International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

An Intelligent Hybrid GA-PI Feature Selection Technique for Network Intrusion Detection Systems

[1]Sowmya, [2]T. Mary Anita

Submitted: 21/03/2023 **Revised**: 25/05/2023 **Accepted**: 10/06/2023

Abstract: The development of Network Intrusion Detection Systems (NIDS) has become increasingly important due to the growing threat of cyber-attacks. However, with the vast amount of data generated in networks, handling big data in NIDS has become a major challenge. To address this challenge, this research paper proposes an intelligent hybrid GA-PI algorithm for feature selection and classification tasks in NIDS using support vector machines (SVM). The proposed approach is evaluated using two sub-datasets, Analysis and Normal, and Reconnaissance and Normal, which are generated from the publicly available UNSWNB-15 dataset. In this work, instead of considering all possible attacks, the focus is on two attacks, emphasizing the importance of the feature selection agent in determining the optimal features based on the attack type. The experimental results show that the proposed hybrid feature selection approach outperforms existing methodologies in terms of accuracy and execution time. Moreover, the selection of features can be subjective and dependent on the domain knowledge of the researcher. Additionally, the proposed approach requires computational resources for feature selection and classification tasks, which can be a limitation for resource-constrained systems. To be brief, this research paper presents a promising approach for feature selection and classification tasks in NIDS using an intelligent hybrid GA-PI algorithm. While there are some challenges and limitations, the proposed approach has the potential to contribute to the development of effective and efficient NIDS.

Keywords: *Network Intrusion Detection Systems (NIDS), Feature selection, Support vector machines (SVM), Hybrid GA-PI algorithm*

1. Introduction

In today's digital world, computer networks play a critical role in the transfer of information, but many of the interconnected devices are susceptible to attackers. Network security ensures the integrity and confidentiality of data transferred through these networks. It is crucial to provide a framework that can protect both wired and wireless networks [1]. Network security allows us to provide a defense mechanism in the form of either hardware or software agents to protect against intrusion mechanisms. Nowadays, machine learning (ML) based mechanisms are popular to detect attacks in an efficient way, compared to traditional intrusion mechanisms. However, with the recent advancement of networks, hackers are identifying loopholes and launching attacks on the network. Therefore, it is essential to build a robust intrusion detection mechanism that can detect these attacks and alert the administrator by raising an alarm [2].

One of the ways to achieve this is through a network intrusion detection system (NIDS), which provides real-time monitoring of data by capturing highly complicated network traffic. Signature-based IDS is the most commonly used and prominent NIDS, but its major disadvantage is that the entire system needs constant updating of new signatures [3]. By constantly updating these signatures, the system can detect new attacks and prevent attackers from exploiting the system. Overall, protecting computer networks is critical in today's digital world. Implementing network security measures, such as NIDS, can help safeguard the network and data from attacks. Constantly updating the system with new signatures can help to detect new attacks, making it harder for attackers to compromise the network[4].

Hackers are becoming increasingly sophisticated and embedding attacks within computer networks, which can compromise the confidentiality of data. Machine Learning (ML) based Intrusion Detection Systems (IDS) have become more efficient due to their accuracy, especially when dealing with large and complex datasets. However, the training of the ML model requires a significant amount of time, and as the size of the datasets increases, so does the CPU execution time. Therefore, it is advisable to select the best optimal features that reduce time complexity when dealing with large and complex datasets. Optimal feature selection refers to selecting distinguishing features that can separate two classes [5][6].

The process of selecting the best optimal feature is part of the feature selection phase of data preprocessing,

[1]*Research scholar, Christ University & Assistant professor , CMRIT, Bangalore. Karnatka, India*
Email ID: sowmya.t@res.christuniversity.in , sowmya.t@cmrit.ac.in
[2]*E A professor, Christ University Bangalore, Bangalore, Karnatka, India ; Email ID: maryanita.ea@christuniversity.in*

which is one of two phases, the other being feature extraction. During feature selection, different features are removed or added to the initial set based on their relevance, while feature extraction involves transforming data from a high-dimensional space to a lower-dimensional space, where the transformed data represents the original data in a more meaningful and efficient way [6].

Feature selection is an important technique that can increase the efficiency of a system in terms of detection speed and accuracy by selecting the nearest subset of features [7]. There are three types of feature selection methods: Filter, Wrapper, and embedded based methods. Filter methods select features by correlating with the target, while Wrapper methods select features based on inference from the previous model. Embedded methods select features by combining filter and wrapper-based methods. However, these methods may not be effective in improving the accuracy and detection speed of complex datasets. Thus, it is mandatory to propose a model that can select the best features out of all the features to improve the overall performance of the system. Time complexity can also be a major problem in several fields, especially in the security domain, where the time complexity of some machine learning algorithms is very high, which degrades the overall performance of the system. In short to increase the efficiency of a system in terms of detection speed and accuracy, it is important to select the best features out of all the features in the dataset. This can be achieved using three types of feature selection methods, namely Filter, Wrapper, and embedded based methods. However, these methods may not be effective for complex datasets, and time complexity can be a major problem in some fields. Therefore, it is necessary to propose a model that can address these issues and improve the overall performance of the system.

For the evaluation purpose, UNSWNB15 dataset is considered which contains nine attacks. It is important to consider each and every attack and to detect the attack individually. Especially based on the attacks the features also will be different for different attacks Here feature selection is acting as an intelligent agent which selects the features corresponding to the attacks. This research work proposes an Intelligent agent based IDS that detects attacks namely Analysis and Reconnaissance. It is important to detect those attacks even though it is not popular, it is a passive attack it will gather the information or analyze the entire network before injecting the actual attack.[8]Passive attacks try to capture all the information passing through the network and may crack the password. UNSWNB15 is a complex dataset that consists of 43 features that contribute to all nine types of attacks. Relevant features of each attack may vary and our proposed model considers only two attacks namely Analysis and Reconnaissance. Hence the proposed IDS passes through a feature selection phase based on two attacks and finally the detection phase. This paper proposes a hybrid intelligent agent based feature selection approach that uses a bio inspired search space methodology called a Genetic algorithm. Although a Genetic Algorithm for feature selection is already existing, some improvement to the existing algorithm increases the efficiency of the entire system. Hence in this paper, a hybrid GA-PI algorithm is proposed for selecting the optimal features corresponding to Analysis and Reconnaissance attack. It removes all the reductant features and increases the detection speed and accuracy in a better way.

The main contributions of the study can be summarized as follows:

1. The study proposes a hybrid GA-PI algorithm for selecting the minimal features that combines the strength of the Genetic Algorithm and Permutation Importance. This algorithm is designed to identify the most important features for the given problem and select them for further analysis.
2. The study evaluates the performance of the hybrid GA-PI algorithm for Analysis and Reconnaissance attacks. This means that the algorithm was tested and validated on datasets related to these types of attacks.
3. Finally, the study compares the performance of hybrid GA-PI on Analysis and Reconnaissance attacks with other feature selection algorithms. This comparison allows for a better understanding of the strengths and weaknesses of different feature selection methods and provides insights into which algorithm performs better under different scenarios.

The remainder of this paper is structured as follows: Section 2 provides a thorough review of the literature relevant to the targeted research area, followed by Section 3 which outlines the proposed methodology used in this study. In Section 4, the results of our analysis are presented, and a detailed discussion of these findings is provided. Finally, Section 5 concludes the paper by summarizing the main findings, highlighting their implications, and discussing potential avenues for future research.

2. Literature Review

Many researchers have proposed several feature selection algorithms for the intrusion detection system. Al-Safi et al [9] utilized Information gain for selecting the effective features in the NSL KDD dataset and employed SVM for classification strategy. For hyperparameter tuning Artificial Bee Colony and Cuckoo Search algorithms are combined effectively to perform classification with Support Vector Machine. The authors said that while comparing with other methods the model outperformed well in terms of classification accuracy.

Orieb Abu Alghanam et al [10] proposed a modified version of pigeon inspired optimizer integrated with the Tabu Local Search algorithm. Here for feature selection and for classification one class approach is used and the model achieves very good accuracy. For the sake of classification ensembled model integrated with one class approach is proposed. Here OC-SVM and OC-IF are used in an ensembled manner and the framework showed better results in terms of FPR and TPR. The proposed LS-PIO feature selection method outperformed well when compared with Hill climbing PIO algorithm.

Thaseen et al [11] devised an intrusion detection framework by utilizing an effective Chi Square feature selection mechanism with a Support Vector Machine algorithm. Apart from these, for parameter optimization, a variance based tuning technique is used and the entire model achieved outstanding performance.

In [12] Khammassi et al applied mainly wrapper based feature selection algorithm called Genetic Algorithm in combination with Logistic Regression for feature selection. The model is tested and validated on UNSWNB15 and KDD CUP 99 datasets. For the attack classification, Decision Tree Classifier is applied and for the visualization of results, Weka tool is used. The suggested approach reduced the features to 18 for KDD CUP 99 and 20 for UNSWNB15 and achieved a better detection rate. The results showed a detection rate of 99.9% for KDD CUP99 and 81.24% for UNSWNB15 dataset. However, the authors didn't note the accuracy and the execution time.

In [13] Ambusaidi et al introduced a flexible Mutual Information feature selection and the reduced features are applied for the Least Square Support Vector Machine based Intrusion Detection System. The experimental results revealed better Detection rate and FAR on three different datasets namely Kyoto 2006, KDD CUP 99, and NSL-KDD datasets. [14] Singh et al adopted a filter based feature selection called Information Gain approach by combining rule based classifiers for the classification purpose. Information gain derived 22 features from the UNSWNB15 dataset and gained an accuracy of 84.83%.

In[15] proposed and studied the effect of feature selection and SMOTE oversampling on UNSWNB15 dataset to detect the attacks. Considerably feature selection approach is used to enhance the performance of the system. The combined effect of SMOTE and RFE feature selection method is applied on mainly four machine learning classifiers namely Random Forest, Decision Tree, KNN and LR algorithm. The model achieved better accuracy of 84.13% with the Random Forest algorithm.

In [16] have proposed effective IDS by integrating univariate feature selection with machine learning classifiers for detecting the attacks in the NSL KDD dataset. The selected features from the feature selection are passed as input for machine learning algorithms namely KNN, DT, rule-based systems, neural networks, and SVM. The proposed workout performed well in comparison with other existing works.

In another work[17] Shalini Subramani et al proposed an intelligent IDS utilizing PSO-based feature selection and enhanced multiclass SVM based classification mechanism for intrusion detection. The entire machine learning model is validated on two datasets namely the KDD'99 Cup data set and the CIDD data set. It enhances the performance in terms of accuracy and balances False Positive Rate.

Table 1. Summary of the literature provided, along with some comparison parameters:

Literature	Feature Selection Algorithm	Classification Algorithm	Dataset	Accuracy/Performance	Remarks
Al-Safi et al [9]	Information Gain	SVM	NSL KDD	Outperformed others in classification accuracy	Used ABC and Cuckoo Search for hyperparameter tuning
Orieb Abu Alghanam et al[10]	LS-PIO (modified Pigeon Inspired Optimizer with Tabu Local Search)	OC-SVM and OC-IF ensemble	N/A	Very good accuracy in one class approach	Outperformed Hill climbing PIO algorithm

Thaseen et al [11]	Chi Square	SVM	N/A	Outstanding performance	Used variance-based tuning for parameter optimization
Khammassi et al [12]	Genetic Algorithm (wrapper based)	Logistic Regression	UNSWNB15 and KDD CUP 99	Better detection rate (99.9% for KDD CUP 99 and 81.24% for UNSWNB15)	Reduced features to 18 for KDD CUP 99 and 20 for UNSWNB15
Ambusaidi et al [13]	Mutual Information	Least Square SVM	Kyoto 2006, KDD CUP 99, and NSL-KDD	Better detection rate and FAR	Used flexible feature selection approach
Singh et al [14]	Information Gain	Rule-based classifiers	UNSWNB15	Accuracy of 84.83%	Derived 22 features using Information Gain
Barkah et.al [15]	SMOTE and RFE	Random Forest, Decision Tree, KNN, LR	UNSWNB15	Better accuracy of 84.13%	Used feature selection to enhance performance
Walling et.al [16]	Univariate feature selection	KNN, DT, rule-based systems, neural networks, and SVM	NSL KDD	Performed well in comparison with other existing works	Used machine learning classifiers for detection
Shalini Subramani et al [17]	PSO-based feature selection	Multiclass SVM	KDD'99 Cup and CIDD	Enhanced performance in terms of accuracy and balance False Positive Rate	Utilized intelligent IDS with enhanced multiclass SVM based classification mechanism

It is clear that several researchers have proposed various feature selection algorithms for intrusion detection systems. Some of these methods use popular machine learning algorithms such as SVM, Decision Tree, and KNN, while others use ensembled models with one class approach. Most of the methods reported high accuracy and better detection rates on various datasets such as NSL KDD, UNSWNB15, KDD CUP 99, and Kyoto 2006.

Our proposed work, "An Agent for Feature Selection in Network Intrusion Detection Systems Using a Hybrid GA-PI and Intelligent Techniques", aims to improve the performance of intrusion detection systems by using a hybrid GA-PI algorithm for feature selection. This approach combines the Genetic Algorithm (GA) with Permutation Importance (PI) to enhance the accuracy and reduce the number of features required for the detection process. Additionally, we integrate intelligent techniques such as fuzzy logic and decision trees to enhance the accuracy and efficiency of the classification process. Therefore, our proposed work builds on the existing literature by introducing a novel approach to feature selection that uses a hybrid GA-PI algorithm and incorporates intelligent techniques to enhance the performance of network intrusion detection systems.

3. Proposed Methodology

Our proposed methodology comprises three components: Data reprocessing, a Feature selection agent, and a classification module. We introduce a novel approach based on a hybrid of Genetic Algorithm and Permutation Importance (GA-PI) for feature selection, which selects the most effective features based on the type of network attack

3.1 Dataset Description

The UNSWNB15 [18] dataset is a network traffic dataset that contains nine different types of attacks and normal traffic data captured in a controlled network environment. The dataset was created to be used for research purposes in the field of network intrusion detection. In the context of proposed work the UNSWNB15 dataset serves as the main dataset for evaluating the performance of the proposed feature selection agent. By selecting the most effective features from the dataset, the agent can improve the accuracy of intrusion detection systems. It's important to note that using the UNSWNB15 dataset for evaluation purposes allows for the reproducibility of results and facilitates comparisons with other research studies that have used the same dataset. [19] This dataset was created by the Australian centre for cyber security and contains both normal data and nine different types of attacks commonly found in the digital environment today. These attacks include Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms. The dataset has a total of 49 features

For the purpose of our research, from the UNSWNB15 dataset 2 attacks namely Analysis, Reconnaissance, and Normal data are filtered out and evaluated. Each attack is evaluated separately and performance metrics are computed. Here we are considering two sub datasets 1. Reconnaissance and Normal(RN) 2. Analysis and Normal(AN) from UNSWNB15 for evaluation purposes. Hence the total number of records in the AN dataset is 2221437 and for RN it is 2220519.For the proposed approach 60:40 is considered a train test split ratio. The number of records for each attack is depicted in Fig 1.

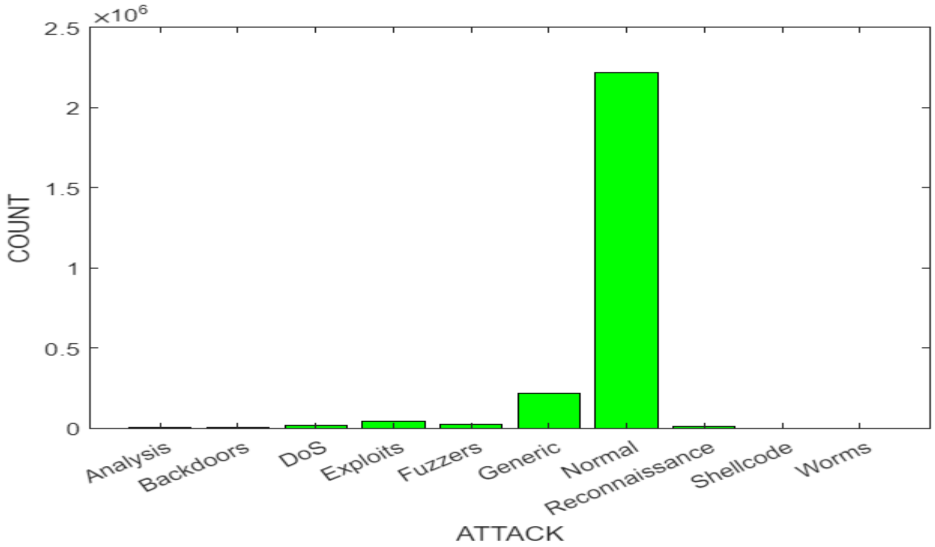

Fig 1: UNSWNB15 attack distribution

3.2 Data Preprocessing

Data preprocessing is an important step in machine learning to get an accurate result by making the data consistent. Here the process prepares the data into a format suitable for machine learning models and data mining processes. [9] Data scaling plays a major role in predicting the final output and reducing the accuracy. UNSWNB15 dataset is having multiple features and that features fluctuate to different ranges. In order to attain an optimal machine learning model feature scaling plays a significant role in our proposed model. The filtered data(AN and RN) are scaled down to some range to compare with one another. In this paper, the data are scaled down using the Standard Scaler method to a variance of 1 and the mean value is reduced to 0. Hence all the features are now in a comparable range to perform the analysis.

Data preprocessing is an essential step in machine learning that prepares data for model training by making it consistent and suitable for analysis. It involves a series of steps to clean, transform, and standardize the data, ensuring that it is in a format that machine learning models can work with. Data scaling is a crucial component of data preprocessing, particularly in predicting the final output and reducing inaccuracies. For example, the UNSWNB15 dataset, which has multiple features, has varying ranges that can affect the accuracy of machine learning models.

In order to create an optimal machine learning model, feature scaling plays a significant role. One way to achieve this is by scaling the filtered data using the Standard Scaler method to a variance of 1 and reducing the mean value to 0, which puts all features in a comparable range for analysis. Overall, data preprocessing is a critical step in machine learning that improves accuracy, removes errors, and prepares data for analysis [20]. It is the first step in preparing raw data for machine learning models, as raw data often contains various errors, anomalies, and redundancies [21]. Automated machine learning can also help streamline the time-consuming process of model development while maintaining high efficiency and productivity [22].

The mathematical model for Standard Scaler is given as follows:

$$x' = \frac{x - \mu}{\sigma} \qquad (1)$$

where x is a column vector of data points and x' is a column vector of transformed data points

Equation (1) represents a transformation of a variable x to a standard normal distribution. This transformation is accomplished by subtracting the mean of the distribution, μ, from each data point and then dividing by the standard deviation of the distribution, σ This transformation is commonly used in statistics to normalize data, making it easier to compare data sets with different means and standard deviations. The resulting transformed data points have a mean of 0 and a standard deviation of 1.

The transformation is based on the assumption that the underlying distribution of the data is normal. The normal distribution is a commonly used probability distribution that describes a bell-shaped curve. The general form of its probability density function is given by

$$\mathrm{f(x)} = \left(\frac{1}{\sigma\sqrt{2\pi}}\right) e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad (2)$$

Where μ the mean and σ is the standard deviation.

In summary, equation (1) represents a transformation of data to a standard normal distribution, which is based on the assumption that the underlying distribution of the data is normal. The transformation is commonly used in statistics to normalize data, making it easier to compare data sets with different means and standard deviations. The transformation is related to the concept of maximum likelihood estimation, which is a statistical method used to estimate the parameters of a probability distribution based on observed data.

3.3. Intelligent Feature Selection Agent

Genetic Algorithm: A genetic algorithm (GA)[23] is a metaheuristic optimization algorithm inspired by the process of natural selection. It is a population-based algorithm that mimics the evolution process of living organisms. The algorithm starts with a population of random solutions to the problem, and then iteratively selects the fittest individuals (those with the best fitness score) to reproduce and generate the next population. The process continues until the population converges to the optimal solution or a stopping criterion is met. The key components of the GA are the representation of the individuals, the fitness function, the selection method, and the genetic operators (crossover and mutation).

The mathematical notation for the GA can be represented as follows:

- Population:

$$P = \{p1, p2, \dots, pn\} \qquad (3)$$

- Fitness function:

$$f(P) = \{f(p1), f(p2), \dots, f(pn)\} \qquad (4)$$

- Selection operator:

$$S\big(f(P)\big) = \{s1, s2, \dots, sm\}(where\ m \le n) \quad (5)$$

- Crossover operator:

$$C\Big(S\big(f(P)\big)\Big) = \{c1, c2, \dots, cn\} \qquad (6)$$

- Mutation operator:

$$M\bigg(C\Big(S\big(f(P)\big)\Big)\bigg) = \{m1, m2, \dots, mn\} \qquad (7)$$

Permutation Importance: Permutation importance is a method for feature importance calculation in machine learning. It measures the decrease in model performance when the values of a feature are randomly permuted, while holding all other features constant [24]. The feature importance score is then calculated as the difference between the original performance and the performance with permuted features. This score represents the contribution of the feature to the model's accuracy.

The mathematical notation for permutation importance can be represented as follows:

Original performance: L

Feature importance score:

$$\mathrm{PI_i} = \mathrm{L} - \mathrm{L_i}, \qquad (8)$$

where L_i is the performance with the feature i permuted

Normalized feature importance score:

$$\mathrm{NPI_i} = \frac{\mathrm{PI_i}}{\mathrm{sum(PI)}} \qquad (9)$$

where sum(PI) is the sum of all feature importance scores

Hybrid of Genetic Algorithm and Permutation Importance (GA-PI) ***:*** The hybrid of Genetic Algorithm and Permutation Importance (GA-PI) is a promising approach for feature selection in Network Intrusion Detection Systems (NIDS) using the UNSWNB15 dataset. GA-PI combines the benefits of both Genetic Algorithm (GA) and Permutation Importance (PI) methods to create an intelligent feature selection agent.

The GA component of the algorithm uses a population-based approach to evolve and select the best feature subset for the NIDS[25]. It employs crossover and mutation operators to produce new candidate solutions for the next generation. The fitness function of the GA is defined based on the classification accuracy of the selected feature subset using a machine learning algorithm.The PI component, on the other hand, uses a permutation-based approach to determine the importance of each feature in the dataset. It evaluates the effect of permuting each feature on the classification accuracy of the machine learning algorithm. The features are then ranked based on their importance scores.

GA-PI integrates the outputs of the GA and PI methods to identify the optimal feature subset for the NIDS. The GA produces a subset of features with high classification accuracy, while the PI provides a measure of the importance of each feature. The GA-PI then selects the features with high classification accuracy and high importance scores. Overall, GA-PI has shown promising results in feature selection for NIDS using the UNSWNB15 dataset. It has achieved high classification accuracy and reduced the number of features required for the NIDS.

The GA-PI algorithm with UNSW-NB15 dataset:

Let X be the set of all features in the UNSW-NB15 dataset and Y be the binary classification target variable indicating whether a network traffic record is either normal or anomalous. The goal is to find a subset of features X' that maximizes the classification performance on Y.

Input:

- UNSWNB15 dataset
- Hyper parameters: population size (pop_{size}), mutation rate (mut_{rate}), crossover rate ($cross_{rate}$), number of iterations (num_{iter}), and permutation importance threshold ($pi_{threshold}$)

Output:

- The optimal set of features selected for Network Intrusion Detection Systems using the UNSWNB15 dataset

1. **Initialization:**
   - 1.1 Initialize $\mathrm{pop_{size}}, \mathrm{mut_{rate}}, \mathrm{cross_{rate}}, \text{and } \mathrm{num_{iter}}$
   - 1.2 Set the feature set as the entire set of features in the UNSWNB15 dataset
2. ***Permutation Importance:***
   - 2.1 Compute the permutation importance scores for all features in the feature set
   - 2.2 Select the top features whose permutation importance scores are greater than $\mathrm{pi_{threshold}}$
   - 2.3 Let S be the set of selected features
3. **Genetic Algorithm:**
   - 3.1 Initialize the population P with random binary strings of length |S|
   - 3.2 Evaluate the fitness of each individual in P using a fitness function f that measures the classification accuracy of a model trained on the selected features
   - 3.3 Select the top performing individuals based on their fitness score to create a mating pool M
   - 3.4 Perform crossover and mutation operations on the mating pool M to create a new generation of individuals P′
   - 3.5 Evaluate the fitness of each individual in P′
   - 3.6 Repeat steps 3.3-3.5 for $\mathrm{num_{iter}}$ iterations
   - 3.7 Let $\mathrm{P_{final}}$ be the final generation of P′
4. **Output:**
   - 4.1 The feature set that has the highest fitness score in $\mathrm{P_{final}}$ is selected as the best set of features for UNSWNB15 dataset

Here are the definitions of the variables used in the algorithm:

- $|S|$: the number of selected features
- S: the set of selected features
- P: the population of binary strings representing the feature sets
- M: the mating pool of individuals selected for crossover and mutation
- P': the new generation of individuals after crossover and mutation
- P_{final}: the final generation of individuals after num_{iter} iterations
- $f(x)$: the fitness function that measures the classification accuracy of a model trained on the selected features x.

Pseudo code for the above algorithm

```
Function $GA - PI\begin{pmatrix} UNSWNB15\ dataset, pop_{size}, mut_{rate}, cross_{rate}, \\ num_{iter}, pi_{threshold} \end{pmatrix}$:

// Step 1: Initialization
    $feature_{set} <- set\ of\ all\ features\ in\ UNSWNB15\ dataset$
    $population$ <- create population of size $pop_{size}$ with random binary strings of length equal to number of features in $feature_{set}$

// Step 2: Permutation Importance
    $pi_{scores} <-$ Compute permutation importance scores for all features in $feature_{set}$
    $selected_{features}$ $<-$ Select features whose permutation importance scores are greater than $pi_{threshold}$
// Step 3: Genetic Algorithm
        $for\ i\ from\ 1\ to\ num_{iter}$:
// Evaluation
    $fitness_{scores}$ $<-$ Evaluate fitness of each individual in population using a fitness function that measures classification accuracy of a model trained on selected features
    $top_{performers}$ $<-$ Select top performing individuals from population to create mating pool

// Crossover and Mutation
    $offspring$ $<-$ Perform crossover and mutation operations on $top_{performers}$ to create new generation of individuals
    Population $<-$ replace population with new offspring
// Step 4: Output
    $best_{individual}$ $<-$ Select individual with highest fitness score in final population
    $best_{features}$ $<-$ features selected in $best\_individual$
    return $best_{features}$
```

3.4 Mathematical Model

The GA-PI (Genetic Algorithm-Permutation Importance) algorithm with the UNSW-NB15 dataset is a feature selection algorithm that aims to find a subset of features that maximizes the classification performance of the binary classification target variable indicating whether a network traffic record is either normal or anomalous. The algorithm starts with the initialization of the hyperparameters, including the population size, mutation rate, crossover rate, number of iterations, and permutation importance threshold. The feature set is set as the entire set of features in the UNSW-NB15 dataset, and permutation importance scores are computed for all features. The top features with permutation importance scores greater than $\mathrm{pi}_{\mathrm{threshold}}$are selected to create a set of selected features. The GA-PI algorithm begins with selecting the top features whose permutation importance scores are greater than $\mathrm{pi}_{\mathrm{threshold}}$.

Let $\mathrm{S} = \{\mathrm{s1}, \mathrm{s2}, \ldots, \mathrm{sm}\}$ be the set of selected features.The Genetic Algorithm is then performed by initializing the population P with random binary strings of length equal to the number of selected features. The fitness of each individual in P is evaluated using a fitness function f that measures the classification accuracy of a model trained on the selected features. The top performing individuals based on their fitness score are selected to create a mating pool M, and crossover and mutation operations are performed on the mating pool M to create a new generation of individuals P'. The fitness of each individual in P' is evaluated, and the process is repeated for $\mathrm{num}_{\mathrm{iter}}$iterations. The feature set with the highest fitness score in $\mathrm{P}_{\mathrm{final}}$is selected as the best set of features for the UNSW-NB15 dataset.

Let $\mathrm{X} = \{\mathrm{x1}, \mathrm{x2}, \ldots, \mathrm{xn}\}$ be the set of features in the UNSW-NB15 dataset, and Y be the binary classification target variable[26]. The goal is to find a subset of features X' that maximizes the classification performance of Y. The GA-PI algorithm can be modeled as follows:

1. **Initialization:** Initialize hyperparameters including the population size (pop_{size}), mutation rate (mut_{rate}), crossover rate ($cross_{rate}$), number of iterations (num_{iter}), and permutation importance threshold ($pi_{threshold}$). Set the feature set as the entire set of features X in the UNSW-NB15 dataset.
2. **Permutation Importance:** Compute permutation importance scores for all features in X. Select the top features whose permutation importance scores are greater than $pi\_threshold$ to create a set of selected features S.
3. **Genetic Algorithm:** Initialize the population P with random binary strings of length equal to the number of selected features$|S|$. Evaluate the fitness of each individual in P using a fitness function f that measures the classification accuracy of a model trained on the selected features. Select the top performing individuals based on their fitness score to create a mating pool M. Perform crossover and mutation operations on the mating pool M to create a new generation of individuals P'. Evaluate the fitness of each individual in P'. Repeat the previous two steps for num_iter iterations. Let P_final be the final generation of P'.
4. **Output:** The feature set that has the highest fitness score in P_{final} is selected as the best set of features for the UNSW-NB15 dataset.

The fitness function f can be defined as the classification accuracy of a model trained on the selected features. Let M be the training dataset consisting of m records, where each record is represented as $(xi, yi), xi$ being the feature vector and yi being the binary label. Let T be the test dataset consisting of t records. $Let\, f(X')$ be the classification accuracy of a model trained on the selected feature set X'. Then, $f(X')$ can be calculated as follows:

$$f(X') = \left(\frac{1}{t}\right) * \; sum(i = 1 \; to \; t) \left[yi \; = \; predicted_{yi(X')}\right] \qquad (10)$$

where $predicted_{yi(X')}$ is the binary label predicted by the model trained on the selected feature set X' for the i^{th} Record in the test dataset. The binary classification model used for training can be any

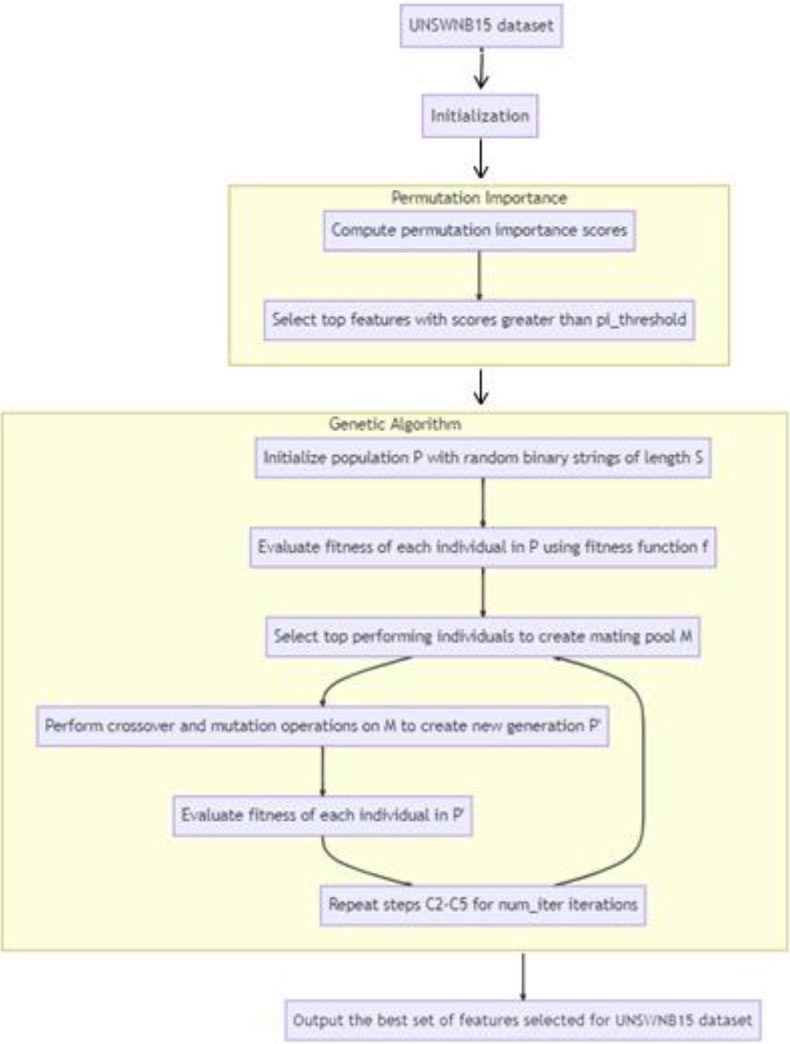
UNSWNB15 dataset
Initialization
Permutation Importance
Compute permutation importance scores
Select top features with scores greater than pi_threshold
Genetic Algorithm
Initialize population P with random binary strings of length S
Evaluate fitness of each individual in P using fitness function f
Select top performing individuals to create mating pool M
Perform crossover and mutation operations on M to create new generation P'
Evaluate fitness of each individual in P'
Repeat steps C2-C5 for num_iter iterations
Output the best set of features selected for UNSWNB15 dataset

Fig 2: Flow model for Hybrid GA-PI Feature Selection Agent

The main goal of the study is to select the optimal features that can increase the detection speed and accuracy. Hence in order to get the best optimal features a hybridization methodology is proposed. In the second stage wrapper based Permutation Importance is used in the Hybrid GA-PI algorithm. In the first step of PI algorithm, the performance is evaluated by randomly shuffling the features. Secondly, feature Importance is computed based on the performance evaluation of the previous step. A larger score corresponds to a decrease in the performance of the model, hence all those features are removed the selected number of features is reduced from 18 to 13. Five features are to be removed and the remaining 13 features undoubtedly contributed to increasing the performance in terms of accuracy and detection speed. Subsequently, the Hybrid GA-PI selects the 13 optimal features by integrating Genetic Algorithm and Permutation Importance for Analysis attack.

For the detection of the Reconnaissance attack, the entire steps of the hybrid GA-PI algorithm are replicated on Reconnaissance and Normal dataset. In the first stage, Genetic Algorithm reduced the features to 20 and PI algorithm selects 17 final sets of features. The working of the GA-PI algorithm is depicted in a detailed way in Algorithm 1.

Here the entire feature selection module works as an intelligent agent for selecting effective features by finding the dependencies between the features and the target attack classes. The major goal of the IDS framework is to provide an efficient classification mechanism and it is achieved by means of an intelligent feature selection agent.

3.5 Intrusion Detection System

The proposed Intrusion Detection System (IDS) with Hybrid GA-PI is designed to identify network anomalies in the UNSW-NB15 dataset. The system employs a feature selection approach using a hybrid of a Genetic Algorithm (GA) and Permutation Importance (PI) to find the most relevant features for classification.

Once the optimal subset of features is selected, the classification module uses a Support Vector Machine (SVM) algorithm to classify network traffic records as normal or anomalous. SVM is a popular supervised learning algorithm for classification tasks that maps data points to a high-dimensional space and finds the hyperplane that best separates the classes. In this work, the SVM algorithm is used to classify the selected network traffic features based on a predefined set of criteria.

The performance of the IDS is evaluated using various classification metrics such as accuracy, computation time. The system aims to achieve high classification performance on the validation set by selecting the optimal subset of features and using the SVM algorithm for classification. The hyperplane is separated in such a way that the gap between the two categories should be slightly high, which means that the two classes are well separated and there is a low chance of misclassification.

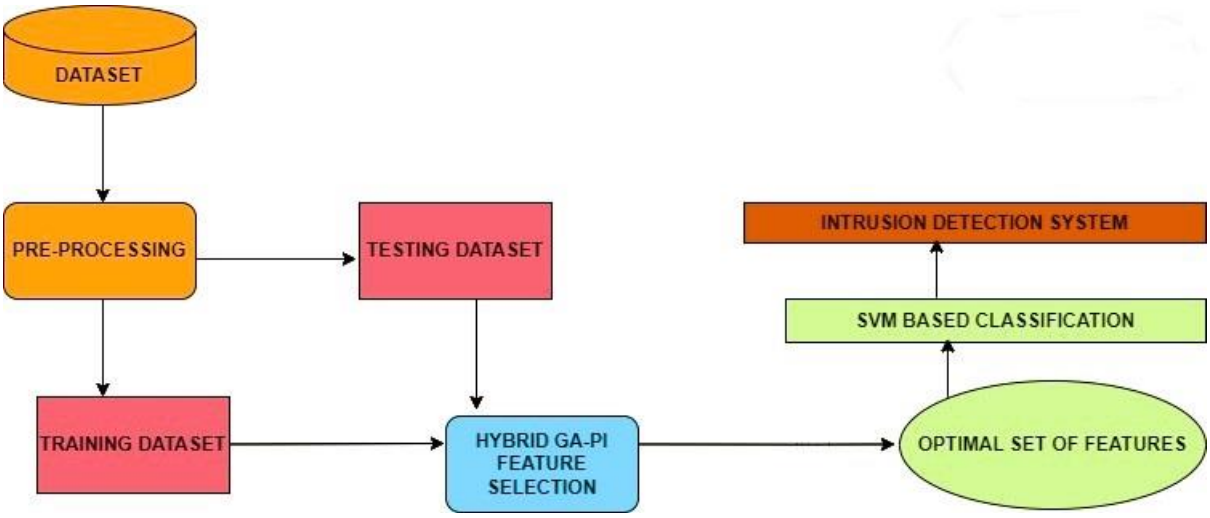

Fig 3: Proposed IDS

4. Results and Discussion

This section includes the experimental study of the UNSWNB15 dataset which was created by the Australian center for cybers security. In this paper, an intelligent feature selection agent plays a significant role in the performance of the entire framework. Hence stage by stage performance evaluation is done to test the IDS framework. Here we are evaluating the performance of two sub datasets 1. Analysis and Normal (AN) 2.Reconnaissance and Normal (RN). Finally, we compared the performance of feature selection algorithms on each attack dataset on the proposed approach as well as with the existing methods. Depending upon the attack type the features will always vary. Fig 4 shows a comparison of the number of features for the existing algorithms and the proposed hybrid GA-PI algorithm. Table 2 shows a detailed comparative study of the number of features for each algorithm.

Table 2: Feature Selection of Hybrid GAPI and Existing Methods

Data Categories	**No of Features** **Genetic Algorithm(GA)**	**No of Features Permutation Importance (PI)**	**No of Features** **Hybrid GA-PI**
Analysis and Normal (AN)	18	22	13
Reconnaissance and Normal(RN)	20	24	17

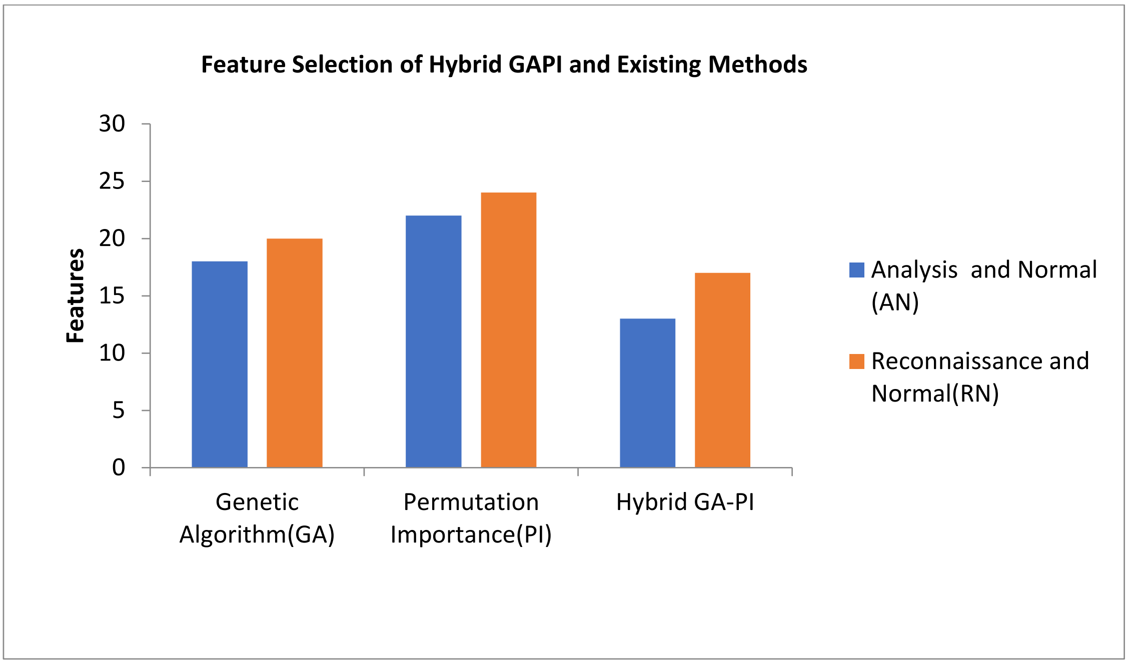

Fig 4: Feature Selection of Hybrid GAPI and Existing Methods

Indeed, the table 2 shows the number of features obtained by using three different feature selection algorithms for two sub-datasets of the UNSWNB15 dataset: Analysis and Normal (AN) and Reconnaissance and Normal (RN). The algorithms compared are Genetic Algorithm (GA), Permutation Importance (PI), and the proposed Hybrid GA-PI approach.

For the AN sub-dataset, GA selected 18 features, PI selected 22 features, and the Hybrid GA-PI approach selected only 13 features. This indicates that the proposed Hybrid GA-PI approach was able to select a smaller subset of features while still achieving comparable or better classification performance than the other two algorithms. Similarly, for the RN sub-dataset, GA selected 20 features, PI selected 24 features, and the Hybrid GA-PI approach selected 17 features. Again, the proposed approach selected a smaller subset of features while still achieving good classification performance.

Overall, the results suggest that the proposed Hybrid GA-PI approach is effective in selecting a smaller subset of features that can still achieve good classification performance, compared to the other two feature selection algorithms.

4.1 Performance Metrics

The proposed IDS is evaluated based on its accuracy and execution time. The evaluation is performed on two sub-datasets which are further divided into training and test data. The model's performance is evaluated using confusion metrics. Accuracy is considered a crucial indicator for evaluating the model's performance, despite considering execution time. The intelligent feature selection agent is implemented using Python and standard ML libraries, including Scikit Learn, are used for modeling. For this study, two sub-datasets are considered, and the performance metrics are based on a binary classification strategy.

4.1.1 Execution Time

To further evaluate the performance the execution time is also depicted in Table 3. The table compares the existing feature selection algorithms and the proposed hybrid GA-PI algorithm on the SVM classifier with respect to the number of features and the execution time. To further extend the evaluation graphs are also plotted in Fig 5.

Table 3: Time comparison between Hybrid GA-PI and Existing Methods

Data Categories	Execution Time Genetic Algorithm(GA)	Execution Time Permutation Importance(PI)	Execution Time Hybrid GA-PI
Analysis and Normal (AN)	20	21	15.8
Reconnaissance and Normal(RN)	20	18	16.2

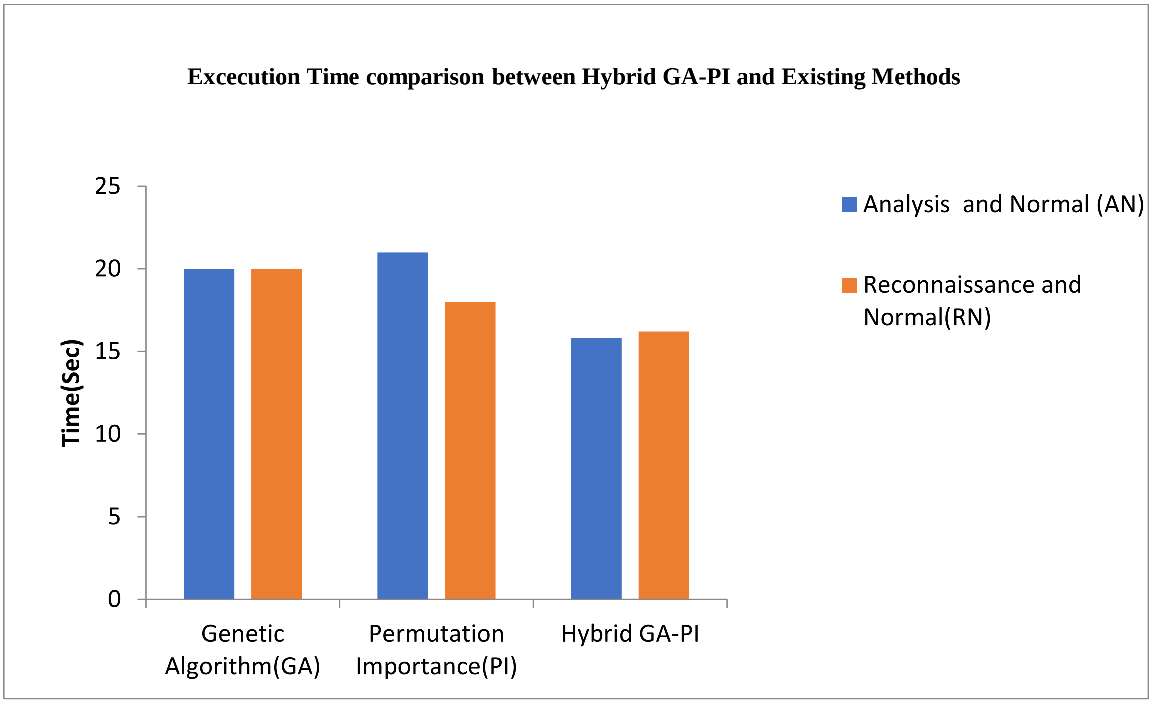

Fig 5: Time comparison between Hybrid GA-PI and Existing Methods

The table 3. provides the execution time comparison of three different feature selection algorithms on two sub datasets of the UNSW-NB15 dataset. The first algorithm is the Genetic Algorithm (GA), the second is Permutation Importance (PI), and the third is the proposed Hybrid GA-PI algorithm. The two sub datasets are Analysis and Normal (AN) and Reconnaissance and Normal (RN). For the AN dataset, the GA algorithm took 20 seconds to execute, while the PI algorithm took 21 seconds. The proposed Hybrid GA-PI algorithm outperformed both algorithms, taking only 15.8 seconds to execute. Similarly, for the RN dataset, the GA algorithm took 20 seconds, while the PI algorithm took 18 seconds. The Hybrid GA-PI algorithm took 16.2 seconds to execute, showing a better performance compared to both algorithms.

Based on the execution time results, the Hybrid GA-PI algorithm performed better than the GA and PI algorithms on both sub datasets. This indicates that the proposed algorithm can select the optimal features more efficiently compared to the existing feature selection algorithms.

4.1.2. Accuracy

Accuracy is the metric in which how the model perfectly classifies the dataset.

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN}*100 \qquad (11)$$

To verify the performance of the proposed GA-PI algorithm as an feature selection agent in IDS, 2 sub datasets (1.Analysis and Normal 2.Reconnaissance and Normal) generated from UNSWNB15 are used for experimentation.

The table 4. Shows the accuracy of three feature selection algorithms: Genetic Algorithm (GA), Permutation Importance (PI), and Hybrid GA-PI. The accuracy is evaluated on two sub-datasets of the UNSWNB15 dataset: Analysis and Normal (AN) and Reconnaissance and Normal (RN). For the AN sub-dataset, the Hybrid GA-PI algorithm outperforms the other two methods with an accuracy of 97%. The GA and PI methods achieve accuracies of 93% and 92%, respectively. Similarly, for the RN sub-dataset, the Hybrid GA-PI algorithm achieves the highest accuracy of 96%, while the GA and PI methods achieve accuracies of 88% and 90%, respectively.

These results suggest that the Hybrid GA-PI algorithm is more effective in selecting relevant features for network intrusion detection compared to the existing methods. The proposed algorithm achieves higher accuracies for both sub-datasets, indicating its potential to generalize well on different types of attacks.

Table 4: Accuracy comparison between Hybrid GA-PI and Existing Methods

Data Categories	**Genetic Algorithm(GA)**	**Permutation Importance(PI)**	**Hybrid GA-PI**
Analysis and Normal (AN)	93	92	97
Reconnaissance and Normal(RN)	88	90	96

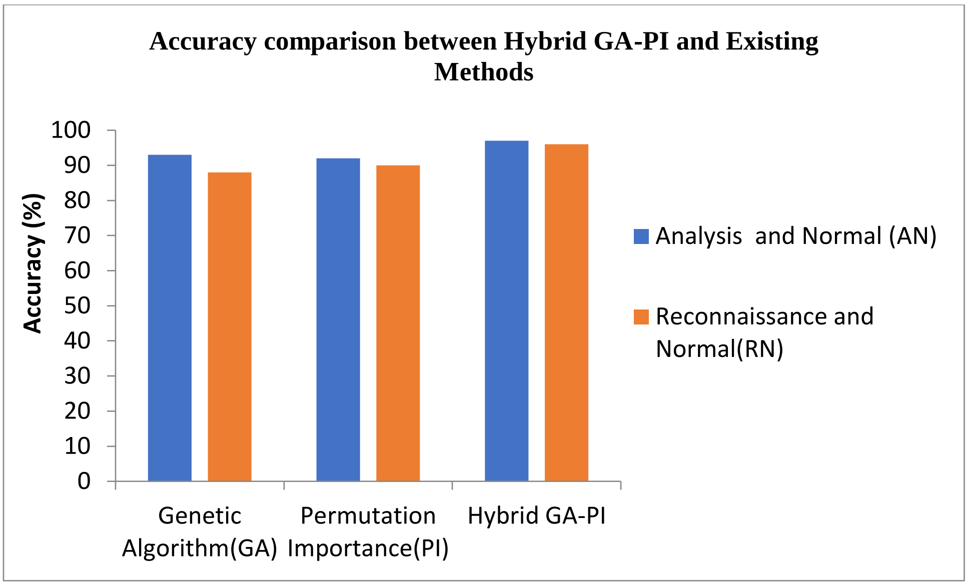

Fig 6: Accuracy comparison between Hybrid GA-PI and Existing Methods

5. Conclusion

An Intrusion Detection System (IDS) is crucial in maintaining cyber security by detecting attacks on networks. However, handling complex datasets can be challenging and can negatively impact the analysis of the entire IDS. This paper proposes using an intelligent feature selection agent to reduce irrelevant features, resulting in better accuracy and reduced execution time. The proposed approach focuses on analyzing two attacks, Analysis and Reconnaissance, from the UNSWNB 15 dataset. Compared to existing methods such as GA and PI, the hybrid GA-PI approach achieved higher accuracy rates of 97% for Analysis attack and 96% for Reconnaissance attack, while also having a faster execution time. The Analysis attack performed better than the Reconnaissance attack using this approach. Future research directions could consider other performance metrics such as Precision, Recall, and F1 score. Furthermore, introducing novelty in classification algorithms can make the classification module an intelligent agent, leading to better performance. Although this paper mainly focused on two attacks, the proposed method can be extended to detect more attacks in the future.

References

[1] Wang, C. R., Xu, R. F., Lee, S. J., & Lee, C. H. (2018). Network intrusion detection using equality constrained-optimization-based extreme learning machines. Knowledge-Based Systems, 147, 68-80.

[2] Subramani, S., & Selvi, M. (2023). Multi-objective PSO based feature selection for intrusion detection in IoT based wireless sensor networks. Optik, 273, 170419.

[3] Qu, L., He, W., Li, J., Zhang, H., Yang, C., & Xie, B. (2023). Explicit and Size-adaptive PSO-based Feature Selection for Classification. Swarm and Evolutionary Computation, 101249.

[4] K Thejeswari, K Sreenivasulu, B Sowjanya.(2022). Cyber Threat Security System Using Artificial Intelligence for Android-Operated Mobile Devices. International Journal of Computer Engineering in Research Trends.9(12),275-280.

[5] P Sandeep Kumar Reddy, M SriRaghavendra, K Sreenivasulu, T N Balakrishna. (2022). Cyber Threat Security System Using Artificial Intelligence for Android-Operated Mobile Devices. International Journal of Computer Engineering in Research Trends.9(12),269-274.

[6] Pasha, M. J., Pingili, M., Sreenivasulu, K., Bhavsingh, M., Saheb, S. I., & Saleh, A. (2022). Bug2 algorithm-based data fusion using mobile element for IoT-enabled wireless sensor networks. Measurement: Sensors, 24, 100548.

[7] Ramana, K. V. ., Muralidhar, A. ., Balusa, B. C. ., Bhavsingh, M., & Majeti, S. . (2023). An Approach for Mining Top-k High Utility Item Sets (HUI). International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 198–203. https://doi.org/10.17762/ijritcc.v11i2s.6045

[8] Samunnisa, K., Kumar, G. S. V., & Madhavi, K. (2023). Intrusion detection system in distributed cloud computing: Hybrid clustering and classification methods. Measurement: Sensors, 25, 100612.

[9] A.Rebekah Johnson, N.Parashuram .S.Prem Kumar, (2014). Organizing of Multipath Routing For

Intrusion Lenience in Various WSNs. International Journal of Computer Engineering in Research Trends.1(2),104-110.

[10] Al-Safi, A. H. S., Hani, Z. I. R., & Zahra, M. A. (2021). Using a hybrid algorithm and feature selection for network anomaly intrusion detection. J Mech Eng Res Dev, 44(4), 253-262.

[11] Alghanam, O. A., Almobaideen, W., Saadeh, M., & Adwan, O. (2023). An improved PIO feature selection algorithm for IoT network intrusion detection system based on ensemble learning. Expert Systems with Applications, 213, 118745.

[12] Thaseen, I. S., & Kumar, C. A. (2017). Intrusion detection model using fusion of chi-square feature selection and multi class SVM. Journal of King Saud University-Computer and Information Sciences, 29(4), 462-472.

[13] Pise, D. P. . (2021). Bot Net Detection for Social Media Using Segmentation with Classification Using Deep Learning Architecture. Research Journal of Computer Systems and Engineering, 2(1), 11:15. Retrieved from https://technicaljournals.org/RJCSE/index.php/journal/article/view/13

[14] Khammassi, C., & Krichen, S. (2017). A GA-LR wrapper approach for feature selection in network intrusion detection. computers & security, 70, 255-277

[15] Ambusaidi, M. A., He, X., Nanda, P., & Tan, Z. (2016). Building an intrusion detection system using a filter-based feature selection algorithm. IEEE transactions on computers, 65(10), 2986-2998.

[16] Singh, P., & Tiwari, A. (2015, May). An efficient approach for intrusion detection in reduced features of KDD99 using ID3 and classification with KNNGA. In 2015 second international conference on advances in computing and communication engineering (pp. 445-452). IEEE.

[17] Barkah, A. S., Selamat, S. R., Abidin, Z. Z., & Wahyudi, R. (2023). Impact of Data Balancing and Feature Selection on Machine Learning-based Network Intrusion Detection. JOIV: International Journal on Informatics Visualization, 7(1).

[18] Walling, S., & Lodh, S. (2023). Performance Evaluation of Supervised Machine Learning Based Intrusion Detection with Univariate Feature Selection on NSL KDD Dataset.

[19] Subramani, S., & Selvi, M. (2023). Multi-objective PSO based feature selection for intrusion detection in IoT based wireless sensor networks. Optik, 273, 170419.

[20] Figueiredo, J., Serrão, C., & de Almeida, A. M. (2023). Deep Learning Model Transposition for Network Intrusion Detection Systems. Electronics, 12(2), 293.

[21] Talukder, M. A., Hasan, K. F., Islam, M. M., Uddin, M. A., Akhter, A., Yousuf, M. A., ... & Moni, M. A. (2023). A dependable hybrid machine learning model for network intrusion detection. Journal of Information Security and Applications, 72, 103405.

[22] Famili, A., Shen, W. M., Weber, R., & Simoudis, E. (1997). Data preprocessing and intelligent data analysis. Intelligent data analysis, 1(1), 3-23.

[23] Kiran, B. R., Thomas, D. M., & Parakkal, R. (2018). An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. Journal of Imaging, 4(2), 36.

[24] Gaddam, A., Wilkin, T., Angelova, M., & Gaddam, J. (2020). Detecting sensor faults, anomalies and outliers in the internet of things: A survey on the challenges and solutions. Electronics, 9(3), 511.

[25] Huang, J., Cai, Y., & Xu, X. (2007). A hybrid genetic algorithm for feature selection wrapper based on mutual information. Pattern recognition letters, 28(13), 1825-1844.

[26] S. Anto, S. Chandramathi. (2015). An Expert System based on SVM and Hybrid GA-SA Optimization for Hepatitis Diagnosis. International Journal of Computer Engineering In Research Trends, 2(7), 437-443.

[27] V. Kishen Ajay Kumar, M. Rudra Kumar, N. Shribala, Ninni Singh, Vinit Kumar Gunjan, Kazy Noor-e-alam Siddiquee, Muhammad Arif, "Dynamic Wavelength Scheduling by Multiobjectives in OBS Networks", Journal of Mathematics, vol. 2022, Article ID 3806018, 10 pages, 2022. https://doi.org/10.1155/2022/3806018

[28] Ramana, Kadiyala, et al. "Leaf disease classification in smart agriculture using deep neural network architecture and IoT." Journal of Circuits, Systems and Computers 31.15 (2022): 2240004. https://doi.org/10.1142/S0218126622400047