This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2019.2927886, IEEE Transactions on Network and Service Management

A Hybrid Deep Learning based Model for Anomaly Detection in Cloud Datacentre Networks

Sahil Garg, *Member, IEEE,* Kuljeet Kaur, *Member, IEEE,* Neeraj Kumar, *Senior Member, IEEE,*
Georges Kaddoum, *Member, IEEE,* Albert Y. Zomaya, *Fellow, IEEE,* and Rajiv Ranjan, *Senior Member, IEEE*

***Abstract*—With the emergence of the Internet-of-Things (IoT) and seamless Internet connectivity, the need to process streaming data on real-time basis has become essential. However, the existing data stream management systems are not efficient in analyzing the network log big data for real-time anomaly detection. In this context, the existing anomaly detection approaches are not efficient because they cannot be applied to networks, are computationally complex, and suffer from high false positives. Thus, in this paper a hybrid data processing model for network anomaly detection is proposed that leverages Grey Wolf Optimization (GWO) and Convolutional Neural Network (CNN). To enhance the capabilities of the proposed model, GWO and CNN learning approaches were: (i) enhanced with improved exploration, exploitation and initial population generation abilities and (ii) revamped dropout functionality. These extended variants are referred to as Improved-GWO (ImGWO) and Improved-CNN (ImCNN), respectively. The proposed model works in two phases for efficient network anomaly detection. In the first phase, ImGWO is used for feature selection in order to obtain an optimal trade-off between two objectives, *i.e.*, reduced error rate and feature-set minimization. In the second phase, ImCNN is used for network anomaly classification. The efficacy of the proposed model is validated on benchmark (DARPA'98 and KDD'99) and synthetic datasets. The results obtained demonstrate that the proposed cloud-based anomaly detection model is superior in comparison to the other state-of-the-art models (used for network anomaly detection), in terms of accuracy, detection rate, false positive rate and F-score. In average, the proposed model exhibits an overall improvement of 8.25%, 4.08% and 3.62% in terms of detection rate, false positives, and accuracy, respectively; relative to standard GWO with CNN.**

***Index Terms*—Anomaly detection, Convolutional Neural Network, Cloud Computing, Feature selection, and Grey Wolf Optimization.**

I. Introduction

THE need for increased computational power and on-demand services as per the user's requirements has paved the way to one of the most powerful technologies of the modern era, Cloud Computing (CC). According to Gartner, CC has been growing at a rate of 40% and will continue to rise at a rate of more than 25% per year [1]. However, transition from traditional client-server architectures to CC is not straightforward and there are a number of operational and security challenges induced due to its underlying virtualized environment. These risks further aggravate with the emergence of the Internet of Things (IoT) in which smart devices communicate with each other using an open channel, Internet. Moreover, these connected devices, deployed across different enterprises, generate large volumes of streaming data, ranging from micro-blog feeds and financial information to complex network monitoring logs [2].

Recent studies have shown that intruders have successfully launched several attacks, which have caused unprecedented levels of disruption in various CC-hosted application services. Recent insights on Cloud Adoption and Security by Forbes says that 49% of businesses are delaying cloud deployment due to cybersecurity issues [3]. According to existing proposals and reports, more than 20% of enterprises in the world witnessed at least one form of Denial of Service (DoS) attack on their infrastructures. For instance, DoS attack on the Amazon cloud infrastructure caused the BitBucket site to be unavailable for a substantial amount of time [4]. Likewise, Dropbox was rendered un-operational for more than 15 hours [5]. Apart from this, researchers from Symantec have discovered that the growing dependence on Cloud services has opened doors for more severe forms of intrusions. Thus, in order to remain resilient, the cloud needs to possess the ability to react not only to the known threats, but also to new emerging threats which may target its underlying networking infrastructure. To combat these challenges, researchers have extensively used Intrusion Detection Systems (IDSs) as a defensive strategy for cloud security [6]. IDSs used in cloud environments include misuse detection, anomaly detection, hypervisor introspection (HVI), virtual machine introspection (VMI), and a combination of these. Among all these techniques, anomaly detection with respect to heterogeneous traffic flow data generated due to diverse application types, is still in its infancy.

More recently, different variants of anomaly detection techniques, amalgamated with IDSs, were proposed in the literature [7], [8]. For instance, Pandeeswari *et al.* [9] proposed an IDS at the hypervisor layer to detect attacks in cloud environments using Fuzzy C-Means clustering algorithm along with Artificial Neural Network (FCM-ANN). Similarly, Watson *et al.* [10] proposed an online cloud anomaly detection technique which uses one-class Support Vector Machine

S. Garg, K. Kaur, and G. Kaddoum are with the Electrical Engineering Department, École de technologie supérieure, Université du Québec, Montréal, QC H3C 1K3, Canada. (E-mail: sahil.garg@ieee.org, kuljeet.kaur@ieee.org, and georges.kaddoum@etsmtl.ca)

N. Kumar is with the Department of Computer Science & Engineering, Thapar Institute of Engineering & Technology (Deemed to be University), Patiala (Punjab), India. (E-mail: neeraj.kumar@thapar.edu).

A. Y. Zomaya is with the School of Information Technologies, J12, University of Sydney, NSW 2006, Australia (E-mail: albert.zomaya@sydney.edu.au).

R. Ranjan is with the Computer Science Department, Newcastle University, UK and China University of Geosciences, China (E-mail: raj.ranjan@ncl.ac.uk).

(SVM) algorithm to detect various types of malware and DoS attacks in CC infrastructures. Further, Ye *et al.* [11] proposed an anomaly detection framework based on Software-Defined Networks (SDN) for cloud setups. Sha *et al.* [12] designed a multi-order Markov chain based model for anomaly detection using DARPA dataset. In [13], Tan *et al.* used Multivariate Correlation Analysis (MCA) for accurate characterization of known and unknown DoS attacks. Although competent in general anomaly detection, most of these approaches suffer from high false alarm rates and elevated computational complexity. Hence, these schemes are not efficient particularly for network anomaly detection in streaming data, which requires real-time analysis [14].

Recently, another trend has grabbed the attention of researchers for network anomaly detection, namely deep-learning (DL). It is a widely-accepted machine learning approach that plays a significant role in detecting the most relevant features from huge datasets using back propagation. Ever since its inception, different architectures have been proposed in the literature such as-Deep Neural Networks, Deep Belief Networks, Recurrent Neural Networks and Convolutional Neural Networks (CNN) [15]. Among these techniques, CNNs are widely utilized for data classification due to their inherent ability to be trained with minimum pre-processing requirements; which makes them suitable for network anomaly detection.

A. Motivation

It is evident from the above discussion that a number of proposals have been suggested to detect anomalous behavior in network traffic using a wide variety of techniques such as-SVM, MCA, FCM-ANN, *etc*. However, these techniques are inefficient because of their reduced accuracy and high false positive alarms. Additionally, due to the heterogeneous and diverse nature of cloud environments, existing techniques may not be applicable to handle the challenges induced due to the existence of virtualized environments and different types of application workloads [16]. In order to tackle these exploding security risks, an efficient anomaly detection technique for streaming data needs to be designed. It should involve careful examination of both historical and real-time data streams with high accuracy and minimal computational complexity [17], [18].

Hence, an anomaly detection model particularly for heterogeneous data in CC networking environments is designed in this paper. Two important issues are explored in the proposed hybrid model (see Fig. 1), *i.e.*, relevant feature set selection from the traffic stream repository and their classification into benign and anomalous classes. In the proposed model, feature extraction is achieved using Grey Wolf Optimization (GWO) [19], a meta-heuristic approach based on evolutionary computation which is widely accepted for its simplicity, flexibility and ability to yield optimal results. On the other hand, anomaly classification is done using CNN, a promising deep learning approach. In addition to this, the proposed work also enhances the capabilities of the proposed model with (i) improved exploration, exploitation and initial population generation abilities for GWO and (ii) revamped dropout functionality for CNN. The improvised version of CNN with dropout functionality not only helps avoid over-fitting but also increases the weights of the most relevant features of the network. This in turn, simultaneously enhances the accuracy of the architecture while helping it converge faster.

GWO and CNN are powerful techniques that have been exploited by the research community in the networking domain to address various problems. For instance, Yang and Zhou [20] used GWO to design an effective IDS based on cloud with improved exploration and exploitation capabilities. Mao *et al.* [21] proposed the use of CNN for path prediction in SDN by learning from the past experiences and pro-actively updating the routing paths. Here, CNN was deployed at the controller and was specifically used to overcome the issues induced by fixed path routing decisions. In another work, Ji *et al.* [22], [23] employed CNN for network fault prediction by effectively analysing the log files. In this work, the log files were treated as textual files for monitoring the realtime status of the network and predicting any network faults using CNN. Likewise, CNN has also been employed for network intrusion detection in different forms.

B. Contributions

The major contributions of the proposed work are summarized as follows:

- We design an efficient hybrid model using GWO and CNN for efficient network anomaly detection in cloud setups. GWO is used for multi-objective feature extraction, while CNN is used for anomaly classification.
- We propose an improvised version of GWO (ImGWO) which enhances the exploration, exploitation, and initial population generation abilities of the standard GWO on streaming data.
- The capabilities of the standard CNN are improved by revamping the functionality of dropout layer using uniform distribution approach. The modified version of CNN is referred as ImCNN.
- We provide qualitative and quantitative comparison of the proposed hybrid model with the current state-of-the-art models on benchmark and synthetic datasets for network anomaly detection.

C. Organization

The rest of the paper is structured as follows. Section II presents the system model followed by an illustrative description of ImGWO and ImCNN in Section III. The proposed hybrid model for network anomaly detection is described in Section IV. Simulation results are summarized in Section V followed by conclusion and future directions in Section VI.

II. PROPOSED HYBRID MODEL

This section provides an overview of the proposed hybrid model used for network anomaly detection in cloud setups in the context of streaming network traffic data. The detailed architectural diagram is depicted in Fig. 1. The individual

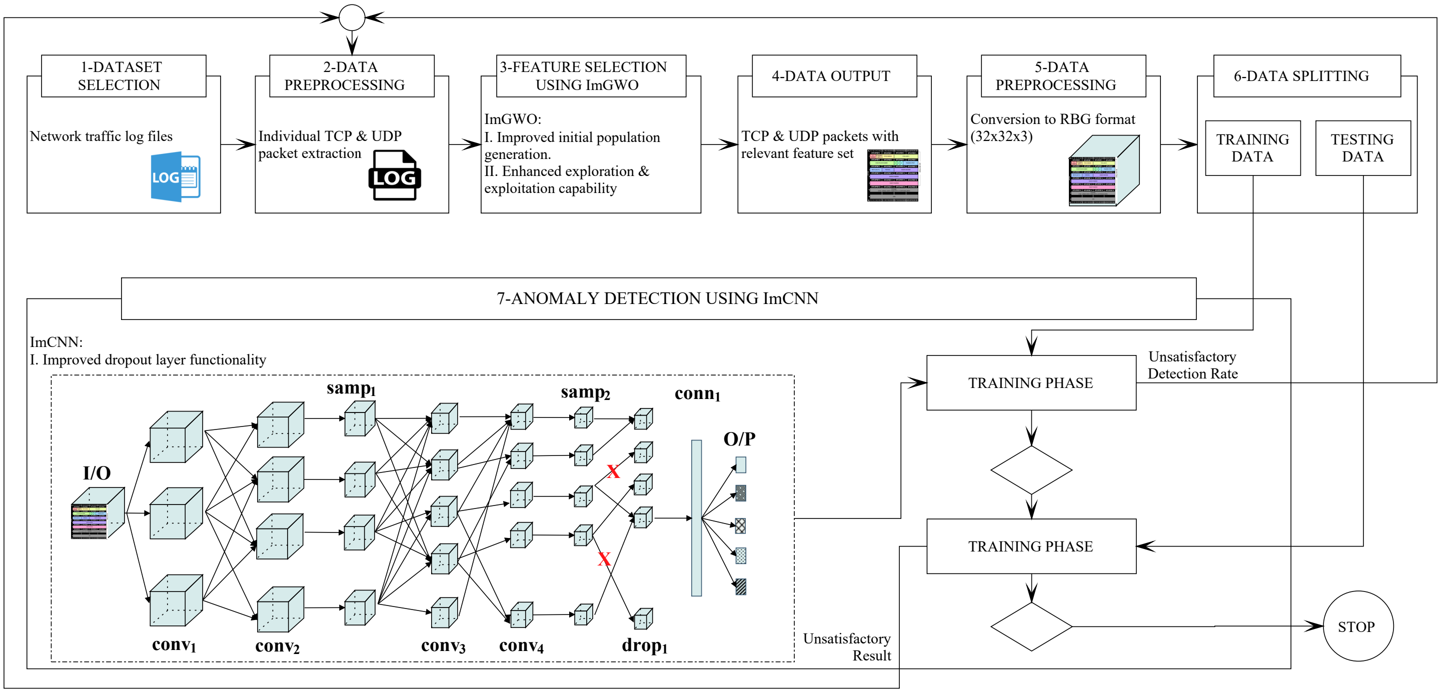

1-DATASET SELECTION
Network traffic log files
LOG
2-DATA PREPROCESSING
Individual TCP & UDP packet extraction
LOG
3-FEATURE SELECTION USING ImGWO
ImGWO:
I. Improved initial population generation.
II. Enhanced exploration & exploitation capability
4-DATA OUTPUT
TCP & UDP packets with relevant feature set
5-DATA PREPROCESSING
Conversion to RBG format (32x32x3)
6-DATA SPLITTING
TRAINING DATA
TESTING DATA
7-ANOMALY DETECTION USING ImCNN
ImCNN:
I. Improved dropout layer functionality
I/O
samp1
samp2
conn1
O/P
conv1
conv2
conv3
conv4
drop1
X
X
TRAINING PHASE
Unsatisfactory Detection Rate
TRAINING PHASE
STOP
Unsatisfactory Result

Fig. 1: Proposed hybrid model using ImGWO and ImCNN for network anomaly detection in cloud setup.

data processing phases are namely-1) dataset selection, 2) data preprocessing, 3) feature selection using ImGWO followed by 4) data output, 5) data preprocessing, 6) data splitting, and 7) anomaly detection using ImCNN. Their detailed description is provided below.

Dataset selection is the first phase of the proposed hybrid model. In this paper, three different datasets were utilized which belong to two categories, *i.e.*, benchmark and synthetic. From these datasets, the tcpdump logs are extracted as they predominantly contain the traffic information pertaining to CC infrastructure. These logs comprise of TCP and UDP packets which constitute almost 90% of the datacenter traffic and thus, are used to represent the network traffic flow data. For these reasons, the considered log files have been considered to detect anomalies over the Internet traffic. During the second phase, the proposed model processes the input data for ImGWO. The individual TCP and UDP packets are extracted from the tcpdump logs and are given as input to the ImGWO. Following this, feature extraction phase is executed which is considered as an important prerequisite in any classification problem ranging from complex images and videos to textual and audio contents. Hence, ImGWO is particularly used to extract the relevant feature sets from the given input dataset such as-source IP address and port number, destination IP address and port number, etc. ImGWO is a multi-objective feature extractor that helps to find the optimal number of features from the available dataset with high classification performance. The improvised packets with the relevant features are provided as the output of this phase. Next, the output acquired from the previous step is preprocessed and converted to RBG format images ($32 \times 32 \times 5$). These images serve as the input to the next phase. Finally, the dataset comprising of RBG images is split in the ratio of 70:30; wherein 70% data is used during the training phase, whereas the rest is utilized in the testing phase of the ImCNN. In the former phase, the hybrid model is trained to detect anomalous activities in the network traffic data, while in the latter phase, it identifies the anomalous activities by applying the underlying logic on the historical data and current input data. Finally, during the anomaly detection phase, ImCNN, a multi-class classifier is used to classify the anomalies of the traffic streams. It is comprised of 8 layers to achieve the desired level of classification. The detailed description of the layers is provided in Section III-B.

III. EXTENSIONS TO GWO AND CNN

This section presents the detailed description of the improvements done to the standard GWO and CNN techniques to accelerate convergence and maximize the accuracy. These modifications are discussed as follows. Related symbols and notations are defined in Table I.

A. Improved-GWO Variant

The existing GWO suffers from several problems like random initial population generation and limited exploration and exploitation capabilities which hamper the local search capability of the algorithm and affect the convergence. The improvements corresponding to these shortcomings are discussed herewith:

1) Improved Initial Population Generation: In traditional GWO, the initial population is generated randomly over the search space which may lead to lack of diversity of the pack of wolves in the considered search space. Numerous studies have suggested that the initial population plays a significant role in the global convergence speed and the optimality of the obtained solution. Motivated by this fact, this work generates

TABLE I: List of symbols and their meanings.

NOMENCLATURE	
x_i^j	Positions of the wolves generated using uniform distribution
μ	Mean of the population
σ	Standard deviation of the population
$\vec{A}$ and $\vec{C}$	Coefficient vectors in GWO
$\mathcal{P}_m$	Non-linear function
t	Present iteration in $\mathcal{P}_m$
T	Maximum number of iterations in $\mathcal{P}_m$
r_2 and $r_2^{'}$	Random variables generated using statistical distribution
r_2^{max}	Upper bound for r_2
r_2^{min}	Lower bound for r_2
l	Hidden layer in dropout
i	Hidden units in dropout
$w_i^{(l)}$	Weights associated with underlying layer l
$b_i^{(l)}$	Biases associated with underlying layer l
$r^{(l)}$	A vector of random numbers
$\tilde{y}^{(l)}$	Thinned outputs
$\tilde{y}^{(l)}$	Input to the next layer
f	Activator function
$\mathcal{L}$	List of all the hidden layers (with weights ≤ 0.5)
m	Number of hidden layers employed in dropout
D	Dataset
F	Feature set
$F(.)$	Feature selection algorithm
A_{de}	Decisive attribute
$F^{'}$	Number of selected features
$\mathcal{E}$	Error Rate
$\mathcal{F}$	Fitness function
α	Grey wolf with maximum fitness
β	Grey wolf with second maximum fitness
δ	Grey wolf with third maximum fitness
γ	Any constant value between [0,1]
$\mathcal{E}_c^{F^{'}}$	Classification error rate involved in selecting the feature-set
$\mathcal{E}_c^{F}$	Error rate with all the actual feature set

an appropriate initial population using uniform distribution, wherein the positions of the wolves (x_i^j) are likely to be equally distributed [24]. The computation of x_i^j is achieved as follows:

$$x_i^j = x_{min}^j + \mathcal{U}(\mu, \sigma) \times (x_{max}^j - x_{min}^j) \tag{1}$$

here, positions of wolves, x_i^j are generated using the uniform distribution with the mean and standard deviation of population (μ and σ) respectively.

2) Improved Exploration and Exploitation Capability: The existing coefficient vectors $\vec{A}$ and $\vec{C}$ in GWO are used for the exploration and exploitation, respectively. Using $\vec{A}$ in every iteration, population of wolves are segregated, wherein half of the iterations are devoted to exploration (when $|A|>1$), while half is dedicated to exploitation (when $|A|<1$). However, this division of population may lead to faster convergence with false pareto front. In order to resolve these problems, adaptive mutation is applied to extend the exploration ability of GWO. To control the probability and range of mutation on each wolf, a non-linear function ($\mathcal{P}_m$) is incorporated which is given as:

$$\mathcal{P}_m = 0.5e^{-10*t/T} + 0.01 \tag{2}$$

where t is the present iteration and T denotes the maximum number of iterations. It can be seen from Eq. (2) that increasing iterations causes $\mathcal{P}_m$ to increase exponentially. If it exceeds a random number in the range of [0,1], the mutation is performed as shown in Eq. (3) below; where $\mathcal{N}$ elements from the pack of wolves are picked to control the mutation range within the search space.

$$\mathcal{N} = \max\left\{1, \left\lceil D - \left(\frac{t}{T}\right)^{\gamma} \times \mathcal{P}_m \right\rceil\right\} \tag{3}$$

Further, $\vec{C}$ is not linearly related to $\vec{A}$. This component provides random weights to prey in order to stochastically emphasize ($C>1$) or de-emphasize ($C<1$). Hence, to further increase the randomness of $\vec{C}$ at all times, this paper suggests the use of a statistical distribution as mentioned below [25]:

$$r_2 = r_2^{'} + \left[\alpha \times N(0,1)_t^3 \frac{(r_2^{\max} - r_2^{\min})}{t}\right] \tag{4}$$

where, r_2 is the random variable generated during the present iteration and $r_2^{'}$ is the random number generated during the previous iteration. The variables r_2^{max} and r_2^{min} are the upper and lower bounds on r_2 and the power of generating random number using $N(0,1)$ is set to 3 based on extensive numerical experimentation. This is helpful in avoiding the local optima stagnation especially during the final iterations.

B. Improved-CNN Variant

CNNs are widely utilized in the domain of image classification due to their limited pre-processing capability. This implies that, in contrast to classical algorithms involving manual intervention, a CNN evolves to learn the filters by itself analogous to classical algorithms involving manual intervention. Hence, this trait of CNN can be regarded as its major advantage over the existing schemes in addition to its ability to provide separation from the prior knowledge. However, the concept of "Dropout" plays an essential role in deep CNN as well as CNN in general. One of the serious issues with CNNs is overfitting, which is induced due to the large network logs (big data). Such networks make it difficult for CNN deep learning technique to learn the relevant features quickly. The main ideology behind dropout is to randomly dropout a few units and their respective connections from the network. This is done during the training phase so that the units do not co-adapt a lot. This is achieved by configuring the output of the hidden layers (with probability=0.5) to zero. The dropped neurons are thus eliminated from the process and do not contribute in back propagation.

Mathematically, the conventional dropout scheme can be understood using the below mentioned equations:

$$\begin{aligned} r^{(l)} &= Bernoulli(p) \\ \tilde{y}^{(l)} &= r^{(l)} * y^{(l)} \\ z_i^{(l+1)} &= w_i^{(l+1)} \tilde{y}^{(l)} + b_i^{(l+1)} \\ y_i^{(l+1)} &= f(z_i^{(l+1)}) \end{aligned}$$

In the above equations, the indices l and i denote the hidden layer and hidden units, respectively. Every layer l is associated with a vector of inputs and outputs which are represented using $z_i^{(l)}$ and $y_i^{(l)}$, respectively. The symbols $w_i^{(l)}$ and $b_i^{(l)}$ refer to the weights and biases associated with underlying layer l. In the conventional dropout approach, a vector of random numbers ($r^{(l)}$) is initially generated using Bernoulli distribution, which is then multiplied element-wise with $y_i^{(l)}$, to yield $\tilde{y}^{(l)}$ (thinned outputs). The obtained value of $\tilde{y}^{(l)}$ acts as input to the next layer, and is used to compute the value of $z_i^{(l+1)}$. This process is repeated for all the layers using the activator function f. It is worth noting here that the output of Bernoulli distribution is either '0' or '1'; which suggests that a particular hidden unit is either completely dropped or taken forward, respectively.

Unlike the conventional dropout, the proposed dropout approach is based on the uniform distribution. It can be viewed as the extension of the conventional dropout scheme which focuses on enhancing the weights of the relevant feature maps. In other words, the proposed dropout scheme not only abandons some of the units and connections from the network like conventional dropout, but also alters the weights of some of the units (which have respective weights below 0.5). This task not only helps avoid the over-fitting but also increases the weights of the most relevant features of the network. This in turn, simultaneously enhances the accuracy of the architecture while helping it converge faster. Mathematically, the overall scheme is presented by Algorithm 1. Initially, list $\mathcal{L}$ is initialized for all the hidden layers with weights less than or equal to 0.5 (Line 1). Then, vector r^l is initialized with the random numbers using the uniform distribution (Line 2). Following this, thinned outputs are computed using pair-wise multiplication of r^l and y^l, followed by z_i^{l+1}'s computation (Line 3-4). Finally, the outputs of the next layer, *i.e.*, $(l+1)$ are estimated and the process is repeated for all the layers in $\mathcal{L}$ (Line 5).

Algorithm 1 Modified dropout in ImCNN

```
1: Initialize List $\mathcal{L} = \{l_1, ...., l_m\}; \forall y^{(l)} <= 0.5$  ▷ Initialize the hidden layer list with weights $\leq 0.5$
2: Compute $r^{(l)} \in [0, 1]$ using uniform distribution  ▷ A vector of random numbers is computed using uniform distribution
3: Compute $\tilde{y}^{(l)} = r^{(l)} \times y^{(l)}; \forall l_i \in \mathcal{L}$  ▷ Thinned output computation for the $l^{th}$ hidden layer
4: Compute $z_i^{(l+1)} = w_i^{(l+1)} \tilde{y}^{(l)} + b_i^{(l+1)}; \forall l_i \in \mathcal{L}$  ▷ Input computation for the next $(l+1)^{th}$ hidden layer
5: Compute $y_i^{(l+1)} = f(z_i^{(l+1)}); \forall l_i \in \mathcal{L}$  ▷ Output computation for the next $(l+1)^{th}$ hidden layer
```

1) Complexity Analysis: The overall complexity of the algorithm is O(m); wherein m denotes the number of hidden layers employed in dropout.

IV. A ROBUST HYBRID MODEL FOR ANOMALY DETECTION

The hybrid model for network-wide anomaly detection works in two phases and the their detailed operation is provided in what follows.

A. Feature Selection using ImGWO

Since the performance of the classifier highly depends on the number of features (such as-source IP address and port number, destination IP address and port number, etc.), the problem consists of finding the most relevant features to maximize its performance. Let, $D = \{x_1, x_2, \cdots, x_n\}$ be a given dataset with n objects and $F = \{f_1, f_2, \cdots, f_m\}$ be the feature set with m number of features. Now, the feature selection process can be considered as a mapping of $S(D, F, A_{de}) \rightarrow F^{'}$, where $F(.)$ is the feature selection algorithm, A_{de} is the decisive attribute that represents class labels and $F^{'} \subset F$, where $|F^{'}| = k$ $(k<m)$ are the number of selected features. The aim of the proposed feature selection technique is to compute $F^{'}$ which are highly relevant to the dataset D as well as less related to each other.

In the proposed model, ImGWO is used to formulate the multi-objective feature selection problem; wherein the best solution for each wolf is to be determined from a set of potential non-dominated solutions. In this context, the fitness function of the participating wolf swarm is mathematically described below.

1) Fitness Function of Wolf Swarm: Feature selection in the context of network anomaly detection typically suffers from two major conflicting objectives: to minimize the number of features and to reduce the error rate of classification. Due to the presence of trade-offs between two or more conflicting objectives, optimal decisions becomes difficult. Thus, a single objective problem with several constraints may not be able to adequately represent this problem. In this case, it is mandatory to use multi-objective optimization which operates under a certain set of constraints in order to minimize or maximize the set of objective functions.

The proposed technique aims to compute a subset of features that yields the lowest Error Rate $(\mathcal{E})$ for the classifier. Several methods have been adopted to determine the classifier performance such as-Hamming loss, ranking loss, accuracy, *etc*. In order to evaluate the classification error rate of a grey wolf, this paper uses accuracy as an evaluation metric. The fitness function to minimize $(\mathcal{E})$ is given in Eq. (5). During the evolutionary training process, this function tests each possible subset of features to find the one which minimizes the classification error involved in feature selection.

$$\mathcal{E} = (FP + FN)/(TP + TN + FP + FN) \tag{5}$$

where FP, FN, TP and TN denote the *False Positive*, *False Negative*, *True Positive* and *True Negative* rates, respectively. These are typically real valued numbers in the range of [0, 100].

This is the basic fitness function which only considers the classification performance but does not take number of features into consideration. Thus, a multi-objective fitness function $(\mathcal{F})$ is used; where the first objective function $(\mathcal{F}_1)$ aims to minimize the classification error rate, whereas the second objective function $(\mathcal{F}_2)$ tends to minimize the number of features. This function is defined as [26]:

$$\mathcal{F} = \begin{cases} \text{Error Rate}(\mathcal{E}) & (\mathcal{F}_1) \\ \gamma \times \frac{\#F^{'}}{\#F} + (1-\gamma) \times \frac{\mathcal{E}_c^{F^{'}}}{\mathcal{E}_c^{F}} & (\mathcal{F}_2) \end{cases} \tag{6}$$

The above defined fitness function is expected to ensure the minimization of the number of features while maintaining a high classification performance. In the defined function, γ is any constant value lying between [0,1], $F^{'}$ denotes the number of selected features, F represents the total number of available features, $\mathcal{E}_c^{F^{'}}$ is the classification error rate involved in selecting the feature-set and $\mathcal{E}_c^{F}$ represents the error rate involved by using all the available features for classification.

The detailed operation of the ImGWO for feature selection in the context of network anomaly detection is illustrated by Algorithm 2. During Step 1, different parameters such as-pop, T, F and pos are initialized (Lines 2-6). In Step 2, the initial population is generated using uniform distribution as discussed above. Along with this, the coefficient vectors ($\vec{A}$ and $\vec{C}$) and the random vectors ($\vec{r}_1$ and $\vec{r}_2$) are initialized (Lines 7-11). Following this, fitness functions are calculated for all the wolves to determine the optimal solution for the considered problem. Based on the obtained fitness values, the participating wolves are categorized into α, β and δ. The rest of the wolves are marked as ω which follow α, β and δ (Lines 14-16). Finally, the process of improved exploration and exploitation capability as discussed above is repeated, till an optimal solution to the problem is reached (Lines 17-33).

2) Complexity Analysis: The overall complexity of the proposed algorithm was found to be O(mnp); wherein variables m, n and p refer to the number of iterations, population size and number of features in the dataset, respectively. The proposed ImGWO gives better convergence with improved exploration, exploitation and initial population generation abilities than its standard counterpart.

B. Anomaly Detection using ImCNN

The structure of the ImCNN used in the proposed model for effective anomaly classification is described as under.

The logical structure of the proposed ImCNN is described using Fig. 2. As shown in the figure, the ImCNN architecture comprises of 8 layers, namely 4 convolutional layers ($conv$), 2 sub-sampling layers ($samp$), 1 dropout layer ($drop$) and 1 fully connected layer ($conn$). The sequence of the layers is as under: $conv_1, conv_2, samp_1, conv_3, conv_4, samp_2, drop_1$, and $conn_1$. The detailed operation of these layers is provided in what follows.

The images acquired from streaming data traffic serve as the input to the ImCNN with the size of $32 \times 32 \times 3$; wherein the spatial dimension are represented using 32×32 pixels and the channel dimensions are fixed to 3. In the first layer, *i.e.*, $conv_1$ a 2D kernel of size 3×3 is applied to the input. Here, a 2D kernel is utilized to extract the relevant feature set. Since, a single kernel is capable of extracting a single feature, thus a total of 12 2D kernels are applied on the data set to generate a holistic feature map of 12 size in the very first layer. Moreover, a total of 6 2D kernel of dimensions 2×2 are utilized as part of the $conv_2$ layer. Subsequently, with an aim to reduce the spatial resolution, sub-sampling is carried out in the next layer, *i.e.*, $samp_1$. This layer helps to enhance the robustness of even the minute spatial distortions. Here, the sampling is performed with the factor of 2×2 which doesn't affect the size of the feature map.

In order to generate a more optimized feature map, another layer of convolution, *i.e.*, $conv_3$ is utilized with the 2D kernel (3×3). This layer in turns generates a set of 3 feature maps. Finally, another convolution layer ($conv_4$) for deep feature identification is employed next. Like the previous layer, the same kernel is used in this layer producing a total of 3 feature maps. Subsequently, sub-sampling is performed on the data as part of $samp_2$ layer; without affecting the size of the feature map. Finally, the modified dropout approach as discussed in Section III-B is carried out as part of $drop_1$ layer; wherein the ImCNN tends to learn the robust features of the underlying network. In the next layer, the proposed ImCNN tends to learn high-level features of the input datasets using convolution in $conn_1$. It is a fully-connected layer which utilizes 3D kernel (size = $5 \times 5 \times 3$), reducing the feature

Algorithm 2 *ImGWO*: Proposed Feature Selection Technique

```
Input: Dataset $D$.
Output: Optimal feature subset $F^{'}$.
 1: procedure FUNCTION(ImGWO)
 2:     Step 1: Initializing Parameters
 3:     pop: size of population
 4:     T: maximum number of iterations
 5:     F: total number of features
 6:     pos: position of grey wolf
 7:     Step 2: Initial Population Generation
 8:     Generate the initial population using uniform distribution
 9:     Initialize $\vec{A}$
10:     Compute $r_2$ using Eq. (4)
11:     Initialize $\vec{C}$ using $r_2$
12:     Step 3: Fitness Function Calculation
13:     Calculate the fitness function $(\mathcal{F})$ of grey wolves using Eq. (6)
14:     Set $\alpha$=the grey wolf with maximum fitness
15:     Set $\beta$=the grey wolf with second maximum fitness
16:     Set $\delta$=the grey wolf with third maximum fitness
17:     while $t<T$ do
18:         for $i = 1$ to $pop$ do
19:             Update the $pos$ of the current grey wolf
20:         end for
21:         for $i = 1$ to $pop$ do
22:             for $j = 1$ to $F$ do
23:                 Compute $\mathcal{P}_m$ using Eq. (2)
24:                 if $\mathcal{P}_m>r_1$ then
25:                     Calculate $\mathcal{N}$ using Eq. (3)
26:                     Set $F^{'} = \{f_1, f_2, \cdots, f_{\mathcal{N}}\}$
27:                     for $k = 1$ to $\mathcal{N}$ do
28:                         Re-initialize the $k^{th}$ feature of the grey wolf
29:                     end for
30:                 end if
31:             end for
32:         end for
33:     end while
34: end procedure
```

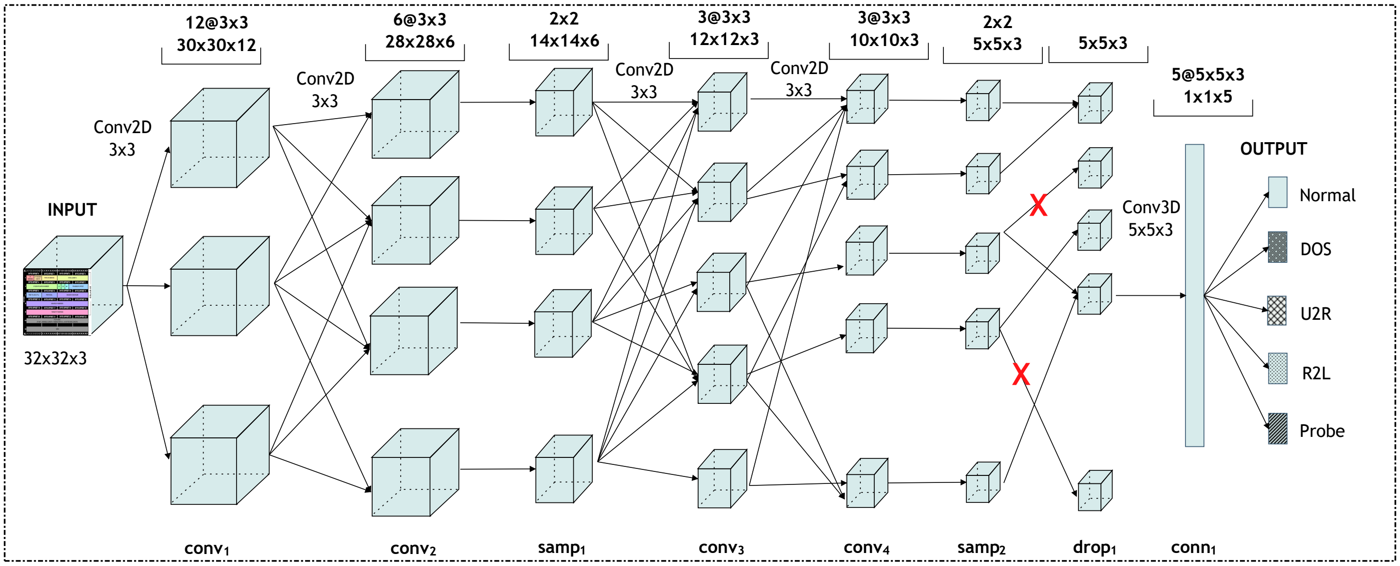


Fig. 2: The architecture of the ImCNN used for network anomaly detection for streaming data in cloud setup.

map to $1 \times 1 \times 5$ size. The number of outputs of this layer is 5 which corresponds to different classes of normal and anomalous traffic streams namely-normal, DoS, U2R, R2L and Probe. This output classifies the traffic stream into the above classes with a definite probability which is chosen in accordance with the benchmark datasets [27], [28].

V. Numerical Simulation Results

This section demonstrates the performance of the proposed model compared to the current state-of-the-art schemes for network anomaly identification. It is implemented using i3-6100U CPU @ 2.30 GHz with 4 GB of RAM on MATLAB R2016a. For the extensive evaluation of the proposed model, three sets of case studies have been considered which measure the performance of the proposed model on different datasets, *i.e.*, benchmark and synthetic datasets.

A. Evaluation metrics

In order to evaluate the performance of the proposed model, the following parameters are used: Detection Rate (DR) or recall, False Positive Rate (FPR), precision, accuracy and F-score [7], [24]. The mathematical derivation of these parameters is illustrated using the below equations.

$$\text{DR (Recall)} = \frac{TP}{TP + FN}$$

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

$$\text{Precision} = \frac{TP}{TP + FP}$$

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

$$\text{F-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

In the above equations, the parameters TP, TN, FP and FN refer to *True Positive*, *True Negative*, *False Positive* and *False Negative*, respectively. TP refers to the case when the considered class (network traffic in our case) is actually normal and is classified as normal. On the similar lines, an anomalous class classified as normal is referred to as FP. On the contrary, a normal class may be classified as anomalous, while an anomalous class may be predicted anomalous. These cases are respectively ascribed as TN and FN.

TABLE II: Illustration of confusion matrix

Predicated class Actual class	Anomaly class	Normal class
Anomaly class	TN	FP
Normal class	FN	TP

B. Datasets Used

1) Benchmark dataset-DARPA'98: The benchmark dataset used for evaluation purpose is acquired from Defense Advanced Research Projects Agency (DARPA) comprising of 58 features [27]. This benchmark dataset is widely accepted and is used for network anomaly detection. It comprises of 4 set of files namely-tcpdump files, tcpdump list files, Solaris BSM audit data files, and ps monitoring data files. These files contain the network traffic log information, however, amongst these files only the tcpdump files contain the traffic log information pertaining to cloud environment. Hence, the tcpdump files are used for evaluating the performance of the proposed hybrid model. Moreover, this raw data (in the form of bytes/packets from tcp dump file) is converted into images for evaluation purposes during the preprocessing phase as explained in Section IV.

2) Benchmark dataset-KDD'99: KDD Cup 1990 is a benchmark data that is acquired from UCI machine learning repository for Case study-II [28]. It comprises of nearly 5 million records and a total of 41 features. Like DARPA'98 dataset, the traffic in this dataset can also be classified into 5 classes namely-normal, DoS, U2R, R2L and Probe.

3) Synthetic dataset: In oder to perform a more comprehensive evaluation of the proposed model, a simulated environment has been set up to generate synthetic network traffic streams. For this purpose, two machines were setup, wherein the first machine was a typical Windows PC, while the other was a dummy server. On the former machine, different kinds of malicious files were executed to generate the anomalous traffic, while on the later machine INetSim2 was used to set an imitation of Internet. The main advantage of using INetSim2 is that it can be used to generate common Internet services data (HTTP, SMTP, DNS, FTP, *etc.*). Subsequently, the generated data from the Windows PC is sent to the server, to which the server responds back with the appropriate queries. The communication between the two machines carries both the anomalous and benign traffic and the same has been employed for the performance evaluation of the proposed model. The anomalous traffic was injected into the traffic stream for following attack vectors: DoS, Generic, Shell code and CLET [29]. Hence, the generated synthetic traffic streams can be classified into 2 classes, *i.e.*, normal and anomalous.

C. Results & Comparisons

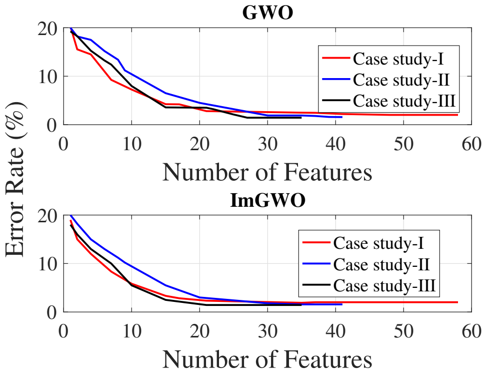
For the extensive evaluation of the proposed model, three case studies were taken into account. These case studies evaluate the performance of the model on different datasets, *i.e.*, *Case study-I* on DARPA'98 dataset, *Case study-II* on KDD'99 dataset and *Case study-III* on synthetic dataset. The results obtained are highlighted in Figs. 3 and 4 respectively.

For the sake of clarity, the obtained results are illustrated in two parts namely-*for ImGWO* and *for proposed hybrid model (ImGWO+ImCNN)*. The relative comparison of the former was carried out against the standard GWO; while the latter was compared with the hybrid combination of GWO and CNN (GWO+CNN). Their detailed description is as follows.

1) For ImGWO: ImGWO was used for the optimal feature set selection from dataset. In the considered case studies, ImGWO was able to attain optimal results as shown in Fig. 3. The trade-off between the competing functions, *i.e.*, number of features and error rate is depicted in the figure. It is evident from the figure that ImGWO leads to improved feature set selection while minimizing the error rate relative to the standard GWO. A total of 37, 34 and 21 features were selected out of 58, 41 and 35 in Case study-I, II and III, respectively, by ImGWO. For instance, important features like the duration of the connection, the number of bytes transferred from sources to destination, the number of bytes transferred from destination to sources, the number of failed logins, protocol type, the status of connection, the number of failed login attempts, the number of compromised conditions, etc. were selected by ImGWO.

2) For proposed hybrid model (ImGWO+ImCNN): The performance evaluation of the proposed hybrid model for network anomaly detection across the considered case studies is discussed as under. In total, 6 parameters have been used for the evaluation purpose of the ImCNN architecture for anomaly detection.

The obtained results in terms of Case study-I are detailed as under. Fig. 4a depicts the high DR achieved by the proposed hybrid model corresponding to normal and anomalous classes (DoS, U2R, R2L and Probe attacks) on DARPA'98 benchmark dataset. The FPR corresponding to the considered set of classes is depicted in Fig. 4b. It is evident from the figure that the proposed model yields FPR values as low as 4.167, 3.448, 3.846, 3.846, 2.703 with respect to different classes. Fig. 4c indicates the proposed model's precision in achieving the desired results. It achieves high precision in detecting both normal (99.98) and anomalous classes (99.98, 99.93, 99.93, 99.98). Similarly, evaluation results with respect to accuracy, F-score and ROC curves are shown in Figs. 4d, 4e and 4f, respectively. The results clearly indicate good performance of the proposed model across all the parameters relative to its existing counterpart, *i.e.*, GWO+CNN.



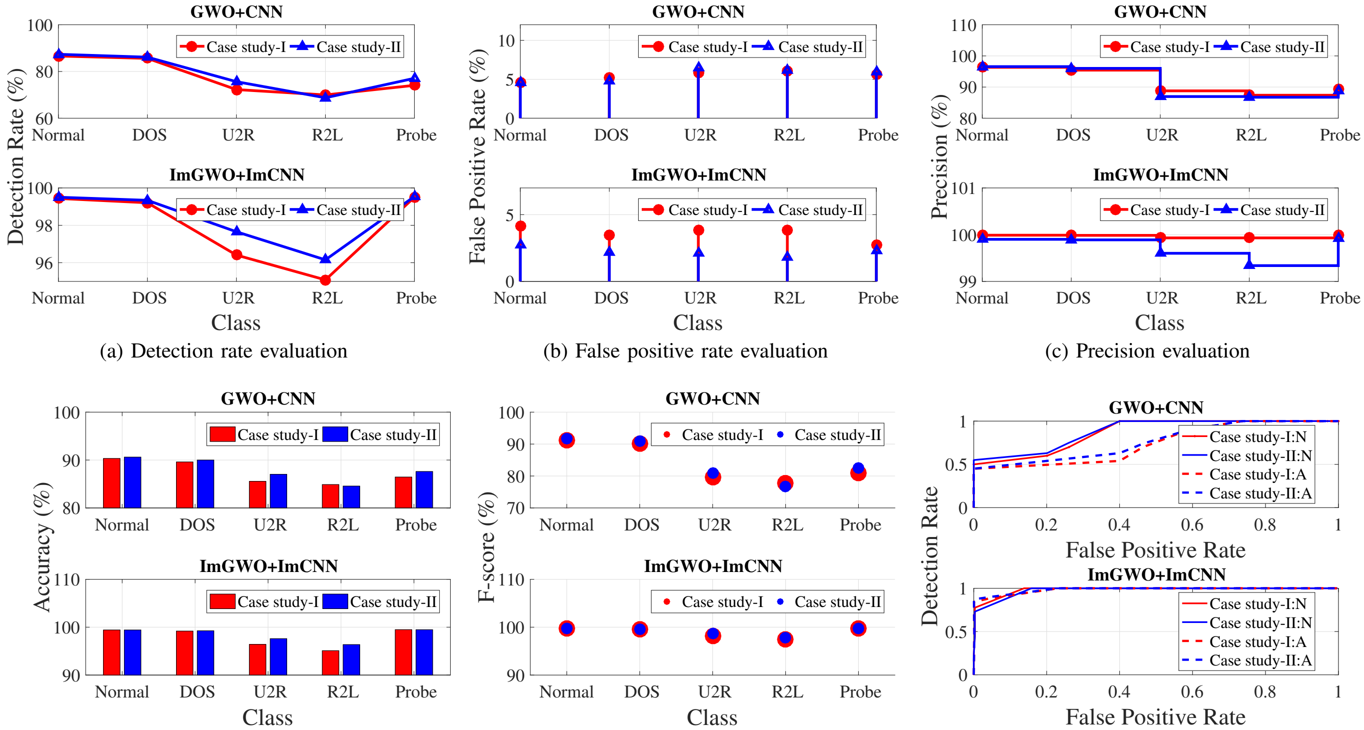
(a) Pareto front for the feature selection phase

Fig. 3: Performance evaluation of the proposed ImGWO on benchmark & synthetic datasets.

Next, we illustrate the performance of the ImCNN architecture on KDD'99 dataset for Case study-II. The obtained results are also depicted in Fig. 4. The proposed model is found to be effective enough to achieve higher DR, precision and accuracy in comparison with Case study I and the same is evident from the results depicted in Figs. 4a, 4c and 4d respectively. Moreover, the proposed model achieves FPR values as low as 2.70, 2.20, 2.10, 1.80 and 2.30 in detecting normal, DoS, U2R, R2L and probe attack classes. Figs. 4e and 4f depict the F-score and ROC curves which clearly indicate the capability of the model of achieving satisfactory performance. Overall, the proposed model is found to perform better on KDD'99 dataset relative to the DARPA'98 dataset. Further, during this case study as well, the proposed scheme performs better than the combination of GWO+CNN as indicative from the results (shown in Fig. 4).

The evaluation results for Case study-III are depicted in Figs. 5a and 5b. The results clearly indicate that the proposed scheme achieves quality results even in case of the synthetic dataset. High DR, precision, accuracy and F-score with low FPR are an indicative of the performance of the proposed scheme on synthetic dataset. The related results are highlighted in Fig. 5a. The corresponding ROC evaluation for this case study are summarized in Fig. 5b. The obtained results imply that the proposed model is efficient enough to be implemented in real-time.

In addition to this, the timing analysis of the proposed



(a) Detection rate evaluation

(b) False positive rate evaluation

(c) Precision evaluation

(d) Accuracy evaluation

(e) F-score evaluation

(f) ROC evaluation

Fig. 4: Performance evaluation of the proposed hybrid model on benchmark datasets.

scheme across all the datasets is depicted in Fig. 6. The obtained results indicate that the proposed model executes in a reasonable amount of time across all the case studies considered. The obtained results are indicative of the fact that the proposed approach is reasonably fast and its performance doesn't fluctuate much with the change in datasets. Contrastingly, the existing scheme based on the amalgamation of conventional GWO and CNN, requires greater execution time. Additionally, the choice of the dataset also affects its execution time adversely. On average, the proposed model exhibits an overall improvement of 8.25%, 4.08% and 3.62% in terms of DR, FPR, and accuracy, respectively.

3) Comparison with the existing schemes: The detailed comparison of the proposed model with the current state-of-the-art techniques [9], [30], [31], [32], [33], [34] is depicted in Table-III. As evident from the table, the results obtained by the proposed model show an indicative improvement over the existing schemes. For instance, the proposed model performs far better than the existing schemes in terms of FPR, accuracy, and F-score for DARPA'98 dataset, and in terms of DR and F-score for KDD'99 dataset.

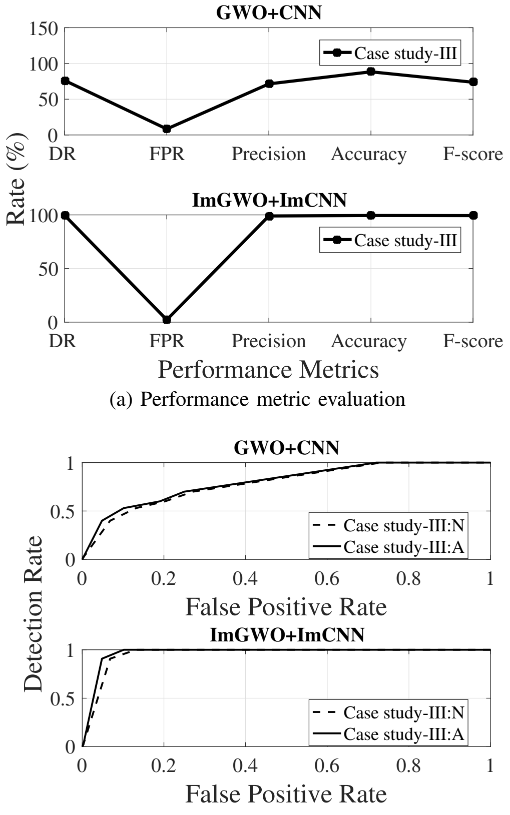
TABLE III: Performance comparison of the proposed model with the state-of-the-art techniques.

DARPA'98 Dataset				
Technique	**DR(%)**	**FPR(%)**	**Accuracy(%)**	**F-score(%)**
Elfeshawy *et al.* [30]	98.43	4.6	95.39	–
Ahmed *et al.* [31]	99.23	–	92.82	96
David & Thomas [33]	98	–	99.5	–
Proposed Model	98.62	3.60	97.92	98.92
KDD'99 Dataset				
Technique	**DR(%)**	**FPR(%)**	**Accuracy(%)**	**F-score(%)**
Sharma *et al.* [32]	93.41	0.275	99.05	93
Pandeeshwari *et al.* [9]	98	3.05	–	83.20
Guo *et al.* [34]	91.86	0.78	93.29	–
Proposed Model	98.72	2.22	98.42	99.07

VI. Conclusion

This work presents a robust hybrid model for network anomaly detection in cloud environments, particularly for streaming data. The model leverages the advantages of multi-objective optimization and deep learning, particularly for feature extraction and anomaly detection on real-time network traffic streams. For this purpose, two computationally efficient techniques were employed namely-GWO and CNN. The amalgamation of these techniques is further improved by revamping their respective standard strategies. For instance, GWO is improvised with respect to enhance initial population, exploration and exploitation capabilities, while CNN is modified in terms of dropout layer functionality. Additionally, the proposed hybrid model was extensively evaluated on benchmark and synthetic datasets. The results obtained clearly indicate the supremacy of the proposed model relative to the existing models.

In the future, we will extend the present work for malware detection, particularly for cloud environments. The inherent complexity in the cloud environment is induced due to the heterogeneity of incoming traffic and underlying hardware; which makes the task of anomaly detection more cumbersome.



(a) Performance metric evaluation

(b) ROC evaluation

Fig. 5: Performance evaluation of the proposed hybrid model on synthetic dataset.

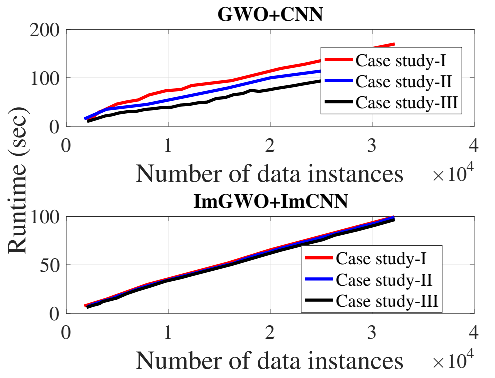


Fig. 6: An illustration of the timing analysis

References

[1] (2016, June) Market Insight: Cloud Computing's Drive to Digital Business Creates Opportunities for Providers. Gartner. [Online]. Available: http://www.gartner.com/newsroom/id/3354117

[2] S. Garg, A. Singh, K. Kaur, G. S. Aujla, S. Batra, N. Kumar, and M. S. Obaidat, "Edge Computing-Based Security Framework for Big Data Analytics in VANETs," *IEEE Network*, vol. 33, no. 2, pp. 72–81, 2019.

[3] L. Columbus. (2017) State Of Cloud Adoption And Security. Forbes. [Online]. Available: https://www.forbes.com/sites/louiscolumbus/2017/04/23/2017-state-of-cloud-adoption-and-security/#4337a38a1848

[4] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in cloud," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 42–57, 2013.

[5] P. Mishra, E. S. Pilli, V. Varadharajan, and U. Tupakula, "Intrusion detection techniques in cloud environment: A survey," *Journal of Network and Computer Applications*, vol. 77, pp. 18–47, 2017.

[6] S. Garg, K. Kaur, G. Kaddoum, S. H. Ahmed, and D. N. K. Jayakody, "SDN based Secure and Privacy-preserving Scheme for Vehicular Networks: A 5G Perspective," *IEEE Transactions on Vehicular Technology*, 2019, DOI:10.1109/TVT.2019.2917776.

[7] S. Garg and S. Batra, "A novel ensembled technique for anomaly detection," *International Journal of Communication Systems*, vol. 30, no. 11, p. e3248, 2017.

[8] ——, "Fuzzified cuckoo based clustering technique for network anomaly detection," *Computers & Electrical Engineering*, vol. 71, pp. 798–817, 2018.

[9] N. Pandeeswari and G. Kumar, "Anomaly detection system in cloud environment using fuzzy clustering based ANN," *Mobile Networks and Applications*, vol. 21, no. 3, pp. 494–505, 2016.

[10] M. R. Watson, A. K. Marnerides, A. Mauthe, D. Hutchison *et al.*, "Malware detection in cloud computing infrastructures," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 2, pp. 192–205, 2016.

[11] X. Ye, X. Chen, H. Wang, X. Zeng, G. Shao, X. Yin, and C. Xu, "An anomalous behavior detection model in cloud computing," *Tsinghua Science and Technology*, vol. 21, no. 3, pp. 322–332, 2016.

[12] W. Sha, Y. Zhu, M. Chen, and T. Huang, "Statistical Learning for Anomaly Detection in Cloud Server Systems: A Multi-Order Markov Chain Framework," *IEEE Transactions on Cloud Computing*, vol. 6, no. 2, pp. 401–413, 2018.

[13] Z. Tan, A. Jamdagni, X. He, P. Nanda, and R. P. Liu, "A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 2, pp. 447–456, 2014.

[14] S. Garg, A. Singh, S. Batra, N. Kumar, and L. T. Yang, "UAV-Empowered Edge Computing Environment for Cyber-Threat Detection in Smart Vehicles," *IEEE Network*, vol. 32, no. 3, pp. 42–51, 2018.

[15] Z. X. Yang, X. L. Qin, W. R. Li, and Y. J. Yang, "A DDoS detection approach based on CNN in cloud computing," in *Applied Mechanics and Materials*, vol. 513. Trans Tech Publ, 2014, pp. 579–584.

[16] S. Garg, K. Kaur, N. Kumar, S. Batra, and M. S. Obaidat, "HyClass: Hybrid Classification Model for Anomaly Detection in Cloud Environment," in *IEEE International Conference on Communications (ICC), Kansas City, USA*, May 2018.

[17] S. Garg, A. Singh, S. Batra, N. Kumar, and M. S. Obaidat, "EnClass: Ensemble-Based Classification Model for Network Anomaly Detection in Massive Datasets," in *IEEE Global Communications Conference (GLOBECOM'17), Singapore*, Dec 2017.

[18] S. Garg, K. Kaur, N. Kumar, and J. J. P. C. Rodrigues, "Hybrid Deep-Learning-Based Anomaly Detection Scheme for Suspicious Flow Detection in SDN: A Social Multimedia Perspective," *IEEE Transactions on Multimedia*, vol. 21, no. 3, pp. 566–578, 2019.

[19] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[20] H. Yang and Z. Zhou, "A novel intrusion detection scheme using Cloud Grey Wolf Optimizer," in *2018 37th Chinese Control Conference (CCC)*. IEEE, 2018, pp. 8297–8302.

[21] B. Mao, F. Tang, Z. M. Fadlullah, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "A Novel Non-Supervised Deep-Learning-Based Network Traffic Control Method for Software Defined Wireless Networks," *IEEE Wireless Communications*, vol. 25, no. 4, pp. 74–81, 2018.

[22] W. Ji, S. Duan, R. Chen, S. Wang, and Q. Ling, "A CNN-based network failure prediction method with logs," in *2018 Chinese Control And Decision Conference (CCDC)*. IEEE, 2018, pp. 4087–4090.

[23] Y. Xiao, C. Xing, T. Zhang, and Z. Zhao, "An intrusion detection model based on feature reduction and convolutional neural networks," *IEEE Access*, 2019, DOI: 10.1109/ACCESS.2019.2904620.

[24] H. Garg, "A hybrid PSO-GA algorithm for constrained optimization problems," *Applied Mathematics and Computation*, vol. 274, pp. 292–305, 2016.

[25] B. Xing and W.-J. Gao, *Innovative computational intelligence: a rough guide to 134 clever algorithms*. Springer Science & Business Media, 2013, vol. 62.

[26] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimization for feature selection in classification: A multi-objective approach," *IEEE transactions on cybernetics*, vol. 43, no. 6, pp. 1656–1671, 2013.

[27] "Intrusion detection dataset," DARPA, 1998, [Accessed on: Mar. 2017]. [Online]. Available: https://www.ll.mit.edu/ideval/data/1998data.html

[28] "KDD Cup. dataset," 1999, [Accessed on: Mar. 2017]. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[29] R. Perdisci, D. Ariu, P. Fogla, G. Giacinto, and W. Lee, "McPAD: A multiple classifier system for accurate payload-based anomaly detection," *Computer networks*, vol. 53, no. 6, pp. 864–881, 2009.

[30] N. A. Elfeshawy and O. S. Faragallah, "Divided two-part adaptive intrusion detection system," *Wireless networks*, vol. 19, no. 3, pp. 301–321, 2013.

[31] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, 2016.

[32] N. Sharma and S. Mukherjee, "A novel multi-classifier layered approach to improve minority attack detection in IDS," *Procedia Technology*, vol. 6, pp. 913–921, 2012.

[33] J. David and C. Thomas, "Efficient DDoS flood attack detection using dynamic thresholding on flow-based network traffic," *Computers & Security*, vol. 82, pp. 284–295, 2019.

[34] C. Guo, Y. Ping, N. Liu, and S.-S. Luo, "A two-level hybrid approach for intrusion detection," *Neurocomputing*, vol. 214, pp. 391–400, 2016.

Sahil Garg (S'15, M'18) received his B.Tech degree from Maharishi Markandeshwar University, Mullana, India, in 2012; his M.Tech degree from Punjab Technical University, Jalandhar, India in 2014; and his Ph.D. from Thapar Institute of Engineering & Technology (Deemed to be University), Patiala, India, in 2018, all in computer science and engineering. He is currently working as a Postdoctoral Research Fellow with Department of Electrical Engineering, École de technologie supérieure, Université du Québec, Montréal, Canada. He has many research contributions in the area of Machine Learning, Big Data Analytics, Knowledge Discovery, Cloud Computing, Internet of Things, and Vehicular Ad-hoc Networks. Some of his research findings are published in top-cited journals such as IEEE TII, IEEE TMM, IEEE TVT, IEEE TSUSC, IEEE IoT Journal, IEEE Systems Journal, IEEE Communications Magazine, IEEE Network Magazine, IEEE Wireless Communications, IEEE Consumer Electronics Magazine, FGCS, Information Sciences, and CAEE including various International conferences of repute such as-IEEE Globecom, IEEE ICC, IEEE WCNC, IEEE VTC, IEEE Infocom Workshops, ACM MobiCom Workshops, ACM MobiHoc Workshops, etc. He was the recipient of prestigious Visvesvaraya PhD fellowship from the Ministry of Electronics & Information Technology under Government of India (2016-2018). For his research, he also received the IEEE ICC best paper award in 2018 at Kansas City, USA. He is a member of IEEE, IEEE ComSoc, IEEE Computer, IEEE IES, IEEE Smart Grid Community, ACM and IAENG.

Kuljeet Kaur (S'13, M'18) received the B.Tech degree in computer science and engineering from Punjab Technical University, Jalandhar, India, in 2011 and the M.E. (Information Security) and PhD (Computer Science and Engineering) degrees from Thapar Institute of Engineering and Technology (Deemed to be University), Patiala, India, in 2015 and 2018, respectively. She is currently working as a NSERC Postdoctoral Research Fellow with Department of Electrical Engineering, École de technologie supérieure, Université du Québec, Montréal, Canada. Her main research interests include Cloud Computing, Energy Efficiency, Smart Grid, Frequency Support, and Vehicle-to-Grid. Dr. Kaur has secured a number of research articles in top-tier journals such as IEEE Wireless Communications, IEEE TII, IEEE TVT, IEEE TSG, IEEE Sensors Journal, IEEE Communications Magazine, IEEE TMM, IEEE TSUSC, IEEE PS, Springer PPNA, etc., and various International conferences including IEEE Globecom, IEEE ICC, IEEE PES GM, IEEE WCNC, IEEE Infocom Workshops, ACM Mobicom Workshops, ACM MobiHoc Workshops, etc. During her PhD, she received two prestigious fellowships including INSPIRE fellowship from Department of Science & Technology, India (in 2015) and research scholarship from Tata Consultancy Services (TCS) (from 2016-2018). Dr. Kaur also received the IEEE ICC best paper award in 2018 at Kansas City, USA. She is a member of IEEE, IEEE Communications Society, IEEE Computer, IEEE Women in Engineering, IEEE Software Defined Networks Community, IEEE Smart Grid Community, ACM and IAENG.

Neeraj Kumar (M'16, SM'17) received his Ph.D. in CSE from Shri Mata Vaishno Devi University, Katra (J & K), India, and was a postdoctoral research fellow in Coventry University, Coventry, UK. He is working as an Associate Professor in the Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology (Deemed to be University), Patiala (Pb.), India. He has published more than 200 technical research papers in leading journals and conferences from IEEE, Elsevier, Springer, John Wiley etc. Some of his research findings are published in top cited journals such as IEEE TIE, IEEE TDSC, IEEE TITS, IEEE TCE, IEEE TII, IEEE TVT, IEEE ITS, IEEE Netw., IEEE Comm., IEEE WC, IEEE IoTJ, IEEE SJ, FGCS, JNCA, and ComCom. He has guided many research scholars leading to Ph.D. and M.E./M.Tech. His research is supported by funding from UGC, DST, CSIR, and TCS. He is an Associate Technical Editor of IEEE Communication Magazine and an Associate Editor of IJCS, Wiley, JNCA, Elsevier, and Security & Communication, Wiley. He is senior member of the IEEE.

Georges Kaddoum (M'11) received the Bachelor's degree in electrical engineering from the École Nationale Supérieure de Techniques Avancés (ENSTA Bretagne), Brest, France, and the M.S. degree in telecommunications and signal processing(circuits, systems, and signal processing) from the Université de Bretagne Occidentale and Telecom Bretagne (ENSTB), Brest, in 2005 and the Ph.D. degree (with honors) in signal processing and telecommunications from the National Institute of Applied Sciences (INSA), University of Toulouse, Toulouse, France, in 2009. He is currently an Associate Professor and Tier 2 Canada Research Chair with the École de Technologie Supérieure (ÉTS), Université du Québec, Montréal, Canada. He was awarded the ÉTS Research Chair in physical-layer security for wireless networks in 2014, and the prestigious Tier 2 Canada Research Chair in wireless IoT networks in 2019. Since 2010, he has been a Scientific Consultant in the field of space and wireless telecommunications for several US and Canadian companies. He has published over 150+ journal and conference papers and has two pending patents. His recent research activities cover mobile communication systems, modulations, security, and space communications and navigation. Dr. Kaddoum received the Best Papers Awards at the 2014 IEEE International Conference on Wireless and Mobile Computing, Networking, Communications (WIMOB), with three coauthors, and at the 2017 IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), with four coauthors. Moreover, he received IEEE Transactions on Communications Exemplary Reviewer Award for the year 2015 and 2017. In addition, he received the research excellence award of the Université du Québec in the year 2018. In the year 2019, he received the research excellence award from the ÉTS in recognition of his outstanding research outcomes. Prof. Kaddoum is currently serving as an Associate Editor for IEEE Transactions on Information Forensics and Security, and IEEE Communications Letters.

Albert Y. Zomaya (M'90, SM'97, F'04) is currently the Chair Professor of High Performance Computing & Networking in the School of Information Technologies, The University of Sydney. He also serves as the Director of the Centre for Distributed and High Performance Computing. Professor Zomaya published more than 550 scientific papers and articles and is author, co-author or editor of more than 20 books. He is the Founding Editor in Chief of the IEEE Transactions on Sustainable Computing and serves as an associate editor for more than 20 leading journals. Professor Zomaya served as an Editor in Chief for the IEEE Transactions on Computers (2011-2014). He is the recipient of the IEEE Technical Committee on Parallel Processing Outstanding Service Award (2011), the IEEE Technical Committee on Scalable Computing Medal for Excellence in Scalable Computing (2011), and the IEEE Computer Society Technical Achievement Award (2014), and the ACM MSWIM Reginald A. Fessenden Award (2017). He is a Chartered Engineer, a Fellow of AAAS, IEEE, and IET. Professor Zomaya's research interests are in the areas of parallel and distributed computing and complex systems.

Rajiv Ranjan (SM'15) is a Chair Professor in Computing Science and Internet of Things at Newcastle University, United Kingdom. He has received two IEEE research excellence awards (2018 IEEE TCCPS Early Career Award and 2016 IEEE TCSC Award for Excellence in Scalable Computing), which recognised his leading expertise in algorithms, resource management models and distributed system architectures for Cloud computing, Internet of Things (IoT) and Data Science. Another testimonial of his international research leadership is his appointment by IEEE Computer Society as the Advisory Board Chair and Lead Editor (2014-2019) for the Blue Skies department of IEEE Cloud Computing. In this appointment, Prof Ranjan's main role is to develop a vision for the research community to guide future research at the intersection of Cloud computing, IoT and Data Science. Additionally, he also serves on the editorial boards of top quality international journals including IEEE Transactions on Cloud computing, ACM Transactions on Internet of Things, IEEE Transactions on Computers (2014-2016), IEEE Cloud Computing, Springer Computing, The Computer Journal (Oxford University Press), among many others. His research outcomes include 240+ academic peer-reviewed articles and multiple open source software toolkits – stemming from funded research projects worth over $12 Million AUD (£6 Million GBP). He is one of the highly cited authors (top 0.05%) in computer science and software engineering worldwide (h-index=46, g-index=121, and 12700+ google scholar citations).