An Investigation of SVM, RFC, and Extreme Machine Learning Techniques for Intrusion Detection Systems

[1]A. RANGAMMA, [2]G. MANASA, [3]D. KEERTHI REDDY

[123]*Assistant Professor, Department of CSE, Sri Indu College of Engineering and Technology Hyderabad-Telangana*

ABSTRACT

The detection of intrusions is a critical aspect of security systems, such as intrusion detection systems, firewalls, and mobile security devices. However, the performance of various intrusion detection techniques is a concern, as accuracy needs improvement to reduce false alarms and increase the detection rate. To address this issue, previous studies have utilized multilayer perception, support vector machines (SVM), and other techniques, but they have limitations and are not efficient for large datasets, such as system and network data. In this paper, we investigate the use of well-known machine learning techniques, namely SVM, random forest, and extreme learning machine (ELM), for efficient *classification of intrusion data. We evaluate their performance on the NSL KDD dataset, which is considered a benchmark in the assessment of intrusion detection tools. The results indicate that ELM outperforms other methods in terms of detection rate and reducing false alarms.*

Index Terms: Intrusion detection, extreme learning machine, false alarms, NSL KDD dataset, random forest, support vector machine.

Keywords: Intrusion detection rate, extreme learning machine, false alarms, NSL KDD dataset, random forest, support vector machine.

INTRODUCTION

Interruption poses a severe threat to security and is a significant cause of security breaches as it can quickly steal or erase data from computer and network systems. Interruption can also cause hardware damage, financial losses, compromise IT infrastructure, and lead to data inadequacy in cyber warfare. Hence, it is crucial to detect and prevent interruption. Various intrusion detection techniques are available, but their accuracy remains an issue, which depends on the detection and false alarm rate. Therefore, to reduce the false alarm rate and increase the detection rate, this research employs support vector machines (SVM), random forest (RF), and extreme learning machine (ELM) techniques, which have proven effective in classification. These techniques are validated on the NSL KDD dataset, an improved version

of the KDD, which is considered a benchmark in evaluating intrusion detection techniques[1].

RELATED WORK

In today's world, securing computer and network data is of utmost importance for both individuals and organizations. The use of intrusion detection systems is crucial to prevent compromised data. Recently, machine learning techniques have been proposed to enhance the performance of these systems. In this regard, Wang et al. [1] proposed an intrusion detection system based on support vector machine (SVM) and achieved 99.92% effectiveness rate. However, they did not mention dataset statistics, number of training and testing samples. Also, SVM's performance decreases with the inclusion of large data and is not suitable for analysing massive network traffic for intrusion detection. Kuang et al. [2] used a hybrid model of SVM and kernel principal component analysis (KPCA) with genetic algorithm (GA) for intrusion detection, and their system showed a 96% detection rate. They used the KDD CUP99 dataset, which is characterized by limitations such as repetition and feature selection based on the top percentage of the essential column from the essential space. SVM is also not a good choice for large data analyses such as high bandwidth network monitoring[4]. To detect, prevent, and resist unauthorized access, interruption detection systems are a significant aid. Hence, Broman and Reza [3] proposed an ensemble classifier method, which is a combination of particle swarm optimization (PSO) and SVM. This classifier outperformed other methods with 92.90% accuracy. They used the KDD99 dataset, which has the limitations. Additionally, SVM is not a good choice for large data analyses. Raman et al. [4] proposed an intrusion detection tool based on hypergraph genetic algorithm (HG-GA) for parameter setting and feature selection in SVM. They claimed that their method outperformed other existing methods with a 97.14% detection rate on an NSL KDD dataset. Teng et al. [5] developed their model based on decision trees (DTs) and SVMs, and they tested it on a KDD CUP99 dataset. The results showed an accuracy of 89.02%. However, SVM is not preferred for large datasets due to its high computation cost and poor performance. Farnaz and Jabbar [6] developed a model for an intrusion detection system based on random forest (RF). They tested the effectiveness of their model on an NSL KDD dataset, and their results showed a 99.67% detection rate compared to J48. The main limitation of the RF algorithm is that many trees may make the calculation slow for real-time prediction. Elbasan et al. [7] proposed a model of intrusion detection based on RF and

weighted k-means and validated their model over the KDD99 dataset. The system showed results with 98.3% accuracy. However, RF is not suitable for predicting real traffic due to its slowness, which is caused by the formation of many trees. Additionally, the KDD99 dataset has a few limitations as mentioned earlier[7].

Conceptual Model

The proposed framework consists of several crucial stages including dataset preparation, pre-processing, classification, and evaluation of results. Each stage plays a significant role in determining the overall performance of the model. The main objective of this study is to assess the performance of SWM, RF, and ELM classifiers in detecting interruptions. The interruption detection framework model proposed in this study is illustrated in Figure 1.

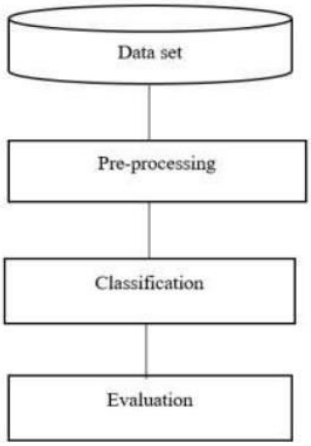


Figure 1 illustrates the proposed model for the intrusion detection system.

A. DATASET

Dataset choice for experimentation is a huge undertaking in light of the fact that the performance of the framework depends on the rightness of a dataset. The more accurate the information, the more prominent the viability of the framework. The dataset can be gathered by various methods, for example, 1)cleaned dataset, 2) simulated dataset, 3)testbed dataset, and 4)standard dataset [8]. Be that as it may, difficulties happen in the use of the initial three techniques. A genuine trap c technique is costly, though the sterilized strategy is perilous. The improvement of a reproduction framework is likewise intricate and testing. Moreover, extraordinary kinds of trap care required to show different system assaults, which is unpredictable and expensive. To defeat these dive societies, the NSL KDD dataset is utilized to approve the proposed framework for interruption detection[13].

B. PRE-PROCESSING

The classifier can't process the crude dataset in view of a portion of its representative highlights. In this manner, pre-handling is basic, in which non-numeric or emblematic highlights are killed or supplanted, on the grounds that they don't demonstrate imperative cooperation in interruption detection. Be that as it may, this procedure generates overhead including all the more preparing time; the classifier's design ends up complex and squanders memory and processing assets. Therefore, the non-numeric highlights are avoided from the crude dataset for enhanced performance of interruption detection frameworks[14].

C. CLASSIFICATION

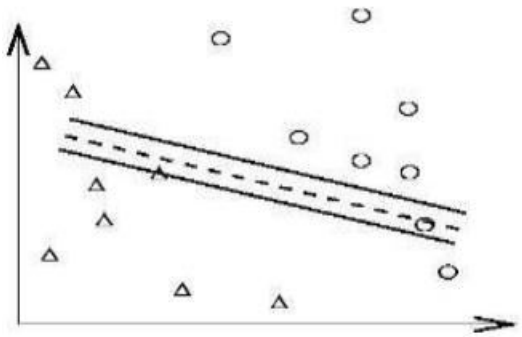
Setting a movement into ordinary and nosy classifications is the centre capacity of an interruption detection framework, which is known as a meddlesome examination motor. In this manner, distinctive classifiers have been connected as meddlesome investigation motors in interruption detection in the writing, for example, multilayer perceptron, SVM, credulous Bayes, self-sorting out guide, and DT. Be that as it may, in this investigation, the three unique classifiers of SVM, RF, and ELM are connected dependent on their demonstrated capacity in classification issues. Subtleties of every classification approach are given[13].

Support Vector Machine

The Support Vector Machine (SVM) is a machine learning technique that was introduced in 1992 by Boser, Guyon, and Vapnik. SVMs are widely used for classification and regression problems along with other supervised learning techniques. They are a type of linear classifiers that belong to a generalized group. The primary aim of SVM is to maximize the predictive accuracy of the learning algorithm, while avoiding overfitting to the data. The basic form of SVM is to maximize the distance between two different classes. When the classes are already known, it is called classification. The data set used to compute the boundary between classes is called the training set, while the data set used to test the efficacy of the method is called the validation set.SVMs are systems that use the hypothesis space of a linear function in a higher-dimensional feature space[14].

These systems are trained with a learning algorithm that uses a learning bias taken from the theory of statistical learning. SVM was initially famous for being used in image processing, pattern recognition, and medical diagnosis, but now it is playing a significant role in machine learning research. SVM is also used for many applications such as face analysis, handwriting analysis, engineering, business management, and much more. SVMs are also being used for regression-based applications in various domains. The Support Vector Machines SVM have

been developed by Vatnik and are yielding good results due to many challenging features and better empirical performance. SVM mainly uses the Structural Risk Minimization (SRM) rule, which is better than the traditional Empirical Risk Minimization (ERM) rule used by ordinary neural networks. This difference makes SVM perform with high accuracy to generalize the training data and make predictions. SVMs were developed primarily to solve the classification problem, but they are now also being used to solve regression problems.



The classifier in Figure 2(a) is overfitting to the data, as indicated by the significant overlap between the training data and the decision boundary. On the other hand, the classifier shown in Figure 2(b) is a better model, as there is very little overlap between the training data and the decision boundary

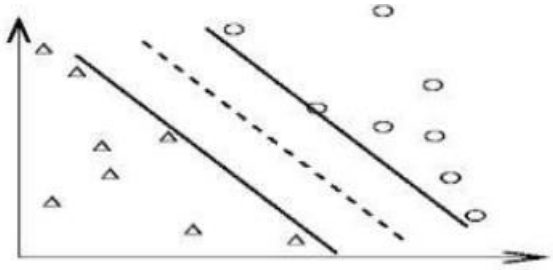


Fig 3: Over-Fitting classifier

Random Forest:

Random Forest is a classification technique based on decision trees that consists of a collection of tree predictors. Each tree is based on the values of a vector independently with the same distribution over all trees in the forest. The error rate of the model decreases as the number of trees in the forest increases. The accuracy of the classifier model mainly depends on the quality of individual trees in the forest and their interaction. Random Forest uses random selection of features to split each node resulting in error rates that can be compared. This method is robust even in the presence of high noise in the training data. Its internal workings include evaluating error, quality, and correlation, which are used to model the response to increasing the number of features used in data partitioning. Internal assessments

can also be used to find variable importance. Random Forest also provides regression functionality with training and testing datasets.

Extreme Learning Machine:

Extreme Learning Machine (ELM) is another name for single or multiple hidden layer feedforward neural networks that can be used to solve various classification, clustering, regression, and feature engineering problems. This learning algorithm involves the input layer, one or more hidden layers, and the output layer. In traditional neural networks, the tasks of adjusting the input and hidden layer weights are computationally expensive and time-consuming because it requires multiple rounds to converge. To overcome this issue,

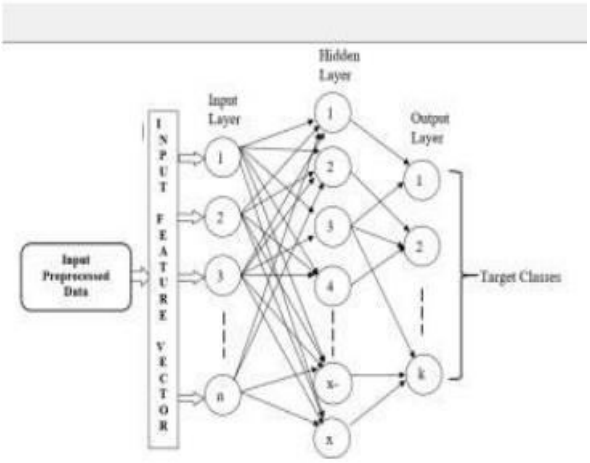


Fig-4: The architecture of the extreme learning machine for intrusion detection.

Huang et al. proposed a SLFN by selecting input weights and hidden layer biases subjectively to minimize training time. A detailed explanation of ELM can be found in Huang et al. and Qayyum et al. . The authors claim that these models can adapt faster and achieve higher generalization capability compared to other feedforward network models. ELM performs comparably to SVM or other state-of-the-art machine learning classifiers and has the best ability to perform well in highly complex datasets. The architecture of the proposed system is shown in Figure 4.

Assume that there are N input tests .zi; yi/ available, where zi is the ith test with n distinct features and you portrays the actual labels of xi. A taditional SLFN with K hidden neurons is defined as follows:

$$\sum_{m=1}^{K} \beta_i h\,(w_m . x_i + c_m) = \alpha_i, \quad i = 1, \ldots\ldots\ldots, N \qquad (5)$$

The proposed system employs a single hidden layer feedforward neural network (SLFN) with K hidden neurons. The weight vector wm represents the association of the ith hidden neuron with the input nodes, while the load vector i denotes the connection of the ith hidden neuron

with the output nodes. The parameter cm is the threshold of the ith hidden neuron, and k represents the kth output neuron. The function h:/ serves as the activation function and is used for M hidden neurons. The activation function can achieve zero errors when applied to the training samples. Various techniques have been utilized to detect and classify network intrusions in both wired and wireless environments.

D. EVALUATION

The performance of the proposed system is assessed using the standard dataset NSL KDD, which is randomized and divided into three parts: the full dataset, the half dataset, and the 1/4 dataset. The full dataset consists of 65,535 examples, the half dataset includes 32,767 examples, and the 1/4th dataset contains 18,383 examples. To evaluate the system, accuracy, precision, and recall are used as assessment metrics. These metrics are defined as follows:

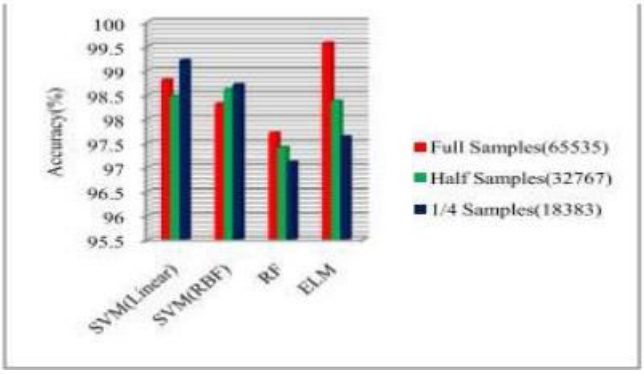


Fig-5: The accuracy of SVM, RF, and ELM (80% training and 20% testing).

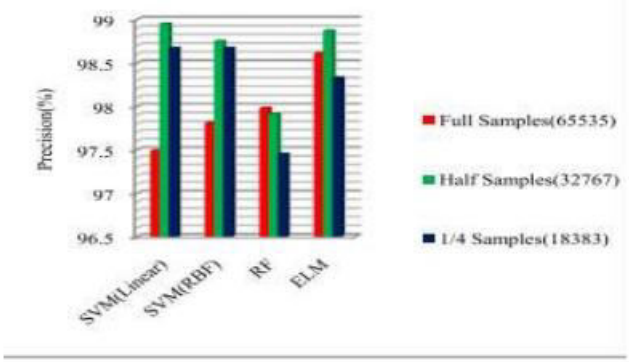


Fig-6: The precision of SVM, RF, and ELM (80% training and 20% testing).

Precision, accuracy, and recall are used as evaluation metrics for the proposed system, which is tested on the standard dataset NSL KDD. The dataset is randomized and divided into three parts: the full dataset, the half dataset, and the 1/4 dataset. These metrics are defined as follows:

- Precision: It is calculated as "the total number of correct predictions, True Positive (TP) C True Negative (TN) divided by the total number of a dataset Positive (P) C Negative (N)".

- Accuracy: It is computed as "the number of correct positive predictions (TP) divided by the total number of positive predictions (TP C FP)". Accuracy is also known as positive predictive value.
- Recall: It is calculated as "the number of correct positive predictions (TP) divided by the total number of positives (P)". Recall is also known as the true positive rate or sensitivity.

CONCLUSION

In conclusion, interruption detection and anticipation are crucial for current and future networks and information systems. Machine learning techniques have been widely used in intrusion detection systems, and in this work, SVM, RF, and ELM are compared. ELM outperforms the other methods in terms of accuracy, precision, and recall on the full dataset tests containing both normal and intrusive activities. SVM showed better results than other methods in half and 1/4 of the data tests. Therefore, ELM is a suitable method for intrusion detection systems that analyse large amounts of data. In the future, ELM will be further investigated for its performance in feature selection and feature transformation techniques, and these techniques will be explored to enhance the accuracy of the intrusion detection system.

REFERENCES

1. *Wang, H., Gu, J., & Wang, S. (2017). An effective intrusion detection framework based on SVM with feature augmentation. Knowledge-Based Systems, 136, 130-139. doi: 10.1016/j.knosys.2017.09.014.*
2. *Kuang, F., Xu, W., & Zhang, S. (2014). A novel hybrid KPCA and SVM with GA model for intrusion detection. Applied Soft Computing, 18, 178-184. doi: 10.1016/j.asoc.2014.01.028.*
3. *Aburomman, A. A., & Reaz, M. B. I. (2016). A novel SVM-kNN-PSO ensemble method for intrusion detection system. Applied Soft Computing, 38, 360-372. doi: 10.1016/j.asoc.2015.10.011.*
4. *M Kiran Kumar, K Balakrishna, M NagaSeshudu ,A Sandeep. Providing Privacy for Numeric Range SQL Queries Using Two-Cloud Architecture. International Journal of Scientific Research and Review. 2018;p.39*
5. *Raman, M. R. G., Somu, N., Kirthivasan, K., Liscano, R., & Sriram, V. S. S. (2017). An efficient intrusion detection system based on hypergraph genetic algorithm for parameter optimization and feature selection in support vector machine. Knowledge-Based Systems, 134, 1-12. doi: 10.1016/j.knosys.2017.07.005.*
6. *Teng, S., Wu, N., Zhu, H., Teng, L., & Zhang, W. (2018). SVM-DT-based adaptive and collaborative intrusion detection. IEEE/CAA Journal of Automatica Sinica, 5(1), 108-118. doi: 10.1109/JAS.2017.7510730.*
7. *M. Kiran Kumar , S. Jessica Saritha. AN EFFICIENT APPROACH TO QUERY REFORMULATION IN WEB SEARCH, International Journal of Research in Engineering and Technology. 2015;p.172.*
8. *Farnaaz, N., & Jabbar, M. A. (2016). Random forest modeling for network intrusion detection system. Procedia Computer Science, 89, 213-217. doi: 10.1016/j.procs.2016.06.047.*

9. *Elbasiony, R. M., Sallam, E. A., Eltobely, T. E., & Fahmy, M. M. (2013). A hybrid network intrusion detection framework based on random forests and weighted k-means. Ain Shams Engineering Journal, 4(4), 753-762. doi: 10.1016/j.asej.2013.01.003.*
10. *Ahmad, I., & Amin, F. e. (2014). Towards feature subset selection in intrusion detection. In Proceedings of IEEE 7th Joint International Information Technology and Artificial Intelligence Conference (pp. 68-73).*
11. *Jha, J., & Ragha, L. (2013). Intrusion detection system using support vector machine. International Journal of Applied Information Systems, ICWAC, 3, 25-30.*
12. *Bamakan, S. M. H., Wang, H., Yingjie, T., & Shi, Y. (2016). An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization. Neurocomputing, 199, 90-102. doi: 10.1016/j.neucom.2016.02.025.*
13. *M. Kiran Kumar, Dr. K. Bhargavi. An Effective Study on Data Science Approach to Cybercrime Underground Economy Data. Journal of Engineering, Computing and Architecture.2020;p.148.*
14. *K Bala Krishna, M Nagaseshudu, M kiran kumar. An Effective Way of Processing Big Data by Using Hierarchically Distributed Data Matrix. International Journal of Research.2019;p.1628*
15. *Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2(3), 27:1-27:27.*