IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 20, Issue 6, Ver. II (Nov - Dec 2018), PP 01-07
www.iosrjournals.org

Enhanced Detection Rate for Network Intrusion Detection System by Using Chaotic Firefly Algorithm

Dr. Sefer Kurnaz[1], Rafid Faeq Ahmed[2]
Corresponding Author: Dr. Sefer Kurnaz

***Abstract**: - Regarding the security of computer systems, the intrusion detection systems (IDS) are essential components for the detection of attacks at the early stage.*
Therefore, the main goal of this thesis is to choose the differentiating features the development of an optimal machine learning algorithm with respect to high detection rates, fast training and testing processes. So a proposed machine learning model containing a feature selection algorithm (wrapper type) based on the integration of Firefly algorithm (FA) with Naïve Bayesian Classifier (NBC) were proposed. 1999 KDDCUP and NSL- KDD data sets .
The proposed model been developed and tested over two types of feature selection objectives single objective fitness function (accuracy) and multiple objective fitness function (accuracy and number of features).
***Keywords:-** Security , Intrusion detection systems, Training , Testing , Feature selection algorithm , Firefly algorithm , Naïve Bayesian Classifier , Accuracy.*

Date of Submission: 04-11-2018 Date of acceptance: 18-11-2018

I. Introduction

Intrusion detection is classification task, consisting of developing a predictive model with the capability of identifying attack instances.

Intrusion detection involves the discovering and detection of network events or traffic on host machines which behaves abnormally or violating the network regulations. It helps in the analysis and monitoring of the daily activities within the computer systems to detect the presence of security threats. Meanwhile, the intrusion techniques are posing several security challenges to the security tools due to their sophisticated nature. Thus, an efficient and reliable IDS that will protect computer networks from all forms of attack is necessary

II. Related Work

Many studies have been reported on the accuracy of classification enhancement problems. Most of the related works are listed below.

The PSO has been merged with SVM for feature selection using the one-versus-rest approach as a classifier to compute the fitness for the PSO [37]. In this experiment, the standard dataset KDD Cup 1999 intrusion detection dataset was used. The results showed an accuracy of (96.11%).

Mukherjeea and Sharmaa have achieved a significantly reduced input features using three methods of feature selection which are Correlation-based feature selection Gain ratio, and Information Gain.

They used the NB data mining algorithm to reduce the data set for the detection of intrusion. The

empirical results show that the selected reduced attributes performed better in terms of the accuracy. The Correlation-based feature selection, Gain ratio, and Information Gain recorded accuracies of 96.5 %, 95.2 %, and 94.5 %, respectively [38].

Arif and Farrukh [39] suggested the binary version of multi objective PSO (MPSO) approach for the detection of network PROBE attacks. Two objectives were used with the PSO approach (i.e., the rate of intrusion detection and rate of false detection) to guide the F process The experiments were carried out using the original KDD Cup 1999 dataset. MPSO was used for FS and Random Forests (RF) was used for the classification task. The classifier achieved a higher IDR of 90.7% while the suggested technique achieved an IDR of 96.66%.

A new wrapper features selection based optimization method named Linear Genetic Programming and Bees Algorithm (LGP_BA) has been proposed by [36] to gain an efficient feature selection algorithm. The modify LGP was used to produce first candidate chromosome and then BA applied neighborhood search to perform the adjustment. In this paper, three sample datasets from the NSL-KDD including 4000 random raw data are used for the training and testing process and the Support Vector Machine as the classifier. The accuracy of three datasets is (DS1-82.60, DS2-94.20, DS3-96.70).

The wrapper feature selection methodology has been proposed based on the Binary Bat Algorithm with

Lévy Flights (BBAL) [40]. Further tests were performed on the NSL_KDD dataset through its combination with NB and SVM. The accuracy of the (NB+BBAL) model was 91.62%, while the accuracy of the (SVM+BBAL) model was 95.78%.

Feature selection based on Genetic algorithm (GA) has been proposed by [41]. The GA was for relevant features selecting from the full NSL KDD data set. It showed a tremendous increase in the accuracy of NBC (89.5%) with a decrease in both the time and number of features. The covering feature selection method based on the Binary Bat Algorithm with slight improvement (BBASI) has been proposed by [42]. Further tests were performed on the NSL_KDD dataset by combining BBASI with NB and SVM. The accuracy of the (c4.5+BBASI) model was 96.02%, while that of (SVM+BBASI) model was 96.88.

III. Methodology

3.1 NSL-KDD DATASET

In IDS research, the KDD Cup 1999 is the commonest data set used. This data contains about 4,900,000 connection records and each record consists of 41 features [28]. This data has been statistically analyzed and presented [29] . NSL-KDD dataset, it has been reported that the KDD dataset has many problems; for example, it contains several redundant features, and the difficulty level of the different records and the percentage of records in the original KDD data set are not inversely proportional. These deficits result in a poor evaluation of different proposed ID techniques. The NSL-KDD dataset was proposed to overcome some of these issues of the KDD Cup 1999 data set. The proposed new dataset is made up of selected complete KDD dataset records [29].

3.2 THE PROPOSED SYSTEM

The major aim of this work is to propose a wrapper feature selection algorithm, which has the ability to select the most relevant features from the NSL_KDD data set. which enhances the detection rate of the Network Intrusion Detection System. In addition, the selection process reduces the size of the dataset, which leads to decrease the required memory space and less search space for classification models .The proposed IDS system is composed of 3 main stages (preprocessing. feature selection and evaluation stages).

3.2.1 Stage 1: Preprocessing

The whole dataset is preprocessed in this stage. It consists of two steps, scaling and normalization. In the scaling step, the dataset is converted from string representation into a numerical representation. For example, the class label in the dataset contains two different categories 'Normal' and 'Attack', after implementing this step this label is changed into '1' and '0', where '1' means a normal case, while '0' means attack.

The second step is normalization. The normalization cleans the noises from the dataset and decreases the differences in the ranges between the features. In this work, we have used Max-Max normalization method, as follows:

$$F_i = \frac{(F_i - Min_i)}{(Max_i - Min_i)}$$

Where F_i represents the current feature needs to be normalized, Min_i and Max_i represent the minimum and the maximum value for that feature respectively.

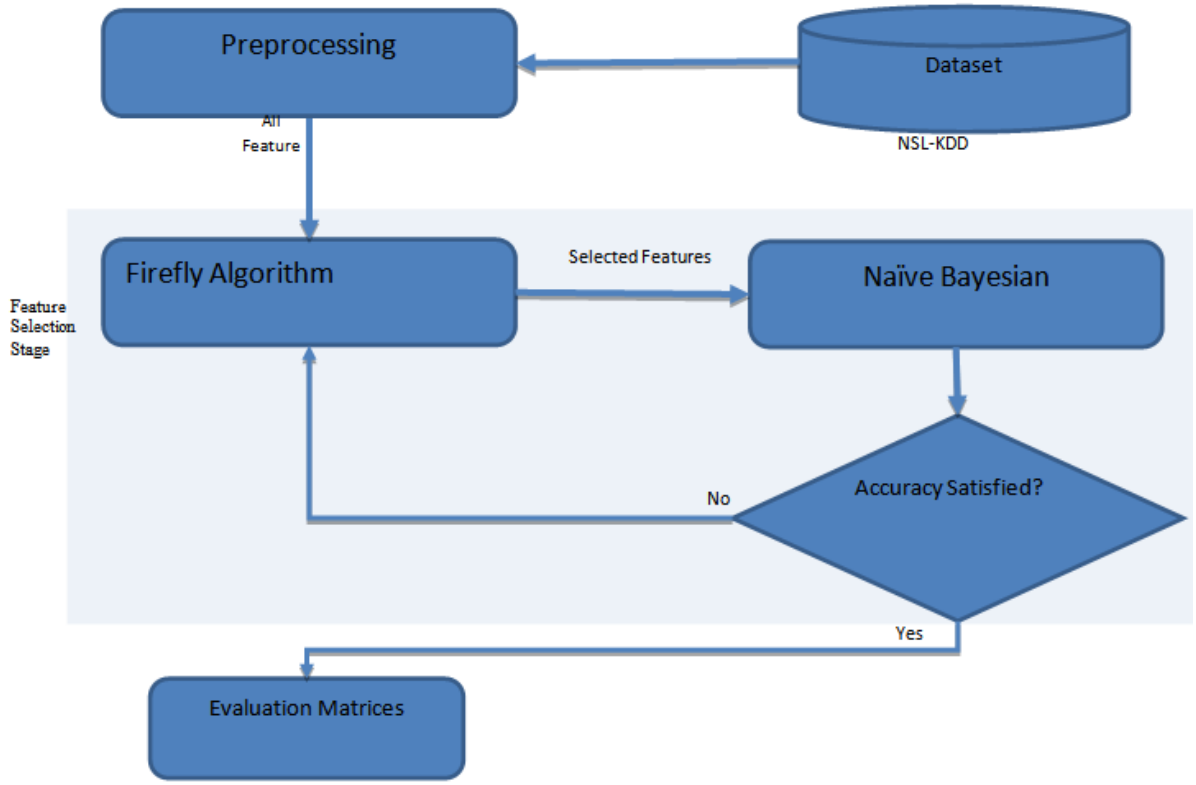

Figure 1: Block diagram of the proposed System

3.2.2 Stage 2: Feature Selection
This stage is responsible for choosing the most relevant feature from the preprocessed dataset. It consists of two steps, Firefly algorithm, and Naïve Bayesian Classifier. In the first step, the algorithm will generate N firefly algorithm (i.e. Swarm size), all these fireflies are evaluated by using Naïve Bayesian classifier which represents the fitness function in this work.

3.2.3 Stage 3: Evaluation Matrices
The evaluation process estimates validity and accuracy of these constructed models which are obtained by the following classification counters:
True Positive (TP): infected e-mail that is correctly categorized as spam.
False Positive (FP): e-mail that is incorrectly categorized as spam.
True Negative (TN): e-mail that is correctly categorized as e-mail.
False Negative (FN): infected e-mail that is incorrectly categorized as e-mail.

There are four evaluation measures that are used to evaluate the results of classifiers which depend on four classification counters as follows :
Accuracy evaluates the classifier effectiveness through its correct predictions percentage.

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

Error rate evaluates the classifier through its incorrect predictions percentage

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

Precision measure estimates the probability of correct positive prediction.

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

Recall, as well it is recognized as true positive rate or sensitivity, which is the rate of examples belonging to a positive class that is correctly positively predicted.

$$\text{Recall} = \frac{TP}{TP+FN}$$

3.3 CHAOTIC FIREFLY ALGORITHM FOR FEATURE SELECTION

The major motivation towards building a hybrid model for feature selection is to find a better balance between the filter models' computational efficiency and the wrapper models' performance accuracy. With the traditional wrapper model like the FA, all fireflies are initialized with randomly selected features. In the proposed hybrid model, all flies in the swarm will be initialized with good position, in other words, the swarm will begin the searching process from a good starting point. In addition to the good features was aimed at strategizing some aspect of the search effort to facilitate the rate of swarm convergence towards the established best-known solution. The major steps in the proposed algorithm as follows:

3.3.1 Initialization:

This step initiates all fireflies in the swarm by using logistic chaotic maps, as follows:

$$X_{i+1} = \mu X_i \ (1 - X_i)$$

where X_i represents the initial parameter of the logistic map, μ represents the controlling parameter of lthe ogistic map which is called "Mutation".

Equation 3.6 is performed t times to generate single firefly position. Therefore, to generate N fireflies, the initialization step will be performed N × t times.

The output of all the positions is a real number which should be converted into the binary number since the proposed method employs binary digits to represent features. The non-selected and selected features are represented by the bit values 0 and 1, respectively. The following equation was used to convert the gain ratio into binary [0,1]:

$$F_i = \begin{cases} 1, \text{sigmoid } (Xi) > U\ (0,1) \\ 0, \text{otherwise} \end{cases}$$

Where Fi is the feature of a firefly, the sigmoid (Xi) is 1 / (1 + e -GRI), and U is the uniform distribution.

The fireflies are initialized through these steps. Each firefly has its own position based on the weight of each feature calculated by the gain ratio.

3.3.2 Fitness Function

The fitness purpose of the proposed scheme is to reduce the error rate of the algorithms' classification performance over the authentication set of a given training data set as shown in Equation 6 while maximizing the number of non- selected features (irrelevant features). To calculate the fitness function, a classifier should be used. In this case, Naïve Bayesian Classifier has been applied to get the accuracy.

$$\boldsymbol{Error} = \sigma * \frac{[\#Features]}{[\#All\ Features]} + (1 - \sigma) * \frac{Err[\#Features]}{Err[\#All\ Features]}$$

where #Features is the selected features; Err is the classifier error rate, in other words, the 5-fold cross-validation error rate after training the Naïve Bayesian; and σ is a constant value limited to the range [0,1], regulating the importance of the classification presentation to the number of selected features. After calculating the error. the intensity of each firefly is calculated using the following Equation:

$$I\ (F_i) = \frac{1}{1 + Error^2}$$

3.3.3 Attractiveness Calculation

The attractiveness β of each firefly can be defined by the following equation:

$$\beta(r) = \beta_0 \times e^{-\gamma r^2}$$

Where r represents the distance between two fireflies and can be calculated by equation 9, and β_0 represents the attractiveness at r =0 (Initial Case).

$$r_{ij} = |X_i - X_j|$$

Where X represents the real values of the location of the fireflies which have been calculated by the information gain ratio equation.

3.3.4 Position Updating

Each firefly in the swarm moves towards the brighter firefly, in other words, the Firefly (*Fi*) is attracted by brighter firefly or has more intensity, this step can be called position updating. This updating can be determined by the following equation: -

$$P_i = P_i + \beta \times \left(P_j.Gr - P_i.Gr \right) + \alpha \times \left(Rnd - \frac{1}{2} \right)$$

Where P_i in the first part of the equat ion represents the current position, while the second part contains the attractiveness between the position of F_i and F_j, Gr represents the information gain ratio values for all features which have calculated in the first step. The third part contains the randomization with α, where $\alpha \in [0,1]$.The randomness parameter is decremented by another constant rate δ, where $\delta \in [0.95 , 0.97]$, so that at the final stage of the optimization α has its lowest value as in equation (3.13).

$$\alpha = \alpha \times \delta$$

IV. Experimental Results

The main code of the proposed chaotic firefly algorithm and Naïve Bayesian classifier have been developed by using visual C#.net version 6.0 – Visual Studio 2017 community version. The developed program has been implemented in an environment with the following specification: Operating System is windows 10 with 64-bit architecture, CPU Intel 2.4 GHz, and RAM 8GB.

As mentioned in the previous chapter, the dataset used in this thesis is an NSL-KDD dataset. It has been downloaded from [XX]. The original version of NSL-KDD consists of five classes (Normal, DDOS, R2L, U2L, and Prop), 41 features, and around 125 samples. In this work, we have converted the dataset into binary classes (Attack and Normal), and 10,000 samples selected randomly for validating the system. In order to execute the Chaotic Firefly Algorithm, several parameters need to be initialized, Table 4-1 shows those parameters with their values

Table 4-1 Parameter Settings

N	Parameter	Symbol	Value
1	Logistic Map Initial value	X_0	Random [0,1]
2	Initial Attractiveness	B_0	0
3	Randomization Factor	α	0.2
4	Gamma	γ	1.0
5	Delta	δ	0.96

The proposed feature selection algorithm has tested based on different scenarios. These scenarios are:
1 .Swarm size, to test the effect of the swarm size on the searching process, four cases were used in the experiments (10, 20, 30, and 40).
2. Number of Iterations. to test the effect of the iterations on the searching process, three cases were examined, (100, and 250) number of iterations.
As mentioned before, the firefly algorithm is initialized randomly (i.e. based on the generated chaotic sequence), each runtime has different results. Therefore, all scenarios have been implemented 15 run times, the maximum, the minimum and the average accuracy are measured.

4.1 DISCUSSION

Summary of results for this section after running the algorithm 15 times run its proposal.

Table 4-9: The results of 15 run times for scenario 5

No.of Iterations	Swarm Size	Best Accuracy	Worst Accuracy	Average Accuracy	Average Features
100	10	95.43333	93.53333	94.5	15.8
	20	95.60000	94.00000	94.9	16.9
	30	96.30000	94.60000	95.1	16
	40	96.10067	94.76667	95.36	15.9
250	10	95.80000	94.03333	95.05	16.3
	20	96.00000	94.53333	95.22	14.9
	30	96.21000	95.20000	95.5	15.8
	40	96.40667	95.23333	95.7	15.2

As can be seen Table 4-9, the results are increased gradually with the increase of the swarm size. Thus, the swarm size effect on the accuracy. In addition, the number of iterations also affect the searching process. We can conclude that both swarm size and the number of repetitions are effect on the prediction accuracy when they are increased

4.2 Benchmarking the proposed method with other algorithms

There are two major stages of the proposed anomaly-based IDS model - the pre-processing stage, which involved the wrapper FS process via a combination of BBAL and NBC; the second stage is the detection step where the obtained classifier performance is compared to that of a previously selected feature subset. The proposed model was tested on a PC with a core i7 processor. speed 2.2 GHz. and 4 GB of RAM running under Windows 10 operating system was used. Also, for the ranking, the proposed algorithm was benchmarked with two other algorithms (BPSO and BBA). The results of these other two algorithms were lifted from previous studies [reference]. The three algorithms had their individual parameters and use specific values, as follows:

Swarm size = 10, Maximum number of iteration = 200.

For BFA:
Bmin = 0.0, G = 1.0, A = 0.2, D = 0.96.

For BBA:
– the maximum loudness A0 = 0.5, and the minimum pulse rate r0 = 0.5.
– the incidence ranges between 0.8 and 1.0.
– = 0:1 and _ = 0:9

BPSO: -
– learning factors are c1 = 2.3 and c2 = 1.8.
– inertia weight was reduced from 0.9 to 0.5.

Table 4-10 showed the results of all the algorithms

Algorithm	**Acc. Rate**	**Err. Rate**	**No. Features**
BPSO	90.63%	9.37%	22
BBAL	91.61%	8.09%	15
Proposed FA	**96.02%**	**3.98%**	**14**

V. Conclusion

A wrapper feature assortment method was proposed in this thesis and applied to intrusion detection system. Further tests on the performance of the proposed method were done using Naïve Bayesian Classifier. The NSL_KDD data set was used and it empirically proved that the movement and randomization of the Firefly algorithm were enhanced by chaotic initiation through logistic map method since the Firefly algorithm was initialized by a binary sequence, unlike the standard Firefly algorithm.
This enhancement can offer better results in terms of performance accuracy and the number of selected features.

The proposed firefly algorithm in this work showed a superior performance when applied to NSL-KDD dataset. The experiments showed the proposed algorithm to have the ability to select the most pertinent features, The performance of algorithm increased when there is an increase in the swarm size and iteration number. The best-attained result in terms of accuracy was 96.4206 with 14 features.

References

[1]. KDD Cup 1999 Dataset, (n.d.).
[2]. M. Tavallaee, E. Bagheri, W. Lu, A.A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in: IEEE Symp. Comput. Intell. Secur. Def. Appl. CISDA 2009, 2009. doi:10.1109/CISDA.2009.5356528.
[3]. S.R. Hasani, Z.A. Othman, S.M.M. Kahaki, Hybrid feature selection algorithm for intrusion detection system, J. Comput. Sci. 10 (2014) 1015–1025. doi:10.3844/jcssp.2014.1015-1025.
[4]. S. Srinoy, Intrusion Detection Model Based On Particle Swarm Optimization and Support Vector Machine, in: Proc. 2007 IEEE Symp. Comput. Intell. Secur. Def. Appl., 2007. doi:1-4244-0700-1/07/$20.00.
[5]. S. Mukherjee, N. Sharma, Intrusion Detection using Naive Bayes Classifier with Feature Reduction, Procedia Technol. 4 (2012) 119–128. doi:10.1016/j.protcy.2012.05.017.
[6]. A.J. Malik, F.A. Khan, A Hybrid Technique Using Multi-objective Particle Swarm Optimization and Random Forests for PROBE Attacks Detection in a Network, in: 2013 IEEE Int. Conf. Syst. Man, Cybern., 2013: pp. 2473–2478. doi:10.1109/SMC.2013.422.
[7]. A.C. Enache, V. Sgârciu, An improved bat algorithm driven by support vector machines for intrusion detection, in: Adv. Intell. Syst. Comput., 2015: pp. 41–51. doi:10.1007/978-3-319-19713-5_4.
[8]. K.S. Desale, R. Ade, Genetic algorithm based feature selection approach for effective intrusion detection system, in: 2015 Int. Conf. Comput. Commun. Informatics, ICCCI 2015, 2015. doi:10.1109/ICCCI.2015.7218109.
[9]. A.C. Enache, V. Sgarciu, A. Petrescu-Nita, Intelligent feature selection method rooted in Binary Bat Algorithm for intrusion detection, in: SACI 2015 - 10th Jubil. IEEE Int. Symp. Appl. Comput. Intell. Informatics, Proc., 2015: pp. 517–521. doi:10.1109/SACI.2015.7208259.

Dr. Sefer Kurnaz. " Enhanced Detection Rate for Network Intrusion Detection System by Using Chaotic Firefly Algorithm" IOSR Journal of Computer Engineering (IOSR-JCE) 20.6 (2018): 01-07.