Anomaly Intrusion Detection based on a Hybrid Classification Algorithm (GSVM)

Salima Benqdara
University of Benghazi
Benghazi, Libya

ABSTRACT
One of the major problems in support vector machines (SVM) is the selection of optimal parameters that can establish an efficient SVM to achieve better output with an acceptable level of accuracy. In this paper, proposed a hybrid classification algorithm (GSVM) based Gravitational Search Algorithm (GSA) and support vector machines (SVM) to optimize the accuracy of the SVM classifier by detecting the subset of the best values of the kernel parameters for the SVM classifier. In the GSVM classifier, the GSA is introduced as an optimization technique to optimize the SVM parameters. The GSVM algorithm evaluated using KDD CUP 99 data set and compared to the outperformance of the original SVM algorithms. The results show that the performance of GSVM algorithm has a higher detection rate with lower false positive rate.

General Terms
Security, Algorithms.

Keywords
Network Intrusion Detection, ensemble clusters, unlabeled data.

1. INTRODUCTION
The importance to safeguard computer network against confidentiality, integrity and availability breaches is an important issue and intrusion detection plays vital role in ensuring a secured network. Security policies or firewalls have difficulty in preventing such attacks because of the hidden vulnerabilities contained in software applications. Therefore, intrusion detection system (IDS) is required as an additional wall for protecting systems despite the prevention techniques. The choice of classifiers to classify the data traffic is an issue because they can affect the accuracy and classification of an attack. [13, 14].

Various machine learning methods are used to classify intrusion detection datasets such as the decision tree, nave Bayesian, neural network and (SVM). The SVM is a margin-based classifier based on small sample learning with good generalization capabilities, and is commonly used in the application of classification [3, 15]. The SVM outperforms in the important aspect of robustness and efficiency in the network classification. It can manage the problem of imbalanced attacks which can otherwise lead to poor detection performance. This problem occurs due to the small learning sample size of low-frequent attacks compared to high-frequent attacks. Moreover, SVMs outperform the neural network in the important aspects of scalability, training time and prediction accuracy [16]. SVM is commonly used in IDSs because of its robustness and efficiency in the network classification [9]. However, one of the primary problems of SVM is how to select the kernel function and its parameter values. This problem is a crucial step in handling a learning task with SVM since it has an impact on the classification accuracy [3, 10].

In this paper, a hybrid classifier is designed based on a combination of the GSA and SVM algorithms. The main purpose of designing the GSVM classifier is to optimize the accuracy of the SVM classifier by detecting the subset of the best values of the kernel parameters for the SVM classifier. The performance of the proposed approach has been tested on KDD CUP 99 data set, and the results have been compared with an original SVM algorithm. The rest of the paper is organized as follows: Section 2 discusses the related works on the hybrid approach in IDS. In section 3 present a brief overview of the gravitational search algorithm to provide a proper background. Section 4 and 5 present proposed approaches and data used. Section 6 describes the flow of the experiment. The results and discussion of findings are presented in Section 7. Finally, Section 8 concludes the paper.

2. RELATED WORK
In this section discuss the published papers related to work on the hybrid classification approach in IDS

Peddabachigari et al. (2007), used two hybrid approaches for modelling IDS. Decision trees (DT) and support vector machines (SVM) are combined as a hierarchical hybrid intelligent system model (DT– SVM) and an ensemble approach combining the base classifiers. In this model, the training set is passed through the DT classifier to generate leaf-node information. Then, the SVM classifier is trained using the training set together with leaf-node information (as an additional attribute) to produce the final output.

Shih et al. (2008), a particle swarm optimization-based approach, capable of searching for the optimal parameter values for SVM to obtain a subset of beneficial features. The PSO SVM approach is applied to eliminate unnecessary or insignificant features, and effectively determine the parameter values, in turn improving the overall classification results.

Kuang et al. (2014) proposed a new intrusion detection system composed of kernel principal component analysis (KPCA) and GA with SVM. The N-KPCAGA-SVM system consists of two stages. In the first stage, KPCA is used to reduce the dataset and extract the features of the normalized data. The second stage deals with the detection classifier. The GA is used to optimize the accuracy of the SVM classifier by detecting the subset of the best values of kernel parameters for the SVM classifier. The results showed that the classification accuracy of the proposed system achieved a faster convergence speed and better detection accuracy compared with a single SVM classifier.

Dastanpour et al. (2014) presented an approach for an IDS composed of the ANN algorithm and GSA optimization. The proposed system consists of two stages. In the first stage, the ANN algorithm is executed on the training data set and the recognition results of the ANN are sent to the next stage. In the second stage, the recognition results of the ANN are classified by the hybrid GSA-ANN algorithm. The KDD 99 dataset was used to evaluate the proposed system, with the

results showing that the GSA-ANN hybrid approach achieved high accuracy compared with a single ANN algorithm.

Manekar and Waghmare (2014) proposed an IDS based on the machine learning technique. The proposed system consists of two machine learning algorithms: SVM and PSO. In the first step in the proposed system, the PSO algorithm is used to optimize the value of the C and parameters and important features for the SVM. In the second step, the parameters and features are used to train the SVM. The results showed that the proposed system k improved the detection accuracy compared to a single SVM classifier.

3. Gravitational Search Algorithm (GSA)

Gravitational search algorithm is one of the latest heuristic optimization algorithms, which was first introduced by Rashedi et al. (2009) as a new stochastic population-based optimization tool based on the metaphor of gravitational interaction between masses. The GSA is constructed on the law of Newtonian Gravity; every particle in the universe attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them. In the algorithm, all the individuals can be viewed as objects with masses. The objects attract each other by the gravity force, and the force makes all of them move towards the ones with heavier masses. The objects transform information by the gravitational force, and the objects with heavier masses become heavier [6].

To describe the GSA, consider a system with N masses (agents) in which the position of the ith mass is defined as follows:

$$X_i = \left(x_i^1, \dots\dots, x_i^d, \dots\dots\dots, x_i^n\right) \quad (1)$$

The mass of each agent is calculated after computing a current population's fitness as follows

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \quad (2)$$

Where

$$m_i(t) = \frac{fit_i(t) - worst(t)}{\sum_{j=1}^{N}(fit_i(t) - worst(t))} \quad (3)$$

Where fit_i (t) represent the fitness value of the agent i at time t. best (t) and worst (t) are the best and worst fitness of all agents, respectively and defined as follows:

$$best(t) = \min_{j\in\{i,\dots,N\}} fit_j(t) \quad (4)$$

$$worst(t) = \max_{j\in\{i,\dots,N\}} fit_j(t)$$

To compute the acceleration of an agent, the total forces from a set of heavier masses that act on it should be considered based on the law of gravity (Equation 5), followed by the calculation of an agent acceleration using a law of motion (Equation. 6. After that, the next velocity of an agent is calculated as a fraction of its current velocity added to its acceleration (Equation 7). Then, its next position can be calculated using Equation8.

$$F_d^i(x) = \sum_{j\in kbest, j\neq i} rand_j G(t) \frac{M_j(t) M_i(t)}{R_{i,j}(t) + \varepsilon} \left(x_j^d(t) - x_j^d(t)\right) \quad (5)$$

$$a_i^d(t) = \sum_{j\in kbest, j\neq i} rand_j G(t) \frac{M_j(t)}{R_{i,j}(t) + \varepsilon} \left(x_j^d(t) - x_i^d(t)\right) \quad (6)$$

$$V_i^d(t+1) = rand_i \times V_i^d(t) + a_i^d(t) \quad (7)$$

$$X_i^d(t+1) = X_i^d(t) + V_i^d(t+1) \quad (8)$$

4. PROPOSED APPROACH

In this paper, a hybrid classifier is designed based on a combination of the GSA and SVM algorithms. The main purpose of designing the GSVM classifier is to optimize the accuracy of the SVM classifier by detecting the subset of the best values of the kernel parameters for the SVM classifier. In the GSVM classifier, the GSA is introduced as an optimization technique to optimize the SVM parameters. The GSA starts with n-randomly selected agents and searches for the optimal agent iteratively. Each agent is an m-dimensional vector and represents a candidate solution. The SVM classifier is built for each candidate solution to evaluate its performance through evaluation of the fitness function. The fitness function value is based on the classification accuracy of the SVM classifier. The GSA guides the selection of potential subsets that lead to the best prediction accuracy. The detailed steps of the algorithm are explained in the Algorithm

5. EXPERMENT DATA

The KDD Cup1999 dataset was obtained from the 1998 DARPA Intrusion Detection, Evaluation Program and prepared by MIT Lincoln Labs. It is the largest publicly available sophisticated benchmark for researchers to evaluate intrusion detection algorithms or machine learning algorithms. The KDD Cup 1999 dataset contains nine weeks of raw transmission control protocol (TCP) dump data from simulated US Air Force local area network which is injected with multiple attacks. Each TCP/IP connection has a total of 41 qualitative and quantitative features where some are derived features. Features were labelled from 1 to 41 and they are termed as f1, f2, f3,… and f41. The type of attacks belongs to four main categories, namely, Denial of Service (DOS), Remote to Local (R2L), User to Root (U2R) and Probing. This study, as in most of the research in the literature, used the 10 % version of the data set consisting of 494,020 traffic connections with a similar ratio of attacks as in the full dataset [7, 8].

6. EXPERMENTAL SETUP

The training and testing data used in this study was comprised of 5,092 and 6,890 records respectively. The composition of these sample data maintains the actual distribution of KDD Cup 1999 data. In this paper, the experiments were performed separately for all four attack classes (probe, DOS, R2L and U2R) by randomly selecting data corresponding to that particular attack class and normal data only. Data scaling was done to ensure the training dataset was within the range of [0, 1]. The number of iterations was 500 iterations and all the experiments were repeated 500 times (iterations) and the results were averaged. The parameter settings used in the experiment are shown in Table 1.

In this paper, the GSVM was designed to improve the classification process. This section describes the experimental setup to evaluate the GSVM classifier. In order to evaluate and compare the effectiveness of the proposed classifier, this study used the KDD cup 1999 dataset. The dataset contains 4,940,000 traffic connections consisting of normal network

traffic and 24 types of attacks from four categories of attacks, namely, probe, DOS, U2R and R2L attacks. . In this study, the experiments were performed separately for all four attack classes (probe, DoS, R2L and U2R) by randomly selecting data corresponding to that particular attack class and normal data only. Data scaling was done to ensure the training dataset was within the range of [0, 1]. In this study, the number of iterations was 500 iterations and all the experiments were repeated 500 times (iterations) and the results were averaged. The training data set is feeding into the hybrid GSVM classifier, the GSA algorithm is used to seek the optimal parameters C, σ in the SVM. Through the training process, the parameter values, and training dataset are used for building the SVM classifier. Then feed the test dataset into the GSVM classifier. Standard measurements, such as the detection rate (DR), false positive rate (FPR), and detection accuracy rate (ACC), for evaluating the performance of GSVM classifier are shown in Table 2.

7. RESLUTS AND DISCUSSION

The GSVM-classifier was evaluated in terms of the overall accuracy, detection rate and false positive rate. In order to evaluate the effectiveness of the detection classifier. The GSVM classifier was validated using the KDD Cup 1999 test dataset. The performance results of the GSVM-classifier benchmarked against the performance results of the SVM classifier using the KDD Cup 1999 test dataset. Table 3 presents a summary of the results achieved by the GSVM classifier and SVM classifier for detection accuracy, detection rate, and false positive rate for all traffic classes. The results showed that the GSVM classifier outperformed the SVM classifier in terms of detection rate and detection accuracy in all five traffic classes. The GSVM classifier achieved a high detection rate and detection accuracy with an average rate of 96.85 % and 97.05 %, respectively. However, the SVM classifier achieved 90.10 % and 77.16 % for the detection rate and detection accuracy, respectively. According to the results, the GSVM classifier achieved a lower false positive rate compared to the SVM classifier in all five classes (with an average rate of 0.03 %). In the experiments, the detection accuracy improved by 6.95 % while the false positive rate reduced by 0.07 % when using the GSVM classifier.

Figure 1 illustrates the detection accuracy of the GSVM classifier and SVM classifier with respect to the five traffic classes. The results showed that the GSVM classifier obtained high detection accuracy compared to the SVM classifier. The results also showed that the GSVM classifier achieved the highest detection accuracy on the DoS and normal classes against the SVM classifier. The GSVM classifier and SVM classifier obtained the lowest detection accuracy on the U2R class; however, the GSVM classifier obtained similar detection accuracy on the normal and DoS classes. The GSVM classifier had the highest detection accuracy for all classes. In addition, the results showed an improvement in the U2R class and R2L class. This occurs because the difficulty of correctly detecting the imbalanced dataset is reduced by optimizing the accuracy of the SVM classifier. Thus, the detection effectiveness is improved when the GSVM classifier implements the GSA to optimize the kernel function parameters for the SVM classifier.

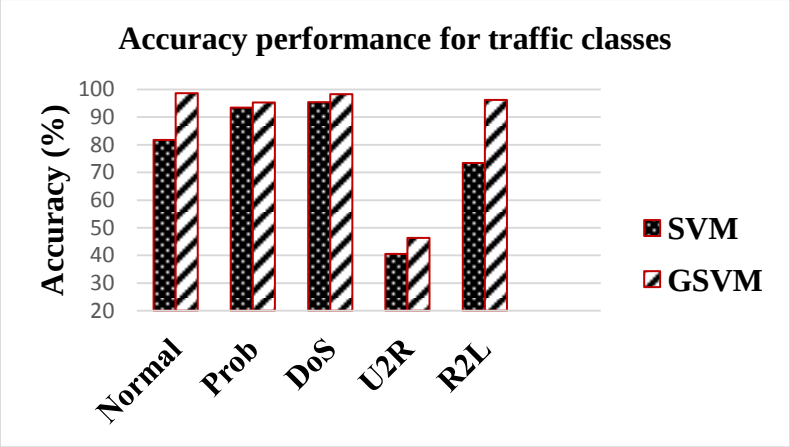

Fig. 1. Detection accuracy performance of GSVM and SVM Classifiers

Figure 2 and Figure 3 illustrate the comparison in terms of overall detection accuracy and false positive rate for the GSVM classifier and the SVM classifier. The results on the detection accuracy (Figure 2) showed that the GSVM classifier achieved a high accuracy with an average rate of 97.05 %. However, the SVM classifier achieved 77.16 % for the detection accuracy. The detection accuracy for the GSVM classifier improved by 6.95% as compared to the SVM classifier. The results on the false positive rate (Figure 3) showed that the GSVM classifier achieved a lower false positive rate compared to the SVM classifier (with an average rate of 0.03 %).The false positive rate for the GSVM classifier reduced by 0.07 % compared to the SVM classifier. The results, as presented in the figures, showed that the GSVM classifier outperformed the SVM classifier in terms of detection accuracy and false positive rate because it included the GSA as an optimization technique to optimize the SVM parameters. The results indicated that the detection effectiveness was improved by optimizing the accuracy of the SVM classifier to enhance the classification process.

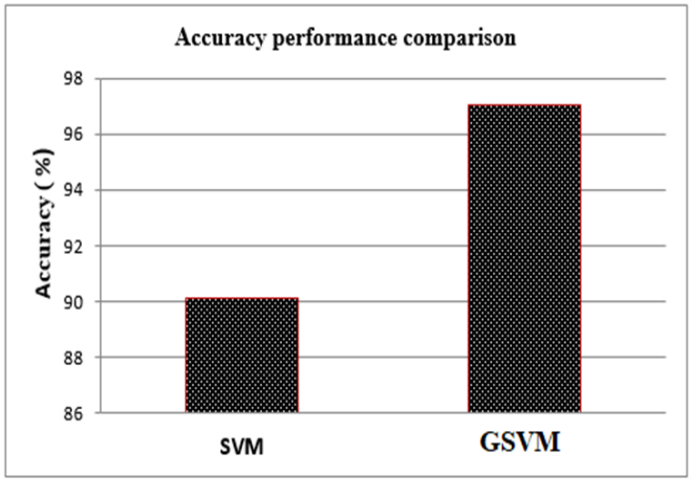

Fig 2: Detection accuracy comparison of GSVM and SVM

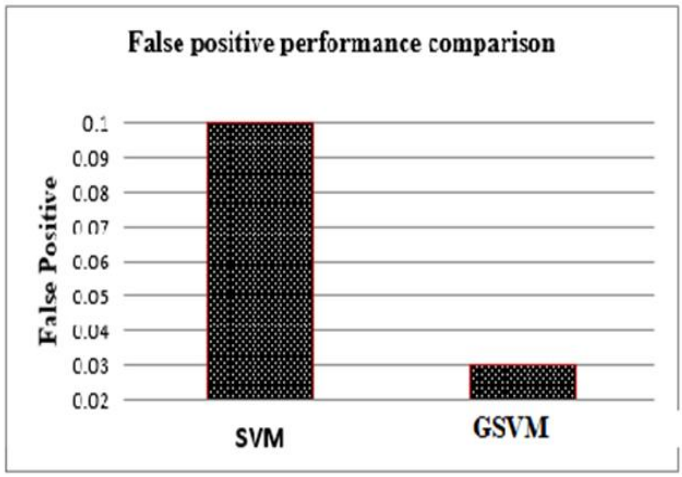

Fig 3:. False positive rate comparison of GSVM and SVM

8. CONCLUSION

In this paper, propose a new hybrid GSVM classifier was designed to enhance the classification process of the detection classifier. In the GSVM classifier, the GSA is used to optimize the accuracy of the SVM classifier by detecting the subset of the best values of kernel parameters for the SVM classifier. The GSA avoids being trapped in the local optima and, by following the best results obtained by every individual object, obtains accurate results. Moreover, it has the capability to optimize and improve the performance of a classification classifier. In the experiments, the detection accuracy improved by 6.95 % while the false positive rate reduced by 0.07 % when using the GSVM classifier. In addition, the results showed an improvement in the U2R class and R2L class. This occurs because the difficulty of correctly detecting the imbalanced dataset is reduced by optimizing the accuracy of the SVM classifier. Thus, the detection effectiveness is improved

when the GSVM classifier implements the GSA to optimize the kernel function parameters for the SVM classifier.

9. REFERENCES

[1] Dastanpour, A., Ibrahim, S., Mashinchi, R. and Selamat, A.," Using Gravitational Search Algorithm to Support Artificial Neural Network in Intrusion Detection System", SmartCR, VOL4 (6), (2014), 426–434.

[2] Hongying Zheng.,"An Efficient Hybrid Clustering-PSO Algorithm for Anomaly Intrusion Detection", JOURNAL OF SOFTWARE, VOL 6(12), (2011), 306-313.

[3] Kuang, F., Xu, W. and Zhang, S., "A novel hybrid KPCA and SVM with GA model for intrusion detection", Applied Soft Computing. VOL.18, (2014), 178–184.

[4] Manekar, V. and Waghmare, K. ,"Intrusion Detection System using Support Vector Machine (SVM) and Particle Swarm Optimization (PSO) ", International Journal of Advanced Computer Research, VOL 4(3), (2014),25–30.

[5] Mukkamala, S., Sung A. and Abraham, A. (2003). Intrusion Detection Using Ensemble of Soft Computing Paradigms. Proceedings of 3rd. International Conference on Intelligent Systems Design and Applications. 239-248.

[6] Rashedi,E., Nezamabadi,H. and Saryazdi, S., "GSA: A gravitational search algorithm", Information Sciences, VOL. 179,(2009), 2232-2248.

[7] Peddabachigari, S. Abraham, A. Grosan, C. J., "Thomas: Modeling intrusion detection system using hybrid intelligent systems", In Journal of Network and Computer Applications, VOL. 30, (2007), 114-132.

[8] Shih.W, Kuo.C , Shih.C and Zne. L. , "Particle swarm optimization for parameter determination and feature selection of support vector machines", Expert Systems with Applications,(2008) 1817–1824.

[9] Tsai, C., Hsu, Y., Lin, C. and Lin, W. , " Intrusion Detection by Machine Learning: A Review", Expert Systems with Applications. VOL 36(10), (2009)11994-12000.

[10] Ranaee, V., Ebrahimizadeh, A., Ghaderi, R., "Application of the PSO–SVM model for recognition of control chart patterns", ISA Transactions, VOL 49 (4), (2010), 577–586.

[11] Vapnik, V., "Statistical learning theory", Wiley, New York, (1998).

[12] Wang, J., Li, T. and Ren, R. (2010b). A real Time IDS Based on Artificial Bee Colony-Support Vector Machine Algorithm. In The Third International Workshop 133 on Advanced Computational Intelligence (IWACI). IEEE, 91–96.

[13] Manikandan, R., Oviya, P. and Hemalatha, C.,"A New Data Mining Based Network Intrusion Detection Model. Journal of Computer Application", VOL. 5, (2012), 1–10.

[14] Tsai, C.-F. and Lin, C.-Y. , "A triangle Area based Nearest Neighbors Approach to Intrusion Detection", Pattern Recognition. VOL 43(1), (2010), 222–229.

[15] Majid, A., Khan, A. and Mirza, A. M., "Combination of support vector machines using genetic programming", International Journal of Hybrid Intelligent Systems. VOL3 (2), (2006), 109–125.

[16] Srinivas, M. and Andrew, H., "Feature selection for intrusion detection using neural networks and support vector machines", Transportation Research Board, winter, (2003), 1–11.

[17] Tsai, C.-F., Hsu, Y.-F., Lin, C.-Y, and Lin, W.-Y, "Intrusion Detection by Machine Learning: A review ", Expert Systems with Applications, VOL 36 (10), (2009), 11994– 12000.

[18] Saini, G. and Kaur, H. , "A Novel Approach Towards K-Mean Clustering Algorithm With PSO", International Journal of Computer Science and Information Technologies, VOL 5(4), (2014), 5978–5986.

10. APPENDIX

Algorithm GSVM

Initialize the position and velocity of agents randomly,Set the parameters of GSA-SVM (N, G0, ε, tmax)
Repeat
For each mass $i = 1, 2, ..., Ndo$
Train SVM
Evaluate fitness function of each agent
Calculate mass for all of the agents
Calculate force for all of the agents
Calculate acceleration for all of the agents
Update the velocity position of agents
Update the position of the agents
End For
Until: cluster centroid not change or max-iter
Retrain SVM and classification results

Table 1. Key parameter values used in GSVM

Parameter	Value/Qty	Description
N	5	Number of agents
max_it	500	Maximum number of iterations
Threshold	2.9	Based on the experiment
rand	0–1	Two uniformly distributed random numbers between 0 and 1
G0	1	Gravitational constant
ε	1	Small value to avoid division by zero
v_i^d	Variable	The velocity of *i*th agent in the *d*th dimension
a_i^d	Variable	The acceleration of the agent *i* in direction *d*th
x_i^d	Variable	The position of *i*th agent in the *d*th dimension
$R_{i,j}$	Variable	Euclidean distance between two agents *i* and *j*
F_i^d	Variable	The total force that acts on agent *i* in a dimension
ξ_i	Variable	Non-negative slack variable.
C	Variable	Penalty parameter, that control of the decision function and the number of training samples.

Table 2. Description of performance measures

Performance Measures		Description
Percentage (%) Classification	Accuracy	Correctly classified as normal and attacks into their respective classes. It quantifies the discriminating capability of the classifier/model when presented with input data. $\frac{TN + TP}{TN + TP + FN + FP}$
	True Positive Rate (TPR) also known as Detection Rate (DR)	Measure the frequency of the targeted data correctly classified by the classifier/model as normal. $\frac{TP}{TP + FN}$
Error Percentage (%)	False Positive Rate (FPR) also known as False Alarm Rate (FAR)	Average number of normal traffic wrongly identified as malicious traffic (false alarm rate) $\frac{FP}{TN + FP}$

Table 3: Performance results for GSVM and SVM

Class	SVM classifier			GSVM classifier		
	ACC (%)	FPR (%)	DR (%)	ACC (%)	FPR (%)	DR (%)
Normal	81.64	0.19	49.97	98.52	0.03	99.99
Prob	93.40	0.11	96.67	95.22	0.09	100
DoS	95.29	0.00	84.85	98.28	0	94.74
U2R	40.55	0.41	26.30	46.26	0.41	46.3
R2L	73.31	0.19	22.25	96.17	0	92.67
AVG	**90.10**	**0.1**	**77.16**	**97.04**	**0.03**	**96.85**

Legend:
In bracket is %; ACC=Detection accuracy, FP=False positive rate; DR=Detection rate
AVG: this average excluded the up normal value