Enhancing intrusion detection system using rectified linear unit function in pigeon inspired optimization algorithm

Agus Tedyyana[1], Osman Ghazali[2], Onno W. Purbo[3]
[1]Department of Informatic Engineering, Politeknik Negeri Bengkalis, Bengkalis, Indonesia
[2]School of Computing, College of Arts and Sciences, University Utara Malaysia, Sintok, Malaysia
[3]Department of Informatic, Institute Technology Tangerang Selatan, Tangerang Selatan, Indonesia

Article Info

Article history:

Received Feb 9, 2023
Revised Oct 12, 2023
Accepted Nov 3, 2023

Keywords:

Feature selection
Intrusion detection system
neural network
Machine learning
Pigeon inspired optimization
Rectified linear unit

ABSTRACT

The increasing rate of cybercrime in the digital world highlights the importance of having a reliable intrusion detection system (IDS) to detect unauthorized attacks and notify administrators. IDS can leverage machine learning techniques to identify patterns of attacks and provide real-time notifications. In building a successful IDS, selecting the right features is crucial as it determines the accuracy of the predictions made by the model. This paper presents a new IDS algorithm that combines the rectified linear unit (ReLU) activation function with a pigeon-inspired optimizer in feature selection. The proposed algorithm was evaluated on network security layer - knowledge discovery in databases (NSL-KDD) datasets and demonstrated improved performance in terms of training speed and accuracy compared to previous IDS models. Thus, the use of the ReLU activation function and a pigeon-inspired optimizer in feature selection can significantly enhance the effectiveness of an IDS in detecting unauthorized attacks.

Corresponding Author:

Agus Tedyyana
Department of Informatics Engineering, Politeknik Negeri Bengkalis
Sungai Alam, Bengkalis, Bengkalis, Riau 28714, Indonesia
Email: agustedyyana@polbeng.ac.id

1. INTRODUCTION

The rapid advancements in network technology have led to the creation of various types of networks, including wireless, cellular, cable, and satellite networks. With wider internet connectivity and high-speed capabilities, today's networks are equipped to handle the increased demand for communication and data exchange. Moreover, the integration of the internet of things (IoT) devices, such as smart homes and cities, has added a new dimension to network technology. The introduction of 5G network technology, offering faster speeds and reduced latency, is another significant milestone in the evolution of networks. However, with the growing global network of millions of computers and devices, the increased use of technology has also brought forth security challenges. In this context, it is crucial to have effective measures in place to safeguard against unauthorized access and attacks [1].

In addition, the development of the internet has also presented various challenges, such as internet crime, privacy, and ethical issues. Therefore, the development and management of the Internet must be carried out by taking into account various related aspects. The development of the internet has also led to the emergence of various types of internet crimes or cybercrimes. This includes illegal activities such as hacking, phishing, spamming, and ransomware. Internet crimes can cause financial loss to individuals and companies, as well as damage reputation and trust. Therefore, prevention and eradication of internet crime are important. Security technology also continues to develop to ward off internet crimes, such as strong threat detection, encryption, and authentication systems. However, despite progress in this field, internet crime is still a serious problem that society, companies, and governments have to deal with.

Intrusion detection system (IDS) is a system used to detect unwanted or unauthorized activity on a computer network [2], [3] IDS can use signature-based or behaviour-based techniques to detect threats. Signature-based techniques use lists of recognized "signatures" or patterns as indicators of unwanted activity, while behavior-based techniques observe network activity to determine whether it is normal or not [4]. The development of IDS continues along with the development of network technology and internet crime. Today's IDSs are also becoming more sophisticated, with the ability to detect more specific threats and provide a more targeted response. In addition, IDS is now easier to integrate with other security systems, such as firewalls and wireless threat detection systems. IDS have also become more scalable, with the ability to monitor larger networks with greater efficiency.

Machine learning can be used to increase the effectiveness of IDS by leveraging the system's ability to learn automatically from data [5]. IDS that use machine learning can learn from the data collected during training to understand patterns of legitimate and unauthorized activity. After training, an IDS can use the knowledge it has acquired to predict unauthorized activity more accurately than an IDS that does not use machine learning, using machine learning in IDS can also help reduce the number of false positives that occur, which are cases where an IDS flags legitimate activity as invalid. By leveraging machine learning capabilities to learn more complex patterns [6].

Pigeon inspired optimization (PIO) is an optimization method based on the pigeon inspired algorithm [7], [8]. PIO is an optimization technique that uses several virtual "pigeons" that represent potential solutions to the problem being solved. Each dove will be updated regularly according to the desired behaviour and will be completed to determine the best solution. PIO can be used to solve many types of optimization problems, such as scheduling, routing, and mapping [9]. This method has shown good results in several research studies and has been applied in various applications, such as production scheduling, truck routing, and network mapping. However, PIO is still classified as a new technique and is still being developed by researchers.

Rectified linear unit (ReLU) is an activation function that is often used in neural networks. The activation function is used to change the output of the neuron layer to the desired value. ReLU is one of the simplest serial activation functions [10]. The ReLU function has several advantages compared to other activation functions, such as not having the gradient vanishing problem that often occurs with sigmoid functions, having faster computations, and simplifying the neural network training process.

2. RESEARCH METHOD

As explained in the previous section, PIO is an optimization algorithm inspired by the behaviour of pigeons. This algorithm can be used to find the best solution to an optimization problem. This section will explain how PIO can be used to solve various kinds of optimization problems, including feature selection problems. Feature selection is the process of selecting relevant features from a data set for use in a machine-learning model.

Selection of the right features can help improve model accuracy and reduce computation time. PIO can be used for feature selection by optimizing the model's performance function by using the selected features. Each feature will be represented by a dove, and the dove will be optimized through PIO algorithm iterations. At each iteration, the pigeon will update its position and speed according to predetermined rules, and its performance will be measured using the selected features.

PIO can be used for feature selection by optimizing the model's performance function by using the selected features. However, in some cases, the selected features may still have unwanted negative values. One way to overcome this problem is to add ReLU activation to the PIO algorithm. ReLU activation will ensure that every value generated by the model is positive so that the selected features will be truly relevant and have a positive value. By adding ReLU activation, PIO can produce better and more accurate solutions in the feature selection process.

2.1. Related works

For this purpose, several investigators have also provided theoretical solutions to prevent data breaches, privacy, and confidentiality on the web, namely; for example [5] who ran an enhanced IDS development study using feature selection methods and ensemble learning algorithms to improve accuracy in detecting attacks using the NSL-KDD dataset. IDS that use an ensemble classifier and a hybrid classifier can improve performance compared to an intrusion detection system that uses only one classifier. However, the authors also note that an intrusion detection system that uses an ensemble classifier and a hybrid classifier requires more computation and complexity compared to an intrusion detection system that uses only one classifier.

Evaluate the performance of three different machine learning methods for detecting intrusions on a network. The methods analyzed in this article are support vector machine (SVM) [11], random forest (RF), and extreme learning machine (ELM). The three methods use different datasets consisting of normal network activity and abnormal network activity. The performance of each method is measured using several measures such as the level of accuracy, sensitivity, and specificity. ELM has a better performance compared to SVM and RF in detecting intrusions on the network. ELM has a higher level of accuracy and faster computation time compared to SVM and RF [12], [13]. The PIO algorithm was tested on the KDDCUP 99, NLS-KDD, and University of New South Wales-Network Benchmark 2015 (UNSW-NB15) data sets used in intrusion detection. The test results show that the PIO algorithm can improve IDS performance compared to existing feature selection methods such as information gain and correlation-based feature selection. This algorithm can help improve IDS performance by selecting the most important features from the available data [14].

IDS development using particle optimization feature extraction (PSO) techniques. IDS is a system used to detect attacks on computer networks or systems. In this paper, an enhanced IDS is developed by adding the PSO technique for feature extraction from data used to detect attacks. PSO is used to extract the most relevant features from the NSL-KDD dataset used as test data. This study tested an IDS system that was enhanced with the PSO technique using the NSL-KDD dataset. The results showed that the IDS enhanced with the PSO technique showed better performance compared to the previous method. The level of accuracy produced by IDS which is enhanced by the PSO technique is 97.8%, while the previous method only achieves an accuracy rate of 94.4% [5].

2.2. Pigeon-inspired optimization

PIO is an optimization algorithm based on the behaviour of pigeons [8]. This algorithm is used to select the most important features in the data to be used in the machine learning model. PIO works by using several doves that represent features in the data [7]. Each dove will have a position and speed that will be updated at each iteration according to predetermined rules. The performance of each pigeon will be measured using the features selected by that pigeon and the pigeon with the best performance will be selected as the feature to be used in the machine learning model. PIO is an optimization algorithm that is quite effective in selecting the most important features in the data. This algorithm is also quite easy to understand and its implementation is relatively simple.

2.3. Feature selection

Feature selection plays a pivotal role in data analysis, specifically during the pre-processing phase. The primary aim of this technique is to minimize the volume of features or attributes used in a data model, usually by discarding irrelevant or redundant data. Feature selection is one of the important techniques and is often used in the pre-processing stage [15]. This technique reduces the number of features involved in determining a target class value. Ignored features are usually irrelevant features and excess data [16]–[18]. The main purpose of feature selection is to select the best features from a feature dataset.

Figure 1 feature selection proses explains that the feature selection process, which is an integral step in the machine learning process, plays a significant role in amplifying both the accuracy and efficiency of a model. This process can be best understood by examining the workflow detailed in Figure 1. Feature selection proses, the crucial first task is to select an appropriate dataset for the intrusion detection system (IDS).

Several datasets exist that are widely used in IDS, each offering various data points and features pertinent to the task at hand. The next step is "data sampling." At this point, a subset of data is selected from the chosen dataset to facilitate the feature selection process. This subset of data, or sample, should ideally be representative of the entire dataset to ensure that the findings from the feature selection process are valid and accurate. Next comes the "data pre-processing" stage. During this phase, the sampled data undergo several operations to clean and standardize it. These operations may include transformations, the removal of unnecessary data, and general data-cleaning tasks. The goal here is to refine the data to its most useful state, free from errors and anomalies that might adversely affect the subsequent stages. Following preprocessing, we delve into the core of this process – "feature selection with information gain." In this step, the system identifies the features that are most relevant for the intrusion detection system. The information gain method is employed to gauge the importance of each feature in classifying data as an attack or not. Those features that register a high information gain value are selected for use in the IDS.

Next comes "classification," where an IDS is developed using the features that have been selected in the previous stage. The IDS applies these features to the preprocessed data to detect potential intrusions. Once the system is developed and has been run, the "result of feature selection performance analysis" stage begins. The outcomes of the IDS are meticulously analyzed and tested with the help of metrics such as accuracy, recall, and precision. These performance metrics provide insights into the system's success in identifying intrusions, thereby informing potential improvements to the system. Finally, having analyzed the performance and made necessary adjustments, the process concludes at the "finish" stage. At this point, an IDS, tailored for optimal feature selection and maximum intrusion detection, has been successfully created and validated.

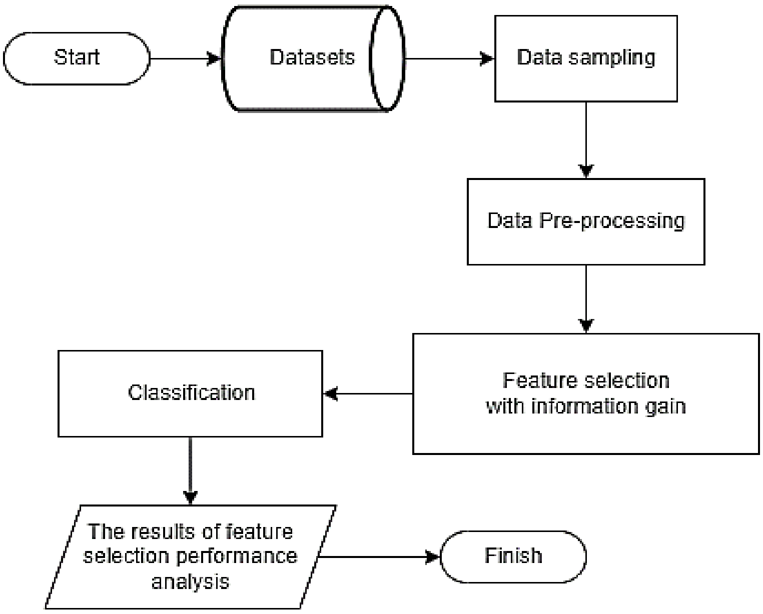

Start
Datasets
Data sampling
Data Pre-processing
Feature selection with information gain
Classification
The results of feature selection performance analysis
Finish

Figure 1. Feature selection proses

2.4. Dataset

Some of the popular datasets used in feature selection for IDS are UNSW-NB15: this dataset contains various types of network attacks, such as a distributed denial-of-service (DDoS) attacks [11], snooping attacks, and counterfeiting attacks [14], [19]. KDD cup 99: this dataset was developed by MIT Lincoln Laboratory and used in KDD Cup 1999, a competition to develop intrusion detection systems [20]. NSL-KDD: this dataset was developed to overcome the drawback of the KDD Cup 99 dataset, namely the dataset is too easy to classify. NSL-KDD contains the same features as KDD Cup 99 and comes with more realistic data and attacks that are harder to detect [21]–[23].

2.5. Modified PIO for feature selection with ReLU

In processing IDS datasets, artificial neural networks can be used to perform data analysis and identify suspicious patterns [19]. The pigeon-inspired optimization algorithm can be used to find the best parameters of the artificial neural network used so that it can make better predictions. Feature selection can be used to select useful features from the data used so that artificial neural networks can focus on only important features and reduce the dimensionality of the data [24], [25] ReLU is an activation function used in artificial neural networks [26].

This function converts the input to a value equal to or greater than 0, truncating all negative values to 0. This is used to avoid the "vanishing gradient" problem and make training faster and more efficient. Based on Figure 2 ReLU_PIO feature selection design, in feature selection, the PIO algorithm is used to evaluate the features used in intrusion detection and select the most important features to be used in the system.

The ReLU function is used in the pigeon's velocity update stage in PIO. This feature evaluation and selection is vital for the efficiency and effectiveness of the intrusion detection system. The choice of relevant features significantly influences the performance of the system, ensuring that it focuses its resources on the most informative aspects of the data. This selective focus not only enhances the accuracy of intrusion detection but also improves the computational efficiency of the system, as it avoids wasting processing power on irrelevant or less significant features.

3. RESULTS AND DISCUSSION

In this study, the quantitative approach is used as the main method because of certain characteristics, such as performance measurement, data set evaluation, and the usefulness of the results. This study uses the deductive cycle because it is more appropriate to test the proposed solution. Based on the proposed framework, the study design is categorized into 3 phases namely pre-processing, processing using PIO, and evaluation, which will be discussed further. in subsequent subsections.

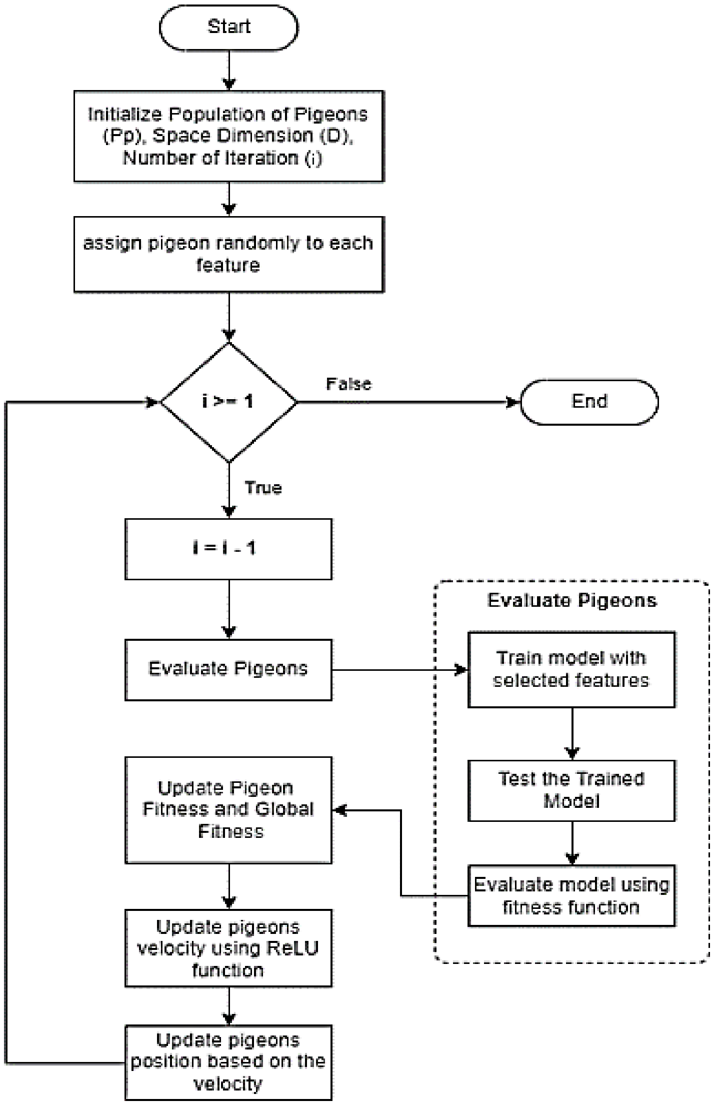

Start
Initialize Population of Pigeons (Pp), Space Dimension (D), Number of Iteration (i)
assign pigeon randomly to each feature
i >= 1
False
End
True
i = i - 1
Evaluate Pigeons
Evaluate Pigeons
Train model with selected features
Test the Trained Model
Evaluate model using fitness function
Update Pigeon Fitness and Global Fitness
Update pigeons velocity using ReLU function
Update pigeons position based on the velocity

Figure 2. ReLU_PIO feature selection design

3.1. Pre-processing phase

In the filtering step, irrelevant or redundant data is removed from the input data stream. This can include removing duplicate packets, filtering known good traffic, and removing data that is irrelevant to IDS detection capabilities. In the normalization step, the data is converted into a standard format that can be easily processed by the detection algorithm. This can include converting data into a unified format, such as converting all internet protocol addresses to canonical form or converting timestamps to the same time zone. In the feature extraction step, we extract all the relevant features from the data that can be used by the detection algorithm. This can include extraction of statistical features, such as average packet size or the number of connections per second, or extraction of structural features, such as the order of packets in a connection or the order of bytes in a packet. Meanwhile, the data pre-processing process itself consists of three main steps, namely label transfer and data transfer, eliminating duplication, and data normalization. In the label transfer and data transfer steps, all symbolic data is transferred to numeric values. Also, class field input is changed to binary class 0 or 1, where 0 indicates normal record while 1 indicates attack record regardless of the attack type.

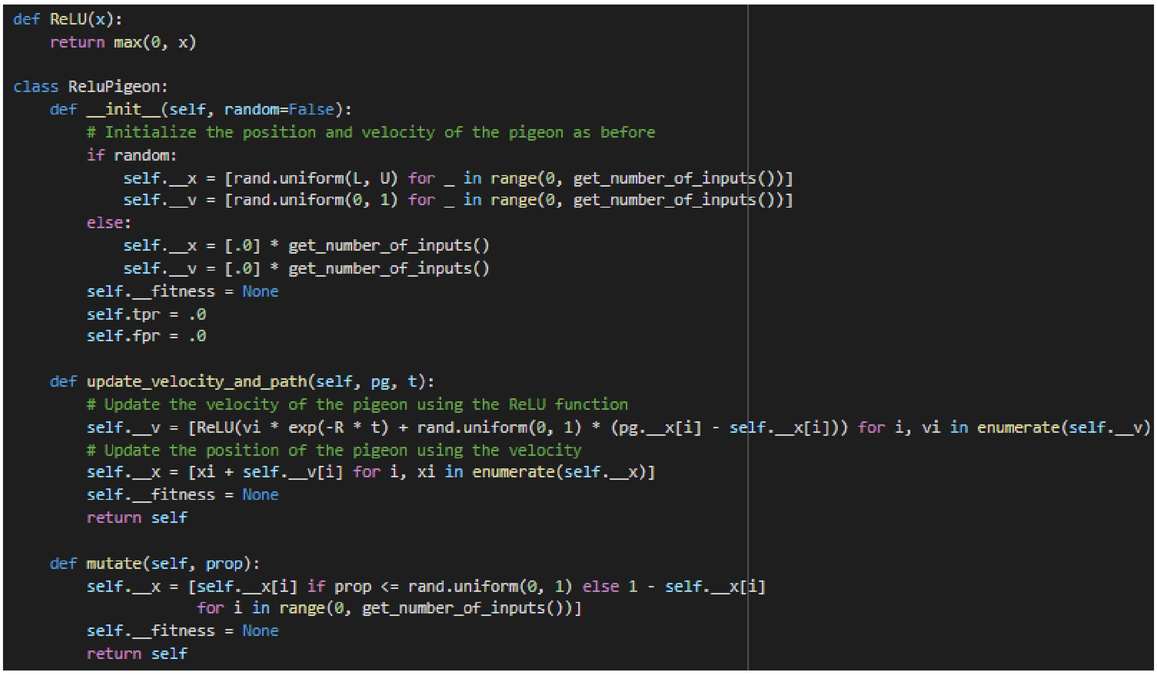
3.2. Processing phase

The PIO algorithm used in this study has been modified by researchers to produce a more effective feature selection process, although it has recently been known that the PIO algorithm itself is effective in solving many optimization problems on feature selection. The modified PIO algorithm is designed to overcome the limitations that occur in commonly used PIO approaches, in this case, the PIO is the rectified linear unit (ReLu) approach. The ReLu activation function has become the most widely used activation function for deep learning applications with state-of-the-art results. It usually achieves better performance and generalization in deep learning compared to sigmoid activation functions. The use of ReLU and PIO can be combined to create effective and fast learning algorithms. In this algorithm, each data input will go through an activation process using the ReLU function until it produces a non-negative activation. Then, the pigeon inspired optimizer algorithm will be used to optimize the weights and biases to minimize the error in each iteration. By combining these two algorithms, it is expected that the neural network can learn quickly and have high accuracy in performing classification or regression. ReLU is a type of activation function used in neural networks to perform classification or regression. This function has the characteristic of giving a value of 0 if the input is negative and maintaining the value of t is positive, explned mathematically as shown in (1).

$$(x) = \max(0, x) = \begin{cases} x_i \; if x_i > 0 \\ 0 \; if \; X_i < 0 \end{cases} \qquad (1)$$

Based on (1), if the found solution is not a neighbor of the global solution, the probability of updating its position regarding the global solution is higher than the probability of the current solution being a neighbor of the global solution. This is illustrated in the following algorithm, which explains how a modified PIO works. This process aids in optimizing solution searching by enhancing the likelihood of finding a solution closer to the global optimum. This faster position updating allows the algorithm to achieve convergence more efficiently and effectively. Thus, improvements in the PIO algorithm assist in attaining optimal solutions more swiftly and accurately.

In Figure 3, the "ReLU_PIO algorithm" is depicted. The code presented is an implementation of ReLU_PIO. The init method initializes a new instance of the ReluPigeon class, assigning either random or zero values for the bird's position and velocity. The update_velocity_and_path method is employed to refresh the bird's position and velocity based on the ReLU function. The mutate method randomly alters the bird's position. The desirable_destination_center method ascertains the ideal destination center for the most optimal birds. Lastly, the update_path method modifies the bird's position based on the desired center position. The ReLU function is described as a function that yields 0 if the input is negative, and returns the input value if the input is positive.



```
def ReLU(x):
    return max(0, x)

class ReluPigeon:
    def __init__(self, random=False):
        # Initialize the position and velocity of the pigeon as before
        if random:
            self.__x = [rand.uniform(L, U) for _ in range(0, get_number_of_inputs())]
            self.__v = [rand.uniform(0, 1) for _ in range(0, get_number_of_inputs())]
        else:
            self.__x = [.0] * get_number_of_inputs()
            self.__v = [.0] * get_number_of_inputs()
        self.__fitness = None
        self.tpr = .0
        self.fpr = .0

    def update_velocity_and_path(self, pg, t):
        # Update the velocity of the pigeon using the ReLU function
        self.__v = [ReLU(vi * exp(-R * t) + rand.uniform(0, 1) * (pg.__x[i] - self.__x[i])) for i, vi in enumerate(self.__v)
        # Update the position of the pigeon using the velocity
        self.__x = [xi + self.__v[i] for i, xi in enumerate(self.__x)]
        self.__fitness = None
        return self

    def mutate(self, prop):
        self.__x = [self.__x[i] if prop <= rand.uniform(0, 1) else 1 - self.__x[i]
                    for i in range(0, get_number_of_inputs())]
        self.__fitness = None
        return self
```

Figure 3. ReLU_PIO algoritm

3.3. Performance metrics

PIO, as a conventional technique, has been in use for years and is widely accepted in various fields. It primarily utilizes proportional, integral, and derivative control, which is often employed for problem-solving and optimization in multiple domains. Despite its straightforwardness and capability to provide stable solutions, PIO may struggle with complex, non-linear systems and might not efficiently handle high-dimensional tasks. On the other hand, ReLU_PIO is an enhanced version of PIO. It incorporates a rectified linear unit (ReLU) to tackle some limitations posed by PIO. The ReLU function, a type of activation function, is known for introducing non-linearity in the network, enabling the learning and representation of more complex patterns. The integration of ReLU with PIO, termed as ReLU_PIO, renders it suitable for handling complex, non-linear problems, thus augmenting the performance of the conventional PIO method.

Based on the results of training with 50 iterations of the two approaches, namely PIO and ReLU_PIO, a comparison is provided in Table 1. Comparison between PIO and ReLU_PIO, the true positive rate (TPR) is the ratio of the number of true positives recognized by the model to the total number of all positives in reality. As depicted, ReLU_PIO has a higher TPR (0.694) compared to PIO (0.661), showcasing its superior ability in identifying true positives.

The comparative analysis between PIO and ReLU_PIO, as illustrated in Table 1, offers insightful observations on their performance metrics. Although both methods showcase substantial effectiveness in various aspects, ReLU_PIO evidently stands out in key areas. It demonstrates a superior true positive rate (0.694) compared to PIO (0.661), highlighting its enhanced ability to correctly identify positive cases. This is crucial for reducing the chances of type II errors in classifications. ReLU_PIO has a slightly higher false positive rate (0.051) than PIO (0.040), a minor trade-off for its other advantages. The slight increase in FPR does not overshadow ReLU_PIO's commendable performance, particularly in terms of accuracy and F-score. With an accuracy of 0.84, ReLU_PIO proves to be more reliable in making correct predictions compared to PIO with an accuracy of 0.79. Additionally, a higher F-score for ReLU_PIO (0.84) compared to PIO (0.79) further reinforces its balanced performance in terms of precision and recall. In essence, the integration of the ReLU function in ReLU_PIO augments its capabilities, contributing to improved performance in identifying true positives, accuracy, and achieving a balanced f-score. Despite a slight increase in the FPR, ReLU_PIO's overall superior performance metrics establish it as a more effective and reliable approach for handling complex, non-linear problems compared to the conventional PIO method [27], [28].

Table 1. Comparison between PIO and ReLU_PIO

Approach	True positive rate	False positive rate	Accuracy	F-score
PIO	0.661	0.040	0.79	0.79
ReLU_PIO	0.694	0.051	0.84	0.84

4. CONCLUSION

The integration of the ReLU function to enhance the pigeon-inspired optimizer algorithm in feature selection demonstrates significant improvements. The proposed algorithm, tested on the NLS-KDD dataset, effectively evaluates the data, manifesting the enhancement brought about by the ReLU activation function. This paper clearly highlights the augmentation in pigeon position and speed due to ReLU, leading to the enhanced performance of the intrusion detection system (IDS) model as seen in the elevated accuracy and F1-score. The computation with the ReLU function is straightforward, involving only comparison and maximum operations. This simplicity in calculation doesn't undermine its functionality. Instead, it infuses the model with the capability to introduce non-linearity, enabling a more intricate learning of the relationships between various inputs and outputs. This advanced learning capability significantly contributes to the model's improved performance, ensuring more accurate and reliable results in complex, real-world tasks. In essence, the incorporation of ReLU into the pigeon-inspired optimizer not only simplifies the computational process but also amplifies the algorithm's efficacy. This results in a more robust and efficient model, capable of adeptly handling the complexity and non-linearity of various problems, showcasing marked improvements especially in the context of the IDS model tested on the NLS-KDD dataset.

REFERENCES

[1] B. B. Zarpelão, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga, "A survey of intrusion detection in Internet of Things," *Journal of Network and Computer Applications*, vol. 84, pp. 25–37, Apr. 2017, doi: 10.1016/j.jnca.2017.02.009.

[2] E. Min, J. Long, Q. Liu, J. Cui, and W. Chen, "TR-IDS: anomaly-based intrusion detection through text-convolutional neural network and random forest," *Security and Communication Networks*, vol. 2018, pp. 1–9, Jul. 2018, doi: 10.1155/2018/4943509.

[3] J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, “CNN-based network intrusion detection against Denial-of-service attacks,” *Electronics*, vol. 9, no. 6, p. 916, Jun. 2020, doi: 10.3390/electronics9060916.

[4] A. Kim, M. Park, and D. H. Lee, “AI-IDS: application of deep learning to real-time web intrusion detection,” *IEEE Access*, vol. 8, pp. 70245–70261, 2020, doi: 10.1109/ACCESS.2020.2986882.

[5] R. O. Ogundokun, J. B. Awotunde, P. Sadiku, E. A. Adeniyi, M. Abiodun, and O. I. Dauda, “An enhanced intrusion detection system using particle swarm optimization feature extraction technique,” *Procedia Computer Science*, vol. 193, pp. 504–512, 2021, doi: 10.1016/j.procs.2021.10.052.

[6] M. Paricherla, M. Ritonga, S. R. Shinde, S. M. Chaudhari, R. Linur, and A. Raghuvanshi, “Machine learning techniques for accurate classification and detection of intrusions in computer network,” *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 4, pp. 2340–2347, Aug. 2023, doi: 10.11591/eei.v12i4.4708.

[7] B. Zhang and H. Duan, “Three-dimensional path planning for uninhabited combat aerial vehicle based on predator-prey pigeon-inspired optimization in dynamic environment,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 14, no. 1, pp. 97–107, Jan. 2017, doi: 10.1109/TCBB.2015.2443789.

[8] Y. Zhang, H. Huang, H. Wu, and Z. Hao, “Theoretical analysis of the convergence property of a basic pigeon-inspired optimizer in a continuous search space,” *Science China Information Sciences*, vol. 62, no. 7, p. 70207, Jul. 2019, doi: 10.1007/s11432-018-9753-5.

[9] Y. Zhong, L. Wang, M. Lin, and H. Zhang, “Discrete pigeon-inspired optimization algorithm with Metropolis acceptance criterion for large-scale traveling salesman problem,” *Swarm and Evolutionary Computation*, vol. 48, pp. 134–144, Aug. 2019, doi: 10.1016/j.swevo.2019.04.002.

[10] M. Chen, H. Jiang, W. Liao, and T. Zhao, “Efficient approximation of deep ReLU networks for functions on low dimensional manifolds,” *Advances in Neural Information Processing Systems*, 2019.

[11] M. Aljanabi, R. Altaie, S. Talib, A. Hussien Ali, M. A. Mohammed, and T. Sutikno, “Distributed denial of service attack defense system-based auto machine learning algorithm,” *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 1, pp. 544–551, Feb. 2023, doi: 10.11591/eei.v12i1.4537.

[12] I. Ahmad, M. Basheri, M. J. Iqbal, and A. Rahim, “Performancecomparison of support vector machine, random forest, and extreme learning machine for intrusion detection,” *IEEE Access*, vol. 6, pp. 33789–33795, 2018, doi: 10.1109/ACCESS.2018.2841987.

[13] R. A. I. Alhayali, M. Aljanabi, A. H. Ali, M. A. Mohammed, and T. Sutikno, “Optimized machine learning algorithm for intrusion detection,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 1, p. 590, Oct. 2021, doi: 10.11591/ijeecs.v24.i1.pp590-599.

[14] H. Alazzam, A. Sharieh, and K. E. Sabri, “A feature selection algorithm for intrusion detection system based on Pigeon Inspired Optimizer,” *Expert Systems with Applications*, vol. 148, p. 113249, Jun. 2020, doi: 10.1016/j.eswa.2020.113249.

[15] C. Torrano-Gimenez, H. T. Nguyen, G. Alvarez, S. Petrovic, and K. Franke, “Applying feature selection to payload-based web application firewalls,” in *2011 Third International Workshop on Security and Communication Networks (IWSCN)*, IEEE, May 2011, pp. 75–81. doi: 10.1109/IWSCN.2011.6827720.

[16] T. M. T. A. Hamid, R. Sallehuddin, Z. M. Yunos, and A. Ali, “Ensemble based filter feature selection with harmonize particle swarm optimization and support vector machine for optimal cancer classification,” *Machine Learning with Applications*, vol. 5, p. 100054, Sep. 2021, doi: 10.1016/j.mlwa.2021.100054.

[17] M. H. Kamarudin, C. Maple, and T. Watson, “Hybrid feature selection technique for intrusion detection system,” *International Journal of High Performance Computing and Networking*, vol. 13, no. 2, p. 232, 2019, doi: 10.1504/IJHPCN.2019.097503.

[18] N. Cleetus and K. A. Dhanya, “Genetic algorithm with different feature selection method for intrusion detection,” in *2014 First International Conference on Computational Systems and Communications (ICCSC)*, IEEE, Dec. 2014, pp. 220–225. doi: 10.1109/COMPSC.2014.7032651.

[19] S. M. Kasongo and Y. Sun, “Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset,” *Journal of Big Data*, vol. 7, no. 1, p. 105, Dec. 2020, doi: 10.1186/s40537-020-00379-6.

[20] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, SCITEPRESS - Science and Technology Publications, 2018, pp. 108–116. doi: 10.5220/0006639801080116.

[21] R. A. Disha and S. Waheed, “Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based weighted random forest (GIWRF) feature selection technique,” *Cybersecurity*, vol. 5, no. 1, p. 1, Dec. 2022, doi: 10.1186/s42400-021-00103-8.

[22] M. Tabash, M. Abd Allah, and B. Tawfik, “Intrusion detection model using naive bayes and deep learning technique,” *The International Arab Journal of Information Technology*, vol. 17, no. 2, pp. 215–224, Feb. 2020, doi: 10.34028/iajit/17/2/9.

[23] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, “Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model,” *Journal of Computational Science*, vol. 25, pp. 152–160, Mar. 2018, doi: 10.1016/j.jocs.2017.03.006.

[24] L. Abualigah and A. Diabat, “Chaotic binary group search optimizer for feature selection,” *Expert Systems with Applications*, vol. 192, p. 116368, Apr. 2022, doi: 10.1016/j.eswa.2021.116368.

[25] W. Rong, B. Zhang, and X. Lv, “Malicious web request detection using character-level CNN,” in *Part of the Lecture Notes in Computer Science book series (LNSC,volume 11806)*, 2019, pp. 6–16. doi: 10.1007/978-3-030-30619-9_2.

[26] P. Bedi, N. Gupta, and V. Jindal, “Siam-IDS: handling class imbalance problem in intrusion detection systems using siamese neural network,” *Procedia Computer Science*, vol. 171, pp. 780–789, 2020, doi: 10.1016/j.procs.2020.04.085.

[27] J. Xu, Z. Li, B. Du, M. Zhang, and J. Liu, “Reluplex made more practical: leaky ReLU,” in *2020 IEEE Symposium on Computers and Communications (ISCC)*, IEEE, Jul. 2020, pp. 1–7. doi: 10.1109/ISCC50000.2020.9219587.

[28] Y. Yu, K. Adu, N. Tashi, P. Anokye, X. Wang, and M. A. Ayidzoe, “RMAF: relu-memristor-like activation function for deep learning,” *IEEE Access*, vol. 8, pp. 72727–72741, 2020, doi: 10.1109/ACCESS.2020.2987829.

BIOGRAPHIES OF AUTHORS

Agus Tedyyana is a senior lecturer at the Politeknik Negeri Bengkalis, Bengkalis, Riau, Indonesia. He has an educational background in computer science. He has worked in education as a lecturer since 2014. He has been continuing his Doctoral (Ph.D.) studies at the Universiti Utara Malaysia (UUM) Campus in Kedah Darul Aman, Malaysia, since early 2020. His research interests are in computer security. He can be contacted at email: agustedyyana@polbeng.ac.id.

Osman Ghazali is an Associate Professor and the Deputy Dean of the School of Computing, Universiti Utara Malaysia. Osman holds a Ph.D. in Information Technology (Networking) from Awang Had Salleh Graduate School, Universiti Utara Malaysia (AHSGS). He was a visiting research fellow at the School of Engineering & Applied Science, Aston University (EAS), in 2012. In 2011, Osman was the Head of the Computer Science Department School of Computing, Universiti Utara Malaysia. Before that, from 2009 to 2011, he was the Technical Chairperson at the University Teaching and Learning Center, Universiti Utara Malaysia. Dr. Osman's research interest is internetworking, cloud computing, and information security. He has more than 100 publications as refereed book chapters and refereed technical papers in journals and conferences. He is a senior member of the InterNetworks Research Laboratory (IRL). He is also a member of the IEEE and the ACM. He can be contacted at email: osman@uum.edu.my.

Onno W. Purbo is graduated from the Department of Electrical Engineering, Bandung Institute of Technology, in 1987. In 1989, he completed his postgraduate education at McMaster University, Canada, in the field of Semi-Conductor Laser. Five years later, he received his Ph.D. from the University of Waterloo, Canada, in the field of Integrated Circuit Technology for satellites, In November 2020, he received the postel service award from the internet society. Postel Service Award was given to Onno for his outstanding contribution to the development of internet technology in Indonesia. He can be contacted at email: onno@indo.net.id.