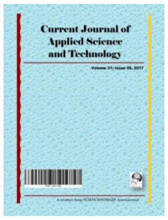

Current Journal of Applied Science and Technology

28(3): 1-14, 2018; Article no.CJAST.42730
ISSN: 2457-1024
*(Past name: British Journal of Applied Science & Technology, Past ISSN: 2231-0843,
NLM ID: 101664541)*

A Machine Learning Algorithm Based on Inverse Problems for Cyber Anomaly Detection

Ali Sever[1*]

[1]Department of Computer Information Systems, Pfeiffer University, Charlotte, NC, 28109, USA.

Author's contribution

The sole author designed, analyzed, interpreted and prepared the manuscript.

Article Information

DOI: 10.9734/CJAST/2018/42730
Editor(s):
(1) Dr. Orlando Manuel da Costa Gomes, Professor of Economics, Lisbon Accounting and Business School (ISCAL), Lisbon Polytechnic Institute, Portugal.
Reviewers:
(1) Lei Zhang, University of Pittsburgh, USA.
(2) Amirhessam Tahmassebi, Florida State University, USA.
(3) Tomasz Andrysiak, University of Technology and Life Sciences in Bydgoszcz, Poland.
Complete Peer review History: http://www.sciencedomain.org/review-history/25552

Original Research Article

Received 23rd April 2018
Accepted 11th July 2018
Published 14th July 2018

ABSTRACT

With the rapid rate of technological advance, digital communications have become an integral part of our lives in e-commerce, healthcare, education, and government. As the cyber world has expanded and become more complex, it has also generated severe threats to cyber security. Adversarial attacks such as anomalies and misuses are hard to detect with conventional methods as these cyber activities look very similar to genuine ones. There are many problems in anomaly and misuse detection of cybersecurity which can be considered as an inverse problem. In this paper, we have modeled anomaly detection system, Inverse Machine Learning Algorithm (IMLA), based on an inverse model approach with Riesz kernel and applying software system development concepts at each phase. For evaluation, the proposed approach IMLA have been compared with other state of the art supervised learning models. The experiments show the effectiveness of the proposed model IMLA.

Keywords: Anomaly detection; inverse problems; machine learning; cyber analytics; data mining.

1. INTRODUCTION

The aim of this paper is to present an efficient approach for anomaly detection in cybersecurity. It is a crucial part for any cybersecurity application because information security personnel have to decide if a system is safe or not at any given moment. The high margin of

*Corresponding author: E-mail: delafener@gmail.com, ali.sever@pfeiffer.edu;

accuracy in the operational cybersecurity is very necessary and quite beneficial for both the users and security personnel. This paper analyses intrusion detection evaluation data set and provides a framework which can discriminate between good cybersecurity operations and the bad ones. This project focuses on the use of knowledge discovery methods with soft computing method to find a solution for this problem. A Hybrid classifier is proposed and used to analyze the data set and extract the decision rules from it. These rules enable cyber security managers to make effective decisions and classify the operations as good or bad. The objective of the model developed here is to maximize the security and minimize the risk on the behalf of the system.

Adversarial attack is a widespread term for a malicious entity using or involving a cyber communication that is interfering with or preventing normal cyber activities [1]. The intention can be to obtain unauthorized access to any cyber medium to compromise the confidentiality, integrity, and availability of an information system. According to the World Economic Forum, the rate of cybercrime, which used to be holding steady during the mid-2000s, had increased by 21 percent in 2017. Around 46% of Americans have been victim to cybercrime in the past years. The Nilson Report estimates that in 2021, losses will top $6 trillion, a 62% increase over the previous five years.

In this paper, we have used DARPA 1999 intrusion detection evaluation dataset that was collected for a total of five weeks. The data for the first three weeks was assigned as the training set and the last two weeks was assigned as the testing dataset [2].

2. ANOMALY DETECTION

Anomaly detection is a challenging field of cybersecurity. It is a very complicated task to distinguish anomalies from normal actions. The aim of every method is to classify the anomalies with normal behavior. It becomes very hard for humans to decide just by inspecting the operations with some detail.

Advancement in information technology allows information systems to electronically store all the details of actions. This led to the urgency of making a system which can automate access granting decisions so that the stress on cybersecurity managers can be reduced and efficiency could be improved. There are lots of methods that have been proposed and covered in the literature that use supervised and unsupervised learning [3-10]. Most of these methods have used different classifying techniques with reasonable accuracy, but these studies do not go beyond classification. Since it plays a crucial role in the anomaly evaluation procedures, there must be a proper justification of the approval and rejection of a specific action and an automated system which can assist the cybersecurity professional in this process.

Fig. 1 shows the general approach that is followed while assessing the anomaly using mathematical modeling. The training dataset which consists of different features and attributes about an event are fed to the network, after this the data is preprocessed. For example, if some data value is missing or corrupted data is found, it is removed to optimize the accuracy. In the next phase, a mathematical modeling technique is used for training the network and evaluating the anomalies and distinguishes between the normal and abnormal events to the cybersecurity manager.

3. MODEL ANALYSIS

Knowledge discovery is finding the patterns from the large dataset and extracting useful knowledge and information from it. It involves discovering patterns from the large datasets using different mathematical models. It is formally the intersection of Machine learning (ML) and artificial intelligence with statistics and data base system. There are different mathematical models available like neural networks, fuzzy, genetic algorithm, and support vectors along with their hybridizations that extract knowledge from the databases [16–20]. These soft computing methodologies are widely accepted for classification, clustering and predictions.

Many data properties estimation as knowledge discovery may be stated as a category of Inverse problems. The category that we are introducing here is composed by problems, when we are interested in properties of an object, as in anomaly detection in cyber platform. We consider the problem of anomaly detection as an inverse problem and discuss how it can be solved by machine learning techniques.

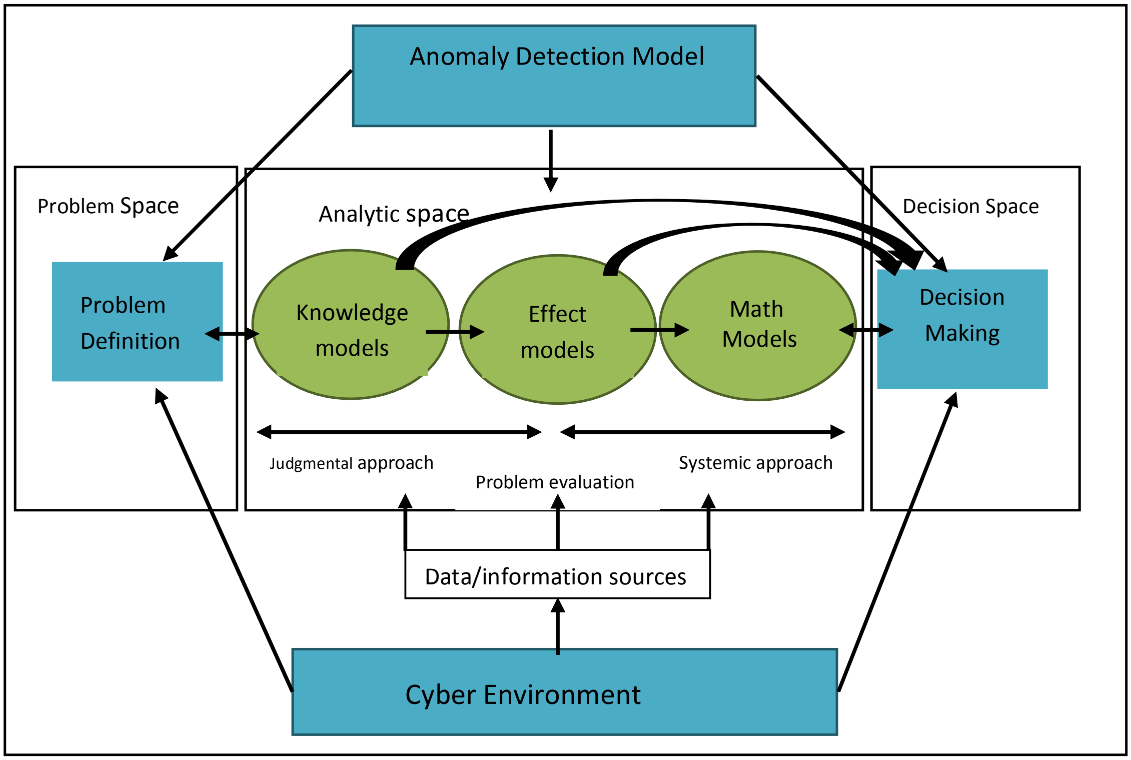

Fig. 1. A general approach of anomaly evaluation using mathematical modeling

In this paper, starting from a reformulation of the anomaly detection as an integral equation of inverse problems, we introduce an alternative machine learning algorithm derived by a well-known regularization method studied in [14] and [15]. Anomaly detection problem in Information Systems may be represented by an operator equation

$$Af = g \qquad (2.1)$$

where A is a compact linear operator with a Riesz type kernel. Here f represents anomaly status of information system and g represents results of the experiments with the training data sets. The eq. (2.1) is a first kind of Fredholm integral equation with a Riesz type kernel. Since the integral operator has an unbounded inverse, the solution of integral equation is ill-posed. Therefore a regularization technique is used to stabilize the solution by adding a new regularization parameter, stabilizer, and the equation (2.1) may be reformulated as a regularized inverse problem

$$(A^*A + \alpha \boldsymbol{I})\mathrm{f}(\alpha) = A^*\mathrm{g} \qquad (2.2)$$

New stabilizer component α in the Eq. (2.2) will adjust input/output relationship. The integral Eq. (2.2) has an unknown function *f* which will be recovered from pre-processed data and will be used to detect anomalies. We refer to [14] for the detailed formulation of Eq. (2.2).

The next step in the model is to discretize the Eq. (2.2) into matrix equation form. Then we show that this discrete reformulation can be employed to design a machine learning algorithm based upon a numerical solution of the equation (2.2).

We have implemented the IMLA algorithms of the anomaly detection using Python programming language. Python is very popular and professional to work with on machine learning projects. Python has particular tools which are very useful in working with machine learning and we have used those tools in our analysis. The following Software and Hardware requirements were used in our computations: Windows 10, Python version 3.6.5, Anaconda version 5, Jupyter notebook/ipython, Python libraries (tensorflow [16], numpy, pandas, sklearn), RAM: 16 GB, Processor i7 and above. We have used Python due to its' efficient and professional structure for implementation of our algorithm. We have used Python libraries such as pandas, numpy, matplotlib for plotting the

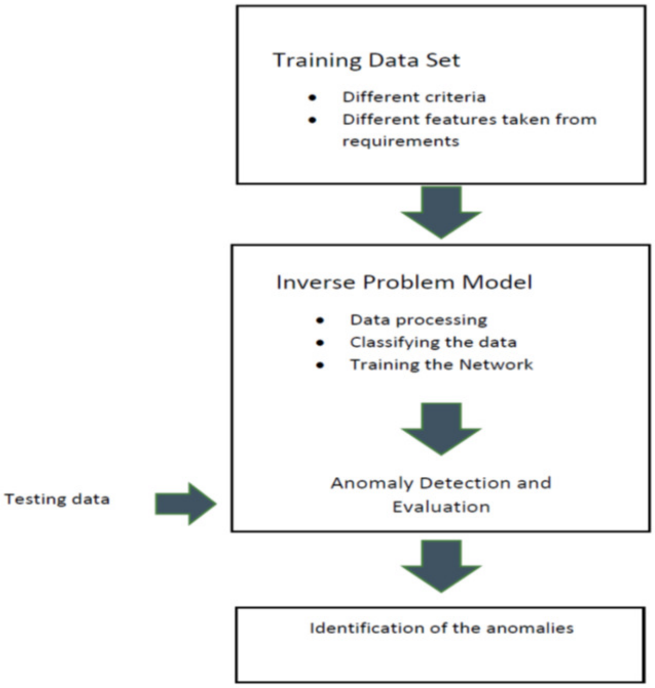

Fig. 2. Core architecture of the system

graphs, pickle etc. This work is divided into five modules such as Input module, Train module, Test module, Performance metrics module and Display module. We have implemented Inverse Machine Learning Algorithm (IMLA) and added some utility methods to measure the performance of the algorithms. The final output in the form of trained model can be hosted as a web service to further anomaly detections. Along with that, diagrams such as Architecture diagram, Sequence diagram of the model and training modelling diagrams are also drawn.

4. RESULTS AND DISCUSSION

To illustrate the functionality of the model, we start getting the data sets for anomaly detection from reliable sources. For the analysis, we used a data set that was used in research on big data and anomaly detection by the Machine Learning Group. The final data set has 125974 rows. It is a very unbalanced dataset which has only numerical input variables created via a PCA data reduction method [16]. This is a necessary step due to a large amount of data being processed.

As depicted in Fig. 3, this data went thru multiple preprocessing steps before it was used. To train and evaluate the model, the preprocessed data is then, randomly, split into two parts. 80 percent of the dataset will be used to actually train the model, while the rest will be used to evaluate the accuracy of this model. Then we followed a common practice in which the trained data is then fed into the classifier and the model is trained using this data as shown in Fig. 3. Once the training is completed, the test data is used to validate the accuracy of the model. The model is retrained if the required accuracy is not obtained. Graphical representations of these accuracies are plotted to get a better visualization of the trained model. The proposed model IMLA has been compared with some of the recent relevant models SVM with RBF kernel, Bayesian Network, and Decision Trees.

For a detailed definition and application of SVM, Bayesian Network, and Decision Trees in the context of intrusion detection we refer to [2].

The proposed model is designed and tested. Then the correctness of the algorithms is checked by comparing the various statistical evaluation metrics, such as accuracy, sensitivity, and so on.

The resulting metrics and graphs of applying SVM, Decision Trees, Bayesian Network, and IMLA algorithms on this dataset are provided to visualize and interpret them. As shown in Fig. 4 –

Fig. 9, all the models give a reasonably high accuracy. As in other machine learning algorithm evaluations, accuracy alone may not be a good indicator, and in some cases it may be misleading, so one has to analyze some other metrics that were obtained and provided as summary statistics from each model implementation since the data set is very balanced.

Table 1 is used for the evaluation of the models on complete or under-sampled data sets.

We, first, have tested the proposed model IMLA against SVM and compared them on the complete data set. The next experiments were to compare our model to SVM, Decision Trees, and Bayesian Network on under-sampled data.

For the first experiment, we start off by applying the IMLA classifier to the complete dataset. Accuracy of 96%, as shown in Fig. 4, was obtained, but the cases of actual anomalies are more important.

As we can see from the false positive count in the Fig. 4 that the number of anomalies is not that low, but in line with results of SVM method.

Since the IMLA did not work very well on the complete dataset because of the imbalanced nature of the dataset, let's apply the same on the Under-Sampled dataset. As we can see, we obtained a low false positive and a good true positive count, and accuracy of 96%. Therefore the proposed model looks good (Fig. 5).

Before moving on to a different classifier we applied Decision Trees from a standard scikit-learn library to check the correctness of the

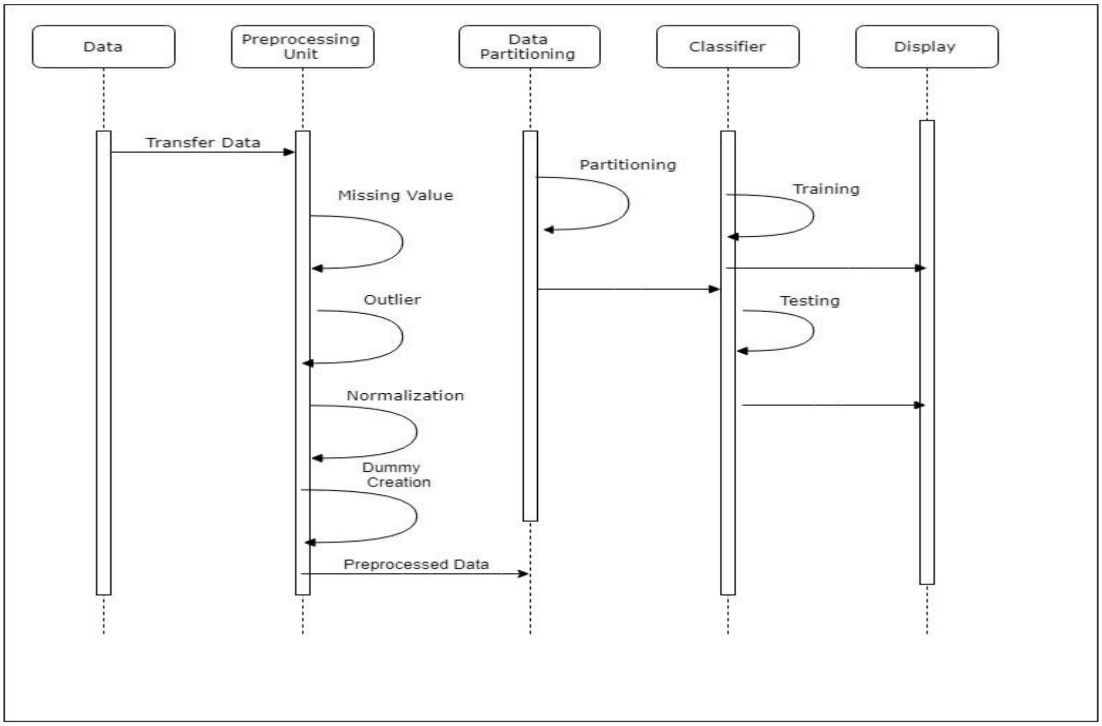

Fig. 3. Sequence diagram of the model

Table 1. Classification table

	Fail-Obs	**Suc-Obs**	
Fail-Pred	TN	FN	PN
Suc-Pred	FP	TP	PP
	ON	OP	

where TN = true negative, FN = false negative, FP = false positive, and TP = true positive.

	1	0
1	186	2883
0	32	73412

Stats	
Accuracy	0.9619
Precision	0.0606
Sensitivity	0.8532
Specificity	0.9622
Accuracy	0.9619
FPR	0.0337

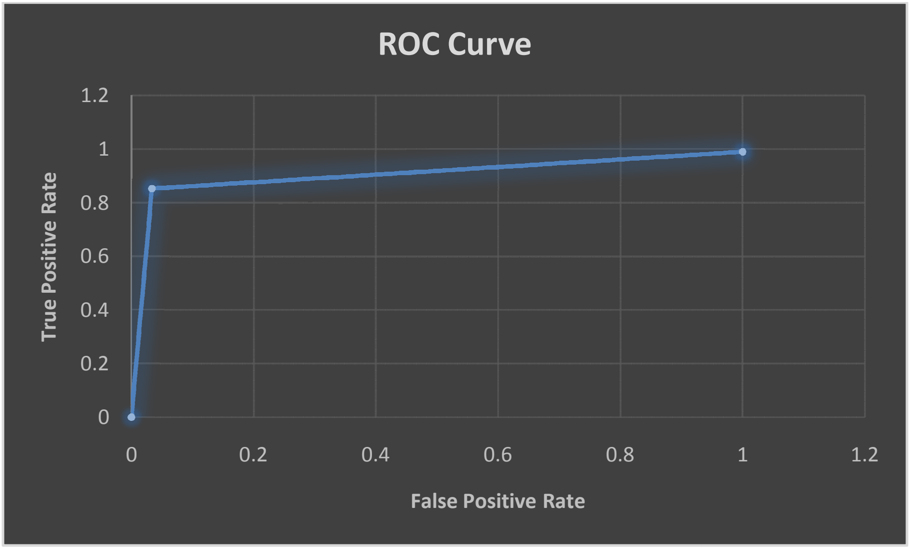

Fig. 4. IMLA on complete dataset

Table 2. Summarized results of the experiments

Experiments results											
IMLA			**Decision trees**			**Bayesian Networking (BN)**			**SVM**		
Under-sampled data											
	1	0		1	0		1	0		1	0
1	142	14	1	138	12	1	142	13	1	140	15
0	15	145	0	17	151	0	9	127	0	16	147
Original data											
	1	0								1	0
1	186	2883							1	176	2870
0	32	73412								42	73425

	1	0
1	142	14
0	15	145

Stats	
Accuracy	0.9619
Precision	0.9103
Sensitivity	0.9045
Specificity	0.9119
FPR	0.0881

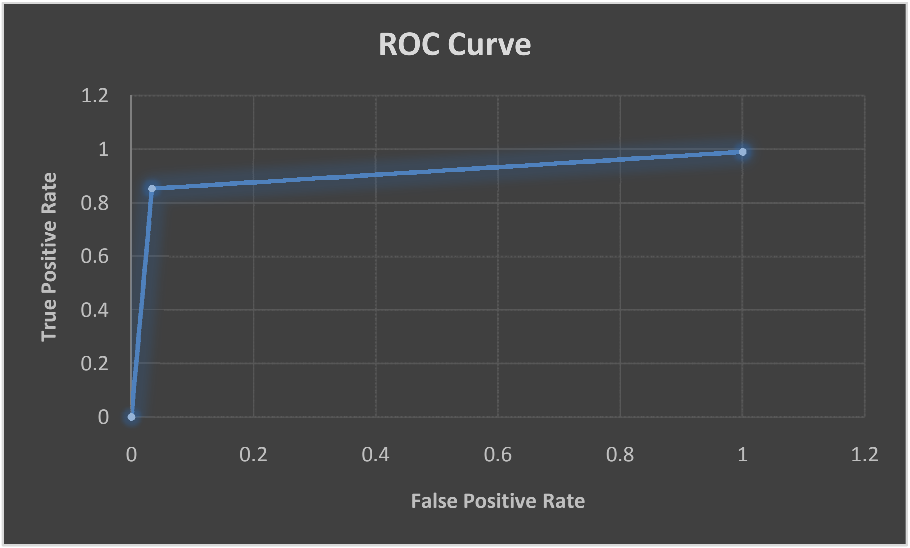

Fig. 5. IMLA on under-sampled dataset

	1	0
1	138	12
0	17	151

Stats	
Accuracy	0.9088
Precision	0.92
Sensitivity	0.8903
Specificity	0.9264
FPR	0.0736

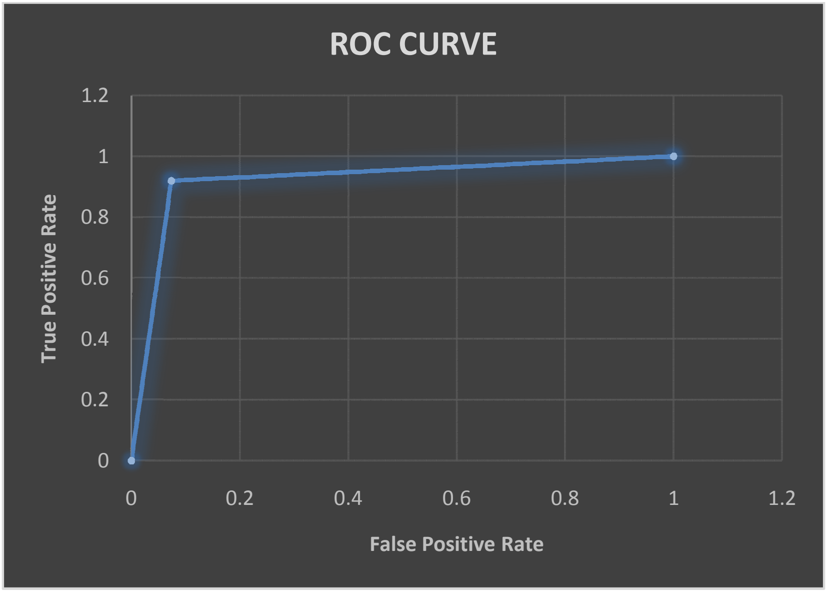

Fig. 6. Decision trees on under-sampled data

	1	0
1	176	2870
0	42	73425

Stats	
Accuracy	0.9619
Precision	0.0578
Sensitivity	0.8073
Specificity	0.9624
FPR	0.0376

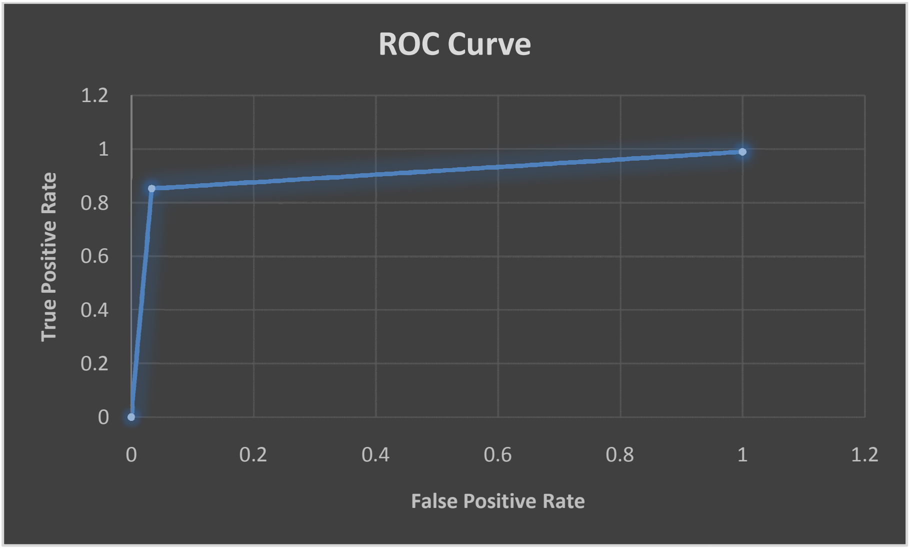

Fig. 7. SVM on original data

	1	0
1	140	15
0	16	147

Stats	
Accuracy	0.9025
Precision	0.9032
Sensitivity	0.8974
Specificity	0.9074
FPR	0.0926

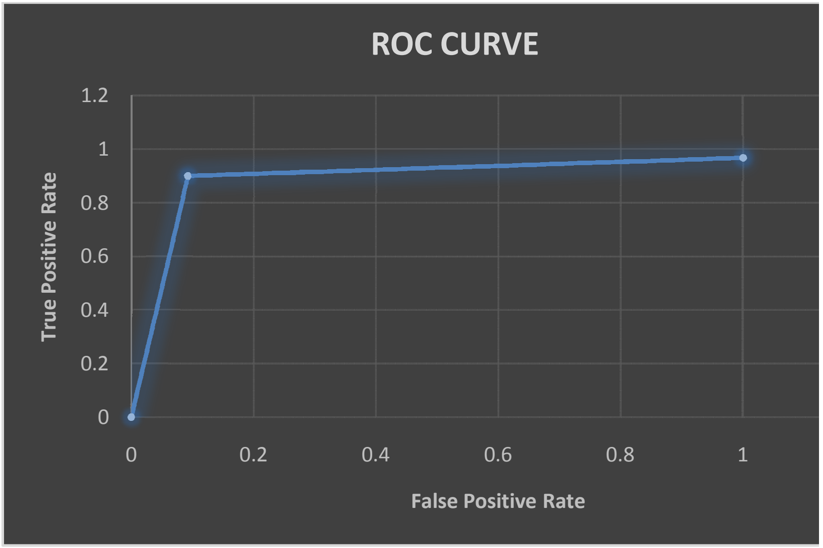

Fig. 8. SVM on under-sampled

	1	0
1	142	13
0	9	127

Stats	
Accuracy	0.9244
Precision	0.9161
Sensitivity	0.9404
Specificity	0.9071
FPR	0.0929

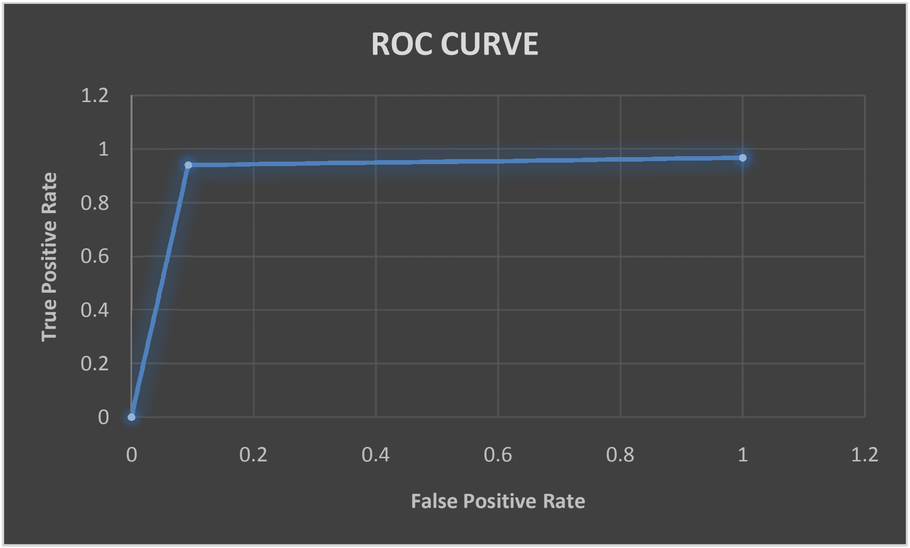

Fig. 9. Bayesian network using Sciket-learn library

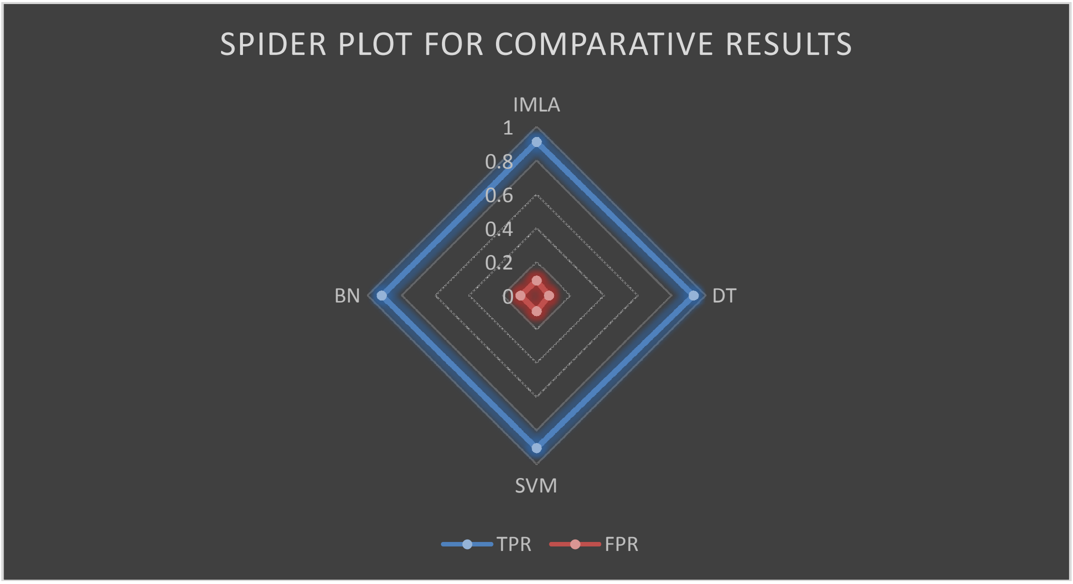

Fig. 10. Comparative study results of the four model

implemented algorithm on under-sampled dataset in Fig. 6. As we can see from the confusion matrix in Fig. 6, the experimental results are in line with those of the implemented version of proposed model IMLA.

The SVM classifier was applied to the complete dataset and an accuracy of 96% was obtained. As mentioned before, the instances that actually matter are the instances which are actually anomalies. As we can see from the false positive count in the below confusion matrix, 2870 instances were classified as normal actions, where as in reality they were anomalies. Therefore, The results in Fig. 7 on the complete data set indicate that the performance of SVM and proposed model IMLA are similar not only in terms of high anomaly detection accuracy of 96%, but also in terms of their low false positive counts while keeping a FPR at 0.0337 and 0.0376 respectively.

When we apply the same on the under-sampled dataset, as it can be seen in Fig. 8, we obtained a very low false positive count, a good true positive count, and accuracy of 90%. This model performs the same as IMLA for equally proportionate dataset.

Before concluding, we applied Bayesian Network algorithm from a standard scikit-learn library to check the correctness of the implemented algorithm. As we can see from the confusion matrix and the stats in Fig 9, they are in line with those of the implemented version of the proposed model IMLA.

When compared with other models on under-sampled data, the Bayesian Network algorithm had lower numbers of false negative and false positive with accuracy of 90% and false positive rate of 0.0926.

In Table 2, next, we summarized all of our findings from Fig. 4-9 in a single diagram to compare the proposed model with the other classifiers and see how it performs. The summarized results show that the proposed model results are in line with other machine learning methods except that the Bayesian Network perform slightly better on under-sample dataset. But the proposed model is appealing by its simplicity.

Finally, we provide a spider plot for the comparative study results in Fig. 10. The visualization results show that they are very much in line with each other.

5. CONCLUDING REMARKS AND FUTURE WORK

In this paper, a machine learning algorithm based on inverse problems for the cyber anomaly detection has been proposed. Comparative results show that the accuracy of the proposed model is reasonable around 90% which is very much in line with contemporary models. All of the algorithms; SVM, Decision

Trees, Bayesian Network, and IMLA over complete data set and under sampled data shows the model trained on under sampled data give high accuracy compared to the counterpart. The results of IMLA are also close to accuracy calculated using the contemporary models which verify the performance of our model. To make the model portable, this trained model can be hosted as a web service to further detect anomalies. As future enhancement, based on this analysis, anomaly detection web-interface schemes can be developed to make security decisions. Also, the answer to which data reduction methods in the preprocessing needs to be used should be investigated. We believe that, to enhance the performance, it is possible to modify the suggested method and devise a scheme using k-folds cross-validation on different number of folds. One has to make further analysis not only on how to improve the efficiency of the proposed model, but also on how to take advantage of parallel and distributed computing for detecting anomalies.

Overall, these results motivate further research of additional empirical studies of its parameters sensitivity and scalability using reinforced learning which enables the model to detect anomalies and train itself at the same time.

COMPETING INTERESTS

Author has declared that no competing interests exist.

REFERENCES

1. Dua S, Du X. Data Mining and Machine Learning in Cybersecurity, CRC Press; 2011.
2. Chen WH, Hsu SH, Shen SH. Application of SVM and ANN for intrusion detection. Computers and Operations Research 2015;32(10):2617–2634.
3. Bhuyan M, Bhattacharyya D, Kalita J. Network anomaly detection: Methods systems and tools. IEEE Commun. Surv. Tuts. 2014;16(1):303-336.
4. Wu SX, Banzhaf W. The use of computational intelligence in intrusion detection systems: A review. Appl. Soft Comput. 2010;10(1):1-35.
5. IBM, Feb; 2015. Available:http://www.iss.net
6. Morel B. Artificial intelligence and the future of cybersecurity. Proc. 4th ACM Workshop Secur. Artif. Intell. 2011;93-98.
7. Brahmi H, Imen B, Sadok B. OMC-IDS: At the cross-roads of OLAP mining and intrusion detection in advances in knowledge discovery and data mining, New York, NY, USA. Springer. 2012;13-24.
8. Benferhat S, Kenaza T, Mokhtari A. A Naïve Bayes approach for detecting coordinated attacks. Proc. 32nd Annu. IEEE Int. Comput. Software Appl. Conf., 2008;704-709.
9. Blowers M, Williams J. Machine learning applied to cyber operations in Network Science and Cybersecurity, New York, NY, USA. Springer. 2014;55-175.
10. Bilge L, Sen S, Balzarotti D, Kirda E, Kruegel C. Exposure: A passive DNS analysis service to detect and report malicious domains. ACM Trans. Inf. Syst. Secur. 2014;16(4).
11. Bilge L, Sen S, Balzarotti D, Kirda E, Kruegel C, Robertson W. Disclosure: Detecting botnet command and control servers through large-scale net flow analysis. Proc. 28th Annu. Comput. Secur. Appl. Conf. (ACSAC'12). 2012;129-138,
12. Khan S. Rule-based network intrusion detection using genetic algorithms. Int. J. Comput. Appl. 2011;18(8):26-29.
13. Li Y, Xia J, Zhang S, Yan J, Ai X, Dai K. An efficient intrusion detection system based on support vector machines and gradually feature removal method. Expert Syst. Appl. 2012;39(1):424-430.
14. Sever A. Neural network algorithm to pattern recognition in inverse problems. Applied Mathematics and Computation 2013;22:484–496.
15. Sever A. A machine leaning algorithm based on inverse problems for software requirements selection. Journal of Advances in Computer Science and Mathematics. 2017;23(2):
16. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Vanderplas J. Scikit-learn: Machine learning in python. Journal of Machine Learning Research. 2011;2825-2830.
17. Tahmassebi A, Gandomi AH, McCann I, Schulte MH, Schmaal L, Goudriaan AE, Meyer-Baese A. An evolutionary approach for fMRI big data classification. In Evolutionary Computation (CEC), IEEE Congress on. IEEE. 2017;1029-1036.
18. Tahmassebi A, Gandomi AH, Schulte MH, Goudriaan AE, Foo SY, Meyer-Baese A. Optimized Naive-Bayes and decision

tree approaches for fMRI Smoking cessation classification. Complexity; 2018.

19. Tahmassebi A. Ideeple: Deep learning in a flash. In Disruptive Technologies in Information Sciences International Society for Optics and Photonics. 2018;10652: 106520S).

20. Tahmassebi A, Gandomi AH. Building energy consumption forecast using multi-objective genetic programming. Measurement. 2018;118: 164-171.

© 2018 Sever; This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sciencedomain.org/review-history/25552