Intrusion Detection System Based on Support Vector Machine Optimised with Cat Swarm Optimization Algorithm

Suleiman Idris
Department of Computer Science,
School of Information & Comm. Tech.,
Federal University of Technology,
Minna, Nigeria.
sidris27@gmail.com

Oyefolahan Ishaq O.
School of Information & Comm. Tech.,
Federal University of Technology,
Minna, Nigeria.
o.ishaq@futminna.edu.ng.

Ndunagu Juliana N.
Department of Computer Science,
Faculty of Sciences,
National Open University of Nigeria
Abuja, Nigeria.
jndunagu@noun.edu.ng

***Abstract*—intrusion detection system (IDS) like firewall, access control and encryption mechanisms no longer provide the much-needed security for systems and computer networks. Current IDS are developed on anomaly detection which helps to detect known and unknown attacks. Though, these anomaly-based IDS feature a high false rate. To reduce this false alarm rate, in this paper, we proposed an intrusion detection model based on support vector machine (SVM) optimized with Cat swarm optimization (CSO) algorithm. We use the information gain (IG) for attribute reduction and perform classification using the optimized Support vector. The result obtained shows that our model performs well with the least false alarm rate and good accuracy value compare with other classification algorithms evaluated using the same datasets.**

***Index Terms*—Intrusion Detection, Support vector machine, Cat Swarm Optimization, Information Gain, NSL-KDD**

I. INTRODUCTION

One of the major technological achievement in recent time is the possibility of connecting computer systems for the purpose of sharing resources. Furthermore, the advent of the internet has made it possible for people to communicate from different part of the globe through connected computer networks. However, these interconnection of computer devices came with its own cons. One of the major issues with this technology is in the area of security. computer networks and the internet at large are faced with many security attacks. These attacks aim to compromise the three security goals confidentiality, integrity and availability of any system, network and their resources.

Many protection techniques have been employed to manage the security risks involved with computers and networks. Techniques like encrypting confidential data, access control and software and hardware firewall policies. However, these techniues are not enough as each one of the techniques possess significant limitations. Therefore, it becomes important to use other additional defense mechanism like intrusion detection system (IDS) [1]. IDS is a software application or a hardware device that is configured to monitors computer system or network for abnormal activities and report or prompt for appropriate action [2]. Many researches have been carried out by researchers to determine an intrusion detection technology with good detection in regards to the accuracy value and minimum training time. Although, many issues still exist with IDS, issues like poor capability for detection, high false positive rate [3].

Many methods have been introduced to improve the performance of IDS in recent times. One of the popular research methods in IDS is support vector machine (SVM). SVM is one of the novels machine learning method that has become a well-known research method in the area of intrusion detection. This is because its generalization performance is good, unavailability of local minimal and it uses minimum time for execution [1]. Although, the performance of Support Vector Machine still depend on how well its parameters are appropriately selected. [1]. If the selection of its parameters are not done appropriately, it will perform poorly. In this study, an IDS that is based on SVM with its parameters optimized using Cat Swarm Optimization algorithm has been proposed.

II. INTRUSION DETECTION SYSTEM

An IDS could be a hardware device or an application software configured to monitor traffics that moves in and out of a computer system or network for activities that are classified as malicious or breach of policy and produces report to a management station. Some of these systems some time may try to completely stop an attempt to get unauthorized access however, it is not compulsory component of a monitoring system [2]. Because of the increasing number of connectivity between computers, intrusion detection has become important in the area of network security [4]. Techniques available for intrusion prevention for instance access control, encryption and firewalls have not provided the security level required to protect systems and networks from increasing security attacks [5]. Therefore, it becomes crucial to deploy an IDS as an additional security measure to detect these security attacks before they course havoc in the system [5]. IDS is developed primarily to detect different kinds of traffics that are malicious and abnormal computer usage that a typical firewall will not be able to detect. The concept of machine learning has been used to develop many IDS. Specifically, integrating two or more learning techniques have yielded better detection performance compare to a single detection technique [6].

Categorizing IDS can be achieved in different ways. The most common categories or types are misuse based IDS and anomaly-based IDS [7][1].

A. Misuse-based Intrusion Detection System

This is also referred to as signature-based intrusion detection system. This type of detection techniques scanned packets or audit logs and compared with commands or events that are previously known to be a sign of an attack [6]. This type of IDS performs very well in detecting attacks that are previously known. It has a low false alarm rate. However, it performs poor when it comes to detecting new attacks that are not previously known or contained in the database [4][7].

B. Anomaly-Based Intrusion Detection System

This category of IDS is developed based on normal behavior features. It uses these identified features with normal traffics to pinpoint any action that significantly deviates from the normal features. It uses data taken from normal usage to identify patterns [7]. Anomaly IDS make use of patterns associated with behavior that could mean unacceptable activities and analyse previous activities to know whether the observed behavior are normal. [6].

III. Categories or Classes of Intrusion Attacks

Intrusion attacks can be classified into: Remote to local (R2L) attack, User to Root (U2R) attack, Probing attack and Dos or DDoS attacks [8].

A. Remote to Local Attacks (R2L)

In an R2L attack, the adversary aim is to acquire a local right to a machine. To achieve this, the attacker send packets that are capable of compromising the target system over the network, the machine loopholes or vulnerability are then exploited to gain unauthorized access. Attackers with ability to communicate with their target device but have no account on that device uses this kind of attack to exploit weaknesses that exist on the target system to acquire local access on the target device [9]. Attacks like this can be carried out by making use of ports that are open on the target system, using the system loopholes, password guessing [7].

B. User to Root Attack (U2R)

In this type of attack, a normal user tries to escalate his/her privileges by taking advantages of weaknesses found in a system to gain administrative access or root access. This attack is like R2L attacks. The difference is that the attacker here is already a normal user and he/she wants to escalate his/her privilege. [7]. User to root (U2R) attack simply refer to a situation where a legit or normal user wants to gain higher privilege in other to carryout illegal or unauthorized activities. [9].

C. Probing Attack

This class of attack has to do with reconnaissance, gathering information by scanning systems and networks to find weaknesses that exist with them. The found loopholes are used to exploit the systems and networks [9].

D. Dos/DDoS Attack

Denial of Service (Dos) attack often involves attacker sending traffics that are more than what the victim system can handle making such system deny legitimate users’ access to services [10]. DoS attack usually originated from a single source. A DoS attack becomes a DDoS attack if the traffics originated from sources more than one [11]. DDoS attacks are usually carried out by deploying many compromised systems (usually called botnet or zombies) to overwhelm their victim [12]. Dos and DDoS attack are attacks targeted at compromising the availability of computer system, router, network and their resources [13]. These attacks are carried out by sending illegitimate traffics capable of draining the system memory or network bandwidth [14][15]. These attacks can be carried out at different layer of the open system interconnection model like the physical layer (the first layer), network layer(the third layer), transport layer (the fourth layer) and application layer (the seventh layer)[12][15] At the physical layer, the attack can simply be to remove a power or network cable connecting a server to the network. Attacks at the network layer are achieved using network layer protocol example of protocol that can be used to achieve this attack at the network layer is the Internet Control Message Protocol (ICMP) [14]. At the transport layer, the attacks can be achieved using layer four protocol like the user datagram protocol which is a connectionless protocol, another protocol that can be used at layer four is the transmission control protocol (TCP) [14]. Hypertext transmission protocol (http) is one major protocol used to carry out denial of service attack this protocol is used at the application layer level. Other protocols used at the application layer to carry out DoS attack are Simple Mail Transmission Protocol, Domain Name System, Voice over internet protocol (VoIP).

IV. Support Vector Machine

Support vector machine is a machine learning algorithm that has gained importance in the area of pattern classification. SVM primarily aim at finding the best hyperplane to divide two classes in a dataset. Several machine learning algorithms exist for dataset classification, SVM standout of these algorithms because of its outstanding generalization capability and its good record for achieving high accuracy level in the training datasets [1].

Classification problem has several major challenges, one of them is the separation of data tending differently, making it difficult for linear separation. [16]. Usually, the dataset is not separable linearly. To overcome this issue of linearly inseparable datasets, the dataset can be mapped into dimension feature space that is higher and then the hyperplane that separate linearly vectors mapped. That is to say x_i will be substituted with where K gives the mapping with the higher dimension (K is also refer to as the kernel function). Commonly, kernel functions are of three main kinds: polynomial, sigmoid and radial-basis kernel function (RBF) [1].

A. *Polynomial kernel function*

This kernel can be used to solve problems were the samples for the training datasets are normalized. It is non-stationary. Using this kernel, some parameters have to be settled. The parameters are the gamma slope, r being the constant term and d being the polynomial degree (hence *d*=3, *r*=0) [17]. The polynomial function is represented as follows.

$$K(x_i, x_j) = (\sigma x_i^T x_j + r)^d, \sigma > 0 \qquad (1)$$

B. *Radial-basis Kernel Function (RBF)*

This family of kernel functions have a distance measure smoothed by an exponential function. It maps samples nonlinearly into space dimension that is higher. It is good with instances where attributes and class label do not have linear relations. In addition, the linear kernel can be described as a subset of RBF because, a linear kernel having the penalty parameter C perform similar way with RBF kernel with some parameters (C, Gamma) [17]. The RBF kernel is represented as follows

$$K(x_i, x_j) = \exp(-\sigma \left\| x_i - x_j \right\|^2), \sigma > 0 \qquad (2)$$

One of the parameters that plays a major role is the adjustable parameter represented as б this parameter should be turned carefully. If it is overestimated, it will cause the exponential to behave like a linear function and the nonlinear power of the higher dimensional projection will begin to. If the adjustable parameter is underestimated, the regulation power of the function will be loss and the boundary for decision will become highly sensitive to noise in training data. Therefore, Support vector machine behavior basically depends on how well the choice of the width parameter б is made [17].

C. *Sigmoid kernel*

One requirement of this kernel is that it must satisfies Mercer’s theorem, for this to happen, the kernel has to be positive definite. Although, this kernel despite its popular acceptance and usage, it is still not positive semi-definite for some of its parameter’s values. Therefore, a carefully chosen parameter for б, *r* is very important. If these parameters are not well chosen, it will lead to a very wrong result [17].

$$K(x_i, x_j) = \tanh(\sigma x_i^T x_j + r) \qquad (3)$$

б can be seen as a parameter that could be measured using scale of the input samples, and *r* as a shifting parameter, the shifting parameter that controls the threshold of mapping (hence *r* =0). Generally speaking, RBF and linear kernels are better than the sigmoid function [23].

V. Cat Swarm Optimization

One of the types of optimization problem is feature selection. It is usually achieved by hybridizing a good an optimization algorithm with a classification algorithm. Two commonly used optimal algorithms are Particle Swarm Optimization (PSO) and genetic algorithm (GA). Recently, another optimization algorithm has been proposed Cat swarm optimization (CSO) and it has been proven to perform better compare to PSO [18][19]. CSO was built putting into considerations the behavior of cats, cats are known for hunting excellently and for also showing great level of alertness even at their resting positions. This behavior exhibited by cats can be described or explained by two modes. These modes are: Seeking and Tracing modes [18][20].

A. *Seeking Mode*

This mode describes the situation of the cat while resting. In this mode, the cat does more of thinking and takes decisions about where to move to next [19]. Four parameters are used to represent seeking mode in the CSO algorithm: one of the parameters is Seeking memory pool (SMP), the second parameter is the Seeking Range of the selected dimension (SRD), third parameter is the count of dimension to change (CDC) and the fourth parameter is Self-Position Consideration (SPC) [21]. Procedure of seeking mode is described below

Step1: produce j replica of the current state of cat$_k$, where j = SMP. Check SPC if it is true, j = (SMP-1), then accept current status to be one among the candidates.

Step2: For each replica, following the CDC, in no order add or subtract SRD percent of the current values and change existing ones.

Step3: determine the values of the fitness (FS) for all candidates points

Step4: in the case where all FS values are not the same, determine the selecting likelihood of every candidate point by (4), else make all selecting likelihood of every candidate point be 1.

Step5: in no order, choose the position to go to next from the candidate points, and change the position of cat$_k$.

$$Pi = \frac{|FSi - FSb|}{FSmax - FSmin} \quad where\ 0 < i < J \qquad (4)$$

In a situation where the fitness function aim is to look for the least solution then *FSb* = *FSmax*, else *FSb* = *FSmin*.

B. *Tracing Mode*

This mode describes the situation of the cat while chasing a target. A cat in a tracing mode changes position in accordance with its own velocity for each dimension [21]. The process of tracing mode is explained as follows

“Step1: Each dimension (vk,d) velocities should be updated following (5).

Step2: velocities should be checked to ensure they are within maximum velocity range. In a situation where the range of the new velocity is over it should be set to be equal to the limit.

$$V_{k,d} = V_{k,d} + r_1 \text{ x } C_1 \left(X_{best\ d} - X_{k,d} \right) \qquad (5)$$

Step3: the position of cat*k* chould be updated following (6).

$$X_{k\,d} = X_{k\,d} + V_{k\,d} \qquad (6)$$

x_{best} represent the state of the cat with the most acceptable fitness value; $X_{k,d}$ is the state of cat$_{k.}$ c_1 represent constant and r_1 represent random value the random values are in the range [0,1]. “

VI. Related Work

In the network intrusion detection algorithm developed by [22], two tree-based classifier models were combined. the random tree and Naïve Bayes tree classifiers. The paper aim is to have a hybrid classifier that can classify traffic entering a network into normal or attack with better accuracy compare to the individual classifiers. The study used the NSL-KDD dataset to assess how well their classifier perform. Detection accuracy of 89.24% was achieved. The future work proposed by the study is to test the effect of reducing the attributes on the training and testing datasets and the detection accuracy.

Also, [23], proposed a framework that detect and mitigate known and unknown distributed denial of service in real time environment using artificial neural network. The study used ANN to detect attack based on some features that separate DDoS attack from normal attack. The ANN was trained with data collected from a network setting that represented a mirror image of a real life network environment. In addition to the data collected from the mirror network, the study used old data to evaluate their work. A detection accuracy of 98% was recorded. The future work would be to train their approach using other dataset and compare the outcome with the outcome they got. Also, their work was not simulated in any network environment, one could simulate their approach to verify the detection accuracy of the work and the false alarm rate.

Bahrami, Bozorg-Haddad and Chu [21] proposed multilayer perception with genetic algorithm to detect DDoS attack at the seventh layer of the OSI model. Four features were considered from traffics entering that exhibit important alteration in their characteristics. The first parameter is the number of hypertext transfer protocol count. Features of hypertext transfer protocol like the GET, POST, OPTIONS, HEAD, DELETE, PUT, TRACE, and CONNECT were analyzed and normal features where recorded. The second parameter is the number of IP address that enters a network within a small-time window. The third parameter is the constant mapping function. The fourth parameter is the fixed frame length. When there is a change in these features, attack will be detected. Experiment result reveal that the technique gave 98.04% accuracy in detecting attacks at the seventh layer (application layer) of the OSI model with high false positive rate of 2.21%. The future work is to work on improving the detection accuracy and lowering the false positive rate.

Enache and Patriciu [1] proposed IDS using SVM combined with information gain. They used Information Gain to select the features of the dataset and the SVM was used for classification. The parameters for Support Vector Machine were selected using Particle Swarm Optimization which optimizes candidate solution through iteration and Artificial Bee Colony developed by observing honey bees behaviour. In other to evaluate the performance NSL-KDD dataset was deployed. The results gotten showed that optimized SVM with PSO or ABC performed better with the dataset compare to the normal SVM. The future work would be to apply other feature selection swarm intelligince that could do better compare to the ones used here.

Rana et, al. [24] uses fuzziness based on semi supervised approach for intrusion detection system. To improve the classifier performance for the IDS, samples that are not labelled supported with supervised learning algorithm were used. Results gotten from experiment using this method reveal that samples that are not labelled belonging to the categories of low and high fuzziness groups provide the most input to increase the performance of the classifier compared to classifiers that are already existing examples random forest naive bayes, support vector machine. They got an accuracy of 84.12%. The future work proposed is to apply this method to increase the effectiveness of IDSs for detecting many types of attacks.

VII. PROPOSED MODEL

Fig. 1 depict our model. Acquiring the NSL-KDD dataset, Preprocessing and attribute selection are pre-requisite in any intrusion detection work. Our contributions begin with optimizing the parameters of the SVM using Cat Swarm Optimization Algorithm.

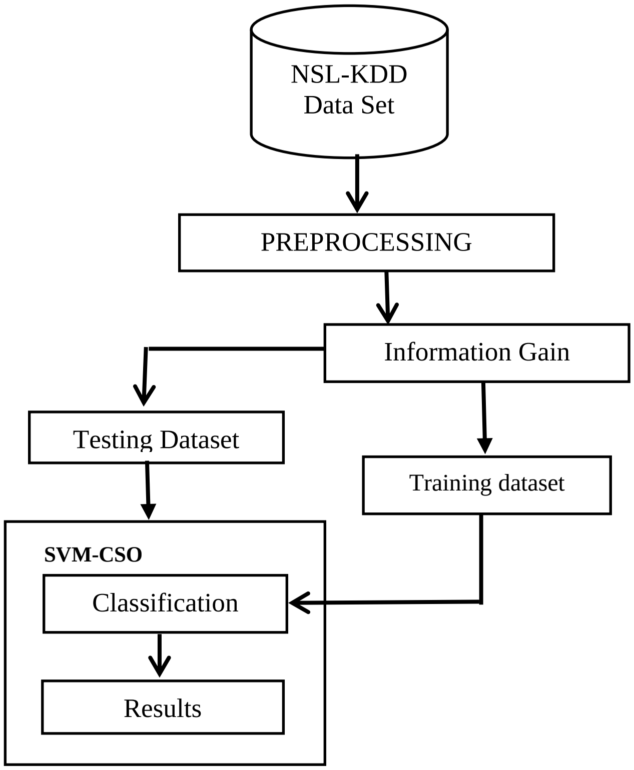

Figure 1: Proposed Model.

A. *Data Processing/Feature Selection*

In other to evaluate our system, NSL KDD dataset was used. The dataset is a KDD dataset that has been improved upon. It has numerous advantages over the KDD Cup 99. The advantages include: unavailability of redundant record in the train dataset, no duplicate records in the test dataset. The NSL KDD dataset consist of 41 features. However not all of the features that are relevant. Therefore, the need for feature selection. In other to carryout feature or attribute selection, Entropy (information gain) was used. Entropy is a common criterion used in machine learning to individually rank features or attributes with respect to class attributes. IG is calculated by a decrease in the uncertainty of knowing the class feature when the value of the feature is not known. Its idea is based on the principle of information theory usually deployed in ranking and chosing attributes with high value to reduce feature vector size and achieve improved classification with less complexity.

The entropy or information gain of a given feature A with relation to the class feature C, represented as I(C/A), is the decrease in uncertainty with the value of C knowing the value of A. Assuming C and A are whole numbers variables that draw from C = (c_1, …. c_k) and A = (a_1, …, a_n). H(C) is the IG, that calculate the uncertainty about the value of C. H(C/A) is the conditional entropy of C given A, that calculates the uncertainty about the value of C knowing the values of A. Therefore I(C/A) = H(C) - H(C/A) [1].

$$\mathrm{H(C)} = -\sum_{I=1}^{K} P(Ci)log2(P(Ci)) \qquad (7)$$

$$\mathrm{H(C/A)} = -\sum_{j=1}^{n} P(aj) \sum_{i=1}^{k} P\left(\frac{Ci}{aj}\right) log2(P(Ci/aj))$$

where, P(ci/aj) is the posterior probabilities of C given the values of A. therefore I(C/A) is given as follows

$$\mathrm{I(C/A)} = -\sum_{I=1}^{K} P(Ci)log2(P(Ci)) - \left(-\sum_{j=1}^{n} P(aj) \sum_{i=1}^{k} P\left(\frac{Ci}{aj}\right) log2(P(Ci/aj))\right)$$

B. *Dataset classification*

After feature selection, the next stage is dataset classification. We carry out the classification using the optimized SVM with CSO. First, we find the best parameters of the SVM using CSO then we use the optimal parameters to build the training sample as follows.

Step 1: Clearly state the parameters of the algorithm
Step 2: create first cats and velocity in no particular order
Step 3: spread the cats into the two modes tracing and seeking
Step 4: Check if cat is in seeking mode if yes start seeking mode otherwise start tracing mode
Step 5: recalculate fitness function and retain the cat with the best solution in the memory
Step 6: check to know if looping condition is satisfied. If it is, Stop looping and give out the peak parameter (C and 6) else, return to step 2.
Step 7: use the peak parameter (C, 6) and training sample to build up SVM prediction model.

C. *Building SVM Prediction Model with Optimal Parameter (C, 6)*

In this work, we use the SVM constructed by Radial basis function (RBF). This family of kernel functions have a distance measure smoothed by an exponential function. It maps samples nonlinearly into space dimension that is higher. It is good with instances where attributes and class label do not have linear relations. In addition, the linear kernel is a subset of RBF because, a linear kernel with penalty parameter C perform the same way with RBF kernel with some parameters (C, Gamma) [23]. The RBF kernel is represented in (2):

VIII. RESULT AND DISCUSSION

The experiment was carried on a java NetBeans platform with Weka.jar libraries to be able to access weka functionalities. First, we carry out feature selection on the 42 attributes in the NSL KDD dataset to know attributes that have high impacts and those without impact on our prediction.

After attribute selection using information gain (Entropy), some of the attributes have good entropy value while others have insignificant or zero (0) entropy value that is they have no impact on the prediction outcome. Attributes with insignificant entropy values were removed, table 1 shows attribute with good entropy value.

Table 1

Attrbutes selected after Information Gain

S/N	Attribute Name	S/N	Attribute Name
1	Arc_bytes	11	Count
2	Dst_bytes	12	Logged_in
3	Services	13	Same_srv_rate
4	Flag	14	Rerror_rate
5	Dst_host_srv_count	15	Srv_rerror_rate
6	Dst_host_same_srv_rate	16	Dst_host_srv_diff_host_rate
7	Dst_host_rerror_rate	17	Dst_host_same_src_port_rate
8	Dst_host_diff_srv_rate	18	Srv_fidd_host_rate
9	Dst_host_srv_rerror_rate	19	Dst_host_serror_rate
10	Diff_srv_rate	20	Dst_host_srv_serror_rate

A. *Performance Evaluation*

The performance of our model was evaluated based on the following metrics:

Accuracy: Proportion of total number of correct predictions

$$\frac{\mathbf{TP + TN}}{\mathbf{P + N}} \tag{8}$$

Precision: proportion of correct positive observation

$$\frac{\mathbf{TP}}{\mathbf{TP + FP}} \tag{9}$$

Recall: Proportion of positives correctly predicted as positive

$$\frac{\mathbf{TP}}{\mathbf{P}} \tag{10}$$

F-Measure: This is derived from precision and recall values. The F-Measure produces a high result when Precision and Recall are both balanced, thus this is very significant.

$$\frac{\mathbf{2 * Recall * Precision}}{\mathbf{Recall + Precision}} \tag{11}$$

FP Rate: with this model we can know if our model has many false alarms. It is calculated by taking the ratio of misclassified instances to normal instances.

The results obtained from applying our optimized support vector machine on the NSL KDD datasets is presented in table 2

Table 2
Results Obtained

Accuracy	Precision	Recall	F-Measures	FP Rate
96.3	95.4	97.9	96.7	0.02

B. *Comparison of Detection Accuracy, Precision, Recall, F-Measure and False Positive Rate with Zero R and Other Classifiers*

We compare the performance of our system with Zero R and some popular classification algorithms namely J48, NaiveBayes, RandomTree applied on the datasets. The performance of the algorithms are presented in fig. 2, 3, 4 and 5 table 3 summarizes the result obtained from each of the algorithms.

Table 3

Summary of Results with other Classification Algorithms

Classifier	Accuracy%	Precision%	Recall%	F-Measure %	FP Rate
J 48	95.7	96.1	94.4	95.2	0.32
RandomTree	95.1	95.7	93.3	94.5	0.035
NaiveBayes	84.3	76.3	94.7	84.5	0.24

Zero R	54.6	29.9	54.7	38.7	0.54
CSO-SVM	96.3	95.4	97.9	96.7	0.02

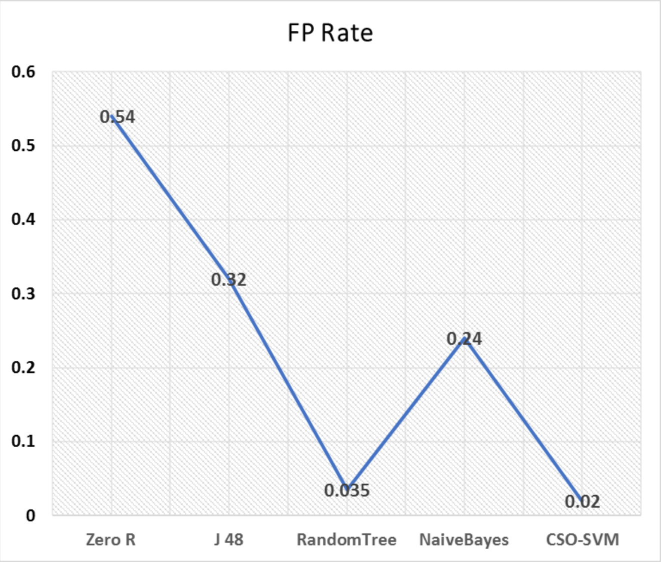

Figure 2: FP Rate

Fig. 2 shows the false positive rate for our classification algorithm and other classification algorithms. CSO-SVM has the lowest false positive rate of 0.02 compare with J48, RandomTree and NaïveBayes with false positive rate of 0.32, 0.035 and 0.024 respectively. While Zero R has the highest value of false positive rate of 0.54.

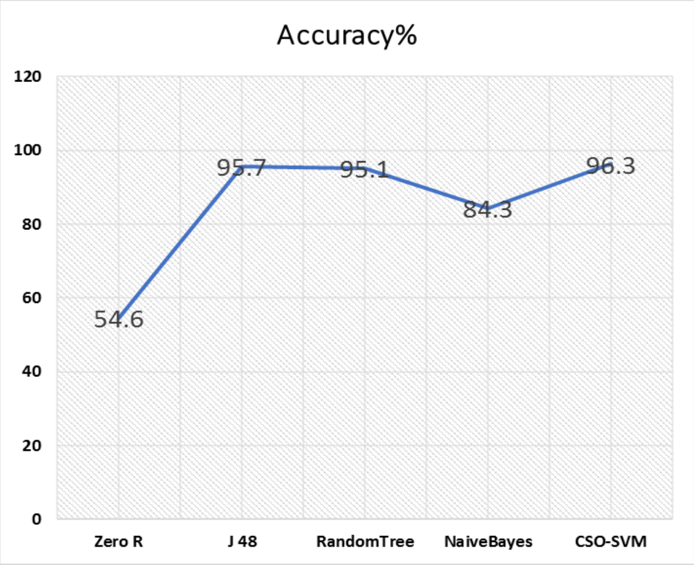

Figure 3: Accuracy

The performance of our algorithm in terms of accuracy in comparison with the Zero R, J48, RandomTree and NaïveBayes is presented in fig. 3, CSO-SVM performs better with accuracy of 96.3 percent compare with Zero R, J48, RandomTree and NaïveBayes with accuracy of 54.6, 95.7, 95.1 and 84.3 respectively.

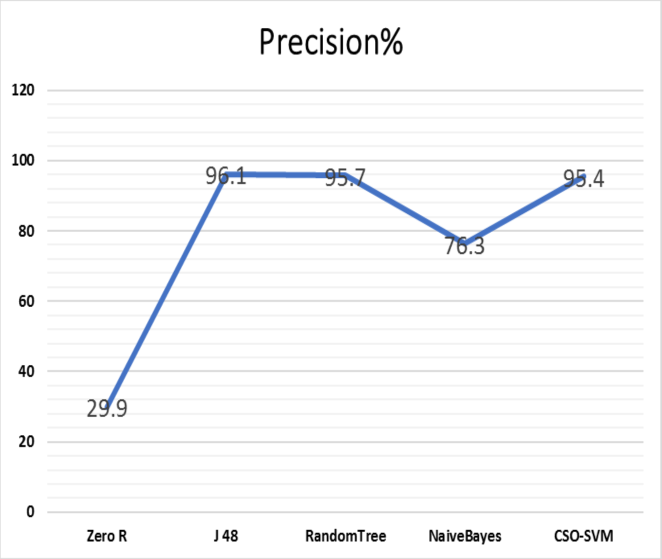

Figure 4: Precision

Interms of precision value, J48 has the highest precision value of 96.1 percent. CSO-SVM has precision value of 95.4. with RandomTree and NaiveBayes having precision values of 95.7 and 76.3 respectively. The base line classifier has precision value of 29.9.

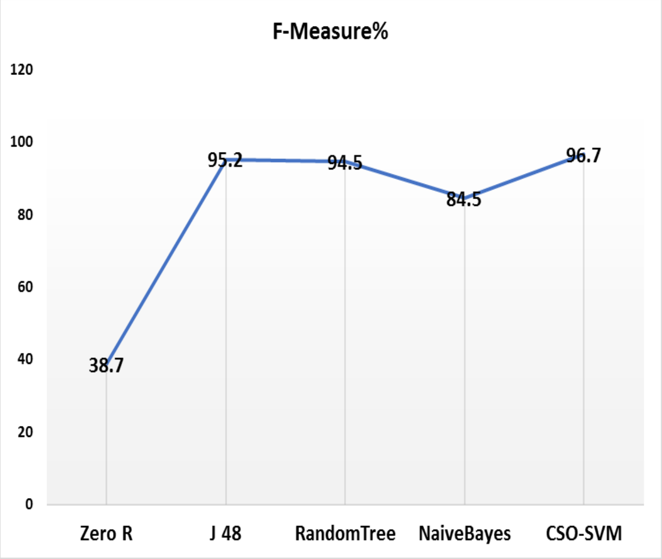

Figure 5: F-Measure

Fig. 5 shows the comparison of the CSO-SVM value compare with the other classification algorithms. F-Measure is high when you have a balanced Precision and Recall value. CSO-SVM has the highest F-Measure value of 96.7 percent followed by J48 95.2. RandomTree and NaiveBayes have F-

Measure values of 94.5 and 84.5 respectively. The baseline classifier present an F-Measure value of 38.7.

IX. Conclusion

In this research work, we have been able to optimize the performance of support vector machine using Cat Swarm Optimization Algorithm. The NSL-KDD dataset was used. the entropy value of each of the attributes was calculated with respect to the class value. Attribute with insignificant entropy value were removed during the preprocessing stage. The classification was done with the optimized SVM-CSO. The classification result shows that the CSO-SVM has better performance in all areas compare to the performance of the baseline classifier (Zero R). In terms of accuracy, and F-measure the CSO-SVM performs better compare to other clarification algorithms like the popular J48, Naïve Bayes and RandomTree. Most importantly, the CSO-SVM has low false positive rate of 0.02 compare to IG-PSO-SVM and IG-ABC-SVM with 0.04 and 0.03 respectively.

References

[1] A. C. Enache, and V. V. Patriciu, "Intrusions Detection Based On Support Vector Machine Optimized with Swarm Intelligence," *9th IEEE International Symposium on Applied Computational Intelligence and Informatics,* pp. 153 – 158, 2014

[2] K. Amit, C. M. Harish, and M. Rahul, "A Research Paper on Hybrid Intrusion Detection System," *International Journal of Engineering and Advanced Technology,* vol. 2 no. 4, pp. 294 – 297, 2013.

[3] F. Kuang, W. Xu, and S. Zhang, "A novel hybrid KPCA and SVM with GA model for intrusion detection," *Elsevier Journal of applied soft computing,* vol. 18, pp. 178 – 184, 2014.

[4] L. A. Wathiq, A. O. Zulaiha, Z. A. N Mohd, "Multi-Level Hybrid Support Vector Machine and Extreme Learning Machine Based on ModiÞed K-means for Intrusion Detection System," *Expert Systems With Applications,* 2016.

[5] A. H. Asma, F. S. Alaa, and M. W. Talaat, "Intrusion Detection System Using Weka Data Mining Tool," *International Journal of Science and Research*. Vol. 6, no. 9, pp. 337 – 342, 2015.

[6] L. Wei-Chao, K. Shih-Wen, T. Chih-Fong, "An intrusion detection system based on combining cluster centers and nearest neighbors," *Knowledge based systems,* 2015.

[7] A. Shadi, A. Monther, and B. Y. Muneer, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *Journal of Computational Science*, 2017.

[8] L. Dhanabal, and S. P. Shantharajah, "A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms," *International Journal of Advanced Research in Computer and Communication Engineering,* vol. 4, no. 6, pp. 446 – 452, 2015.

[9] R. Jamal, "A survey of Cyber Attack Detection Strategies," *International Journal of Security and Its Applications*, vol. 8, no. 1, pp. 247 – 256, 2014.

[10] P. D. Sheetal, R. H. Priti, A. D. Arundhati, "Denial of Service Attack Defense Techniques," *International Research Journal of Engineering and Technology*, vol. 4, no. 10, pp. 1532 – 1535, 2017.

[11] N. Vani, and P. K. Munivara, "Detection of Anomaly Based Application Layer DDoS Attacks Using Machine Learning Approaches," *i-manager's Journal on Computer Science,* vol. 4, no. 2, pp. 6, 2016.

[12] N. Hoque, H. Kashyap, and D. K. Bhattacharyya, "Real-time DDoS attack detection using FPGA," *Elsevier journal of Computer Communications,* vol. 110, pp. 48 – 58, 2017.

[13] V. K. Yadav, M. C. Trivedi and B. M. Mehtre, "An Approach to Handle DDoS (Ping Flood) Attack'" *Proceedings of International Conference on ICT for Sustainable Development, Advances in Intelligent Systems and Computing,* vol. 408, pp. 11 – 23, 2016.

[14] J. S. Khundrakpam, and D. Tanmay, "MLP-GA based algorithm to detect application layer DDoS attack'" *Journal of Information Security and Applications,* vol. 36, pp. 145 – 153, 2017.

[15] T. Zhiyuan, J Aruna, and H. Xiangjian, (2013). "A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis," *IEEE transactions on parallel and distributed systems,* 2013.

[16] A. N. Muhammad, B. S. Kudang N. Dodi and M. Akhiruddin, "A Comparison Study of Kernel Functions in the Support Vector Machine and Its Application for Termite Detection. *Open Access Journal of Information Science*, vol. 9, no. 5, pp. 418 – 424, 2018.

[17] A. Rimah, B. A. Dorra and E. Noureddine, "Practical Selection of SVM Supervised Parameters with Different Feature Representations for Vowel Recognition," *International Journal of Digital Content Technology and its Applications,* vol. 7, no. 9, 2013.

[18] L. Kuan-Cheng, Z. Kai-Yuan, H. Yi-Hung, C. H. Jason and Y. Neil, "Feature selection based on an improved cat swarm optimization algorithm for big data classification. *Springer Science+Business Media New York, 2016.*

[19] I. Israa and S. Mustafa, "Improved Cat Swarm Optimization for Efficient Motion Estimation," International Journal of Hybrid Information Technology. Vol. 8, no. 1, 279 – 294, 2015.

[20] M. Kumar, S. K. Mishra and S. S. Sahu, "Cat Swarm Optimization Based Functional Link Artificial Neural Network Filter for Gaussian Noise Removal from Computed Tomography Images," *Applied Computational Intelligence and Soft Computing,* pp. 1 – 5. 2016

[21] M. Bahrami, O. Bozorg-Haddad, and X. Chu, "Advanced Optimization by Nature-Inspired Algorithms," *Studies in Computational Intelligence*, pp. 9 – 18, 2018.

[22] K. Jasmin, J. Samed, and S. Abdulhamit, "An effective combining classifier approach using tree algorithms for network intrusion detection," *Neural Computing & Applications,* 2016.

[23] S. Alan, E. Richard and T. R. Overill, "Detection of known and unknown DDoS attacks using Artificial Neural Networks," 2015.

[24] A. R. A. Rana, W. Xi-Zhao, Z. H. Joshua, A. Haider and H. Yu-Lin, "Fuzziness based semi-supervised learning approach for intrusion detection system," *Information Sciences*, vol. 378, pp. 484 – 4 97, 2017