International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Ffo-Ann: Firefly-Optimized Artificial Neural Network for High-Accuracy Intrusion Detection

Gavaskar Vincent

Professor, Department of Computer Science and Engineering, Vins Christian College of Engineering, Chunkankadai, Nagercoil, Tamil Nadu 629003.

ABSTRACT

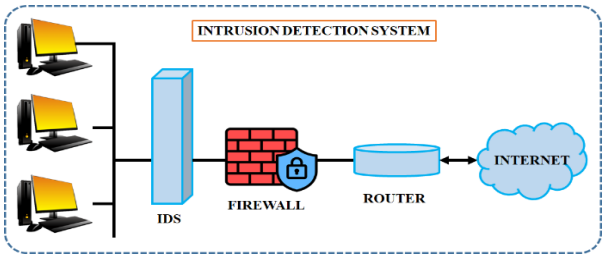
Intrusion Detection Systems (IDS) play a critical role in protecting modern network infrastructures against a range of evolving threats. This work develops a sophisticated deep learning-based intrusion detection system that uses Firefly optimized Artificial Neural Network (FFO-ANN) for accurate attack classification. First, raw traffic data is pre-processed, this includes classifying category variables as data encoding and numerical variables as z-score normalization, to normalize feature scaling so that no one feature is to be more prominent than the others. For feature selection, Exploratory Data Analysis (EDA) is conducted to extract relevant information. The FFO-ANN model is then able to classify the best possible accuracy, dynamically adapting to network patterns (training without the computational complexity), but is still able to increase accuracy. The FFO-ANN model produced more realistic intrusion detection capability, with an improvement in accuracy of 0.964, precision of 0.963, recall of 0.962, FI-Score of 0.9625 and specificity of 0.9655, compared to traditional models developed through python. The results provided evidence that the proposed model is a practical application for an intrusion detection system for cybersecurity.

Keywords: Intrusion Detection System,Data preprocessing,FFO-ANN, Exploratory Data Analysis.

1. INTRODUCTION

In today's connected digital world, protecting and maintaining the integrity of networked systems is of utmost importance [1]. Due to the rise of cyber threats and more sophisticated intrusion techniques, the need for strong Network Intrusion Detection Systems (NIDS) is further intensified [2].The vulnerability of networking systems and the security of network infrastructure are frequently overlooked and make them attractive targets for cyber- attack. The CNCERT/CC report of 2020 supported the issue of a high volume of serious cybersecurity risk, needing immediate attention to the enhancement of defence technologies [3]. The report also indicated that Indonesia was ranked 6th in Southeast Asia and 84th worldwide in regards to cybersecurity, therefore, it is imperative to develop cyber defence technologies in Indonesia [4]. A view of the architecture of Intrusion Detection Systems (IDS) is presented in figure 1.

Figure 1: Diagram of IDS



The IDSs provide essential support in protecting networks, as they continuously monitor networks for signs of malicious activity. However, IDS may exhibit limitations as many IDS inaccurately classify some normal user activity as an attack and this leads to numerous instances of false negatives being generated [5]. This results from working with imbalanced dataset, where the proportion of attack samples to normal user activity is not equal. To address this, the proposed architecture employs a modified version of the First-Order Feed-Forward Artificial Neural Network [6]. The first step of the proposed architecture consists of performing Preprocessing, where the raw data are converted into an encoded and normalised format such that the data are uniform and usable for further modelling purposes [7]. Once the data have been transformed into a uniform numerical form, Exploratory Data Analysis (EDA) can be performed to determine the leading indicators, factors, and anomalies that dictate the model selection and parameter adjustment processes. The processed data is then fed to the classification stage to accurately differentiate between normal and abnormal attacks [8-9].

Table 1: Related works of classification for high accuracy IDS

SI. NO	Author /Year	Methodology	Merits	Demerits	Evaluation metrics
1.	Mourad Benmalek et al [2024] [10]	SVM	SVM is capable of performing well when available labeled intrusion data is limited.	Training SVMs are computationally expensive, especially in the case of datasets that are very large in size of network traffic thatsystemstypically experience in real-time IDS.	It attains the accuracy of 0.9496, precision of 0.9595, recall of 0.9494 and FI-score of 0.9596
2.	Ghanshyam et al [2024] [11]	KNN	KNN can identify small differences in network traffic, thus improving anomaly detection, based on analysis of the local neighborhoods.	KNN performance could be negatively impacted if the dataset contains features with different scales or irrelevant and noisy features, unless	KNN achieves an accuracy, precision, recall and FI- score of 0.927, 0.871, 0.942 and

				normalization and feature subset selection takes place first.	0.905
3.	Hafiz MuhammadSanaullahBadaret al [2025] [12]	Naïve Bayes	Naive Bayes is computationally efficient and easy to implement, making it suitable for a light-weight idioms detection systems with low resource consumption.	NB not perform well in scenarios with imbalanced data as it tends to favour the majority class and result in very low detection rates for rare or novel attacks	The metrics of NB IS 0.86 of accuracy, 0.84 of precision, 0.81 of recall and 0.85 of FI-score.
4.	Nawaf AbdualazizAlmolhis et al [2025] [13]	Random Forest	Random Forest is capable of handling a large number of features, which is not unusual in intrusion detection systems, with each network packet or system log potentially having many attributes.	RF is expensive in computation time, especially when dealing with very large datasets or a large number of trees are being constructed.	The performance metrics of RF is 0.95 of accuracy, 0.93 of precision, 0.92 of recall and precision.
5.	AdenusiDaudaAdeite et al [2024] [14]	Logistic Regression	LR is easy to use and computationally efficient. This property allows it to quickly process network traffic.	In detecting intrusions often involves complex, non-linear relationships between features.	LR attains the accuracy of 0.934, precision of 0.936, recall of 0.934 and FI-score of 0.935.

A challenge with most current IDS is their inability to detect subtle intrusions patterns because they have to operate efficiently in terms of computing resources. To solve these limitations, a new method has been developed FFO-ANN, which uses firefly optimization on the ANN training parameters.

OBJECTIVES:

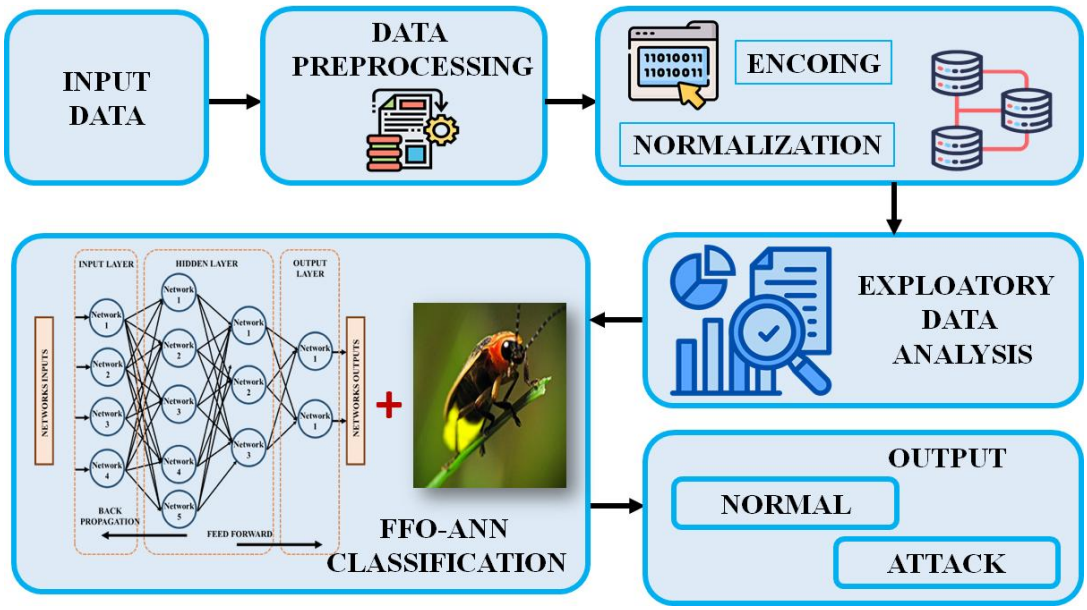
The main objectives of this research includes,

- To clean, encode, and scale the network traffic for efficient feed into the FFO-ANN, providing the required preprocessing and normalization of the data.
- To increase classification accuracy by applying the Firefly optimization algorithm to fine-tune the neural network's parameters and enhance classification performance for network intrusion detection.
- To label the traffic as either Normal or Attack to identify and address security issues in real-time.

2. PROPOSED SYSTEM DESCRIPTION

The proposed system, FFO-ANN aims to address issues with recognizing malicious network activity in a rapid and accurate manner. The system employs a solid FFO-ANN model based on the Firefly algorithm that optimizes the parameters of an ANN to facilitate quick and reliable classification of network traffic. The block diagram of the proposed work is illustrated in figure 2.

Figure 2: Block diagram of the proposed model



Raw traffic data is collected during the initial data-preprocessing stage and then refined by cleaning, restructuring, and formatting it into structured form for analyses. After the data has been cleaned and reformatted into a structured format, it will then undergo encoding and normalization so that all categorical variables are converted into numeric form, and all feature values are scaled to be within the same range so that all input data can be fed into the Neural Network. Exploration of the Data will take place through Exploratory Data Analysis (EDA) which will aid in the interpretation of the data to develop as well as discover new possible relationships between the variables and patterns that exist within the data, which will aid further in determining the classification of the data. The core of the system is based on the Firefly Optimization algorithm's (FFO) function in helping to tune the parameters for the Artificial Neural Network (ANN) classification model. Additionally, utilizing an FFO-ANN classification model optimally will assist the user in classifying traffic as either Normal or an Attack allowing for the timely detection and accurate identification of all security intrusions by a system. Utilizing this type of model will enable the system to classify network behavior in real-time allowing it to be able to react defensively to new potential threats and optimize its processing requirements.The method is scalable, flexible, and can process through large volumes of data, suitable for many different networked settings and environments.

3. PROPOSED SYSTEM MODELLING

A) DATA PREPROCESSING:

The preprocessing of data is of great importance for preparing raw network traffic data for intrusion detection. The data is normalized and passes through a series of preprocessing changes to prepare the dataset for training/testing with the Firefly-Optimized Artificial Neural Network. Normalization is

applied first to ensure that the data all is on the same scale and allows the model to converge faster and improve the overall training performance. The data is cleaned and dependent features are selected to remove unrelated/noisy data, so only output features are placed into the model dataset. Cleaning of the model helps reduce the dimensionality of the model, along with reducing model overfitting as it navigates the training set towards an output without managing a complex dataset. After the dataset was processed it is split into training and testing datasets, usually 80% and 20%, to provide a training dataset and testing dataset that measures model generalization once trained on the training dataset.

B) ENCODING AND NORMALIZATION:

A data encoding method was used to convert the raw network traffic datasets obtained from the open-access networks, NSL-KDD and UNSW-NB15, into numeric data for processing and inference with the neural network. In this case, min–max scaling was used to rescale the datasets from the original range of 0 to 1. Normalization of the data to this range also standardizes the feature values and helps ANN convergence during training. Normal and attack instances of the datasets are mapped to 0 and 1 (attack). In a one-class classification task, to ensure class balance between normal and attack samples, apply an underdamping technique so there are equal clean and attack samples in the training and testing datasets. A training dataset will allow the FFO-ANN model to learn the classification boundaries with minimum confusion between normal and attack instances.

The normalization technique is employed, in which the normalized value Z_{new} of a feature is calculated as,

$$Z_{new} = \frac{Z - Z_{min}}{Z_{max} - Z_{min}} \times (A_{max} - A_{min}) + A_{min} \qquad (1)$$

In this case, Z is the individual's feature value, Z_{min} and Z_{max} are the minimum and maximum feature values, respectively, while Amin and Amax establish the target range for the model (in this case 0 to 1). This normalization standardizes the inputs received by the FFO-ANN model, enhancing the efficiency, accuracy, and stability of the IDS.

C) EXPLORATORY DATA ANALYSIS:

Exploratory Data Analysis (EDA) is an essential process for understanding the structure, distribution, and the quality of the network traffic data prior to using the data to train the neural network. EDA has several purposes: identify patterns, discover anomalies, evaluate the importance of features, and assist in preprocessing steps like feature selection and normalization.

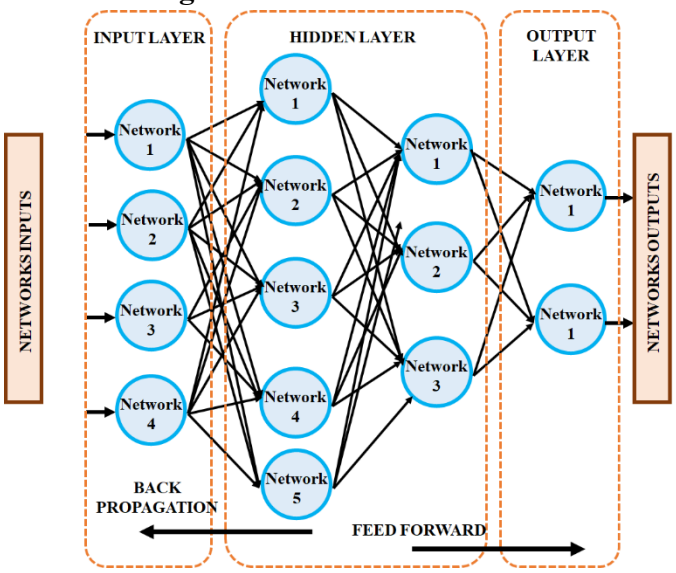
During EDA, the distribution for each feature is examined for skewness, outliers, or other unusual patterns that could change model performance. Histograms, boxplots, and scatter plots are useful visualizations to discover relationships between features and the target labels (normal and attack). For categorical features, frequencies and bar charts are useful to determine if certain classes of features are underrepresented, which is exceptionally important for an intrusion detection system where normal traffic usually over-represents the dataset. EDA also allows the determination of correlation between features. Features that are highly correlated may be redundant, and therefore can be removed, especially if computational complexity is a factor, or to prevent overfitting the model. Features that have a strong and predictive relationship with the target labels will be kept as input features for the model to ensure the FFO-ANN model receives the most informative features.

D) CLASSIFICATION- FFO-ANN:

ANNs are effective methods for high-accuracy intrusion detection, based on their capacity to model complex, non-linear relationships in the data. In network security, ANNs can detect malicious activity by examining the data of network traffic, where each data instance has several features such as the

packet length, protocol type, and behavior of communication. An ANN to detect an intrusion typically has an input layer, several hidden layers, and an output layer. The structure of ANN is shown in figure 3.

Figure 3: Structure of ANN



The neurons in each layer are connected by weighted edges, with the strength of the connections adjusted in training by the weights assigned to them, and with training, they are adjusted to minimize errors in prediction. The neuron output in an ANN is the function of a weight factorization of the input,

$$y = f(\sum_{i=1}^{n} w_i x_i + b) \quad (2)$$

Here x_i denotes the input features (such as characteristics of network traffic), w_i represents the weights of the connections, b is the bias, and f is the activation function such as ReLU or sigmoid, which adds non-linearity to the model. The non-linearity is important for the network to learn the complex relationships among features that are typicalin normal or malicious traffic.

Table 2: Optimized values of FFO

SI.NO	PARAMETER	VALUE
1.	Number of fireflies	30
2.	Maximum iterations	50
3.	Attractiveness (beta)	1.0
4.	Absorption coefficient	0.6
5.	Randomness (alpha)	0.25
6.	Alpha reduction factor	0.97
7.	Batch size	32
8.	Learning rate	0.001

The objective during training is to minimize the gap between the predictions made by the network and the corresponding actual values, which is initially accomplished using back proliferation and optimization methods such as gradient descent. In the case of classification problems (normal traffic vs. attack traffic), a common loss function is cross-entropy loss. Cross-entropy is defined as,

$$\text{Cross} - \text{Entropy Loss} = -\sum_{i=1}^{n} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (3)$$

Where, y_i is the true class label (0 for normal, 1 for an attack) and $\hat{y}_i$ is the predicted probability of the instance belongingto the attack class. The weights are updated to minimize this loss function, whichsubsequently improves the accuracy of the intrusion detection system.

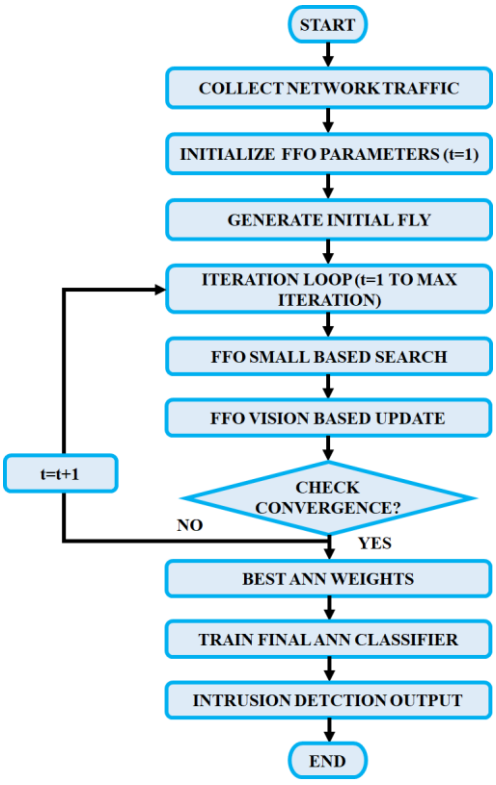
Pseudo Code: FFO

```
1. Initialize number of fireflies n
2. Initialize parameters α, β0, γ, MaxIter
3. Generate initial population of fireflies Xi randomly
4. Evaluate objective function f(Xi) for each firefly
5. FOR iteration = 1 to MaxIter DO
6.    FOR i = 1 to n DO
7.       FOR j = 1 to n DO
8.          IF f(Xj) is better than f(Xi) THEN
9.             Compute distance rij between firefly i and j
10.            Compute attractiveness β = β0 * exp(-γ * rij²)
11.            Move firefly i towards j using β and α
12.         ELSE
13.            Keep firefly i in its current position
14.         ENDIF
15.      ENDFOR
16.   Update α = α * δ (randomness reduction)
17.   Evaluate new solutions f(Xi)
18.   IF best solution improved THEN
19.      Save current best firefly
20.   ELSE
21.      Continue with next iteration
22.   ENDIF
23. ENDFOR
24. Return the best firefly solution
25. Print optimized value
26. Print final accuracy or minimized error
27. END
```

Achieving high accuracy during intrusion detection is reliant on several critical considerations. These considerations include the quality and diversity of the training data, activation functions, prevention of overfitting, and hyperparameter tuning. With careful design and training, ANNs can be taught to detect both known and unknown attacks, making them an invaluable part of modern cybersecurity defense. To enhance the parameters of ANN, FFO algorithm is implemented.

FFO is used to quickly move through a large search space of multiple dimensions, leading to a good solution while using fewer iterations, making it an ideal algorithm for NP-hard optimization problems like intrusion detection, where it is computationally demanding to identify the optimal set of discriminative features and the parameters for an ANN. The block diagram of FFO-ANN block diagram is depicted in figure 4.

Figure 4: Flowchart of FFO-ANN



FFO is inspired by the natural flashing behavior of fireflies, and models the attraction mechanism that a firefly has for another firefly on the basis of brightness which corresponds to the classification accuracy acquiring from evaluating the ANN in the occasion of the intrusion detection. Fireflies that display higher brightness attract the weaker counterparts to guide the population to an optimal solution. As is typical in empirical, conventional FFO we consider that fireflies are unisex and are attracted to each other based on brightness only; the attraction is proportional to the brightness and the fireflies with the most extreme intensity move points to allow exploration. The brightness intensity of each firefly, which in our case is the objective function value corresponding the ANN model encoded by the firefly is given by at the outset,

$$\mathrm{Int}(S) = \mathrm{Int}(r)S^2 \quad (4)$$

Here,$\mathrm{Int}(S)$ refers to the intensity of the light source at a distance of s, while $\mathrm{Int}(r)$ represents the base intensity of the light source. The attractiveness of a firefly depends on the total brightness and also on the absorption coefficient for light from all objects in the firefly's surrounding environment. The amount of attraction depends on the relationship between the two factors above:

$$\mathrm{Int} = \mathrm{Int}_0 e^{-\Omega S^2} \quad (5)$$

Once the brightness has been calculated, the Euclidean distance between two fireflies is calculated based on their respective feature subsets (IDS) or configurations of ANN

$$S = \|x_i - x_j\| = \sqrt{\sum_{z=1}^{x}(x_{i,z} - x_{j,z})^2} \quad (6)$$

The dimensionality of the intrusion detection feature space is defined as r. The movements of Fireflyi particles toward brighter Firefly j's are based onγ, which represents the attraction of Fireflyitoj's, EUR_irepresents random exploration at iteration u. The damping coefficient, which decreases randomness to facilitate convergence, is calculated through the following formula,

$$x_i^{u+1} = x_i^u + \Gamma_0 e^{-(s^2)}\left(x_u^j - x_u^i\right) + \gamma EUR_i \quad (7)$$

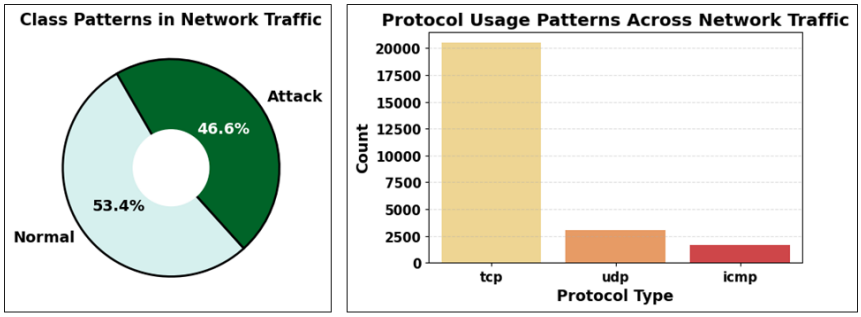
$$\gamma = \gamma_0 \theta^u, 0 < \theta < 1 \quad (8)$$

Iterative processes of the FFOdevelops high-quality feature subsets through optimizing ANNs and increase detection performance. The resulting IDS is based on the minimum percentage of both false positives and false negatives, increasing its ability to differentiate between normal and intrusions types. Trust-aware optimization provides a robust and adaptive blackout for intrusion detection in dynamic networks.

4. RESULT AND DISCUSSION

The proposed FFO-ANN improves the accuracy of IDS. This framework has been trained and validated on the Network Intrusion Detection Dataset consisting of 25,192 records and 44 Features with 43 attributes (input) used to describe connection behaviour, with an output label, indicating the class of each connection. The data set is partitioned into an equal amount of training (80%) and testing (20%) to ensure accurate modelling of the ANN and valid performance measurements against previously unseen network traffic.

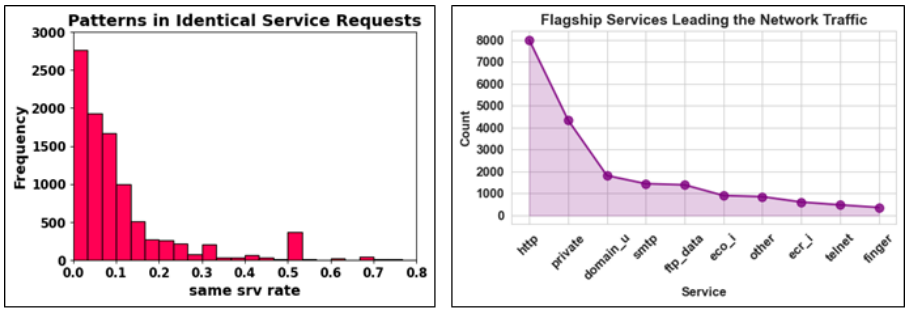
Figure 5:a) Class patterns in network traffic and b) protocol usage patterns across network traffic



a) b)

Figure 5 displays class patterns in network traffic and protocol usage patterns across network traffic. The data that appears in these charts indicates that the ratio of normal (53.4%) and attack (46.6%) instances is about equal. The great majority of the connections were created using the TCP protocol; considerably less were created using UDP and ICMP.

Figure 6: a) Patterns in identical service requests and b) Flagship services leading to network traffic

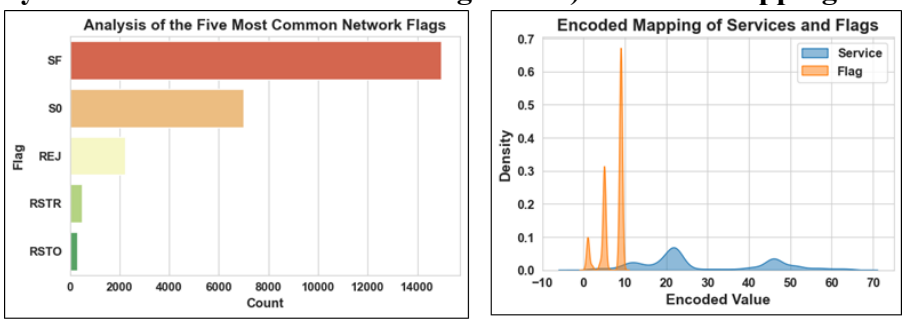


a) b)

Figure 6indicates that in terms of identical service requests this will generally happen at a very low rate and the number of requests will taper off dramatically as the rate continues to increase. The data also

indicates that the flow of any network is primarily different services that fall under two categories, specifically http and private; with every other service represented as falling far below.

Figure 7: a) Analysis of the five most common flags and b) Encoded mapping of services and flags



a) b)

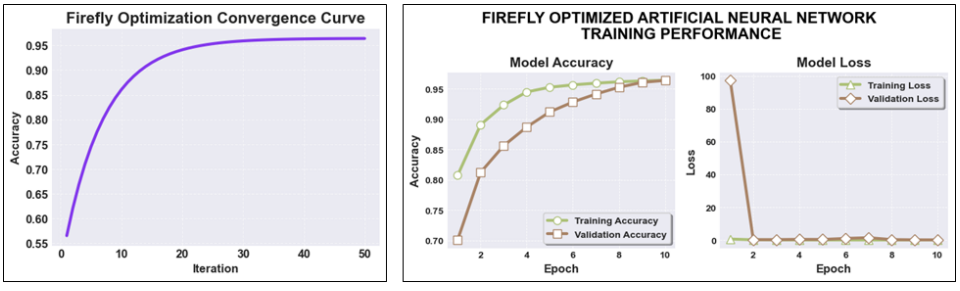
Figure 7specifies that SF is the dominant network flag, while FL and DNS having a lower representation compared to SF and a smaller number of service flags. Additionally, the data shows that while the flag values are clustered in a close range, the service encoding values are more widely distributed.

Figure 8: a) Feature interrelationships across dataset and b) Train Vs Test Split distribution

a) b)

As shown in figure 8, the heatmap, correlations between each attribute in the data and the attributes relate to one another. Most of the correlations are weak, with only a few moderate correlations appearing. The right bar chart shows the results for the train and test dataset splits, where 20,153 instances are in the training dataset and 5,039 instances in the test dataset.

Figure 9: a) Convergence curve of FFO and b) Training performance of FFO-ANN



a) b)

Figure 9evident that the Firefly Optimizer enhances the results of the model and causes a rapid increase in accuracy as well as a steady mode of convergence. The ANN that was optimally trained using F.O.,

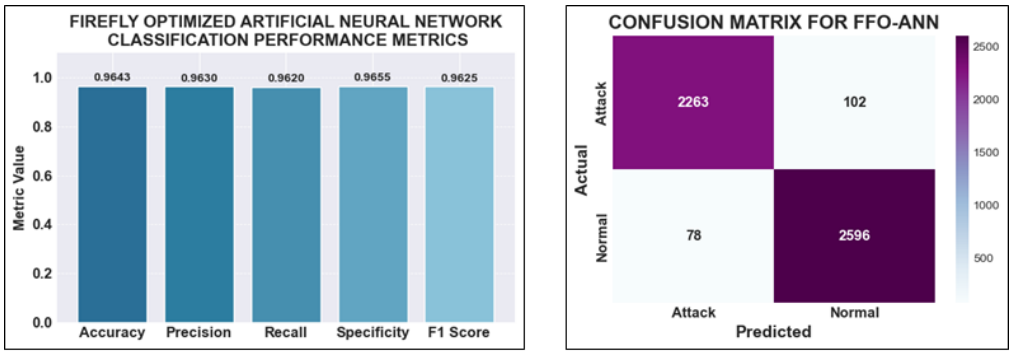
also demonstrates that it trained on data rapidly and provided very accurate predictions in addition to having low/stable training and validation losses.

Table 3: Comparative analysis of accuracy for various methods

METHODS	ACCURACY
KNN [11]	0.92
NB [12]	0.86
RF [13]	0.95
LR [14]	0.93
Proposed	0.96

Table 3 compares the overall accuracy rates from the various methods, it is clear that our approach outperforms the existing methods we evaluated. The proposed method has achieved 0.96 accuracy, which is much better than KNN, NB, RF, or LR.

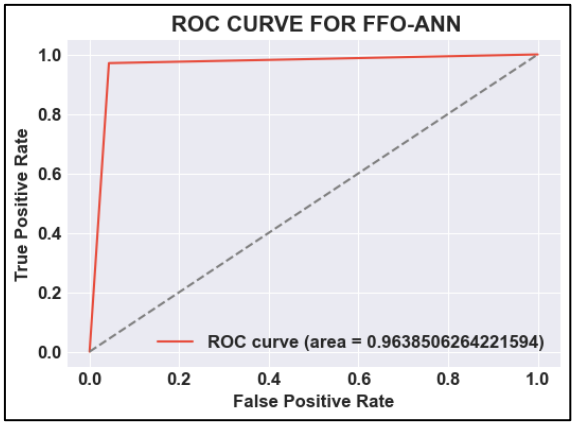
Figure 10: a) Performance metrics of FFO-ANN and b) Confusion matrix of FFO-ANN



a) b)

Figure 10shows that the performance of the Firefly-optimized ANN, in terms of classification performance, is outstanding and balanced with almost equal scores for Accuracy, Precision, Recall, Specificity, and F1-Score all being approximately equal to 0.96 and proving reliable predictions for both classes. Among the two class predictions, this model produced a high number of correct predictions while also yielding a low number of misclassifications, demonstrating overall effectiveness of the model.

Figure 11: ROC curve for FFO-ANN



In figure 11, the ROC curve indicates that the FFO-ANN achieved a high true positive rate and low false positive rate, thus providing strong evidence of the classification effectiveness of the FFO-ANN.

Figure 12: Comparative analysis of proposed and existing methods

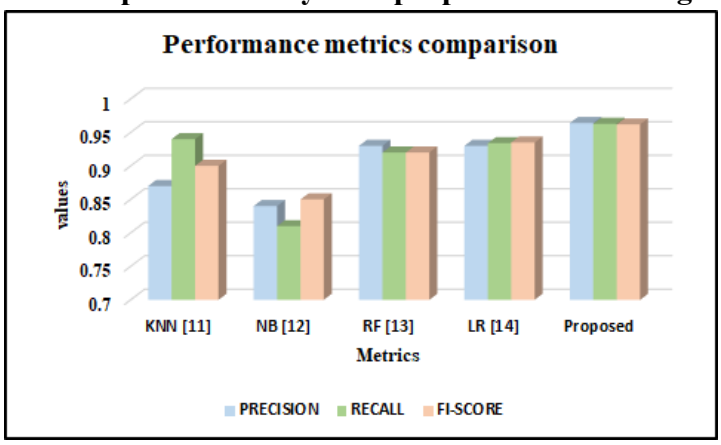


Figure 12indicates that the proposed approach achieves overall better results compared with the alternative approaches as it relates to precision, recall, and F1-score.

5. CONCLUSION

The proposed system allows the system to accurately identify and classify different attacks. The system initially pre-processes the data by using a systematized approach to eliminate noise created while collecting the data and converts the input into a clean single comparison of the data. The system further prepares the features for analysis by providing the same encoded and normalized format. Through an EDA, data patterns fully understood by identifying anomalies that are associated with malicious behaviour. The final classification of an FFO-ANN uses the optimized parameters of a neuron's activation. As of this optimization the FFO-ANN is able to differentiate between normal activity and intrusion attempts precisely, which ultimately allows the proposed system to perform efficiently, accurately and trustworthily under all circumstances as a credible intrusion detection solution and significantly enhance the security of the network. The proposed system is implemented in python and attains the accuracy of 0.964, precision of 0.963, Recall of 0.962, FI-Score of 0.9625 and specificity of 0.9655.

REFERENCES

1. Wang, Xiaosong, Yuxin Qiao, Jize Xiong, Zhiming Zhao, Ning Zhang, Mingyang Feng, and Chufeng Jiang. "Advanced network intrusion detection with tabtransformer." Journal of Theory and Practice of Engineering Science 4, no. 03 (2024): 191-198.
2. Widodo, AkdeasOktanae, BambangSetiawan, and RarasmayaIndraswari. "Machine learning-based intrusion detection on multi-class imbalanced dataset using SMOTE." Procedia Computer Science 234 (2024): 578-583.
3. Javeed, Danish, Muhammad Shahid Saeed, Muhammad Adil, Prabhat Kumar, and Alireza Jolfaei. "A federated learning-based zero trust intrusion detection system for Internet of Things." Ad Hoc Networks 162 (2024): 103540.
4. Amru, Malothu, R. Jagadeesh Kannan, Enthrakandi Narasimhan Ganesh, SuruliveluMuthumarilakshmi, Kuppan Padmanaban, JeyaprakashJeyapriya, and SubbiahMurugan.

"Network intrusion detection system by applying ensemble model for smart home." International Journal of Electrical and Computer Engineering 14, no. 3 (2024): 3485-3494.

5. Panjaitan, MuktarBahruddin, Janaki Sivakumar, Archana Chabra, Menila James, Harpreet Kaur, and ChetnaVaidKwatra. "Intrusion detection system based on machine learning models: An empirical analysis." In AIP Conference Proceedings, vol. 2930, no. 1, p. 020040. AIP Publishing LLC, 2023.
6. Abed, RuqayaAbdulhasan, EkhlasKadhum Hamza, and Amjad J. Humaidi. "A modified CNN-IDS model for enhancing the efficacy of intrusion detection system." Measurement: Sensors 35 (2024): 101299.
7. Rajasekaran, Arun Sekar, S. Shanmuga Priya, Janaki Sivakumar, and Akshay Varkale. "Cybersecurity Measures Using Machine Learning for Business Applications." In 2023 4th International Conference on Computation, Automation and Knowledge Management (ICCAKM), pp. 01-08. IEEE, 2023.
8. Rao, Deepak Dasaratha, Akhilesh A. Waoo, Murlidhar Prasad Singh, Piyush Kumar Pareek, Shoaib Kamal, and Shraddha V. Pandit. "Strategizing IoT network layer security through advanced intrusion detection systems and AI-driven threat analysis." Full Length Article 12, no. 2 (2024): 195-95.
9. Karthika K, Rangasamy DP. Authorization of Aadhar data using Diffie Helman key with enhanced security concerns. Journal of Intelligent & Fuzzy Systems. (2024).
10. Benmalek, Mourad, and Kamel-Dine Haouam. "Advancing network intrusion detection systems with machine learning techniques." Advances in Artificial Intelligence and Machine Learning 4, no. 03 (2024): 2575-2592.
11. Pandey, Pradeep. "A KNN-Based Intrusion Detection System for Enhanced Anomaly Detection in Industrial IoT Networks." International Journal of Innovative Research in Technology and Science 12, no. 6 (2024): 1-7.
12. Badar, Hafiz Muhammad Sanaullah, Nadeem Iqbal Kajla, Jehangir Arshad, NajiaSaher, Manal Ahmad, and Muhammad Ahsan Jamil. "Lightweight intrusion detection for IoD infrastructure using deep learning." Journal of Computing & Biomedical Informatics (2024).
13. Almolhis, NawafAbdualaziz. "Intrusion Detection Using Hybrid Random Forest and Attention Models and Explainable AI Visualization." Journal of Internet Services and Information Security 15, no. 1 (2025): 371-384.
14. Adeite, AdenusiDauda, OladosuOyebisiOladimeji, OyekolaTheophilusAdekunle, and MuminOlatunjiOladipo. "Real-Time Network Intrusion Detection System with Machine Learning (Logistic Regression)." theme: Science, Technology and the Humanities: An Interdisciplinary Perspective (2024): 204.