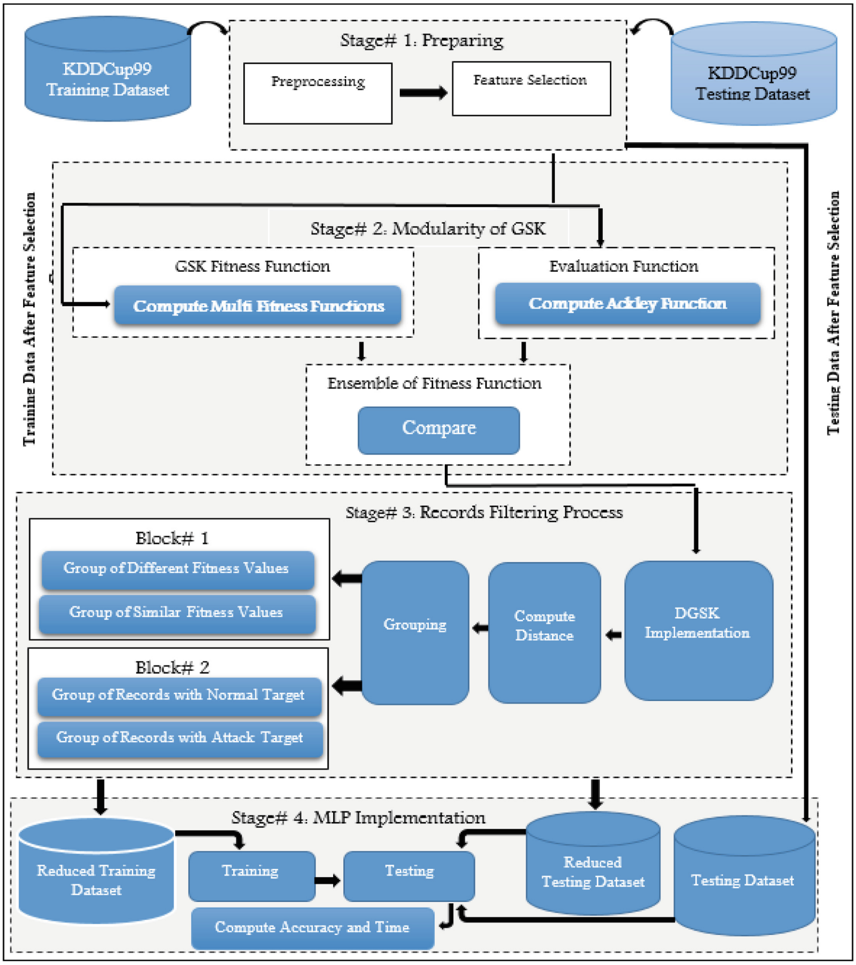
KDDCup99 Training Dataset
Stage# 1: Preparing
Preprocessing
Feature Selection
KDDCup99 Testing Dataset
Training Data After Feature Selection
Testing Data After Feature Selection
Stage# 2: Modularity of GSK
GSK Fitness Function
Compute Multi Fitness Functions
Evaluation Function
Compute Ackley Function
Ensemble of Fitness Function
Compare
Stage# 3: Records Filtering Process
Block# 1
Group of Different Fitness Values
Group of Similar Fitness Values
Block# 2
Group of Records with Normal Target
Group of Records with Attack Target
Grouping
Compute Distance
DGSK Implementation
Stage# 4: MLP Implementation
Reduced Training Dataset
Training
Testing
Reduced Testing Dataset
Testing Dataset
Compute Accuracy and Time

Fig. 2. Block Diagram of EGSK-CIDD

Table 1. The extent of normal and types of attacks in the KDDCup99 datasets

Dataset	Normal	Attack Categories					No. Attack Types
		DoS	R2L	Probe	U2R	Total	
Train Data	97,278	391,458	1,126	4,107	52	494,021	22
Test Data	60,593	229,853	16,347	4,166	70	311,029	39

Normalization. Bad data models may result when the values of many attributes are measured on different scales, so, they are normalized to put all the qualities on the same scale. The Min-Max normalization method is used to compress the values and prepare them for the next step according to the following formula.

$$Xnew = (X - Xmin) / (Xmax - Xmin) \tag{3}$$

Features Selection. To compress the data in terms of feature dimension and choose the important ones only. To do this task**,** Pearson Correlation, Entropy, and Information Gain were employed between each feature and target. After calculating these three metrics, the number of features that have useful information about the target class is 20 from 41. Note that the testing dataset should consist of the same features as a training dataset. Therefore, the process of selecting features in both datasets should result in keeping the same features.

3.3 Modularity of GSK

Modifying the performance of the GSK algorithm by changing its kernel by computing fitness functions such as Recombination methods (Discrete, Extended Line, Extended Intermediate, and Fuzzy) and other functions, then testing their performance using the Ackley function to choose the best one.

Discrete Recombination. Implements a value transaction between the individuals by exchanging variable values, where for each place, every parent who provides its variable to the child is randomly selected with an equal probability [22].

Suppose that X1 = {X11, X12,….,X1n), Y1 = (Y11,Y12,….,Y1n) are the parents, then the offspring is: Z1 = {Z11, Z12,…., Z1n}. The parents are picked with an equal probability by [23].

$$Zi\,\varepsilon\{Xi,\ Yi\} \tag{4}$$

Extended Line Recombination. With this crossover technique, new gene values based on both parents' genes can be obtained rather than only copying genes from one parent [24] as seen in Eq. (5) [25]. It takes the parents as points in the design space and anchors the new gene to the line connecting the two points [24].

$$Zi = Xi \times \alpha + Yi \times (1 - \alpha) \tag{5}$$

whereas, Zi is the variables of the new individual; Xi and Yi reflect the parents' variables; and α is a proportionality constant that is picked at random from the range [–0.25, 1.25] [25].

Extended Intermediate Recombination. The entire hypercube that takes shape between the parents, as opposed to just the line, is a potential candidate for having offspring [24]. as seen in Eq. (6) [26].

$$Zi = (1 - \alpha i)\,Xi + \alpha i\,Yi \tag{6}$$

where i = (1, …, n), αi is chosen at random from the range [–0.25, 1.25], the difference between αi and α in Eq. (5) is that a new α is chosen for each i [26].

Fuzzy Recombination. The operators of Fuzzy Recombination successfully preserved population diversity while accelerating convergence [27]. The likelihood that the offspring will have the value zi is calculated as seen in Eq. (7) [28].

$$P(zi)\,\{\varnothing(xi),\ \varnothing\,(yi)\} \tag{7}$$

The polynomial function is used in this method as a membership function, as seen in Eq. (8) [23].

$$F(z) = \ 1\,/\,\sqrt{}2\pi\ exp\,(-Zi * 2/2) \tag{8}$$

In addition to the recombination methods, Pearson Correlation, Entropy, and Information Gain were also calculated as a fitness function.

Correlation. The correlation between each record and target class will be computed which represents the fitness value of that record in the dataset.

Entropy. The statistical variation for each record will be computed separately by using the entropy law.

Information Gain. For each record in the dataset, the information gain will be computed based on the amount of useful information in that record regarding the target class and determined as the fitness value of that record.

Evaluation by Ackley Function. To test optimization strategies, a widely used function for this purpose is the Ackley function [29]. This test function is highly multimodal, scalable, continuous, and non-separable [30]. An exponential function and an amplified cosine function are typically superimposed to create a nonlinear multimodal function [31]. The Ackley function is as Eq. (9) [32].

$$f(X) = -aexp(-b\sqrt{1/d\sum\nolimits_{i=1}^{d} Xi^2}) - exp(1/d\sum\nolimits_{i=1}^{d} cos(cXi)) + a + exp(1) \tag{9}$$

where: d is the number of features, a = 20, b = 0.2, c = 2 π, Xi = (X_1, X_2,.., X_d).

Ensemble of Fitness Functions. The seven fitness functions that are calculated will be compared and tested under the delusion of the Ackley function to determine the best one that corresponds to it and choose it as the new kernel for the GSK algorithm.

3.4 Records Filtering Process

EGSK Implementation. The new kernel will take its place in the GSK algorithm to update the fitness values of the records according to the percentage of the least important information that will be shared between the nodes.

Grouping. The data will be clustered into groups and two new methods for filtering will be applied. The presence of a number of records with similar characteristics within the same dataset does not provide anything for the algorithm as long as there is a record (at least one) that carries these characteristics, which is useful for training the algorithm on it, and therefore there is no benefit if there is another record with the same characteristics for the algorithm, so, it is preferable to filter these records from the dataset in order to decrease the time and increase the accuracy.

3.5 Multilayer Perceptron Implementation

To build the model that will perform the training and testing process, we used Multilayer Perceptron (MLP), which is a feed-forward neural network that transfers data from the input layer to the output layer in a forward direction [33]. The three layers of MLP constitute the basic structure of the Artificial Neural Network (ANN) [34]. The input layer is responsible for receiving the data to be processed, then sending it to the hidden layer that performs the actual computations, and then to the output layer that gives the result of the required task such as classification or prediction [35]. Neurons in each layer are connected to their neighbors using weights and a bias is used to provide a threshold to activate neurons [36]. Problems that cannot be solved linearly can be solved using MLPs, which are designed to approximate any continuous function [37]. Figure 3 shows a simple architecture of MLP.

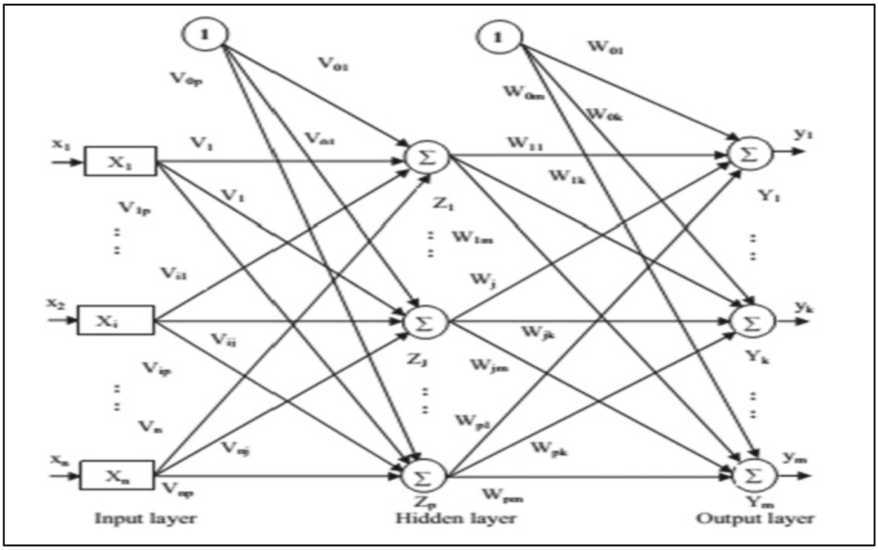

Fig. 3 MLP architecture [38]

The weights are set in random before the training begins. The data in the training set is consists of X_1, X_2 which are the input data, and Y which is the expected output. The calculation of output is by the following Equation:

$$Y = WX + B \tag{10}$$

where W is the weight and B is the bias.

The model was fed with the training dataset after reducing its size in terms of features and records. The accuracy of the model and the time it needs will be calculated.

4 Implementation of the EGSK-CIDD Model

In Stage#1 of the EGSK-CIDD model, after reading the KDDCup99 training and testing dataset files, the first step is to check the datasets for missing values and then convert categorical data to numerical data to make the dataset easy to handle by the algorithm.

Protocol type, service, and flag are three symbolic-valued features that have been transformed to have numerical values in the second step. For instance, the three values for the protocol type feature (ICMP, TCP, UDP) are transformed to 0, 1, and 2, respectively. The same strategy is used for the other features. Attack categories of the same type are given the same numerical code, assuming that the values of main attack names are Normal = 0, DoS = 1, Probe = 2, U2R = 3, and R2L = 4.

The original data is transformed linearly by min-max normalization in the next step taking into account the smallest and largest value in each feature and performing the scaling. All scaled data will be within the range (0, 1). The most and the least important features after performing PC-E-IG which results in a set of 20 features are shown in Table 2.

Table 2. The most and least important features in the KDDCup99 dataset

Most Important Features	Least Important Features
service	duration
flag	protocol
src_bytes	dst_bytes
hot	land
num_failed_logins	wrong_fragment
root_shell	urgent
is_guest_login	logged_in
count	num_compromised
srv_count	su_attempted
serror_rate	num_root
srv_serror_rate	num_file_creations
rerror_rate	num_shells
srv_rerror_rate	num_access_files
diff_srv_rate	num_outbound_cmds
dst_host_count	is_host_login
dst_host_same_src_port_rate	same_srv_rate
dst_host_serror_rate	srv_diff_host_rate
dst_host_srv_serror_rate	dst_host_srv_count

(*continued*)

Table 2. (*continued*)

Most Important Features	Least Important Features
dst_host_rerror_rate	dst_host_same_srv_rate
dst_host_srv_rerror_rate	dst_host_diff_srv_rate
	dst_host_srv_diff_host_rate

During Stage#2, the most important features undergo two parallel phases: calculating the fitness function using the seven metrics described earlier and evaluating the fitness function. Each metric is applied to each record of the important features, resulting in a fitness value for each one. A sample of the fitness values for each metric is presented in Table 3. The main benefit of this approach is that it provides a more comprehensive and accurate method for assessing the fitness of intrusion detection models, which can lead to improved performance and increased security.

Table 3. Fitness function calculations

Entropy	Inf. Gain	Cor	Junior	Senior	Ex. Line	Ex. Int	Fuzzy	Ack
0.44176	3.51E + 0	1	0.1752	0.0013	0.00557	0.00369	0.3989	0.3144
0.41792	3.10E + 0	1	0.1550	0.0008	0.00492	0.00613	0.3989	0.3126
0.44009	3.47E + 0	1	0.1736	0.0013	0.00552	0.00209	0.3989	0.3108
0.46989	4.08E + 0	1	0.2040	0.0023	0.00650	0.00667	0.3987	0.3085
0.49139	4.64E + 0	1	0.2320	0.0035	0.00741	0.01218	0.3986	0.3050
0.51213	5.39E + 0	1	0.2697	0.0051	0.00863	0.00465	0.3985	0.2926
-0.0142	1.83E + 0	1	0.9168	0.0929	0.03067	0.01933	0.3911	0.2900
0.41401	3.03E + 0	1	0.1517	0.0009	0.00482	0.00409	0.3989	0.2926
0.44805	3.62E + 0	1	0.1811	0.0015	0.00576	0.00133	0.3988	0.2925
0.47757	4.29E + 0	1	0.2144	0.0016	0.00681	0.01019	0.3988	0.2924
0.46529	4.00E + 0	1	0.1999	0.0010	0.00635	-0.00025	0.3989	0.2925
-7.59802	4.64E + 0	−1	0.1773	0.0167	0.11845	0.19450	0.3712	0.1966
-7.89533	5.39E + 0	–1	0.1812	0.0172	0.12108	0.20351	0.3705	0.1964
-8.85905	4.29E + 0	–1	0.1934	0.0189	0.12940	0.09078	0.3679	0.1859
-9.10412	4.00E + 0	–1	0.1963	0.0193	0.13146	0.04444	0.3672	0.1842
-9.27305	3.05E + 0	–1	0.1984	0.0197	0.13287	0.01374	0.3667	0.1824

After testing various fitness functions, it is found that the Fuzzy Recombination function is the best fit for the Ackley function and is chosen as the kernel for the GSK algorithm. Figure 4 provides a visual representation for 100 records of the rationale for using the Fuzzy fitness function as the GSK algorithm's kernel. The main benefit of this approach is that it provides a more effective method for identifying and selecting important records for intrusion detection models, improving their accuracy and efficiency.

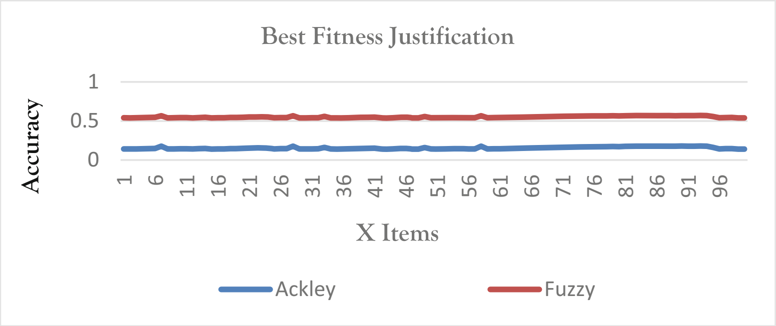

Fig. 4. Fuzzy recombination evaluation by Ackley function

In Stage#3, the GSK algorithm is used to arrange the population based on their fitness values and identify individuals with the best traits and behaviors. The records are then rearranged in ascending order according to their updated fitness values to identify nodes that generate the least percentages of useful information, which are considered attacks. To compress the data in the record dimension, the next step is to cluster the data into groups using Euclidean distance. There are two case studies to consider. The training dataset's size is reduced by filtering records with unequal distances (unequal fitness values) from those with equal distances (equal fitness values), resulting in two groups. The group with unequal fitness values is filtered, reducing the training dataset's size by 20%, while the second group of records with equal fitness values is passed to the next stage. The main benefit of this method is that it reduces the size of the training dataset by identifying and filtering redundant records, resulting in a more efficient and accurate intrusion detection system. Table 4 illustrates a sample of distances while both the training and testing dataset's size is reduced by filtering records with the Normal target. The test dataset undergoes the same stages as the training dataset, including calculating fitness values, evaluating these values using the Ackley function, replacing the kernel of the GSK algorithm, updating fitness values, and grouping. The group that comprises records with a Normal target is filtered from both training and testing datasets and results in reducing the dataset's size by 19.69% and 19.48%, respectively, while the second group is passed to the next stage. The main benefit of this method is that it reduces the size of the datasets by identifying and filtering records with a Normal target, resulting in improved efficiency and accuracy of intrusion detection systems.

In Stage#4, the MLP algorithm is used to train and test the dataset in three cases: the original training dataset after preprocessing, the training dataset after feature selection, and the dataset after record filtering. Accuracy and testing time are measured and compared for the three cases, as shown in Table 5.

Table 4. The distance between the fitness values and their count

Distance	Count	Distance	Count
0	395191	0.000129939	1
0.000108755	1	0.000144297	1
0.000109891	1	0.000147038	1
0.000109974	1	0.000149003	1
0.00011651	1	0.000154448	1
0.000119694	1	0.000154771	1
0.000157773	1	0.000177615	5
0.000164153	1	0.000179542	5
0.000168385	1	0.000180747	13

Table 5. Accuracy and testing time of the MLP model

Dataset	Accuracy	Time/Second
After Preprocessing	91.53	0.15
After Features Selection	92.29	0.12
After Fitness Values Records Filtering	92.45	0.09
After Normal Filtering	93.8	0.07

It is clear from the above table that the proposed method is efficient in terms of increasing accuracy and reducing the time required by the model. Figure 6 and Fig. 6 compare the accuracy and time of the four methods, respectively.

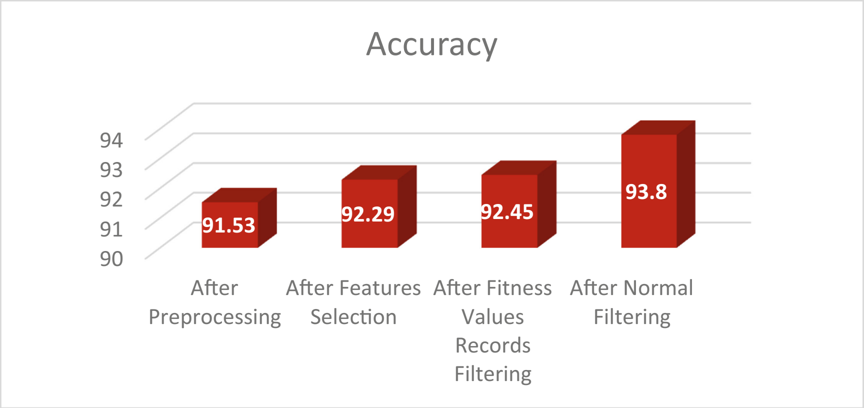

Fig. 5. Accuracy comparison

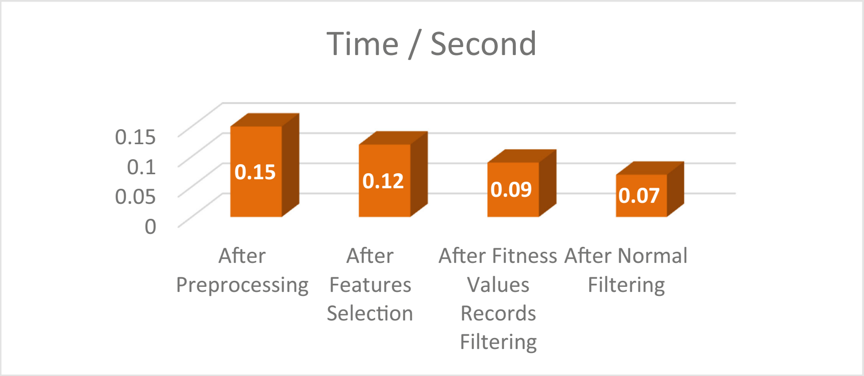

Fig. 6. Time comparison in second

Figure 4 and Fig. 6 clearly show that the proposed model for reducing dataset size has significantly improved accuracy and reduced implementation time. The results also indicate that filtering records with Normal targets yields better results than filtering records with different fitness values. While both methods were effective, the Normal filtering method proved to be superior. The main benefit of this approach is that it provides a reliable and efficient method for reducing dataset size and improving the accuracy of intrusion detection models, which can lead to better system performance and increased security.

5 Conclusion

The study proposes two new methods to reduce the size of intrusion detection datasets, which are known to contain redundant and unimportant data. The GSK algorithm is developed and employed to identify nodes generating the least percentage of useful information, which are considered attacks. The research aims to reduce the dataset size from three dimensions to improve performance by reducing running time. The KDDCup99 datasets are used, and PC-E-IG metrics are used to select the most essential features, resulting in a dataset with only 20 features in both training and testing datasets. The kernel of the GSK algorithm is replaced with a Fuzzy Recombination function chosen based on seven metrics and the Ackley function. The DGSK algorithm is then used to identify unimportant records by finding similar fitness values and filtering records with unequal fitness values, resulting in a 20% reduction in dataset size. Filtering records with Normal targets from both datasets also reduces the training dataset's size by 19.69% and the testing dataset's size by 19.48%.

The MLP algorithm is used to train and test the model in three situations, resulting in increased accuracy and reduced time requirements. Filtering records with a Normal target is found to be a better method than filtering records with different fitness values. The main benefit of this approach is that it improves the efficiency and accuracy of intrusion detection systems by reducing dataset size and running time. Future research can explore the use of another deep learning algorithm to further improve intrusion detection system performance.

References

1. Meftah, S., Rachidi, T., Assem, N.: Network based intrusion detection using the UNSW-NB15 dataset. Inter. J. Comput. Digital Syst. **8**(5), 478–487 (2019). https://doi.org/10.12785/ijcds/080505
2. Huang, S., Lei, K.: IGAN-IDS: An imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks. Ad Hoc Netw. **105**, 102177 (2020). https://doi.org/10.1016/j.adhoc.2020.102177
3. Kocher, G., Kumar, G.: Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges. Soft. Comput. **25**(15), 9731–9763 (2021). https://doi.org/10.1007/s00500-021-05893-0
4. Alazab, A., Hobbs, M., Abawajy, J., Alazab, M.: Using feature selection for intrusion detection system. In: 2012 international symposium on communications and information technologies (ISCIT), pp. 296–301. IEEE (2012). https://doi.org/10.1109/ISCIT.2012.6380910
5. Almasoudy, F.H., Al-Yaseen, W.L., Idrees, A.K.: Differential evolution wrapper feature selection for intrusion detection system. Proc. Comput. Sci. **167**, 1230–1239 (2020). https://doi.org/10.1016/j.procs.2020.03.438
6. Almomani, O.: A feature selection model for network intrusion detection system based on PSO, GWO. FFA and GA algorithms. Symmetry **12**(6), 1046 (2020). https://doi.org/10.3390/sym12061046
7. Al-Janabi, S., Alkaim, A.: A novel optimization algorithm (Lion-AYAD) to find optimal DNA protein synthesis. Egyptian Inform. J. **23**(2), 271–290 (2022). https://doi.org/10.1016/j.eij.2022.01.004
8. Agrawal, P., Ganesh, T., Mohamed, A.W.: A novel binary gaining–sharing knowledge-based optimization algorithm for feature selection. Neural Comput. Appl. **33**(11), 5989–6008 (2021). https://doi.org/10.1007/s00521-020-05375-8
9. Agrawal, P., Ganesh, T., Mohamed, A.W.: Chaotic gaining sharing knowledge-based optimization algorithm: an improved metaheuristic algorithm for feature selection. Soft. Comput. **25**(14), 9505–9528 (2021). https://doi.org/10.1007/s00500-021-05874-3
10. Mohammed, G.S., Al-Janabi, S.: An innovative synthesis of optmization techniques (FDIRE-GSK) for generation electrical renewable energy from natural resources. Res. Eng. **16**, 100637 (2022). https://doi.org/10.1016/j.rineng.2022.100637
11. Khraisat, A., Gondal, I., Vamplew, P., Kamruzzaman, J.: Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity **2**(1), 1–22 (2019). https://doi.org/10.1186/s42400-019-0038-7
12. Aljanabi, M., Ismail, M.A., Ali, A.H.: Intrusion detection systems, issues, challenges, and needs. Inter. J. Comput. Intell. Syst. **14**(1), 560–571 (2021). https://doi.org/10.2991/ijcis.d.210105.001
13. Al-Janabi, S., Majed, H., Mahmood, S.: One step to enhancement the performance of XGBoost through GSK for prediction ethanol, ethylene, ammonia, acetaldehyde, acetone, and toluene. In: Data Science for Genomics, pp. 179–203, Academic Press (2023). https://doi.org/10.1016/B978-0-323-98352-5.00011-2
14. Agrawal, P., Ganesh, T., Mohamed, A. W.: Solving knapsack problems using a binary gaining sharing knowledge-based optimization algorithm. Complex Intell. Syst., 1–21 (2021). https://doi.org/10.1007/s40747-021-00351-8
15. Majed, H., Al-Janabi, S., Mahmood, S.: Data Science for Genomics (GSK-XGBoost) for Prediction Six Types of Gas Based on Intelligent Analytics. In: 2022 22nd International Conference on Computational Science and Its Applications (ICCSA), pp. 28–34. IEEE. (2022). : https://doi.org/10.1109/ICCSA57511.2022.00015

16. Mohamed, A.W., Abutarboush, H.F., Hadi, A.A., Mohamed, A.K.: Gaining-sharing knowledge based algorithm with adaptive parameters for engineering optimization. IEEE Access **9**, 65934–65946 (2021). https://doi.org/10.1109/ACCESS.2021.3076091
17. Hassan, S.A., Ayman, Y.M., Alnowibet, K., Agrawal, P., Mohamed, A.W.: Stochastic travelling advisor problem simulation with a case study: a novel binary gaining-sharing knowledge-based optimization algorithm. Complexity **2020**, 1–15 (2020). https://doi.org/10.1155/2020/6692978
18. Mohamed, A.W., Hadi, A.A., Mohamed, A.K.: Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. Int. J. Mach. Learn. Cybern. **11**(7), 1501–1529 (2020). https://doi.org/10.1007/s13042-019-01053-x
19. Chandrashekhar, A.M., Raghuveer, K.: Performance evaluation of data clustering techniques using KDD Cup-99 Intrusion detection data set. Inter. J. Inform. Netw. Sec. **1**(4), 294 (2012)
20. Vasudevan, A., Harshini, E., Selvakumar, S.: SSENet-2011: a network intrusion detection system dataset and its comparison with KDD CUP 99 dataset. In: 2011 second asian himalayas international conference on internet (AH-ICI), pp. 1–5. IEEE. (2011). https://doi.org/10.1109/AHICI.2011.6113948
21. Al Mehedi Hasan, M., Nasser, M., Pal, B.: On the KDD'99 dataset: support vector machine based intrusion detection system (ids) with different kernels. Int. J. Electron. Commun. Comput. Eng. 4(4), 1164–1170 (2013)
22. Schlierkamp-Voosen, D., Mühlenbein, H.: Predictive models for the breeder genetic algorithm. Evol. Comput. **1**(1), 25–49 (2013). https://doi.org/10.1162/evco.1993.1.1.25
23. Kaghed, N. H., Abbas, T. A., Ali, S. H.: Design and implementation of classification system for satellite images based on soft computing techniques. In: 2006 2nd international Conference on Information & Communication Technologies, vol. 1, pp. 430–436. IEEE. (2006). https://doi.org/10.1109/ICTTA.2006.1684408
24. de Zeeuw, S.: Effective Design Space Exploration: Exploration of the UV curing process for uniform UV distribution in commercial printers
25. Li, P., Wang, H.: A novel strategy for the crossarm length optimization of PSSCs based on multi-dimensional global optimization algorithms. Eng. Struct. **238**, 112238 (2021). https://doi.org/10.1016/j.engstruct.2021.112238
26. Qin, Y., Huangfu, W., Zhang, H., Long, K., Yuan, J.: Rethinking cellular system coverage optimization: a perspective of pseudometric structure of antenna Azimuth variable space. IEEE Syst. J. **15**(2), 2971–2979 (2020). https://doi.org/10.1109/JSYST.2020.2990320
27. Naqvi, F. B., Shad, M. Y.: Seeking a balance between population diversity and premature convergence for real-coded genetic algorithms with crossover operator. Evolutionary Intell., 1–16 (2021). https://doi.org/10.1007/s12065-021-00636-4
28. Voigt, H.M., Mühlenbein, H., Cvetković, D.: Fuzzy Recombination for the Breeder Genetic Algorithm. In: Eshelman, L. (ed.) Proceedings of the Sixth International Conference on Genetic Algorithms, pp. 104–111. Morgan Kaufmann Publishers, San Francisco (1995)
29. Cheng, J., Shi, T.: Structural optimization of transmission line tower based on improved fruit fly optimization algorithm. Comput. Electr. Eng. **103**, 108320 (2022). https://doi.org/10.1016/j.compeleceng.2022.108320
30. Abiyev, R.H., Tunay, M.: Optimization of high-dimensional functions through hypercube evaluation. Comput. Intell. Neurosci. **2015**, 17 (2015). https://doi.org/10.1155/2015/967320
31. Cai, W., Yang, L., Yu, Y.: Solution of ackley function based on particle swarm optimization algorithm. In: 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), pp. 563–566. IEEE (2020).. https://doi.org/10.1109/AEECA49918.2020.9213634
32. Molga, M., Smutnicki, C.: Test functions for optimization needs. Test functions for optimization needs **101**, 48 (2005). http://www.robertmarks.org/Classes/ENGR5358/Papers/functions.pdf

33. Raj, P., David, P.E.: The digital twin paradigm for smarter systems and environments: The industry use cases. Academic Press (2020)
34. Noh, J., Badloe, T., Lee, C., Yun, J., So, S., Rho, J.: Inverse design meets nanophotonics: From computational optimization to artificial neural network. Intell. Nanotechnol., 3–32 (2023)
35. Abinaya, S., Devi, M.K.: Enhancing crop productivity through autoencoder-based disease detection and context-aware remedy recommendation system. In: Application of Machine Learning in Agriculture, pp. 239–262. Academic Press (2022)
36. Rajamanickam, R., Baskaran, D.: Neural network model for biological waste management systems. In: Current Trends and Advances in Computer-Aided Intelligent Environmental Data Engineering, pp. 393–415. Academic Press (2022)
37. Menzies, T., Kocagüneli, E., Minku, L., Peters, F., Turhan, B.: Using goals in model-based reasoning. Sharing Data Models Softw. Eng. **1**, 321–353 (2015)
38. Mohanty, M.D., Mohanty, M.N.: Verbal sentiment analysis and detection using recurrent neural network. In: Advanced Data Mining Tools and Methods for Social Computing, pp. 85–106. Academic Press (2022)