Journal of Theoretical and Applied Information Technology
15th April 2020. Vol.98. No 07

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AN EFFICIENT ANOMALY INTRUSION DETECTION METHOD WITH EVOLUTIONARY KERNEL NEURAL NETWORK RANDOM WEIGHTS

SAMIRA SARVARI [1], NOR FAZLIDA MOHD SANI [1], ZURINA MOHD HANAPI [2], MOHD TAUFIK ABDULLAH [1]

[1]Department of Computer Science, Faculty of Computer Science and Information Technology Universiti Putra Malaysia,43400 UPM Serdang Selangor, MALAYSIA
[2] Department of Communication Technology and Network, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia,43400 UPM Serdang Selangor, MALAYSIA

E-mail: [1]samirasarvari82@yahoo.com

ABSTRACT

Internet security requirements are increasing due to the growth of internet usage. One of the most efficient approaches used to secure the usage of the internet from internal and external intruders is Intrusion Detection System (IDS). Considering that using a combination of ANN and EA can produce an advanced technique to develop an efficient anomaly detection approach for IDS, several types of research have used ENN algorithms to detect the attacks. To enhance the efficiency of anomaly-based detection in terms of accuracy of classification, in this paper, the evolutionary kernel neural network random weight is proposed. This model is applied to the NSL-KDD dataset, an improvement of the KDD Cup'99. The proposed method achieved 99.24% accuracy which shows that the novel algorithm suggested is more superior to existing ones as it provides the optimal overall efficiency.

Keywords: *Intrusion detection systems (IDSs), Multilayer perceptron (MLP), Multiverse optimizer (MVO), NSL-KDD Dataset*

1. INTRODUCTION

Currently, the number of internet networks has grown and resulted in a corresponding increase in the number of internet users. Hence, there will be a dramatic increase in the volume of confidential and sensitive data and information carried across these online networks. This opens the door to the possibility of intrusions and malicious attacks in efforts to hijack such sensitive data and information. Therefore, in such a situation there is an urgent need to ensure the safety and security of such information for users. These proposed security systems must be precise, reliable and configurable safety systems [1]. It is true that there already exist numerous protection approaches in the form of user authentication, encrypting of information, controlled access, and firewalls but unfortunately, all these are essentially a primary line of defense in terms of computer security, but a technique to ensure total and impenetrable network defense is still to be found.

Many researchers have been attempting to develop a
novel security system for attack detection that among the important approaches Intrusion Detection System (IDS) is able to acquire higher security and detect threats from both inside and outside the network [2]. Intrusion detection systems (IDS) in general have two types: (1) misuse-based detection and (2) anomaly-based detection.

Misuse-based detection has the ability to detect the patterns of known attacks as the IDS database has a record of their signatures. The major drawback of the system that depends on pattern matching is the fact that they are incapable of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

detecting unknown intrusions as their signatures do not exist in the IDS database. However, an anomaly-based detection, use patterns of the normal behaviors in its database. This system monitors all network-wide actions and analyses them accurately. This technique identifies any deviation from normal patterns as an attack and the IDS generates an alarm warning to inform the network security manager about the newly detected attacks.

The main benefit of this system is the ability to detect unknown attacks [3]. There are several methods existing that could be incorporated into IDS to make it more accurate, classification algorithms have been shown to be impressively effective in the detection of attacks. Data classification is a supervised machine learning (ML) technique [4]. Several ML techniques have been utilized for classification in anomaly-based detection that among them ANNs offer an appealing alternative method with the capability to implement an approach that learns from the data to understand the system users. Machine learning models follow the function that learned from the data, but at some point, it still needs some guidance. For instance, if machine learning algorithm provides an imprecise result or forecast, an engineer needed to intervene and makes the necessary adjustments, while in ANN models, the algorithms have the capability to arbitrarily decide if the forecasts/results are accurate or not [5]. ANNs need an activation function to deal with nonlinear phenomena and raw data needs to be clearly converted into a feature vector platform through a user-specified feature map. However, this approach is only useful if the available parametric function catalog fits the data nicely. One of the efficient methods to overcome this problem is the kernel-based method. A kernel-based approach converts data in a non-linear way into an abstract space with no computation of the data coordinates in that space [6]. In this article, we aim to propose a new anomaly intrusion detection method to improve the classification accuracy using evolutionary kernel neural network random weights.
The rest of this article is organized as follows. Related works are introduced in Section 2, followed by the details of the proposed method in Section 3. Training of enhanced artificial neural networks in section 4. Validation and the dataset used is described in section 5. The experimental setup and results are explained in Section 6 also conclusion and future work in Section 7.

2. LITERATURE REVIEW

This section provides a brief discussion of researches related to the kernel-used approach in machine learning, followed by a presentation of some recent researches in relation to ENN.

2.1 Machine Learning (ML) and Kernel

Kernel-based clustering, regression and classification have attached much attention. Kernel mapping, for example, can be employed for the non-linear transformation of the description of the problem to a new feature space and in the process adapting traditional classifier paradigms, such as:
"Kernel Principal Component Analysis" (Kernel PCA) [7], "Fisher Discriminant Analysis" (FDA) [8], "kernel K-means clustering " [9] , the "Nearest-Neighbor classifier" [10] , "Quadratic Gaussian-based Bayes classifier" and "Least Mean Square Linear Dichotomizer" [11] or high-dimensional pattern visualization by the "Sammon map" [12]. Many machine learning tools use kernel tricks to improve the efficiency of detecting intrusion. The discussion that follows presents some previous research in this area.

[8] proposed an Intrusion detection system using machine learning, combining cost-functions with Support Vector Machine (SVM) and laminated sampling by weighted SVM works when dealing with a subset of training data containing more samples than others in the same training set. In this approach, the Gaussian Radial Basis Function (RBF) kernel used produced a superior result. The efficiency of an IDS enables the maintenance of higher accuracy and improves the detection of the most threatening intrusion categories. This model has shown excellent performance in small classes such as "User-To-Remote" and "Remote-To-Local" attacks with the well-known dataset NSL-KDD.

[9] proposed a new multi-class SVM technique. "Multi-lined Dimensionality Reduction" (ML-DR) is suggested as a technique for extracting train-time characteristics, while Multi-class SVM (M-SVM) is used to detect multi-attack

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

classes. The "Radial Basis Function Kernel" is employed in this method. False alarm rate and correlation coefficient are evaluated to measure performance metrics such as classification accuracy. The NSL-KDD dataset evaluated the proposed approach and experimental results presented more accurate compared to other methods of detection.

[10] used a direct SVM kernel method and achieved highly accurate results, high detection rates, and low false positives. Kernel's direct approach is described as "the arithmetical and the statistical formulation which can calculate the score or weights of the features." The current research uses KDDCup99, NSL-KDD, and Kyoto 2006 + datasets to evaluate projected system performance. Deleting duplicate methods is a good concept for optimizing datasets that reduce the learning machine's small size.

[11] suggested anomaly-based intrusion detection. New network threats can be identified through the anomaly-based detection. This research suggests that "Real-time Big Data Stream Processing Framework," and "Apache Storm," should be employed for the implementation of network IDSs. "Apache Storm" on the other hand can facilitate the management of generated enormously fast and sizeable network traffic which increases continuously. "Radial Basis Function" (non-linear kernel function) is employed in this study for transformation into higher dimensional space. This research used the 1999 dataset of Knowledge Discovery and Data Mining (KDD CUP 99) to investigate and assess the suggested solution.

[12] proposed a new SVM and "Gaussian Process Latent Variable Model" (GPLVM). GPLVM used to reduce the size of network connection records and "Radial Basis Function" (RBF) kernel for transferring the space computation feature. After dimensional reduction steps, the samples are classified using SVM algorithms to distinguish normal traffic from intrusions. The efficiency of the suggested method in this research is shown in the KDD Cup'99 dataset experiment. This research conducted a comparative analysis with the GPLVM, and ANN based intrusion detection system.

[13] presented a novel competitive method to NN intrusion detection. This research compares the performance of this approach to the popular uncontrolled intrusion detection algorithm of the self-organizing map using different types of kernel functions such as inverse distance, the triangular kernel, the quadratic kernel, and the Gaussian kernel. The proposed approach used only one-fourth of the calculation time. Additionally, clustering results vary with the initial numbers of neurons. It also detects well-known attacks on the network. Experimental results showed that when using the Gaussian kernel, the proposed approach is more accurate and efficient.

[14] used the combination of SVM and ANN for both anomaly and misuse detection. The proposed method recommends a process of new2-stage hybrid classification. The fundamental idea is to combine the advantages of each method with a low likelihood
of false positives to improve the precise classification. The first stage detects anomalous intrusion. The second stage (misuse) further investigates and classifies the type of attack into four classes. Simulation results show that the proposed algorithm exceeds conventional models, including individual SVM with "Radial-basis Kernel Function" and ANN algorithms. Empirical results show that the proposed system reliably detects abnormal activity through network. Both simulation results are used NSL-KDD, an enhanced dataset of KDD CUP 99.

To the best of our knowledge, in respect of anomaly IDS, no attempts have been made to use kernel mapping based on the structure of the feed-forward overall purpose function approximator (commonly referred to as the "Multilayer Perceptron" which is able to make an explicit presentation of the knowledge in weights. Thus, particular emphasis is placed in this study on the kernel multilayer perceptron algorithm for classification.

2.2 Evolutionary Neural Network (ENN)

[15] addressed feed-forward NN training issues employing a recent metaheuristic "Locust Swarm Optimization" (LSO) algorithm to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

enhance an advanced detection system. LSO enables ENN to vigorously develop its weights and adapt its factors to avoid being trap in local minima and to circumvent under- fitting or over-fitting. Tests carried out have demonstrated that the suggested method is efficient in training FNN compared to more commonly known training approaches, like PSO and GA. NSL-KDD and UNSW-NB15 datasets have been used for evaluating the success of the proposed approach. The proposed approach allows alternative IDS solutions and increases the rate of detection.

[16] proposed a novel training algorithm with its basis in the newly suggested whale optimization algorithm (WOA). This algorithm has been shown to be capable of solving numerous optimization issues and outperforming the existing algorithms. The high local optima avoidance and fast convergence speed were the major inspirations for the application of the WOA to the challenge of training MLPs. The WOA was employed to determine the best values for weights and biases to minimalize the MSE. The results indicated that the suggested WOA-based training algorithm was capable of outperforming the existing algorithms on most of the datasets. The outcomes were superior with greater accuracy and improved convergence.

[17] suggested a novel IDS with flexibility in selecting groups of attributes employing ANN. The selection procedure was based on a contribution criterion that the researchers had defined as "a function of precision measures of Heuristics for Variable Selection" (HSV).

The NSL-KDD data- set was employed in training, testing and assessment of their scheme. A comparison was made with other works. The suggested IDS scheme indicated satisfactory results.

[18] discussed a novel NIDS model developed on the basis of the BP NN that is an enhanced "Dempster–Shafer" (D–S) theory. First, the primary characteristics are drawn into several diverse characteristic subsets. Second, various ANN models are trained based on various characteristics of subsets. Lastly, the various ANN results are combined employing the D–S evidence theory to prepare the ultimate outcomes. Empirical outcomes with the Kddcup99 dataset attained acceptable detection rate and low false-positive rate.

[19] designed an IDS model employing the enhanced PSO-BP NN. Their study exploited the benefits of local search competence of BP NN and the universal search capacity of PSO. KDD Cup 99 was used for evaluating the efficiency of proposed IDS model.

3. PROPOSED METHOD

The proposed anomaly-based detection can be separated into two main modules, namely, Kernel Neural Network Random Weights (KNNRW) and Multiverse Optimizer (MVO) module, which are described below:

3. 1 Artificial Neural Network-Multilayer Perceptron (ANN-MLP)

The net a d-dimensional consider as input to pattern x from an input domain X which is typically the Euclidean vector space. There are H neurons in the hidden layer of the net, and all compute the function θ: X → R which is the inner product $\langle w_h|x\rangle = w_h \cdot x = w_h^T x$ of a neuron-specific d-dimensional weight vector w_h and the

$$\theta_h = z(\langle w_h|x\rangle) \ , \mathrm{h} = 1, \ldots, \mathrm{H} \qquad (1)$$

input x.

Every H input to the hidden mappings θ_h of the hidden layer are combined into an H-dimensional feature vector $\theta = [\theta_1 \ldots \theta_H]^T$ which comprises a novel pattern in the domain.
The feature vector θ defines a map based on calculus of the hidden layer.

$$x \to \theta(x) = \left[z(\langle w_{h_1}|x\rangle) \ldots z(\langle w_{h_H}|x\rangle)\right]^T \qquad (2)$$

$$y_i = z(\langle w_h|x\rangle) \ , \ \mathrm{i} = 1, \ldots, \mathrm{c} \qquad (3)$$

The identical functional mapping from the input layer to the hidden layer is replicated, then the final mapping from the hidden layer to the output layer compute for all output neurons of the function.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

where: for any of the c output units w_i are explicitly H-dimensional weight vectors. The output integrates into a c-dimensional feature vector $y = [y_1 \dots y_c]^T$ which comprises a novel pattern in the eventual output domain Y. The calculus of the output layer feature vector y thus

$$\theta \to y(x) = \left[z(\langle w_{i_1}|x\rangle) \dots z(\langle w_{i_c}|x\rangle)\right]^T \quad (4)$$

defines another map

Consequently, we note that the Multilayer Perceptron performs a mapping

$$x \to \theta(x) \to y(\theta(x)) \quad (5)$$

There is often an additional fixed input and a fixed hidden part is presented which enlarges the dimension of the input and weight vectors.

A Neural Network with no activation function would merely be a linear regression model, with power constraints and mostly underperforms. To address the nonlinear phenomena, a range of nonlinear models has been proposed. Several activation function models assume that data follow some parametric class of nonlinear function then fine-tunes the shape of the parametric function to fit observed data. There are several activation functions in ANN such as [20]:

- Linear: This activation function is essentially the identity function, which in practice means that the signal stays unchanged.
- Sigmoid: it permits a decrease an extreme or typical value invalid data but does not remove them. Instead, it transforms autonomous variables of nearly unlimited range into simple likelihoods between 0 and 1. Sigmoid has slow convergence.
- Tanh: Tanh is defined as "the association between the hyperbolic cosine and hyperbolic sine: tanh(x) equal to sinh(x) divided by cosh(x)." The range of tanh is between -1 and 1 that is different from the sigmoid function. The benefit of tanh is that conveniently managed negative numbers, but it suffers from a vanishing gradient problem.
- Softmax: to generalize the logistic regression, rather than classifying in binary it can have limits.
- ReLU: Linear unit rectified by an activation function. (ReLU) is an actual intriguing change which triggers a single node if the input exceeds a specified threshold. it should only be employed within hidden layers.

However, activation functions are real significance for an ANN to learn and understand a complex matter and the non-linear complicated functional mappings between the inputs and response variable it only beneficial if data are fit nicely by the available catalog of parametric functions.

There is also an approach called kernel method which does the non-linear transformation to the input and capable ANN to learn and perform more complex tasks with no requirement to compute the coordinates of the data in that space.
Kernel-based methods function to transform data into an abstract space that calculates distances between observations based on new values or classes. Kernel methods are defined as "a class of machine learning pattern analysis algorithms that avoid explicit mapping needed to obtain linear learning algorithms to learn a decision boundary or nonlinear function, and also provide a convenient method for obtaining high-dimensional data mapping features without computing nonlinear transformation." A user-specified feature map must explicitly transform raw data representation into feature vector representations in ANN. While kernel methods need just a user-specified kernel similarity to raw data point representation pairs [21]. In other words, non-linear activation applies to weighted inputs in ANNs, whereas the nonlinear feature map applies directly to the input instance of kernel-based approaches, and a weighted sample sum estimates the target value.

3. 2 Kernel Neural Network Random Weights (KNNRW)

One of the main parts of ANN-MLP is the designation of the neuron's weight to calculate the output of the system. Weighing in this network is performed in a variety of ways and in most cases it

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

is random. On the other hand, this randomness cannot be ascertained in reaching the answer, and it should be replaced by a method in which the best values for the weight can be found. For this purpose, many methods have been designed. One of the efficient techniques is Neural Network Random Weights (NNRW) algorithm [22]. The basic intuition behind initializing weight layers into small (and different) values is just so that the bias of the system is broken, and weight values can move along and away to different values. More precisely, when the initial weights are to be distinct and have "a small gap" between them, this 'gap' expands out and forces the weights to be a bit larger at every iteration, and this helps the network to converge faster and the learning process speeds up.

NNRW is defined as "a non-iterative training algorithm in which the weights randomly selected between the hidden layer and input layer and analytically obtained weights between the output layer and hidden layer". In NNRW, except the linear model, to selected values of weights randomly also help to determine the optimal value of weights. NNRW possesses significantly reduced training complexity compared with the conventional training of feed-forward neural networks. NNRW is a simple structural system with three layers: the input, output, and the hidden layers, which contains numerous nonlinear processing nodes. The mathematical equation used in this model is as follows:

$$o_k = w^T g\,(w_{in}.\,x_k + b)\,, k \;\; = 1,2,3,\dots,N \qquad (6)$$

Based on the Equation (6), w_{in} is the weights of the input layer and w is a random weight between middle and output layer; g is an activation function that can be slightly linear parameters combination. N is the number of test data used to test the network. x represents input data for testing and o indicates the output of the network.

The NNRW training process is a linear regression process and also uses random weights in this network which can effectively affect the limits of traditional ANNs, but this algorithm still can be improved in terms of performance and generalization. For this purpose, the Kernel Neural Network Random Weights (KNNRW) algorithm presented completes form of NNRW algorithm. Based on the KNNRW algorithm the matrix value of the weights is considered as the input of the system and displayed by H value which ultimately changes these weights and creates the new one.
On the basis of the foregoing equations, the values of C and α are the random values between [-1,1] and I is an identity matrix. To calculate f(x) the following equations are used:

$$\left(\frac{I}{C} + HH^T\right)\alpha = T \qquad (7)$$

$$w = H^T\left(\frac{I}{C} + HH^T\right)^{-1} T \qquad (8)$$

$$f(x) = h(x)H^T\left(\frac{I}{C} + HH^T\right)^{-1} T \qquad (9)$$

where the definition of the kernel matrix of KNNRW is as follows:

$$\Omega_{NNRW} = HH^T :$$
$$\Omega_{NNRW_{ij}} = h(x_i)\,.\,h(x_j) = K(x_i, x_j) \qquad (10)$$

As a result, the output function can be rewritten as:

$$f(x) = \begin{bmatrix} K(x, x_1) \\ . \\ . \\ . \\ K(x, x_N) \end{bmatrix}^T \left(\frac{I}{C} + \Omega_{NNRW}\right)^{-1} T \qquad (11)$$

According to the above calculations, not only kernel is the main component of the structure, but it is also a value of the variable in the final Equation f(x).

Kernel by solving the challengeof transforming the non-linear model into a linear model can facilitate weight adjustment and error reduction. In the kernel implementation of NNRW, the parallel kernel function $K = (u, v)$ typically should be provided. KNNRW has global approximation capability:
According to theorem [23],Universal Approximation Capability: an extensive kind of the hidden node mapping $h(x)$ can be utilized in NNRW to estimate any uninterrupted target functions. With this universal estimation ability, KNNRW can utilize an inclusive range of feature mappings, such as Sigmoid, Radial Basis Function (RBF), trigonometric, and so on. Many problems use the RBF kernel to perform complex

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

nonlinear mapping that allows a robust learning mechanism [24]. RBF kernel function employs the Euclidean distance in the original space to locate the correlation in the augmented space and compare it to other ANN paradigm-based algorithms. It has the advantage of simple computing network parameters. Similar to existing approaches it only focuses either on mapping representations from one domain to the another, or on learning to extract features that are invariant to the domain from which they were extracted.

Moreover, if the focus is confined to only the creation of a mapping or shared representation between the two domains, it would be ignoring each domain's individual characteristics. To get a better result in this research, the linear combination of multiple RBF kernels was used: By applying the linear combination of multiple RBF kernels in terms of kernel function the output function can be rewritten as:

$$f(x) = \sum_n \eta_n exp\left(-\frac{1}{2}\frac{\|x_i - x_j\|^2}{\sigma^2}\right)\left(\frac{I}{c} + \Omega_{NNRW}\right)^{-1} T \quad (12)$$

In above Equation σ^2 is the standard deviation and η_n is the weight for n^{th} RBF kernel and n represent the number of neurons. Using this kernel guarantees the linear combination remains its characteristics while also using standard deviation to compute the kernel can increase the accuracy of this function. The steps of the proposed algorithm are presented in Figure 1.

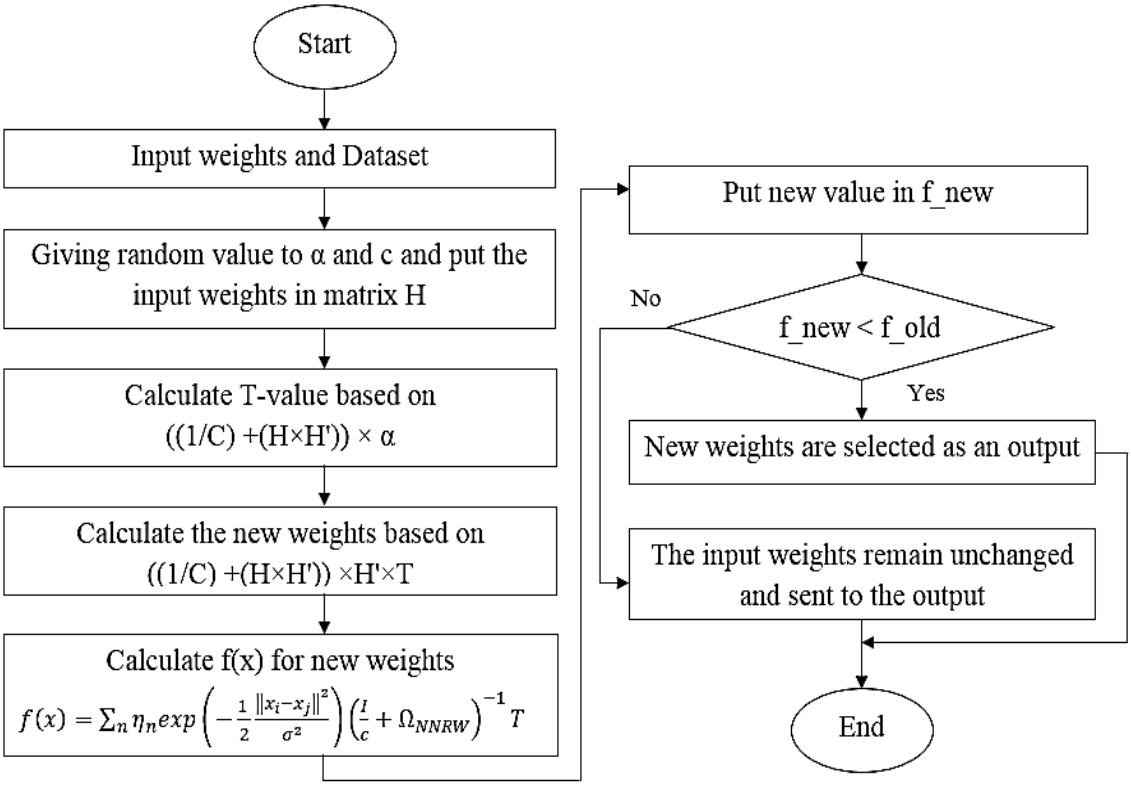

Figure 1: The steps of the proposed algorithm (KNNRW)

Given that there are some drawbacks in implementing the BPANN systems in this research to training the ANN-MLP instead of the BP algorithm the evolutionary algorithm is known as Multiverse Optimizer (MVO) was used.

3. 3 Multiverse Optimization (MVO)

The basis of the MVO algorithm is the three cosmological perceptions: white holes, which are not visible in the universe; black holes, which are visible, and wormholes, which are interconnected parts of the universe. These three

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

concepts constitute the major inspiration for the MVO algorithm, which provides the simulation of the dynamics and universes interaction via black holes, white holes, and wormholes [28].

In MVO, solution for the ultimate optimal universe, available in the corresponding to an object in the universe, the inflation rate is corresponding to the fitness while the term "time" parallels the iteration. The rules below are applicable to all the universes in the course of the optimization process:

- A higher inflation rate has greater probability of an existing white hole.
- A lower inflation rate has greater probability an existing black hole.
- The best universe is the result of objects randomly moving through wormholes.

When objects between universes are exchanged, a universe possessing an object sending to other universes from higher inflation rate to the lower inflation rate. Moreover, a universe possessing a lower inflation rate receives additional objects from better universes to attain status stability and eventually become the optimal universe with an enhanced inflation rate. This exchange activity guarantees the enhancement of average inflation rates for all the universes during the time conducted. In the process of optimization, the universes are aligned according to their inflation rates, and the selected universe is determined, employing the roulette wheel, to be the white hole. The travelling distance rate (TDR) and the wormhole existence probability (WEP) are the two major coefficients; lb_jrepresent the lowest j^{th} variable, ub_j denotes the highest j^{th} variable [25]. The formulas for the two coefficients are given by 13 and 14 respectively:

$$TDR = 1 - \left(\frac{l^{\frac{1}{p}}}{L^{\frac{1}{p}}}\right) \qquad (13)$$

$$WEP = min - l * \left(\frac{min - max}{L}\right) \qquad (14)$$

where: p (=6) is the accurateness of exploring the iterations; where l is the existing iteration, and L indicates the highest number of iterations. WEP and TDR are elevated at all iterations to attain greater accuracy around the best-gained universe in exploring/local search. The standard steps of the MVO algorithm are shown in Figure 2:

The optimization process commences with the creation and initialization of the factors, such as population size and the upper and lower bounds. At that time initialize a set of universes

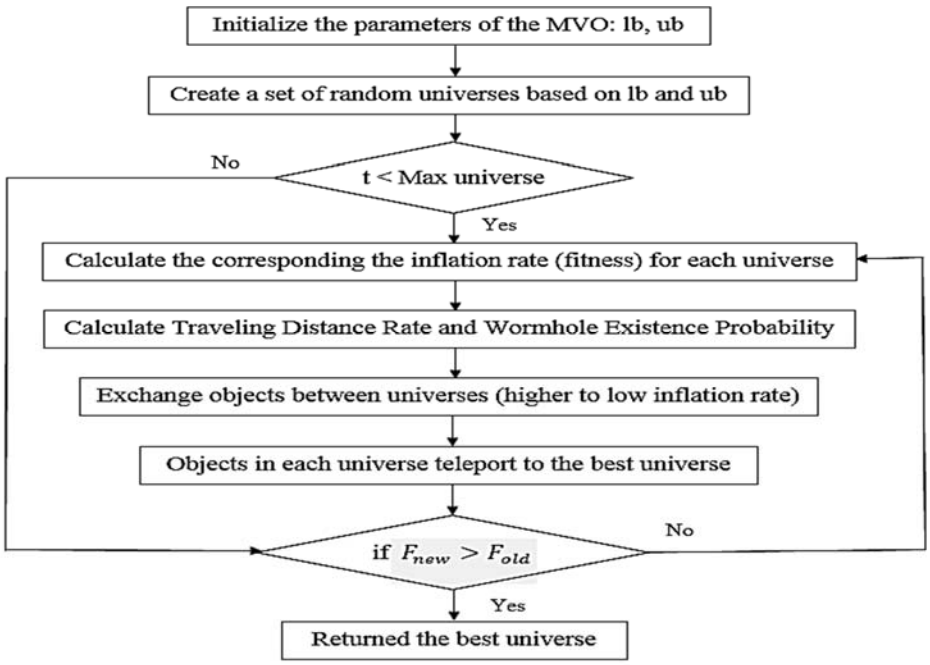

Figure 2: The standard steps of the MVO

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

randomly on the basis of the upper and lower bounds. The corresponding inflation rate (fitness value) is computed for individual universes to describe the optimal potential inflation rate. Then, at individual iterations, high inflation rates objects in the universes tend to migrate to the universes containing low inflation rates through the white or black hole. At the same time, objects in individual universes move randomly to the optimal universe via wormholes. Ultimately, the optima universe is formed at the completion of the operation.

4. TRAINING OF KNNRW WITH THE MVO ALGORITHM

One of the important steps is the training process. The aims of this process are to locate the synaptic weights of the KNNRW and to decrease the MSE to achieve the highest classification and prediction accuracy. The MVO algorithm commences the optimization process with the generation of a list of potential solutions and makes the assumption that individual universes are specific to the solution population, which is created at random, and the size of the solution denotes the magnitude of the challenge.

The number of objects in each individual population is computed as presented in Equation 15:

$$Indv_{nbr} = (n * m) + (2 * m) + 1 \qquad (15)$$

$$MSE = \left(\frac{1}{T_n}\right) * \sum_{i=1}^{T_n} (output - input)^2 \qquad (16)$$

In the context of this research, MSE is utilized as the main cost function of the suggested MVO training algorithm. The training objective is the minimization of the MSE to the highest number of generations to be attained. The MSE can be computed using Equation 16:

There is an actual data as an input and estimated values as an output and $T_n$is the number of Firs the universes values and dataset are received from the input, then in initiate the neurons and define the minimum and maximum of weights. Afterward, the value of α , c is selected between the minimum and maximum values and the new instances in the dataset. The general steps of the MVO-KNNRW training is shown in Figure 3.

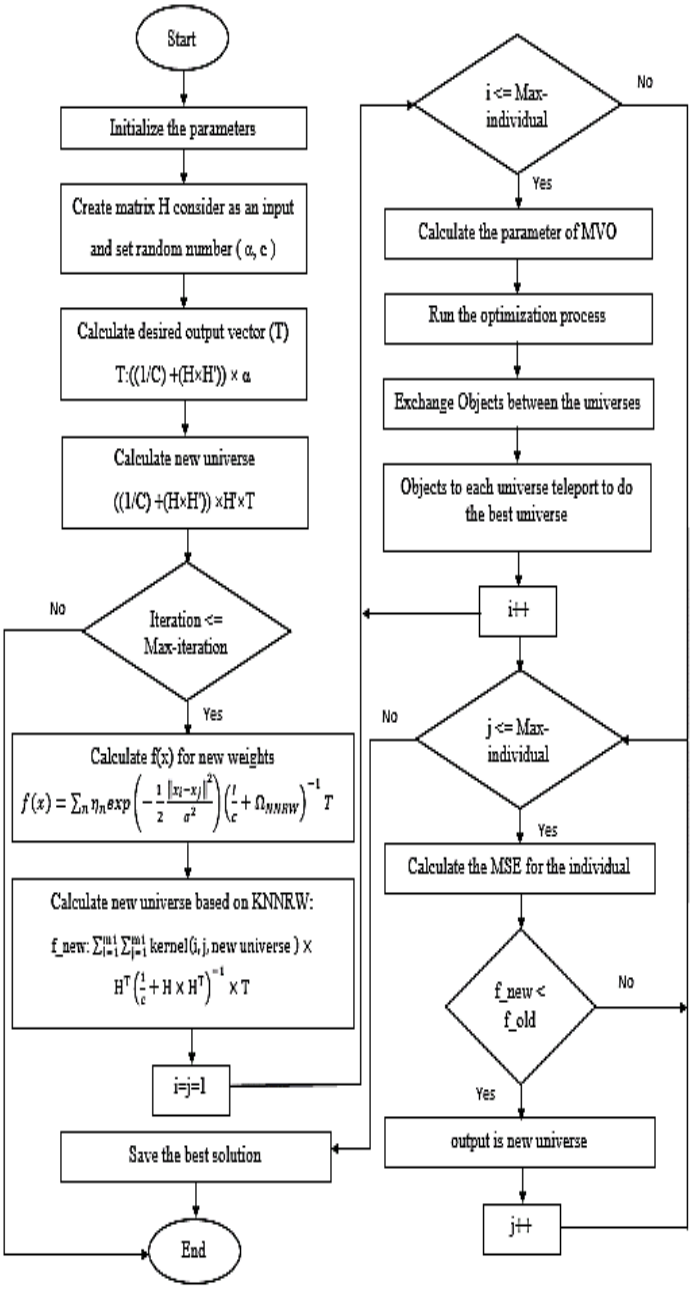

Figure 3: The general steps of the MVO-KNNRW training

universe is calculated based on KNNRW equations. Later the value of function f(x) is calculated based on the new universe and kernel. The f(x) value is calculated based on the initial

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

universe in the input. Eventually, choose a better value of weights using new f(x) and old f(x).

According to the general steps, there are stages of adjusting the weights based on the proposed algorithm. The weighting matrix is firstly taken from the input and is considered as the value of H. After that, based on the KNNRW mathematical Equation calculate new values for input weights. Finally, after comparison the best result is sent to the output.

5. VALIDATION AND EXPERIMENTAL RESULTS

This section offers the findings and discusses the performances of the suggested anomaly- intrusion detection method. One of the successes of any IDS is a complex issue because of its non-linearity and the quantitative or qualitative network traffic data stream with several features. Consequently, to address this issue, in this research, the evolutionary kernel neural network random weights have been proposed to offer possible solutions for improving the classification accuracy of anomaly-based detection. The proposed model was applied and assessed in MATLAB 2017a on a PC with Core I5 3.20 GHz CPU and 8 GB RAM. The standard factor settings were utilized for the multiverse optimization, minimum WEP=0.2; maximum WEP=0.6; number of individuals=4; and number of iterations=100.The efficiency of the proposed method (EKNNRW) for anomaly-based detection was assessed, and a comparison was made of its effectiveness with other existing IDS approaches. For the validation NSL-KDD dataset used. This dataset is specifically the purpose of evaluating IDS offline.

5. 1 NSL-KDD Dataset

The KDD Cup 99 is a manipulated form of the DARPA 98 dataset and is currently the furthermost extensively used benchmark dataset for evaluating IDSs. In the training dataset, there is nearly 5 million overall number of connection records, but they suffer from some unnecessary elements in the testing and training sets. These redundant elements have a significant effect on the performance and lead to an unsatisfactory assessment of the IDSs [30]. Hence, the NSL-KDD dataset was structured to improve the KDD Cup 99 dataset and address the characteristic issues of the last mentioned. This dataset is derived from the various portions of the original KDD Cup 99 dataset, with no unnecessary elements and repetitions. Additionally, the issue of unbalanced dissemination in the testing and training set was resolved, to enhance the precision of the IDS assessment. The NSL-KDD dataset comprises 41 features, labeled as “normal connections” or “attack types.” The NSL-KDD dataset has four attack classes: DoS, U2R, R2L, and probing [31] as shown in Table 1.

Table 1: The attack classes in the NSL-KDD dataset

Attack Class	Attack Type
DOS	Back, Land, Neptune, Pod, Smurf, Teardrop, Apache, Udpstorm, Processtable, Worm
Probing	Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint
R2L	Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named
U2R	Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps

5. 2 Results

Accuracy has a significant role in an IDS, and the effectiveness of an IDS is dependent on robust classification algorithms to accurately identify cyber-attacks against assets on cyber-enabled infrastructures. Accuracy is calculated according to the major elements: TP, FP, TN and FN, which are in the confusion matrix (CM).
The CM denotes the outcomes of the organization with a dimension of NN [26]. The shortened forms of the confusion matrix present in Table 2.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: The shortened form of confusion matrix

TP: Number of connections successfully classified as anomalies by the classifier.
FN: Number of anomalous connections that are misclassified as normal by the classifier.
FP: Number of normal/non-intrusive connections that are misclassified as intrusive as anomalies by the classifier.
TN: Number of normal/non-intrusive connections that are successfully classified as normal by the classifier.

Considering the importance of kernels in improving classification accuracy, the performance test result is measured in terms of accuracy with the help of two experiments. Experiment I, MVO-ANN without using kernel and random weights while experiment II used MVO-ANN with kernel and random weights that called EKNNRW.

Classification rate or Accuracy (ACC): It is characterized as "the ratio of a classified number of correct instances and total instances" as presented in Equation (17).

$$ACC = \frac{Correctly\ classified\ instances}{Total\ number\ of\ instances}$$

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \quad (17)$$

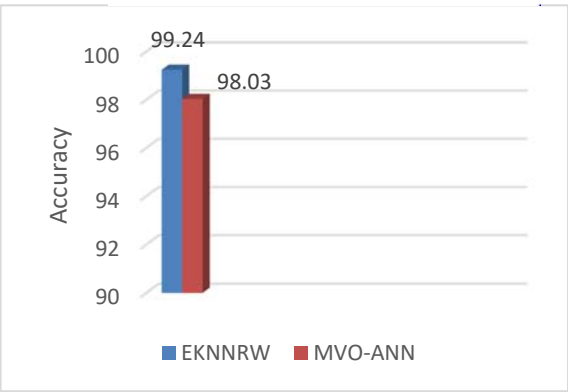

Figure 4: Comparison between the proposed method (EKNNRW) and (MVO-ANN) in terms of accuracy

According to the results obtained from the accuracy, it is obvious that the proposed method (EKNNRW) using kernel and random weights has better performs than the previous method (MVO-ANN).

The loss function is a crucial element in ANNs, which is employed to calculate the irregularities between predicted and actual values. It is a non-negative value, where the strength of the model rises together with the lowering of the loss function value. Mean Squared Error (MSE), or quadratic, loss function is extensively applied in linear regression as the indicator of performance. Figure 5 shows the evolution curve of the MSE function of the proposed model (EKNNRW) for the NSL-KDD dataset within100 iterations.

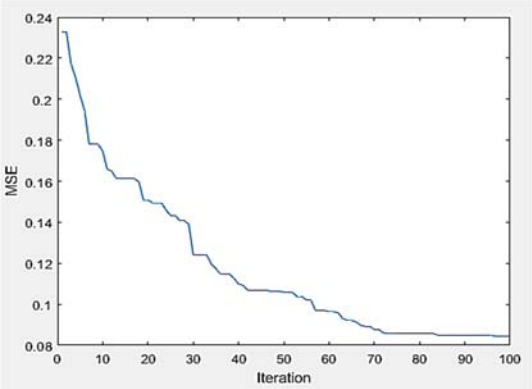

Figure 5: MSE convergence curve of EKNNRW

Furthermore, an ideal IDS has a high attack detection rate with a low false-positive rate, although it is difficult to attain. Detection of illegal behaviors is the major objective of IDS. Considering that being accurate is defined as "the amount of data correctly classified that is known as true positive (TP) and true negative (TN) which effects the performance of an IDS." To calculate the detection rate and false alarm rate the following Equations were used.

Detection rate (DR): It is defined as "The ratio of correctly detected attacks to the total number of attacks."

$$DR = \frac{Correctly\ detected\ attacks}{Total\ number\ of\ attacks}$$

$$DR = \frac{TP}{TP+FN} \quad (18)$$

False positive rate (FPR): is defined as "the ratio of the number of normal instances detected as attack to the overall number of normal instances."

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$FPR = \frac{Number\ of\ normal\ instances\ detected\ as\ attacks}{Total\ number\ of\ normal\ instances}$$

$$FPR = \frac{FP}{FP+TN} \quad (19)$$

Figure 6 and 7 are shown the comparison test results of detection rate and false alarm rate for the proposed method (EKNNRW) and (MVO-ANN).

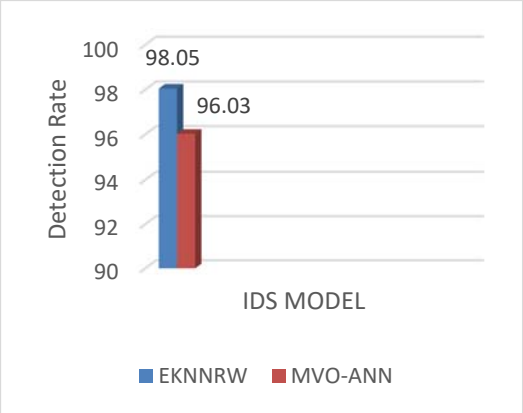

Figure 6: Comparison between the proposed method (EKNNRW) and (MVO-ANN) in terms of detection rate

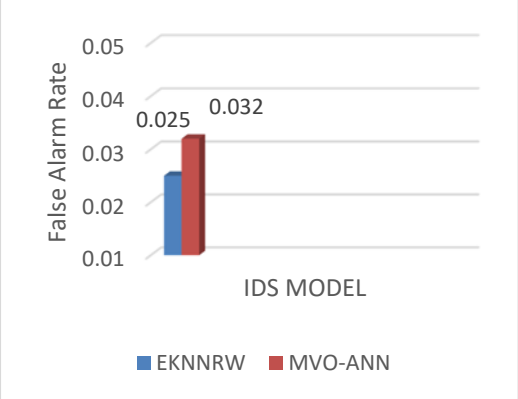

Figure 7: Comparison between the proposed method (EKNNRW) and MVO-ANN in terms of false alarm rate

Comparing the performance outcomes of the proposed model (EKNNRW) with some well-known classification methods such as Logistic Regression (LR), Naive Bayes (NB), Artificial Neural Network-Multilayer Perceptron (ANN-MLP), Multiverse Optimizer-Artificial Neural Network (MVO-ANN) showed that proposed model outperforms previous works in terms of accuracy of detection and data correctly classified by the proposed model as shown in Figure 8.

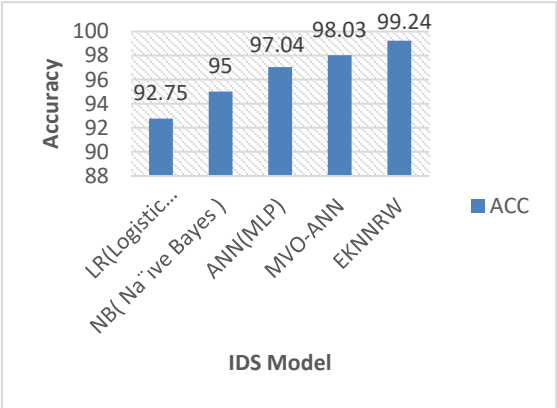

Figure 8: Comparison of accuracy between proposed method (EKNNRW) and previous works

Due to the structure of the ANN, there are random values with different starting points per each simulation during the training phase that cause different results for each running time. There is no rule of thumbs for the number of running code to get an acceptable result. For this reason, by referring to the previous researches in this area the code was running for 30 times to ensure that it provides accurate results [27]. To get the average result, the mean calculation was used. The mean is the most commonly used mathematical measure of average. where $\{x_1, x_2, x_3, \ldots, x_N\}$ are the observed values of the sample items $\bar{x}$ is the mean value and N is the number of observations in the sample. The following Equation shows the calculation of mean value:

$$\text{Mean value} = \frac{\sum x_i}{N} \quad (20)$$

The following results shown in Table 3 are the performance average results of EKNNRW achieved by calculating the mean value for ACC, DR and FAR.

Table 3: The average results of proposed method (EKNNRW)

Results	Average ACC	Average DR	Average FAR
Average EKNNRW	98.90	97.35	0.029

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6. CONCLUSION

In order to have a robust and efficient IDS using a multilayer perceptron neural network, in this research we proposed an anomaly-based detection system with important parts which are: kernel, random weights, and optimization algorithm. Considering the importance of the kernel in improving the classification accuracy linear combination of multiple RBF kernel was used to guarantee that the linear combination remains characteristic, while random weights technique is also used to reduce complexity, besides employing multiverse optimizer to improve the training part in MLP. Results show that the accuracy of the suggested method (EKNNRW) compared with some existing works, is higher and the data accurately classified by the suggested model exceed those by the other models. Furthermore, the experimental results based on two performance metrics, detection rate and false alarm rate, show that proposed method using kernel and random weights achieved lower false positive rate and higher detection rate rather than (MVO-ANN). Given that kernel-based algorithms have the ability to extract domain invariant features and are also able to solve tasks in classification, improve the classification accuracy for each type of attack will be the subject of further research.

REFERENCES:

[1] T. Hamed, J. B. Ernst, and S. C. Kremer, "A Survey and Taxonomy of Classifiers of Intrusion Detection Systems," 2018.

[2] V. Pandu, J. Mohan, and T. S. P. Kumar, "Network Intrusion Detection and Prevention Systems for Attacks in IoT Systems," no. January, pp. 128–141, 2019.

[3] V. Hajisalem and S. Babaie, "A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection," Comput. Networks, vol. 136, pp. 37–50, 2018.

[4] K. Peng, V. C. M. Leung, L. Zheng, S. Wang, C. Huang, and T. Lin, "Intrusion detection system based on decision tree over big data in fog environment," Wirel. Commun. Mob. Comput., vol. 2018, 2018.

[5] L. P. Dias, J. J. F. Cerqueira, and K. D. R. Assis, "Using Artificial Neural Network in Intrusion Detection Systems to Computer Networks," pp. 145–150, 2017.

[6] S. Wang, Y. Jiang, F. Chung, and P. Qian, "Feedforward kernel neural networks, generalized least learning machine, and its deep learning with application to image classification," Appl. Soft Comput. J., pp. 1–17, 2015.

[7] Q. Wang, "Kernel Principal Component Analysis and its Applications in Face Recognition and Active Shape Models," no. July 2012, 2012.

[8] A. Alabdallah and M. Awad, "Using weighted support vector machine to address the imbalanced classes problem of intrusion detection system," KSII Trans. Internet Inf. Syst., vol. 12, no. 10, pp. 5143–5158, 2018.

[9] B. N. Kumar, M. S. V. S. B. Raju, and B. V. Vardhan, "Enhancing the performance of an intrusion detection system through multi-linear dimensionality reduction and Multi-class SVM," Int. J. Intell. Eng. Syst., vol. 11, no. 1, pp. 181–192, 2018.

[10] A. G. Gedam and S. G. Shikalpure, "Direct kernel method for machine learning with support vector machine," 2017 Int. Conf. Intell. Comput. Instrum. Control Technol. ICICICT 2017, vol. 2018-Janua, no. 12, pp. 1772–1775, 2018.

[11] M. A. Manzoor and Y. Morgan, "Network Intrusion Detection System using Apache Storm,"

Adv. Sci. Technol. Eng. Syst. J., vol. 2, no. 3, pp. 812–818, 2017.

[12] H. Moudni, M. Er-rouidi, H. Faouzi, H. Mouncif, B. El Hadadi, and F. Polydisciplinary, "Anomaly Traffic Detection Based on GPLVM and," pp. 2–6, 2016.

[13] K. Karthikeyan and R. Mala, "Artificial Neural Content Techniques for Enhanced Intrusion Detection and Prevention System," Int. J. Innov. Res. Adv. Eng., vol. 3, no. 04, pp. 96–104, 2016.

[14] J. Hussain, S. Lalmuanawma, and L. Chhakchhuak, "A two-stage hybrid classification technique for network intrusion detection system," Int. J. Comput. Intell. Syst., vol. 9, no. 5, pp. 863–875, 2016.

[15] I. Benmessahel, K. Xie, and M. Chellal, "A new evolutionary neural networks based on intrusion detection systems using locust swarm optimization," Appl. Intell., vol. 48, no. 8, pp. 2315–2327, 2019.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[16] I. Aljarah, H. Faris, and S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm," Soft Comput., vol. 22, no. 1, pp. 1–15, 2018.

[17] B. H. L. Djionang and G. Tindo, "A New Networks Intrusion Detection Architecture based on Neural Networks," vol. 17, no. 1, 2017.

[18] H. Hu and W. He, "Research on key technology of network intrusion detection system based on improved GA-BPNN algorithm," Chem. Eng. Trans., vol. 51, no. 2010, pp. 391–396, 2016.

[19] C. Qiu and J. Shan, "Research on intrusion detection algorithm based on BP neural network," Int. J. Secur. its Appl., vol. 9, no. 4, pp. 247–258, 2015.

[20] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning," pp. 1–20, 2018.

[21] Q. Ye, "Kernel-based Approximation Methods for Generalized Interpolations: A Deterministic or Stochastic Problem?," pp. 1–31, 2017.

[22] W. F. Schmidt, M. A. Kraaijveld, and R. P. W. Duin, "Feed forward neural networks with random weights," Proc. - Int. Conf. Pattern Recognit., vol. 2, no. July, pp. 1–4, 1992.

[23] G. Bin Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," IEEE Trans. Syst. Man, Cybern. Part B Cybern., vol. 42, no. 2, pp. 513–529, 2012.

[24] K. Lang, M. Zhang, and Y. Yuan, "Improved Neural Networks with Random Weights for Short-Term Load Forecasting," pp. 1–14, 2015.

[25] I. Benmessahel, K. Xie, and M. Chellal, "A new evolutionary neural networks based on intrusion detection systems using multiverse optimization," Appl. Intell., 2017.

[26] G. Canbek, T. T. Temizel, S. Sagiroglu, and N. Baykal, "Binary classification performance measures/metrics: A comprehensive visualized roadmap to gain new insights," 2nd Int. Conf. Comput. Sci. Eng. UBMK 2017, no. October 2018, pp. 821–826, 2017.