Scientific Research Publishing

Journal of Information Security, 2026, 17(3), 209-220
https://www.scirp.org/journal/jis
ISSN Online: 2153-1242
ISSN Print: 2153-1234

Performance Evaluation of a Genetic Neuro-Fuzzy Intrusion Detection System across Multiple Datasets

Mohammad Hamdan[1], Mohammed Assora[1], Mustapha Dakkak[2]

[1]Telecommunication Department, Higher Institute for Applied Science and Technology (HIAST), Damascus, Syria
[2]Information Department, Higher Institute for Applied Science and Technology (HIAST), Damascus, Syria
Email: Mohammad.hamdan@hiast.edu.sy, Mohammed.assora@hiast.edu.sy, mustapha.dakkak@hiast.edu.sy

How to cite this paper: Hamdan, M., Assora, M. and Dakkak, M. (2026) Performance Evaluation of a Genetic Neuro-Fuzzy Intrusion Detection System across Multiple Datasets. *Journal of Information Security*, **17**, 209-220.
https://doi.org/10.4236/jis.2026.173011

Received: June 22, 2025
Accepted: June 6, 2026
Published: June 9, 2026

Abstract

The paper introduces an IDS that combines a genetic-algorithm feature selector with an Adaptive Neuro-Fuzzy Inference System classifier. A genetic algorithm, one of the most prominent heuristic optimization methods, is utilized to select a set of optimal features to serve as inputs to the IDS. The performance of this hybrid approach is rigorously compared with the widely adopted open-source Snort system using several standard benchmark datasets, including KDDCup99, NSL-KDD, UNSW-NB15, Bot-IoT, and CSE-CIC-IDS2018. The primary objective is to create a system capable of learning and detecting previously unknown attacks by harnessing the strengths of neural networks and fuzzy logic, thereby minimizing erroneous classifications—whether considering benign data as malicious or vice versa. The model is trained and tested on five public datasets and benchmarked against Snort. Across all datasets the GA-ANFIS variant attains higher accuracy (≈99%) and markedly lower false-positive rates (<0.3%) than Snort, implying better adaptability to diverse attack patterns. The empirical results demonstrate that the proposed system exhibits substantial potential in enhancing detection accuracy and adaptability to emerging threats.

Keywords

Network Security, Intrusion Detection Systems, Neural Networks, Fuzzy Logic, Genetic Algorithm

1. Introduction

The propagation and increasing complexity of computer networks have led to an unprecedented rise in security threats. As these networks expand and evolve, so

too does the potential for significant damage to critical infrastructure, necessitating the development of more sophisticated protective measures. Among these, intrusion detection systems (IDS) have emerged as a pivotal component of network security. However, the effectiveness of traditional IDS is often limited, particularly in their ability to identify and respond to previously unknown attacks. This limitation underscores the need for advanced methods that can analyze the behavior and operational sequences of attacks in real-time to enhance detection capabilities.

Machine learning (ML), a subset of artificial intelligence (AI), has become a cornerstone in addressing these challenges. The integration of ML techniques into IDS has demonstrated significant potential in improving their ability to detect and mitigate emerging threats. By enabling systems to learn from vast amounts of network data and adapt to new attack vectors, ML enhances the overall robustness of IDS.

This study reviews several machine learning techniques employed in the realm of intrusion detection, with a particular focus on the Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS combines the learning capabilities of neural networks with the decision-making processes of fuzzy logic, creating a hybrid system that is both adaptive and precise. Additionally, this research explores the application of genetic algorithms for feature selection during the training and testing phases, a method that optimizes the input features, thereby improving the system's performance.

The application of AI and ML in IDS represents a modern approach to enhancing network security, capable of adapting to the rapid evolution of attack types and structures. This study's contributions are threefold:

- **Integration of Weighted Logic:** The proposed system leverages fuzzy logic to enhance decision accuracy, determining whether a data packet constitutes an attack or normal network activity.
- **Neural Network Learning:** The inherent learning ability of neural networks is harnessed to continuously improve the detection capabilities of the IDS.
- **Genetic Algorithm Optimization:** Feature selection is optimized through genetic algorithms, ensuring that the most relevant attributes are used in the training process.

These elements collectively shape the training process of the proposed system, aiming to create an IDS capable of learning and adapting to the rapidly changing landscape of cyber threats. The system's performance is then benchmarked against that of the widely used Snort IDS, providing a comprehensive analysis of its efficacy in real-world scenarios.

2. Literature Review

The application of machine learning techniques to intrusion detection systems (IDS) has been a focal point of research, with numerous studies exploring various approaches to enhance detection capabilities. One early study introduced the use

of feedforward neural networks combined with K-Nearest Neighbor (KNN) classifiers to improve the accuracy of IDS. This approach, however, faced challenges due to the high dimensionality of features, which led to difficulties in optimizing the classifier's performance. To mitigate this issue, the researchers implemented a genetic algorithm for feature selection, which successfully reduced the feature space and enhanced the system's overall efficiency. The effectiveness of this method was demonstrated using the KDD Cup99 dataset, where improvements in detection rates were observed, albeit with an accompanying increase in false positive rates.

Building on this work, Patil *et al.* [1] developed a framework that leverages the Binary Bat algorithm for feature extraction, applying it specifically to the UNSW-NB15 dataset. The Binary Bat algorithm, known for its capability to solve complex optimization problems, was employed to optimize the selection of relevant features, thereby improving the IDS's detection accuracy. Despite the algorithm's strengths, the study highlighted the trade-offs between detection accuracy and computational efficiency, emphasizing the need for further refinement in balancing these aspects.

In a similar vein, researchers have explored the integration of various machine learning classifiers into IDS frameworks. For instance, the work of Tsai *et al.* [2] explored the use of ensemble learning methods, combining classifiers such as Random Forest (RF), Support Vector Machines (SVM), and K-Nearest Neighbors (KNN) to enhance detection performance across multiple datasets. The study demonstrated that ensemble methods, particularly those incorporating boosting algorithms like AdaBoost, were effective in improving detection rates while maintaining a low false positive rate. However, the computational complexity associated with these methods remains a concern, especially when applied to large-scale datasets.

Further research by Khan *et al.* [3] focused on the comparative performance of different machine learning techniques, including Artificial Neural Networks (ANN), SVM, and Decision Trees (DT), in the context of IDS. Their study, which utilized the CSE-CIC-2018 dataset, provided a detailed analysis of each classifier's strengths and limitations. The findings suggested that while deep learning models like ANN showed high accuracy, traditional classifiers such as SVM and DT offered better interpretability and faster processing times, making them more suitable for real-time intrusion detection applications.

In addition to these approaches, studies have also explored the role of feature selection techniques in enhancing IDS performance. For example, Guyon and Elisseeff [4] provided a comprehensive review of feature selection methods, emphasizing their importance in reducing the dimensionality of data and improving model interpretability and accuracy. Their work underscores the critical role that feature selection plays in the development of efficient and effective IDS.

More recent investigations have further demonstrated the advantages of hybrid neuro-fuzzy models in intrusion detection contexts. Aljanabi *et al.* [5] developed

a fuzzy logic-based IDS optimized using neural network learning, demonstrating enhanced detection of distributed denial-of-service (DDoS) attacks in IoT environments. Their work underscores the capacity of neuro-fuzzy integration to handle the inherent uncertainty and noise present in network traffic data while maintaining interpretable rule sets. In a complementary study, Ishaque *et al.* [6] proposed a hybrid IDS framework combining genetic algorithms for feature optimization with a fuzzy inference classifier, achieving notable improvements in both detection accuracy and computational efficiency across multiple benchmark datasets. These studies corroborate the effectiveness of the genetic neuro-fuzzy hybrid paradigm and highlight its applicability to contemporary cybersecurity challenges, including IoT-specific threats and large-scale network monitoring. The model presented in this paper builds upon these foundations by integrating ANFIS with a genetically optimized feature selector, thereby offering a unified framework capable of addressing both feature dimensionality reduction and adaptive classification.

These studies collectively highlight the diverse approaches taken by researchers to refine machine learning methods for IDS, with each contributing to the understanding of the trade-offs and challenges inherent in this field. The ongoing advancements in feature selection, ensemble learning, and optimization algorithms continue to push the boundaries of what is possible in intrusion detection, paving the way for more robust and adaptive security systems.

3. Evaluation Metrics and Datasets

Intrusion Detection Systems (IDS) are essential components of network security, designed to monitor and analyze data packets within a network to detect and respond to potential threats. IDS can be categorized based on their operational mode [7]: Host-Based IDS (HIDS) operate on individual devices, while Network-Based IDS (NIDS) monitors traffic across an entire network. Additionally, IDS can be classified by their detection methods: Signature-Based Detection, which identifies threats by matching network traffic against known attack signatures, and Anomaly-Based Detection, which detects deviations from established patterns of normal behavior, potentially indicating novel or zero-day attacks.

Evaluating IDS performance requires the use of specific metrics to assess its accuracy and reliability. The key metrics include:

- **False Negative (FN):** Occurs when the IDS fails to identify an actual attack, representing a significant security lapse.
- **False Positive (FP):** Occurs when the IDS incorrectly identifies benign activity as malicious, leading to unnecessary alerts.
- **True Positive (TP):** Represents the correct identification of an attack.
- **True Negative (TN):** Indicates the correct identification of non-malicious activity as benign.

The overall effectiveness of an IDS can be quantified using the following metrics [8]:

- **Detection Rate (DR):** The proportion of actual attacks correctly identified by the IDS, calculated as the ratio of TP to the total number of attacks (TP + FN).
- **False Positive Rate (FPR):** The proportion of benign events incorrectly classified as threats, calculated as the ratio of FP to the total number of benign events (FP + TN).

The equations for Accuracy and FPR are given by:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{1}$$

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{2}$$

To evaluate and benchmark IDS performance, several well-established datasets are commonly employed in research:

- **KDD Cup 99:** [9] Originating from the DARPA 1998 benchmark, this dataset comprises approximately 4 GB of TCP dump data, organized into 41 attributes, both continuous and discrete, across 22 attack types. These attacks are classified into four categories: Denial of Service (DoS), Root to Local (R2L), User to Root (U2R), and Probe. Although widely used, the KDD Cup 99 dataset has faced criticism for containing redundant records, which may bias machine learning models and lead to overfitting.
- **NSL-KDD:** [10] Developed as an improvement over the KDD Cup 99, the NSL-KDD dataset addresses the issues of redundancy and data imbalance by removing duplicate records. It retains the original number of attributes and types of attacks, offering a more balanced and reliable benchmark for evaluating IDS performance.
- **UNSW-NB15:** [11] Compiled by the Australian Centre for Cyber Security (ACCS), this dataset offers a realistic portrayal of modern network traffic and includes nine attack types such as Analysis, Backdoors, DoS, Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, and Worms. The dataset is widely recognized for its applicability in testing IDS in contemporary network environments.
- **CSE-CIC-IDS2018:** [12] This dataset contains a wide variety of 14 different attack types, including DoS GoldenEye, Heartbleed, DoS Hulk, DoS SlowHTTP, DoS Slowloris, DDoS, SSH-Patator, FTP-Patator, Brute Force, XSS, Botnet infiltration, PortScan, and SQL injection. It is highly valued for its comprehensive coverage of modern attack vectors, making it an excellent resource for evaluating IDS in realistic scenarios.
- **Bot-IoT:** [12] Created within a simulated environment at UNSW Canberra's Cyber Range Lab, this dataset includes over 72 million records of various IoT-based attacks, such as DDoS, DoS, OS and Service Scan, Keylogging, and Data Exfiltration. Its detailed categorization of DDoS and DoS attacks, particularly in IoT contexts, provides a valuable resource for assessing IDS performance in increasingly prevalent IoT environments.

Recent studies have emphasized the importance of evaluating IDS using these

datasets to reflect contemporary network environments and evolving threat landscapes. For instance, recent work by Sharafaldin *et al.* [10] highlights the use of the CSE-CIC-IDS2018 dataset to develop and benchmark new IDS models, pointing to the dataset's richness in representing modern attack strategies. Similarly, Moustafa *et al.* [11] have explored the UNSW-NB15 dataset's relevance in modeling real-world network traffic and its effectiveness in assessing IDS capabilities against complex, multifaceted threats.

4. Proposed Model

The proposed model for enhancing the performance of Intrusion Detection Systems (IDS) involves several critical steps, each aimed at optimizing the system's ability to detect malicious activities. The first step in this process involves converting textual attributes within the datasets into numerical attributes, a necessary transformation for most machine learning algorithms. All datasets are divided using an 80% training and 20% testing split, with stratified sampling to preserve the original class distribution. A fixed random seed of 42 is employed across all experiments to ensure reproducibility. No k-fold cross-validation was applied in this study; results are reported on the held-out test sets. This conversion ensures that all data are in a format suitable for processing and analysis. Following this, the data undergoes a standardization process, where each attribute is scaled based on its minimum and maximum values. This normalization is crucial to prevent any single attribute from disproportionately influencing the model due to differences in scale, and is performed using the Unit Range (UR) method, expressed as:

$$\mathrm{UR} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

where X represents the value of the attribute, and $X_{\max}$ and $X_{\min}$ are the maximum and minimum values of that attribute, respectively.

After standardization, the next step involves feature selection, a critical process in which the most relevant attributes are identified and retained for model training. The selection process is conducted using a genetic algorithm (GA), which is known for its effectiveness in optimization problems, particularly in reducing the dimensionality of data while preserving its informative value. The GA operates by encoding the attributes as chromosomes, performing operations such as selection, crossover, and mutation to evolve the population of attributes across generations. The goal is to identify a set of features that maximizes the fitness function. The fitness function for the genetic algorithm is defined as the F1-score achieved by a lightweight ANFIS classifier trained on the candidate feature subset. The F1-score—the harmonic mean of precision and recall—is selected because it provides a balanced assessment of classification quality, particularly in imbalanced datasets where attack samples may be significantly outnumbered by normal traffic. This metric directly reflects the discriminative power of the selected features, in contrast to simpler heuristics such as the attack-to-normal ratio, which does not account for actual classification outcomes. This approach is particularly effective in high-speed

networks where rapid processing is essential.

The feature selection process is followed by the implementation of the Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS combines the learning capabilities of neural networks with the decision-making logic of fuzzy systems, creating a hybrid model that can adapt to the complexities of network traffic patterns. The model is trained using the selected features, and its performance is tested across various datasets to ensure robustness and generalizability.

Figure 1 illustrates the architecture of the proposed model, detailing the flow from raw data preprocessing through to feature selection and ANFIS implementation.

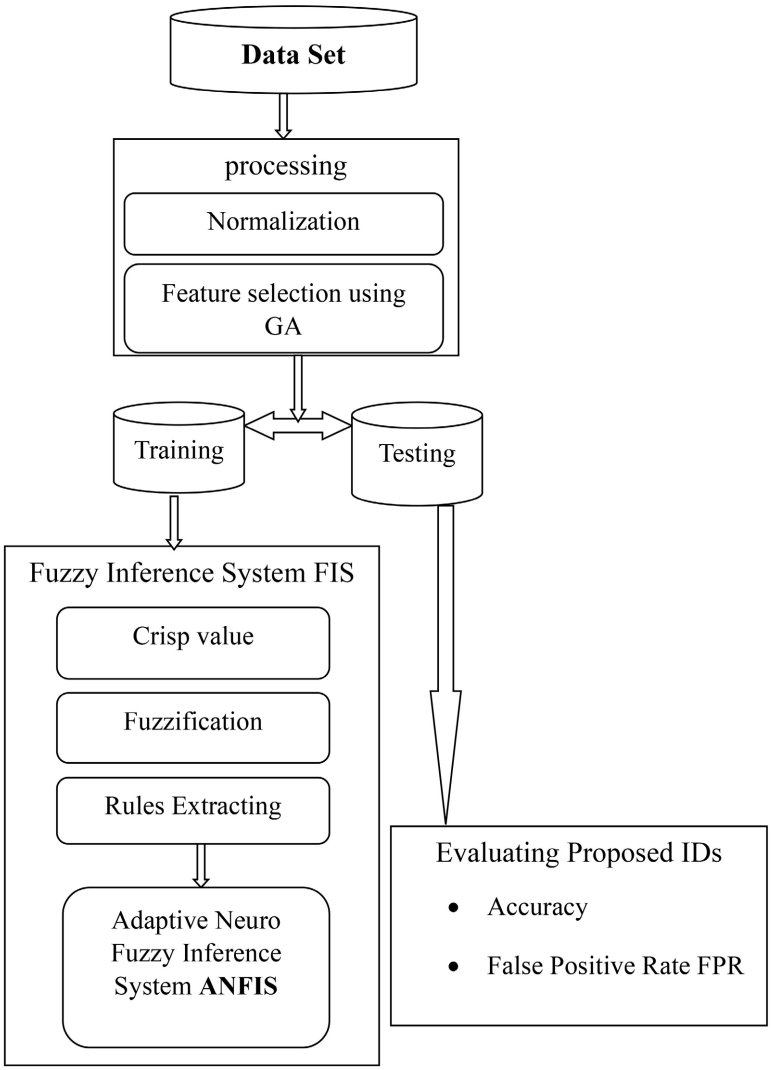

Figure 1. Proposed model architecture.

4.1. Preprocessing Phase

Before any advanced processing, the dataset undergoes a preliminary phase where no treatment is applied. Subsequently, textual attributes are converted into nu-

merical values. Various methods exist for this conversion, but the most common approach is chosen to ensure that attribute values remain consistent across training and testing datasets.

4.2. Standardization of Numerical Features

Once the conversion is complete, the dataset's attributes are standardized. This involves scaling the values to a specific range, typically [0, 1], to facilitate more effective model training and to ensure that the model's performance is not skewed by attributes with larger numerical ranges.

4.3. Feature Selection Process

The genetic algorithm is utilized for feature selection, where attributes are treated as chromosomes. Through iterative processes of selection, crossover, and mutation, the algorithm optimizes the feature set. During crossover, for example, two parent chromosomes are combined to produce offspring that inherit traits from both, as demonstrated by the bit-switching technique. The fitness of these offspring is evaluated, and those with the highest fitness values are selected for the next generation, as determined by the F1-score fitness function described above.

The mutation phase introduces small random changes to the chromosomes to maintain diversity in the population and prevent premature convergence on suboptimal solutions. If Parent 1 = 1[01]00[01]110 and Parent 2 = [0]01[10 10010] when performing the regular pattern, we have offspring1 = 0011001010 and offspring 2 = 1010010110. We also limit the probability of crossover to 60% as this probability gives an effective performance.

The probability of mutation is typically set at 1%, which is effective in balancing exploration and exploitation within the algorithm. At the conclusion of this feature selection process, a set of 21 attributes with the highest impact on model performance is identified. Table 1. Shows the selected features to use as inputs for the ANFIS model, ensuring that the system is both efficient and effective in detecting intrusions.

Table 1. The 21 features selected by genetic algorithm.

No.	Feature Name	Description	No.	Feature Name	Description
1	duration	Length of the connection	13	num_compromised	Number of compromised conditions
2	protocol_type	Type of protocol (tcp, udp, icmp)	14	root_shell	1 if root shell is obtained
3	service	Network service on the destination	15	su_attempted	1 if "su root" command attempted
4	flag	Status flag of the connection	16	num_root	Number of root accesses
5	src_bytes	Bytes sent from source to destination	17	num_file_creations	Number of file creation operations
6	dst_bytes	Bytes sent from destination to source	18	num_shells	Number of shell prompts

Continued

7	land	1 if connection is from/to same host/port	19	num_access_files	Number of operations on access control files
8	wrong_fragment	Number of wrong fragments	20	num_outbound_cmds	Number of outbound commands in an ftp session
9	urgent	Number of urgent packets	21	is_host_login	1 if login belongs to "host" list
10	hot	Number of "hot" indicators			
11	num_failed_logins	Number of failed login attempts			
12	logged_in	1 if successfully logged in; 0 otherwise			

Note: The same feature subset is applied consistently across all five datasets. Where a dataset lacks one of these attributes, a zero-value placeholder is inserted to maintain uniform input dimensionality.

5. Implementation

The implementation of the proposed Adaptive Neuro-Fuzzy Inference System (ANFIS) model involves integrating the 21 selected features identified during the feature selection process. These features represent critical attributes that significantly impact the performance of the IDS. The ANFIS model was implemented using MATLAB, specifically leveraging the Fuzzy Logic Toolbox to manage the membership functions and the rule-based inference system.

For each of the 21 features, we defined five membership functions, representing varying degrees of attribute association: large, medium, small, low, and very small. This granularity allows the model to capture subtle differences in the data, improving its ability to distinguish between normal and malicious activities. The membership functions were constructed by dividing each attribute's domain into five sections, using the calculated arithmetic mean and standard deviation to ensure precise segmentation. Table 2. Shows the model hyper-parameters.

Table 2. GA and ANFIS hyper-parameter configuration.

Parameter (Genetic Algorithm)	Value (Genetic Algorithm)	Parameter (ANFIS)	Value (ANFIS)
Population size	50	Membership function type	Generalized bell-shaped (gbellmf)
Number of generations	100	Number of membership functions per input	5
Crossover probability	0.60 (60%)	Number of epochs	200
Mutation probability	0.01 (1%)	Learning algorithm	Hybrid (LSM + Gradient Descent)
Selection method	Roulette wheel selection	Output membership function	Linear
Chromosome length	41 bits (one per original feature)	AND method	Product
Fitness function	F1-score	OR method	Probabilistic OR (Probor)
		Defuzzification method	Weighted average (wtaver)

Training of the ANFIS model employed a hybrid learning algorithm, combin-

ing the Least Squares Method (LSM) for fine-tuning the linear parameters in the fuzzy inference system and the Gradient Descent (GD) method for optimizing the non-linear parameters. The LSM was particularly effective in adjusting the parameters of the fourth layer of the ANFIS network, where the membership functions are configured. In contrast, the GD algorithm was utilized to model the weighting system based on the training data.

After determining the optimal values for the parameters in the fourth layer, the training process continued with a backward pass, where the membership function parameters were further refined using the training dataset as summarized in Table 3:

Table 3. ANFIS system setup.

Parameter	Value
Number of entrances	21
Number of membership affiliates per entry	5
Output	Constant
Duplicate Count	200
Learning Algorithm	Hybrid Algorithm

To benchmark the performance of the proposed GA-ANFIS system, we implemented a comparative analysis using the Snort IDS, a widely recognized intrusion detection system. The implementation details are summarized in Table 4, which outlines the configurations and computational resources used for both systems.

Table 4. General summary of the systems and tools used in the proposed work.

Tools	Platform
FIS + ANFIS	MATLAB
Snort IDs	a Linux Ubuntu Server

6. Discussion

The performance of the proposed GA-ANFIS system was evaluated using standard datasets, including KDDCup 99, NSL-KDD, UNSW-NB15, CSE-CIC-IDS2018, and Bot-IoT. These datasets were chosen for their widespread use in IDS research and their ability to represent a diverse range of attack types and network conditions.

The comparison between the GA-ANFIS system and the Snort IDS is presented in Table 5. The results indicate that the GA-ANFIS system consistently outperformed Snort in terms of both accuracy and False Positive Rate (FPR). Notably, the GA-ANFIS system achieved an accuracy of 99.72% with an FPR of 0.28% on the KDDCup 99 dataset, compared to Snort's 92% accuracy and 12% FPR. Similarly, significant improvements were observed across the other datasets, with the GA-ANFIS system showing superior performance.

Table 5. Performance comparison between GA-ANFIS and Snort IDS.

Dataset	GA-ANFIS Accuracy (%)	GA-ANFIS FPR (%)	Snort Accuracy (%)	Snort FPR (%)
KDDCup 99	99.72	0.28	92.00	12.00
NSL-KDD	98.83	0.31	91.70	12.54
UNSW-NB15	99.91	0.19	95.62	10.73
CSE-CIC-IDS2018	99.93	0.18	96.23	10.12
Bot-IoT	99.92	0.19	95.81	10.35

The superior performance of the GA-ANFIS system over Snort can be attributed to several key factors. First, the integration of neural networks with fuzzy logic in the ANFIS model allows for more nuanced decision-making, particularly in cases where traditional rule-based systems might fail to capture complex patterns in the data. The genetic algorithm's role in optimizing feature selection further enhances the system's ability to accurately identify relevant attributes, thus improving the overall accuracy and reducing the FPR.

The performance discrepancies between the datasets also highlight the importance of data quality and diversity in training machine learning models. For example, the GA-ANFIS system performed exceptionally well on the CSE-CIC-IDS2018 dataset, likely due to the dataset's comprehensive representation of modern attack vectors. Conversely, the NSL-KDD dataset, which contains some redundant records, posed more challenges, potentially leading to suboptimal training and higher error rates.

7. Conclusions

This research presents a generalized methodology that combines machine learning and data mining techniques to enhance the learning process of IDS. Future research could focus on refining the rule extraction process within the ANFIS model, possibly by exploring alternative methods such as fuzzy association rules or hybrid algorithms that incorporate deep learning components.

Additionally, expanding the model to detect a wider range of attack types, beyond simple binary classification, could improve its applicability in real-world scenarios. Further exploration into unsupervised learning techniques, which do not rely on labeled data, could also provide valuable insights into detecting previously unknown or evolving threats.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Patil, R., Dudeja, H. and Modi, C. (2019) Designing an Efficient Security Framework for Detecting Intrusions in Virtual Network of Cloud Computing. *Computers & Security*, **85**, 402-422. https://doi.org/10.1016/j.cose.2019.05.016

[2] Tsai, C., Hsu, Y., Lin, C. and Lin, W. (2009) Intrusion Detection by Machine Learning: A Review. *Expert Systems with Applications*, **36**, 11994-12000. https://doi.org/10.1016/j.eswa.2009.05.029

[3] Tait, K.A., *et al.* (2021) Intrusion Detection Using Machine Learning Techniques: An Experimental Comparison. arXiv: 2105.13435.

[4] Guyon, I. and Elisseeff, A. (2003) An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, **3**, 1157-1182.

[5] Aljanabi, M. and Ismail, M.A. (2021) Improved Intrusion Detection Algorithm Based on TLBO and GA Algorithms. *The International Arab Journal of Information Technology*, **18**.

[6] Ishaque, M., Johar, M.G.M., Khatibi, A. and Yamin, M. (2023) A Novel Hybrid Technique Using Fuzzy Logic, Neural Networks and Genetic Algorithm for Intrusion Detection System. *Measurement*: *Sensors*, **30**, Article ID: 100933. https://doi.org/10.1016/j.measen.2023.100933

[7] Saxena, N., Roy, S. and Kim, H. (2017) Machine Learning for Intrusion Detection: A Comprehensive Overview. *IEEE Communications Surveys & Tutorials*, **18**, 1155-1176.

[8] Sokolova, M., Japkowicz, N. and Szpakowicz, S. (2006) Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation. In: Sattar, A. and Kang, B., Eds., *AI* 2006: *Advances in Artificial Intelligence*, Springer, 1015-1021. https://doi.org/10.1007/11941439_114

[9] Tavallaee, M., Bagheri, E., Lu, W. and Ghorbani, A.A. (2009) A Detailed Analysis of the KDD CUP 99 Data Set. 2009 *IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, 8-10 July 2009, 1-6. https://doi.org/10.1109/cisda.2009.5356528

[10] Sharafaldin, I., Habibi Lashkari, A. and Ghorbani, A.A. (2018) Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. *Proceedings of the* 4*th International Conference on Information Systems Security and Privacy*, Funchal, 22-24 January 2018, 108-116. https://doi.org/10.5220/0006639801080116

[11] Moustafa, N. and Slay, J. (2016) The Evaluation of Network Anomaly Detection Systems: Statistical Analysis of the UNSW-NB15 Data Set and the Comparison with the KDD99 Data Set. *Information Security Journal*: *A Global Perspective*, **25**, 18-31. https://doi.org/10.1080/19393555.2015.1125974

[12] Ferrag, M.A., Maglaras, L., Moschoyiannis, S. and Janicke, H. (2020) Deep Learning for Cyber Security Intrusion Detection: Approaches, Datasets, and Comparative Study. *Journal of Information Security and Applications*, **50**, Article ID: 102419. https://doi.org/10.1016/j.jisa.2019.102419