2019

Intelligent intrusion detection using radial basis function neural network

Alia AbuGhazleh

Muder Almiani

Basel Magableh

See next page for additional authors

Authors
Alia AbuGhazleh, Muder Almiani, Basel Magableh, and Abdul Razaque

Intelligent Intrusion Detection Using Radial Basis Function Neural Network

Alia AbuGhazleh
Research Lead Engineer
IEEE Jordan Section
Amman, Jordan,
aabughazleh03@eng.just.edu.jo

Muder Almiani
College of Information Technology
Al-Hussein Bin Talal University
Ma'an, Jordan
malmiani@my.bridgeport.edu

Basel Magableh
School of Computer Science,
Technological University
Dublin, Ireland
basel.magableh@dit.ie

Abdul Razaque
Department of Computer Science
New York Institute of Technology
arazaque@nyit.edu

***Abstract*— Recently we witness a booming and ubiquity evolving of internet connectivity all over the world leading to dramatic amount of network activities and large amount of data and information transfer. Massive data transfer composes a fertile ground to hackers and intruders to launch cyber-attacks and various types of penetrations. As a consequence, researchers around the globe have devoted a large room for researches that can handle different types of attacks efficiently through building various types of intrusion detection systems capable to handle different types of attacks, known and unknown (novel) ones as well as have the capability to deal with large amount of traffic and data transferring. In this paper, we present an intelligent intrusion detection system based on radial basis function capable to handle all types of attacks and intrusions with high detection accuracy and precision through addressing the intrusion detection problem in the framework of interpolation and adaptive network theories.**

***Index Terms*— artificial neural network, data approximation, clustering, interpolation, intrusion detection, radial basis function.**

I. Introduction

Data encryption, user authentication and firewalls are most common classical regimes for computer network protection. However, as conducting network cyber activities is in increasing pattern with tendency to high levels of breaches and penetration, the classical protection techniques collapsed in front of these attacks. The situation gets harder if the target organizations and establishments are of highly sensitive data containers such as: banks, telecom, and military agents. Intelligent software intrusion detection systems represent the typical solution for cases cannot be handled by classical security techniques. This is due to the learning capability in one hand and due to adaptivity shown by these systems on the other hand. Most of intelligent intrusion detection systems are built using Artificial Intelligence (AI) such as swarm intelligence, case-based reasoning, neural networks and fuzzy logic, rule-based systems, cellular automata, reinforcement learning, multi-agent systems, and hybrid systems built by hybrid of two or more of these techniques. Artificial neural networks are used extensively to model the nonlinear mapping of intrusion detection problem. Radial Basis Function (RBF) neural network represents a typical type of powerful neural networks that show high capability in binary and class-wise classification problems including intrusion detection.

As an early implementations of basic RBF Network (RBFN) for sake of security and intrusion detection are that proposed by Yang *et al*. [1], Jing *et al*. [2] and Devaraju and Ramakrishnan [3]. Rapaka *et al*. [4] used the classical version of RBF network for combining misuse and anomaly detections. Exploiting the short training time acquired by RBFNN, Jiang *et al*. [5] proposed a hierarchical intrusion detection system composed of multiple layers of RBFN structured in parallel and serial manners for real-time implementation.

On the other hand, for optimized RBF mapping operation, Particle Swarm Optimization (PSO) was used by Chen *et al.* [6] to enhance RBFN parameters in sake of solving issues such as poor generalization and low detection sensitivity. Xu *et al*. [7] used hybrid of PSO technique and kernel principal component analysis to extract the core nonlinear characteristic of dataset whereas Other researchers [8] adapted genetic algorithm-chaos as RBFNN optimization technique.

Instead of using classical clustering techniques such as k-means and Self-Organized Map (SOM) to search for candidate RBFN hidden neurons , Zhong et al. [9] employed a multiple granularities immune neural network algorithm for constructing the hidden layer of RBFN whereas Yichun et al. [10] used immune recognition algorithm as RBFN learning algorithm. For detecting new attacks in continual manner, Tian *et al.* [11] integrate SOM network to generate new nodes in RBF network for on-line intrusion detection. Other intrusion detection structures were built based on a hybrid combination of RBFN and other techniques to enhance the overall performance of intrusion detection. Ma *et al*. [12] proposed a hybrid learning algorithm for RBFN based on combination of Quantum-Behaved Particle Swarm Optimization (QPSO) and gradient descent algorithm. As another example of hybrid techniques, a hybrid scheme was presented by Li-Zhong *et al*. [13] for intrusion detection composed of rough set and RBFN. Most of aforementioned literature addressed the problem of intrusion detection as a sort of nonlinear mapping problem. In our work, we addressed the intrusion detection from pure mathematical perspective leading to high detection performance without the need for complicated hybrid schemes and preliminary stages for RBF parameters optimization.

We first introduce the problem of intrusion detection in the framework of mathematical approximation (interpolation) and introduce our intrusion detection system in Section II. We show our results associated with comparisons and discussion in Section III. Section IV concludes the paper.

II. Radial Basis Function as Intrusion Detection Technique

In this section, we formulate the problem of intrusion detection in a pure mathematical framework ending up with intrusion detection system as a sort of adaptive network.

A. *RBF as Strict Interpolator (Approximator)*

As a concept, radial basis function has its root in approximation theory of mathematics, in particular, in the field of strict multivariate functions interpolation where the treated problem is: Consider a set of $\mathcal{M}$ distinct data items (vectors) as $\{x_i\,; i = 1,2,\cdots \mathcal{M}\}$ in $\Re^d$ and a set of $\aleph$ real numbers $\{f_i\,; i = 1,2,\cdots \aleph\}$ in $\Re$ in such a way, we choose a function (mapping)
U: $\Re^d \rightarrow \Re$ which satisfies the strict interpolation condition as in (1):

$$U(x_i) = f_i,\ i = 1,2,\cdots \mathcal{M} \tag{1}$$

It is worth noting that $U(x_i)$ is constrained to go through all available data points.

The core idea of radial basis function is to establish a linear space of functions. The basis functions of this space depend on the "radial" distance between the data items and the centers of these functions. For sake of constructing the space, a set of $\mathcal{M}$ basis functions $\varphi\|(x_i - x^c)\|$ is used, where, in general, $\varphi\|.\|$ is nonlinear of distance between data item x_i and centers of basis functions x^c.

In order to approximate a function, a linear combination of these $\varphi\|.\|$ functions used as in (2):

$$U(x_i) = \sum_{j=1}^{M} \lambda_j\, \varphi(\|x - x_j^c\|) \stackrel{\text{def}}{=} \sum_{j=1}^{M} \lambda_j\, \varphi_j(r) \tag{2}$$

Referring to (1), $U(x_i) = f_i$, Therefore,

$$f_i = \sum_{j=1}^{M} \lambda_j\, \varphi_j(r) \tag{3}$$

In terms of matrices, equation (3) can be re-written as in (4):

$$\begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_\aleph \end{pmatrix} = \begin{pmatrix} A_{11} & \cdots & A_{\mathcal{M}1} \\ \vdots & \cdots & \vdots \\ A_{\mathcal{M}1} & \cdots & A_{\mathcal{M}\mathcal{M}} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_\aleph \end{pmatrix} \tag{4}$$

Where

$$A_{ji} = \varphi(\|x - x_j^c\|) \tag{5}$$

Thus, if A_{ji}^{-1} exist, then $U(x_i)$ can be *strictly* approximated using (6):

$$\boldsymbol{\lambda} = A^{-1}\boldsymbol{f} \tag{6}$$

Where $\boldsymbol{\lambda}$ and $\boldsymbol{f}$ are vectors.

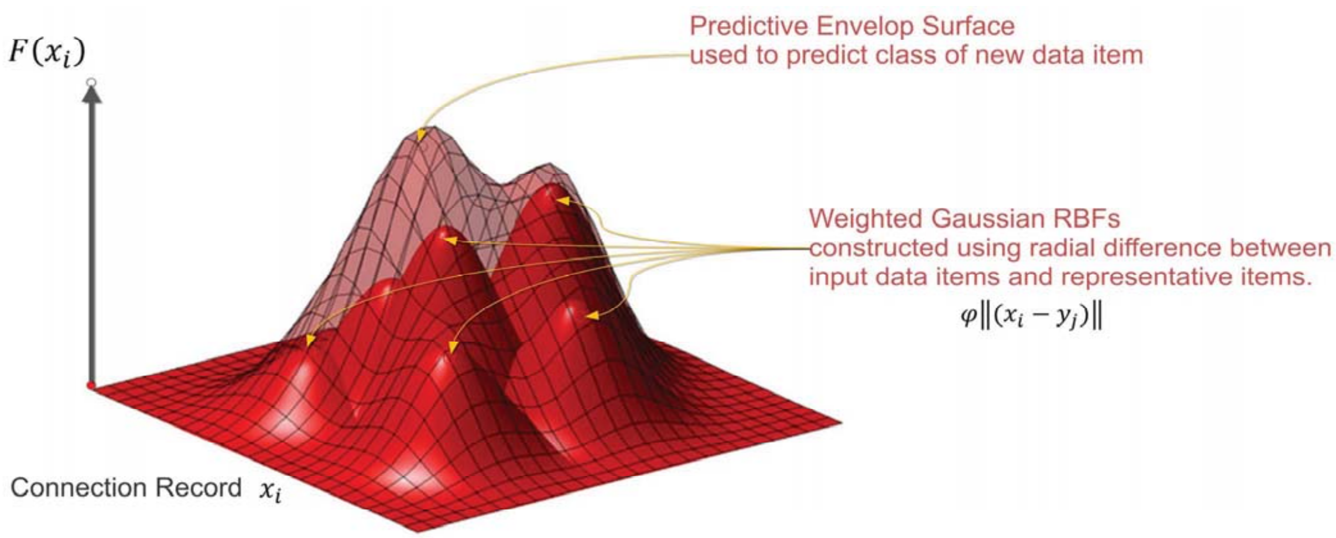

Fig. 1. Explanatory example of predictive surface generated by six weighted and spatially shifted Gaussian bells in 2D space produced by means of RBF neurons [14]. *Note that* $x_i \in \Re^2$ *instead of* $\Re^d$, *for purpose of clarification.*

As can be deduced from (1- 6), we can visualize using radial basis function as approximator as a space made of functions where each function in the space thought of as a point. RBF based network approximate our function (record labels) by stretching and compressing spatially-shifted Gaussian bells based on µs' determined by x_j^c. As a deeper insight, this process emulates surface re-construction using scattered data points. This reconstructed space is our target in the framework of problem of attack detection since it represents the *predictive* surface we look for, namely, the predictive surface form the envelope of all Gaussian bells involved in approximation, which, provides us the complete surface (input-output mapping) that fill the gaps in situ of new data items as can be shown in Fig. 1. [14].

B. Intrusion Detection as Approximation Problem

In order to formulate the problem in the framework of approximation theory, it is an essential step to explore our target data. In this work, we employ NSL-KDD dataset that originally represent the enhanced version of KDD Cup'99 dataset proposed in third international knowledge discovery and data mining tools competition under DARPA 1998 Intrusion Detection Evaluation Program. This dataset is composed of a massive set (about 125,000) network activity recordings includes a board variety of attacks originally simulated in a military network environment. Each connection record represents a sequence of 41 TCP packets which compose input vector x_i. For each input vector, a label of either normal or an attack (referring to the type of attack) is associated with it as:

Normal connection record:
2,tcp,ftp_data,SF,12983,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,134,86,0.61,0.04,0.61,0.02,0.00,0.00,0.00,0.00,**normal**

Attack connection record:
0,icmp,ecr_i,SF,520,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,511,511,0.00,0.00,0.00,0.00,1.00,0.00,0.00,46,59,1.00,0.00,1.00,0.14,0.00,0.00,0.00,0.00,**smurf**

In terms of approximation theory, these connection records represent the data items x_i to be mapped to f_i values. Therefore, we seek a mapping of $U\colon \mathfrak{R}^{41} \rightarrow \mathfrak{R}$.

C. Radial Basis Function as Layered Network

One of the essentials of data fitting is the required numbers of freedom, namely, the minimum number of basis functions that required to establish an acceptable fit to target data. In case of intrusion detection, the number of connection records far exceeds the number of degrees of freedom, gives rise to prodigious redundancy since strict interpolation constrained to use as many radial basis functions as data items. In order to comfort this issue, Brommhead and Lowe [15] suggested to relax the strict interpolation of eq.(1) by weakening the conditions as follows:

Instead of considering the radial distance between a particular data item x_i and the rest of data items x^c, the generalized version of conventional radial basis function considers the radial distance between the target data item x_i and a set of data representative points y_j. For dataset $\mathfrak{D}$ composed of $\mathcal{M}$ items, the representative points can be selected either randomly or in a systematic way through clustering techniques or other techniques, in our system, we use the simple k-means algorithm.

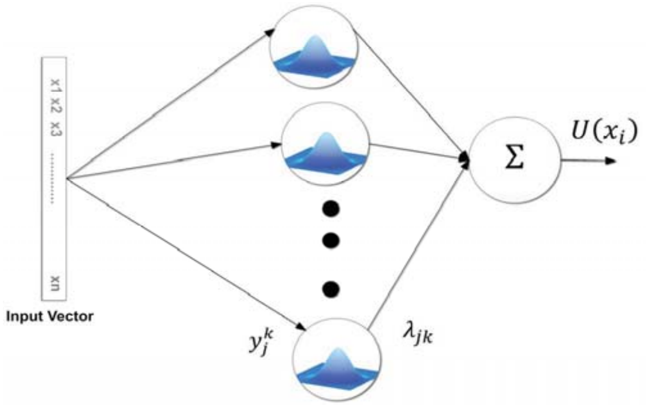

Fig. 2. Graphical representation of (8) where $U\colon \mathfrak{R}^{41} \rightarrow \mathfrak{R}$

Mathematically expressed, instead of using $\varphi(\|(x_i - x_j^c)\|)$ $where\ i,j = 1,2,\cdots,m.\ i \neq j\ \ x_i, x_j^c \in \mathfrak{D}$ the generalized version uses $\varphi(\|(x_i - y_j)\|)$ $where\ i = 1,2,\cdots,\mathcal{M}\ \ j = 1,2,\cdots,\mathcal{N}_0\ \ \mathcal{N}_0 \ll \mathcal{M}\ \ x_i \in \mathfrak{D}\ \ y_j \in \mathfrak{D}'$ where $\mathfrak{D}'$composed of representative data items may or may not belong to the original dataset $\mathfrak{D}$.

Therefore, we end up with limited number of radial basis functions φ equals to $\mathcal{N}_0$. However, this relaxation has a compromise, the matrix A_{ji} is no longer square. Thus, no unique inverse exists. To address this issue, pseudo-inverse matrix A^+ [16] was used instead as in (7):

$$\boldsymbol{\lambda} = A^{+1}\boldsymbol{f} \qquad (7)$$

Subsequently equation (2) is relaxed to (8):

$$U(x_i) = \lambda_o + \sum_{j=1}^{M} \lambda_j\, \varphi(\|x_i - y_j\|) \qquad (8)$$

where $x_i \in \mathfrak{R}^d$, $j\colon number\ of\ cluster\ cneters, j = 1,2,\cdots,\mathcal{N}_0$. Since our input vector is of high independent components $x_i \in \mathfrak{R}^{41}$, λ_o represents the constant offset (bias).

Equation (8) reveals a couple of two important deduction: Firstly, it represents a remarkable reduction from nonlinear mapping to linear algebra. Secondly, it represents one form of adaptive network as can be depicted graphically as shown in Fig. 2.

However, in this work, the attack classification is in type; where we have four major attack types as will be illustrated in next section. Thus, our output vector is a vector composed of four binary components, i.e., $U\colon \mathfrak{R}^{41} \rightarrow \mathfrak{R}^4$. Consequently, we will have λ_{jk} for each output component of $\boldsymbol{U} = \{u_1\, u_2\, \cdots u_n\}$ as in (9):

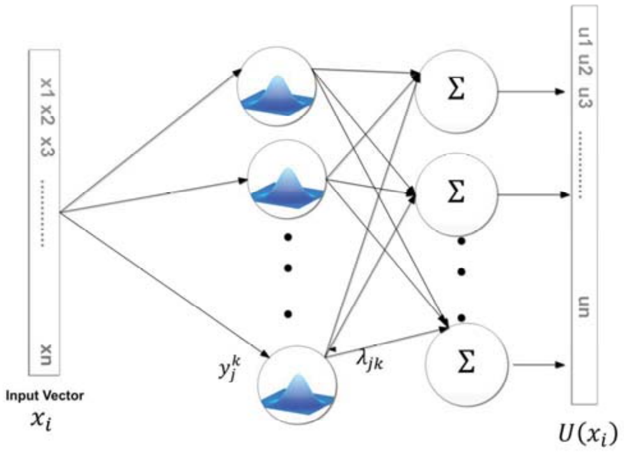

Fig. 3. Graphical representation of (9) where $U: \Re^{41} \rightarrow \Re^n$

$$\boldsymbol{U}(x_i) = \lambda_{ok} + \sum_{j=1}^{M} \lambda_{jk}\, \varphi(\|x_i - y_j^k\|) \tag{9}$$

where $x_i \in \Re^d$, $\boldsymbol{U} \in \Re^n$ $\quad j: number\ of\ cluster\ cneters, j = 1,2,\cdots,\mathcal{N}_0$ $k = 1,2,\ldots,n$. Since our input vector is of high independent components $x_i \in \Re^{41}$, λ_{ok} represents the constant offset (bias) for each k^{th} component of $\boldsymbol{U}$ vector. which can be depicted as in Fig. 3.

The binarization of (9) takes the form as in (10):

$$u_k = \begin{cases} 1, & if\ u_k = max_k\{U\} \\ 0, & otherwise \end{cases} \tag{10}$$

Therefore, the type of attack cab be identified through scoring process of the sum's outputs of (9), i.e., the sum of high score (cumulative output) give rise to a specific attack type.

III. Experimental Analysis and Results

In this section, RBF-based intrusion detection system is applied to NSL-KDD dataset for sake of multi-classification. The proposed work was implemented on MATLAB 2018b, working on a system with an i7 processor having 8GB RAM. In this application, we adapt two-phase RBF training through k-means algorithm of k = 15, where σs' and μs' values of the clusters were found and used to establish Gaussian functions φ's of hidden neurons as illustrated in Fig. 4. In this work, we designed a multi-class intrusion detection scheme, namely, instead of detecting normal or attack, the system can detect the attack and classify the type that belongs to as shown in Table I.

TABLE I
Attack Categories and Sub-Categories Found in Training and Testing Datasets

DOS	Probe	R2L	U2R
Back	Satan	Guess_password	Buffer-overflow
Land	Ipsweep	Guess-passwd	Loadmodule
Neptune	Nmap	ftp-write	Rootkit
Pod	Portsweep	Imap	Perl
Smurf	Mscan	Phf	Sqlattack
Teardrop	Saint	Multihop	Xterm
Mailbomb		Warezmaster	Ps
Processtable		Xlock	
Udpstorm		Xsnoop	
Apache2		Snmpguess	
Worm		Snmpgetattack	
		Httptunnel	
		Sendmail	
		Named	
		Warezclient	
		Spy	

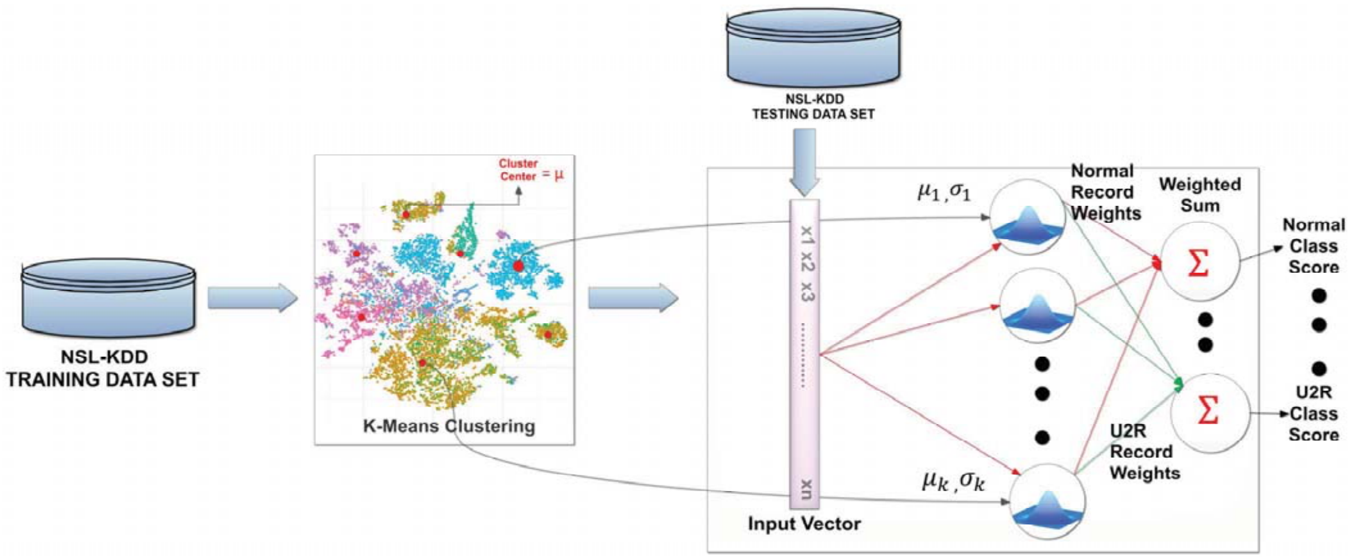

Fig. 4. Graphical pipeline of experimental analysis of proposed RBF-based IDS.

NSL-KDD dataset shows high levels of imbalance between different attacks statistics, for example, the low frequency of U2R and R2L instances in contrast to high frequency of DOS and Probe instances leads to explicit biasing of detection performance towards high frequent classes. Moreover, this imbalance lessens the sensitivity of RBF functions towards low frequent attacks which lowers the detection performance reliability.

As a pre-liminary data pre-processing step, we followed same steps were conducted in [17] except one *core* step impacts our detection performance. In this work, we address the issue of data imbalance by oversampling least frequent instances {U2R, R2L} leading to increase the total statistics of these types of attacks increasing the number of total records that involved in the testing and training phases of proposed model which ends up with a new statistic for our datasets.

The attack label is classified into one of four categories:

DOS: Denial of Service Attacker attempts to not allowing legitimate users from using a service.

Prob: Probing Attacker attempts to discover vulnerable hosts in the internet.

R2L: Remote to Local

Attacker does not have an authorized local access and attempts to gain unauthorized one.

U2R: User to Remote

Attacker have an authorized local access, but attempts to grab the privilege of root (administrator) access.

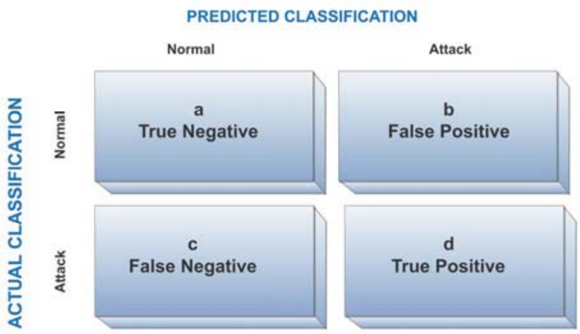

Fig. 5. Confusion matrix of detection response.

Each of attack categories include several attack types as illustrated in Table I. As a final step of preliminary dataset processing, according to Table I, labels of connection records undergo discretization where the different types of attacks grouped into one category and given discrete integer value: {Normal:0, DOS: 1, Probe: 2, R2L:3, U2R: 4}, which, in other hand, represent distinct values of f_i.

In this work, we adopted two types of overall performance evaluation: (1) Binary-wise performance evaluation. (2) Class-wise overall performance evaluation. Where the former is built on binary confusion matrix of intrusion detection outputs, whereas the later is built based on the class-wise confusion matrix where the average of multi-class system responses is calculated and considered as overall performance. Both types of performance evaluation use same measures that are essentially based on the confusion matrix as elaborated in Fig. 5. Based on Fig. 5, the classification results of testing connection records are typically given in terms of the following measures of performance (11- 14):

$$Sensitivity(Detection\ Rate) = \frac{d}{(d+c)} \tag{11}$$

$$False\ Positive\ Rate\ (FPR) = \frac{b}{(b+a)} \tag{12}$$

$$Positive\ Predictive\ Value(Precision) = \frac{d}{(d+b)} \tag{13}$$

$$Accuracy\ (Acc) = \frac{\mathrm{d+a}}{(a+b+c+d)} \tag{14}$$

TABLE II
BINARY-WISE CONFUSION MATRIX

	PREDICTED	
ACTUAL	**Normal**	**Attack**
Normal	19793	551
Attack	1136	18521

TABLE III
CLASS-WISE CONFUSION MATRIX

	PREDICTED				
ACTUAL	**Normal**	**DoS**	**Probe**	**R2L**	**U2R**
Normal	19793	170	126	252	3
DoS	492	13269	90	4	0
Probe	252	127	3067	0	0
R2L	382	0	0	1695	2
U2R	10	0	0	15	252

Using subset of 40,000 records for sake of performance testing and validation and 15 neurons for the hidden layer of RBFNN, simulation results of our intrusion detection system are given as confusion matrix of attack detection in type and the corresponding performance metrics as shown in Table II and Table III respectively.

In this paper, our proposed model is compared with two prominent types of RBF-based intrusion detection systems: (1) Intrusion detection systems built on classical version of RBF as listed in Table V. (2) Intrusion detection systems built on optimized (improved) version of RBF or systems built using RBF network as a part of a hybrid intrusion detection system as listed in TABLE VI. The first type (which our proposed model belongs to) employs the basic version of RBF network to classify the records using simple clustering technique (as k-means or SOM) to set network parameters.

TABLE V
COMPARATIVE ANALYSIS WITH CLASSICAL RBF NETWORK BASED ID SYSTEMS/MODELS

Method	Accuracy	Detection Rate	FP	Precision	Response Time
Rapaka *et al.* [4]	97%	-	-	-	-
Jiang *et al.* [5]	-	Single-Layer RBF-IDS: DoS : 98.4% Probe : 99.6% R2L : 98.0%	Single-Layer RBF-IDS: DoS : 0% Probe: 0% R2L : 0%	-	3 min / 4,900,000 training records
Jiang *et al.* [5]	-	Multi-Layered RBF-IDS: SHIDS[a]: Normal: 99.5% Ipsweep 99.1% Smurf: 99.3% Guess-Pass: 98.2% Buffer-Over: 94.1% PHIDS[b]: Normal: 99.8% Ipsweep 99.5% Smurf: 99.0% Guess-Pass: 99.7% Buffer-Over: 98.8% Portsweep: 86.9%	Multi-Layered RBF-IDS: SHIDS: Normal: 1.2% Ipsweep 1.62% Smurf: 4.5% Guess-Pass: 0% Buffer-Over: 5.4% PHIDS: Normal: 1.2% Ipsweep 0.8% Smurf: 0% Guess-Pass: 4% Buffer-Over: 3.3% Portsweep: 0%	-	-
Yang *et al.* [1]	-	97.1%	1.6%	-	4 sec / 1,200 training records
Bi *et al.* [2]	-	87%	-	-	-
Devaraju and Ramakrishnan [3]	75.4% (400 test records)	-	-	-	-
Shi et al. [8]	-	77.6%	3.27%	-	-
Chen *et al.* [6]	Class-wise: DoS : 85.14% Probe : 81.71% R2L : 88.00% U2R : 86.86%				
Li-Zhong *et al.* [13]	83.5% (1190 testing records) 82.5% (2170 testing record)	-	-	-	-
Proposed RBFN IDS	Class-wise: DoS : 98.04% Probe : 98.37% R2L : 97.13% U2R : 99.94% **Average: 98.37%** Binary-wise: 95.87%	Class-wise: DoS : 96.42% Probe : 92.41% R2L : 81.61% U2R : 96.18% **Average: 92.86%** Binary-wise: 94.22%	Class-wise: DoS : 0.85% Probe : 0.63% R2L : 1.26% U2R : 0.015% **Average: 0.688%** Binary-wise: 2.7%	Class-wise: DoS : 98.74% Probe : 96.05% R2L : 87.06% U2R : 98.82% **Average: 95.16%** Binary-wise: 97.11%	7.11 sec/ 68,000 training records 3.39 sec/ 40,000 testing records

[a]SHIDS Serial Hierarchical RBF-IDS.
[b]PHIDS Parallel Hierarchical RBF-IDS.

On the other hand, the optimized versions of RBF network use optimization techniques to determine the optimal values of these parameters or combine the classical RBF network with other classification technique in a hybrid manner. In our proposed model, instead of optimizing RBF network, we optimized the training dataset that used to determine the parameters of Gaussian functions, we established a balanced grid of high selective Gaussian functions for each type of attacks leading to a balanced bank of filters which, reflect in high detection performance without optimization or hybrid steps. Based on the detection results provided in Table II and Table III the binary and class-wise overall performance evaluation represented by detection accuracy, detection rate, false positive rate, precision and time responses for training and testing stages of our proposed system associated with a comparison to other classical, optimized and hybrid RBFN-based intrusion detection models/systems are presented in Table V and Table VI respectively. Examining the results provided in Table V, it can be observed that although our proposed classifier uses the classical version of RBF network

TABLE VI
COMPARATIVE ANALYSIS WITH OPTIMIZED/HYBRID RBF NETWORK BASED ID SYSTEMS/MODELS

Method	**Accuracy**	**Detection Rate**	**FP**	**Precision**	**Response Time**
Zhong *et al.* [9]	-	94.04%	0.71%	-	-
Yichun et al. [10]	-	-	-	83.7%	-
Xu et al. [7]	-	Class-wise: Normal: 84.14% DoS : 100 % Probe : 48% R2L : 81.6% U2R : 42% Binary-wise: 98.95%	Class-wise: Normal: 1.28% DoS : 0 % Probe : 30% R2L : 0.45% U2R : 30.3% Binary-wise: 2.05%	-	-
Shi et al. [8]	-	Class-wise: DoS : 97.6% Probe : 96.3% R2L : 98.1% U2R : 95.4% Binary-wise: GA-RBF: 86.8% Chaos-GA-RBF: 95.4%	Class-wise: DoS : 0.92% Probe : 0.83% R2L : 0.99% U2R : 0.86% Binary-wise: GA-RBF: 1.21% Chaos-GA-RBF: 1.05%	-	-
Chen *et al.* [6]	Class-wise: DoS : 92.57% Probe : 88.86% R2L : 94.29% U2R : 91.57%	-	-	-	-
Ma *et al.* [12]	-	QPSO-RBFN 92.08%	QPSO-RBFN 5.29%	-	-
Li-Zhong *et al.* [13]	92.8% (1190 testing records) 91% (2170 testing record)	-	-	-	-
Proposed RBFN IDS	Class-wise: DoS : 98.04% Probe : 98.37% R2L : 97.13% U2R : 99.94% Average: 98.37% Binary-wise: 95.87%	Class-wise: DoS : 96.42% Probe : 92.41% R2L : 81.61% U2R : 96.18% Average: 92.86% Binary-wise: 94.22%	Class-wise: DoS : 0.85% Probe : 0.63% R2L : 1.26% U2R : 0.015% Average: 0.688% Binary-wise: 2.7%	Class-wise: DoS : 98.74% Probe : 96.05% R2L : 87.06% U2R : 98.82% Average: 95.16% Binary-wise: 97.11%	7.11 sec/ 68,000 training records 3.39 sec/ 40,000 testing records

as the principal classifier, it has achieved high detection performance, where 40,000 record take about less than 4 seconds to be tested and less than 8 seconds to build the system model using 65,000 training records. Moreover, the system shows high accuracy towards rare attacks {R2L and U2R} reached up to 97.13% and 99.94% and FPR as low as 1% and 0.015% for R2L and U2R respectively. Referring to Table V, among all classical RBF-based ID systems, our proposed model as achieved the best in terms of class-wise accuracy and it is competetive with other systems and is particularly strong in all perfromance measures when dealing with rare and difficult-to-detect U2R attacks.

In terms of detection rate, our proposed model falls behind some other methods such as proposed by Jiang *et al.* [5] and Yang *et al*.[1] . However, for single layer IDS proposed by [5], the system can detect all types of attacks *except* U2R attack. Moreover, in case of multi-layered IDS proposed by same authors, the system was designed to deal with very specific types of major types of attacks, i.e., refering to TABLE I, Ipsweep belongs to Probe attack category, smurf belongs to DoS, Guess-pass belongs to R2L and buffer-overflow belongs to U2R category. In contrast, our proposed system has the ability to deal with a broad band of attack types per each attack category with competitive class-wise detection rates, precisions and FPRs and without the complexity of multi-layered structures. Moreover, both references [5] and [1] used KDD Cup'99 dataset for testing, training and validation purposes whereas, our prposed system was trained, tested and validated using NSL-KDD dataset which is considered a challenging dataset especially for multi-class wise intrusion detection. Interesetingly, according to recent survey done by Chowdhury et al. [18], it was found that detection accuracy of intrusion detection systems built using NSL-KDD dataset was stuck around 85% approximatley for multi-class classifiers that used all features. On the contrary, most of intrusion detection systems were built suing KDD Cup'99had achieved impeccable detection perfromance reached up to 100% for some shemes; espacially for binary classifiers. To sum up, NSL-KDD dataset is more reliable to mimic the challenging nature of real-world security issues. [1] and [5] mentioned nothing about the detection precision or accuracy which represent strong indicators of IDS effeciency. All of these aspects causing [1] and [5] less competetive.

In order to further reflect the superiority of our proposed model, Table VI illustrates the accuracy, detection rate, FPR, precision and response time for optimized/hybrid state-of-the-art RBF-based intrusion detection models/systems validated on KDD Cup'99 dataset, in which, the most competitive work to our proposed work in terms of binary classification are Xu et al. [7] and Shi *et al*. [8]. However, in terms of class-wise classification, our proposed system is better in terms of both detection rate and FPR. In contrary to state-of-the-art RBFN-based schemes, our proposed system shows higher efficiency and reliability, even though it has much lower complexity and it is much easier to implement. That is why high achievable classical RBF-based intrusion detection systems are considered essential blocks in hierarchal on-line adaptive intrusion detection systems that designed to detect novel and hard-to-detect attacks in hostile real-time environment.

IV. Conclusion and Future Work

In this work, we formulate the intrusion detection problem in the framework of approximation(interpolation) theory and applying RBFNN as classifier. The accuracy and PPV value of attack classification reached up to 95.78% and 97.11% respectively with high sensitivity to rare attacks (R2L and U2R). However, for this system, we have worked only with 2 phase-RBF training which reflect in relatively low sensitivity towards frequent attacks (DoS and Probe). Therefore, as future work, this system can be enhanced by extending to 3-phase RBF training using different basis functions other than Gaussians and other data representation technique rather than simple k-means algorithm applying on different datasets other than NSL-KDD dataset.

References

[1] Z. Yang, X. Wei, L. Bi, D. Shi, and H. Li, "An intrusion detection system based on RBF neural network," in *Proceedings of the Ninth International Conference on Computer Supported Cooperative Work in Design, 2005.*, 2005, pp. 873-875.

[2] J. Bi, K. Zhang, and X. Cheng, "Intrusion detection based on RBF neural network," in *2009 International Symposium on Information Engineering and Electronic Commerce*, 2009, pp. 357-360.

[3] S. Devaraju and S. Ramakrishnan, "Performance analysis of intrusion detection system using various neural network classifiers," in *2011 international conference on recent trends in information technology (ICRTIT)*, 2011, pp. 1033-1038.

[4] A. Rapaka, A. Novokhodko, and D. Wunsch, "Intrusion detection using radial basis function network on sequences of system calls," in *Proceedings of the International Joint Conference on Neural Networks, 2003.*, 2003, pp. 1820-1825.

[5] J. Ju, Z. Chunlin, and M. Kamel, "RBF-based real-time hierarchical intrusion detection systems," in *Proceedings of the International Joint Conference on Neural Networks, 2003.*, 2003, pp. 1512-1516 vol.2.

[6] Z. Chen and P. Qian, "Application of PSO-RBF neural network in network intrusion detection," in *2009 Third International Symposium on Intelligent Information Technology Application*, 2009, pp. 362-364.

[7] R. Xu, R. An, and X. Geng, "Research intrusion detection based PSO-RBF classifier," in *2011 IEEE 2nd International Conference on Software Engineering and Service Science*, 2011, pp. 104-107.

[8] Y. Shi, J. Bao, Z. Yan, and S. Jiang, "Intrusion detection for transportation information security systems based on genetic algorithm-chaos and RBF neural network," in *2011 Third Pacific-Asia Conference on Circuits, Communications and System (PACCS)*, 2011, pp. 1-3.

[9] J. Zhong, Z. Li, Y. Feng, and C. Ye, "Intrusion detection based on adaptive RBF neural network," in *Sixth International Conference on Intelligent Systems Design and Applications*, 2006, pp. 1081-1084.

[10] P. Yichun, N. Yi, and H. Qiwei, "Research on Intrusion Detection System Based on IRBF," in *2012 Eighth International Conference on Computational Intelligence and Security*, 2012, pp. 544-548.

[11] L.-Y. Tian and W.-P. Liu, "Incremental intrusion detecting method based on SOM/RBF," in *2010 International Conference on Machine Learning and Cybernetics*, 2010, pp. 2849-2853.

[12] R. Ma, Y. Liu, X. Lin, and Z. Wang, "Network anomaly detection using RBF neural network with hybrid QPSO," in *2008 IEEE International Conference on Networking, Sensing and Control*, 2008, pp. 1284-1287.

[13] L. Li-zhong, L. Zhi-guo, and D. Xian-hui, "Network intrusion detection by a hybrid method of rough set and RBF neural network," in *2010 2nd International Conference on Education Technology and Computer*, 2010, pp. V3-317-V3-320.

[14] "Radial basis function (RBF) approximation for PDE problems," *RBF Research Group. Uppsala University. Available Online: http://www.it.uu.se/research/scientific_computing/project/rbf*, 2019.

[15] D. S. Broomhead and D. Lowe, "Radial basis functions, multi-variable functional interpolation and adaptive networks," Royal Signals and Radar Establishment Malvern (United Kingdom)1988.
[16] G. Golub and W. Kahan, "Calculating the singular values and pseudo-inverse of a matrix," *Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis,* vol. 2, pp. 205-224, 1965.
[17] M. Almi'ani, A. A. Ghazleh, A. Al-Rahayfeh, and A. Razaque, "Intelligent intrusion detection system using clustered self organized map," in *2018 Fifth International Conference on Software Defined Systems (SDS)*, 2018, pp. 138-144.
[18] M. M. U. Chowdhury, F. Hammond, G. Konowicz, C. Xin, H. Wu, and J. Li, "A few-shot deep learning approach for improved intrusion detection," in *Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), 2017 IEEE 8th Annual*, 2017, pp. 456-462.