IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 15, 2018, accepted May 18, 2018, date of publication May 30, 2018, date of current version July 6, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2841987

Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection

IFTIKHAR AHMAD[1], MOHAMMAD BASHERI[1], MUHAMMAD JAVED IQBAL[2], AND ANEEL RAHIM[3]

[1]Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[2]Department of Computer Science, University of Engineering and Technology Taxila, Taxila 47080, Pakistan
[3]School of Computing, Dublin Institute of Technology, Dublin, D08 X622 Ireland

Corresponding author: Iftikhar Ahmad (iakhan@kau.edu.sa)

This work was supported in part by the IT Department, KAU, Saudi Arabia, and in part by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah. The authors, therefore, acknowledge with thanks DSR for technical and financial support.

ABSTRACT Intrusion detection is a fundamental part of security tools, such as adaptive security appliances, intrusion detection systems, intrusion prevention systems, and firewalls. Various intrusion detection techniques are used, but their performance is an issue. Intrusion detection performance depends on accuracy, which needs to improve to decrease false alarms and to increase the detection rate. To resolve concerns on performance, multilayer perceptron, support vector machine (SVM), and other techniques have been used in recent work. Such techniques indicate limitations and are not efficient for use in large data sets, such as system and network data. The intrusion detection system is used in analyzing huge traffic data; thus, an efficient classification technique is necessary to overcome the issue. This problem is considered in this paper. Well-known machine learning techniques, namely, SVM, random forest, and extreme learning machine (ELM) are applied. These techniques are well-known because of their capability in classification. The NSL–knowledge discovery and data mining data set is used, which is considered a benchmark in the evaluation of intrusion detection mechanisms. The results indicate that ELM outperforms other approaches.

INDEX TERMS Detection rate, extreme learning machine, false alarms, NSL–KDD, random forest, support vector machine.

I. INTRODUCTION

Intrusion is a severe issue in security and a prime problem of security breach, because a single instance of intrusion can steal or delete data from computer and network systems in a few seconds. Intrusion can also damage system hardware. Furthermore, intrusion can cause huge losses financially and compromise the IT critical infrastructure, thereby leading to information inferiority in cyber war. Therefore, intrusion detection is important and its prevention is necessary.

Different intrusion detection techniques are available, but their accuracy remains an issue; accuracy depends on detection and false alarm rate. The problem on accuracy needs to be addressed to reduce the false alarms rate and to increase the detection rate. This notion was the impetus for this research work. Thus, support vector machine (SVM), random forest (RF), and extreme learning machine (ELM) are applied in this work; these methods have been proven effective in their capability to address the classification problem.

Intrusion detection mechanisms are validated on a standard dataset, KDD. This work used the NSL–knowledge discovery and data mining (KDD) dataset, which is an improved form of the KDD and is considered a benchmark in the evaluation of intrusion detection methods.

The remainder of the paper is organized as detailed below. The related work is presented in Section II. The proposed model of intrusion detection to which different machine learning techniques are applied is described in Section III. The implementation and results are discussed in Section IV. The paper is concluded in Section V, which provides a summary and directions for future work.

II. RELATED WORK

Securing computer and network information is important for organizations and individuals because compromised information can cause considerable damage. To avoid such circumstances, intrusion detection systems are important. Recently, different machine learning approaches have been proposed to improve the performance of intrusion detection systems. Wang *et al.* [1] proposed an intrusion detection framework based on SVM and validated their method on the NSL–KDD dataset. They claimed that their method, which has 99.92% effectiveness rate, was superior to other approaches; however, they did not mention used dataset statistics, number of training, and testing samples. Furthermore, the SVM performance decreases when large data are involved, and it is not an ideal choice for analyzing huge network traffic for intrusion detection.

Kuang *et al.* [2] applied a hybrid model of SVM and KPCA with GA to intrusion detection, and their system showed 96% detection rate. They used the KDD CUP99 dataset for the verification of their system, but this dataset is characterized by limitations. One example is redundancy, which causes the classifier to be biased to more frequently occurring records. They applied KPCA for feature reduction, and it is limited by the possibility of missing important features because of selecting top percentages of the principal component from the principal space. In addition, the SVM is not appropriate for heavy data such as monitoring the high bandwidth of the network.

Intrusion detection systems provide assistance in detecting, preventing, and resisting unauthorized access. Thus, Aburomman and Reaz [3] proposed an ensemble classifier method, which is a combination of PSO and SVM; this classifier outperformed other approaches with 92.90% accuracy. They used the knowledge discovery and data mining 1999 (KDD99) dataset, which has the previously mentioned drawbacks. Furthermore, the SVM is not a good choice for huge data analyses, because its performance degrades as data size increases.

Raman *et al.* [4] proposed an intrusion detection mechanism based on hypergraph genetic algorithm (HG-GA) for parameter setting and feature selection in SVM. They claimed that their method outperformed the existing approaches with a 97.14 % detection rate on an NSL–KDD dataset; it has been used for experimentation and validation of intrusion detection systems.

The security of network systems is one of the most critical issues in our daily lives, and intrusion detection systems are significant as prime defense techniques. Thus, Teng *et al.* [5] conducted important work. They developed their model based on decision trees (DTs) and SVMs, and they tested their model on a KDD CUP 1999 dataset. The results showed an accuracy reaching 89.02%. However, SVMs are not preferred for heavy datasets because of the high computation cost and poor performance.

Farnaaz and Jabbar [6] developed a model for an intrusion detection system based on RF. They tested the effectiveness of their model on an NSL–KDD dataset, and their results demonstrated a 99.67% detection rate compared with J48. The main limitation of the RF algorithm is that many trees may make the algorithm slow for real-time prediction. Elbasiony *et al.* [7] proposed a model of intrusion detection based on RF and weighted k-means; they validated their model over the KDD99 dataset. The system demonstrated results with 98.3% accuracy. The RF is not suitable for predicting real traffic because of its slowness, which is due to the formation of a large number of trees. Additionally, the KDD99 dataset indicates few limitations as aforementioned.

III. PROPOSED MODEL

The key phases of the proposed model include the dataset, pre-processing, classification, and result evaluation. Each phase of the proposed system is important and adds valuable influence on its performance. The core focus of this work is to investigate the performance of different classifiers, namely, SWM, RF, and ELM in intrusion detection. Figure 1 demonstrates the model of intrusion detection system proposed in this work.

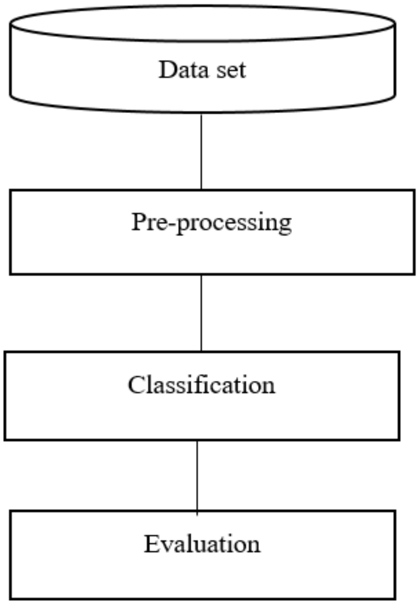


FIGURE 1. Proposed model of intrusion detection system.

A. DATASET

Dataset selection for experimentation is a significant task, because the performance of the system is based on the correctness of a dataset. The more accurate the data, the greater the effectiveness of the system. The dataset can be collected by numerous means, such as 1) sanitized dataset, 2) simulated dataset, 3) testbed dataset, and 4) standard dataset [8]. However, complications occur in the application of the first three methodologies. A real traffic method is expensive, whereas the sanitized method is unsafe. The development of a simulation system is also complex and challenging. Additionally, different types of traffic are required to model various network attacks, which is complex and costly. To overcome these difficulties, the NSL–KDD dataset is used to validate the proposed system for intrusion detection.

B. PRE-PROCESSING

The classifier is unable to process the raw dataset because of some of its symbolic features. Thus, pre-processing is essential, in which non-numeric or symbolic features are eliminated or replaced, because they do not indicate vital participation in intrusion detection. However, this process generates overhead including more training time; the classifier's architecture becomes complex and wastes memory and computing resources. Therefore, the non-numeric features are excluded from the raw dataset for improved performance of intrusion detection systems.

C. CLASSIFICATION

Placing an activity into normal and intrusive categories is the core function of an intrusion detection system, which is known as an intrusive analysis engine. Thus, different classifiers have been applied as intrusive analysis engines in intrusion detection in the literature, such as multilayer perceptron, SVM, naive Bayes, self-organizing map, and DT.

However, in this study, the three different classifiers of SVM, RF, and ELM are applied based on their proven ability in classification problems. Details of each classification approach are provided.

1) SUPPORT VECTOR MACHINE

SVMs were initially proposed by Vapnik (1995) for solving problems of classification and regression analysis [9]. SVM is a supervised learning technique that is trained to classify different categories of data from various disciplines. These have been used for two-class classification problems and are applicable on both linear and non-linear data classification tasks. SVM creates a hyperplane or multiple hyperplanes in a high-dimensional space, and the best hyperplane in them is the one that optimally divides data into different classes with the largest separation between the classes. A non-linear classifier uses various kernel functions to estimate the margins. The main objective of these kernel functions (i.e., linear, polynomial, radial basis, and sigmoid) is to maximize margins between hyper-planes. Recently, many highly promising applications have been developed by researchers because of the increasing interest in SVMs [10]. SVM has been widely used in image processing and pattern recognition applications.

Figure 2 illustrates the architecture of the SVM classification model in the proposed intrusion detection system. We have used the radial basis function (RBF) kernel for the implementation of the SVM model in the proposed system. The kernel function uses squared Euclidean distance between two numeric vectors and maps input data to a high dimensional space to optimally separate the given data into their respective attack classes. Therefore, kernel RBF is particularly effective in separating sets of data that share complex boundaries. In our study, all the simulations have been conducted using the freely available LibSVM package [11]. Given that the chosen problem is a multiclass classification problem, it uses the notion of one vs all for attack classification. In this notion, the multiclass problem is divided into a two-class problem. The radial basis function (RBF) kernel is used in this study, which is represented as follows:

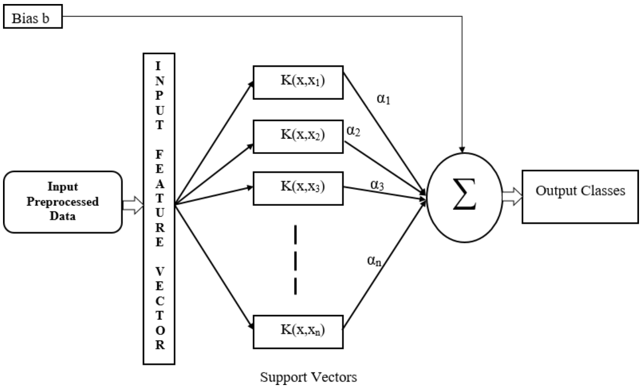


FIGURE 2. Architecture of SVM for intrusion detection.

$$K(x, y) = e^{-\gamma\|x-y\|^2}, \quad \gamma > 0 \tag{1}$$

For given training samples (x_i, y_i), i = 1, 2, ... n, where i is the maximum number of samples in the training data, $x_i \in R^n$ and $y_i \in \{1, -1\}$, where 1 shows samples from a positive class and -1 represents sequences from the negative class. When using SVM, the solution of the following problem is provided.

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \xi_i \tag{2}$$

$$\text{subject to } y_i\left(w^T w\phi\left(x_i\right) + b\right) \geq 1 - \xi_i. \tag{3}$$

Here, ϕ transforms the training vector x_i to the higher dimensional space. Following this, the SVM shows a hyper-plane having a maximum margin to separate different classes of data.

The observed results via the SVM model are not significantly convincing compared with those from the other classifiers. The advantage of SVM is that minimal parameter adjustment is required. The disadvantages of it include the requirements of a Gaussian function for each instance of the training set, thereby increasing training time and performance degradation on very large datasets with thousands of instances, as in the case classification. In case maximum margin classifier fails to find any separating hyperplane, soft margin is used to overcome this problem. Soft margin uses positive slack variables ξ_i, i = 1, 2, ..., N in the constraints, as follows:

$$\begin{aligned} (w.\, x_i - b) &\geq +1 - \xi_i \quad \text{for } y_i = +1 \\ (w.\, x_i - b) &\geq -1 + \xi_i \quad \text{for } y_i = -1 \\ \xi &\geq 0. \end{aligned}$$

When an error occurs, ξ_i must exceed unity. Then, $\sum_i \xi_i$ is an upper bound on the training error. The Lagrange in this situation is as follows:

$$L_p = \frac{1}{2}\left\|w^2\right\| + C\sum\nolimits_{i=1}^{n}\xi_i - \sum\nolimits_i \alpha_i\left\{y_i\left(x_i.w - b\right) - 1 + \xi_i\right\} - \sum\nolimits_i \mu_i\xi_i, \quad (4)$$

where, μ_i represents Lagrange multipliers used to obtain the positive value of ξ_i.

2) RANDOM FORES

RFs are ensemble classifiers, which are used for classification and regression analysis on the intrusion detection data. RF works by creating various decision trees in the training phase and output class labels those have the majority vote [12]. RF attains high classification accuracy and can handle outliers and noise in the data. RF is used in this work because it is less susceptible to over-fitting and it has previously shown good classification results.

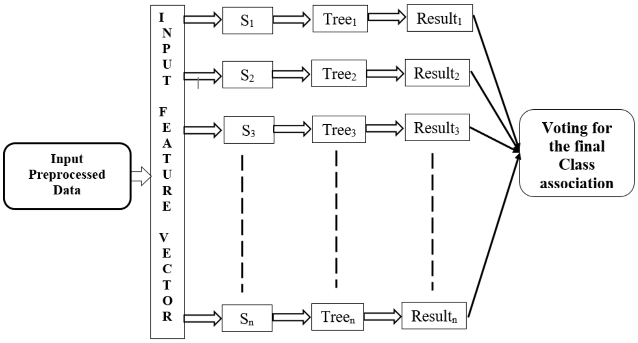


FIGURE 3. Architecture of the RF for intrusion detection system.

Figure 3 shows the implementation of the random forest classification model in the data classification in the proposed system. A pre-processed sample of n samples is fed to the random forest classifier. RF creates n different trees by using a number of feature subsets. Each tree produces a classification result, and the result of the classification model depends on the majority voting. The sample is assigned to the class that obtains highest voting scores. The previously attained classification results indicate that RF is reasonably suitable in the classification of such data because in some cases, it has obtained better results than have other classifiers. Other advantages of the RF include its higher accuracy than Adaboost and fewer chances of overfitting.

3) EXTREME LEARNING MACHINE

ELM is another name for single or multiple hidden layer feedforward neural networks [13]. ELM can be used to solve various classification, clustering, regression, and feature engineering problems. This learning algorithm involves input layer, one or multiple hidden layers and the output layer. In the traditional neural networks, the tasks of adjustment of the input and hidden layer weights are very computationally expensive and time-consuming because it requires multiple rounds to converge. To overcome this problem, Huang *et al.* [13]. proposed an SLFN by arbitrarily selecting input weights and hidden layer biases to minimize the training time. The comprehensive detail of ELM is available in Huang *et al.* [14]. and Qayyum *et al.* [15]. The authors claim that these models learn faster and attain higher generalization capability as compared with other feedforward network models. ELM performance is comparable with SVM or other state-of-the-art machine learning classifiers. ELM has the greatest ability to perform better in highly complex datasets. The architecture of the proposed system is shown in Figure 4.

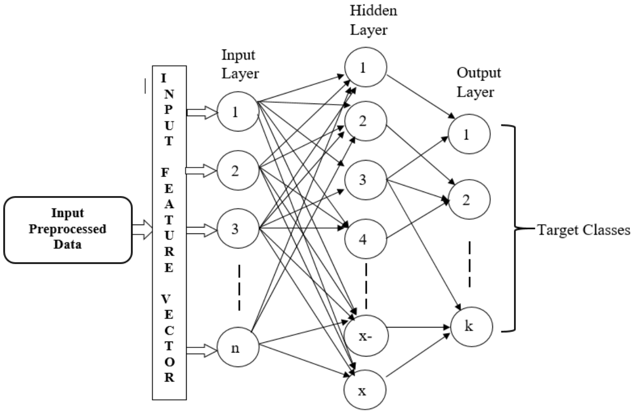


FIGURE 4. Architecture of the extreme learning machine for intrusion detection.

N input samples (z_i, y_i) are present, where $z_i = [x_{i1}, x_{i2}, \ldots\ldots., x_{in}]^T$ is the *ith* sample with n different features and $y_i = [y_{i1}, y_{i2}, \ldots\ldots., y_{im}]^T$ describes the actual labels of x_i with traditional SLFN with K hidden neurons which is defined as follows:

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

where $w_m = [w_{m1},\ w_{m2},\ \ldots\ \ \ldots., w_{mn}]^T$ is the chosen weight vector and indicates an *ith* hidden neuron connection with the input nodes. $\beta_i = [\beta_{i1}, \beta_{i2}, \ldots\ldots., \beta_{im}]^T$ shows the weight vector with connection of *ith* hidden neuron and the output nodes and c_m is the threshold of the *ith* hidden neuron $\alpha_k = [\alpha_{k1}, \alpha_{k2}, \ldots\ldots., \alpha_{km}]^T$ is the *kth* output neuron. $h(.)$ represents the activation function and SLFN used for M hidden neurons and activation function can approach these N training samples with zero error. Various other techniques have been applied to detect and classify intrusion of wired and wireless environment [16]–[20].

D. EVALUATION

The designed system is evaluated based on the standard dataset NSL–KDD, which is randomized and divided into three parts, namely, the full dataset, the half dataset, and the 1/4 dataset. The full dataset consists of 65,535 samples, the half dataset includes 32,767 samples, and the 1/4th dataset consists of 18,383 samples. Accuracy, precision, and recall

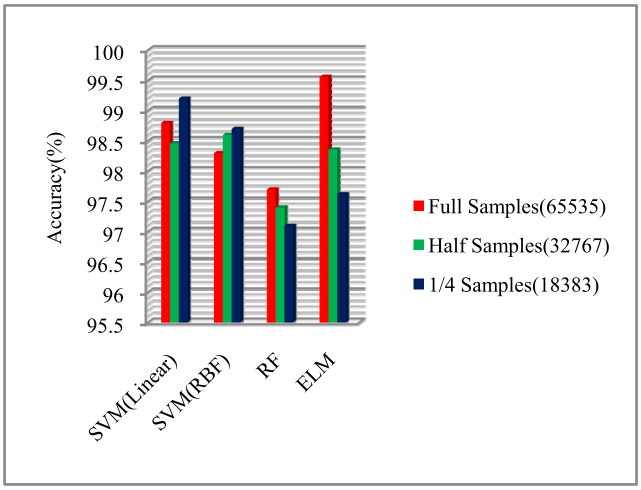


FIGURE 5. Accuracy of SVM, RF, and ELM (80% training and 20% testing).

are used as evaluation metrics. These metrics are described here [21].

Accuracy: Accuracy is computed as "the total number of correct prediction, True Positive (TP) + True Negative (TN) divided by the total number of a dataset Positive (P) + Negative (N)".

$$Accuracy = \frac{TP + TN}{P + N}$$

Precision: Precision is computed as "the number of correct positive predictions (TP) divided by the total number of positive predictions (TP + FP)". Precision is also known as a positive predictive value.

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

Recall: Recall is computed 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.

$$Recall = \frac{TP}{P}$$

IV. RESULTS

The accuracy of SVM (Linear), SVM (RBF), RF, and ELM on 20% testing and 80% training data samples is shown in Figure 5. ELM performs better compared with SVM (Linear), SVM (RBF) and RF on full data samples, whereas SVM (RBF) indicates improved accuracy over RF and ELM on half data samples. SVM (Linear) outperforms other techniques on 1/4 data samples, as depicted in Figure 5.

The precision of SVM (Linear), SVM (RBF), RF, and ELM on 20% testing and 80% training data samples is shown in Figure 6. The precision of ELM is better than that of SVM Linear and RBF on the full data samples, and it also outperforms that of RF. On half data samples, the precision of SVM (Linear) is higher than that of SVM (RBF), ELM, and RF. On 1/4th data samples, the precision of SVM (Linear) is equal to that of SVM (RBF). Furthermore, the SVM performs better than ELM and RF in the 1/4 dataset.

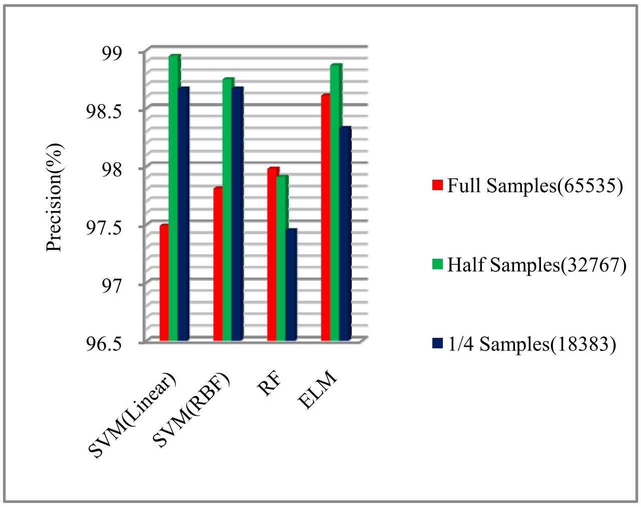


FIGURE 6. Precision of SVM, RF, and ELM (80% training and 20% testing).

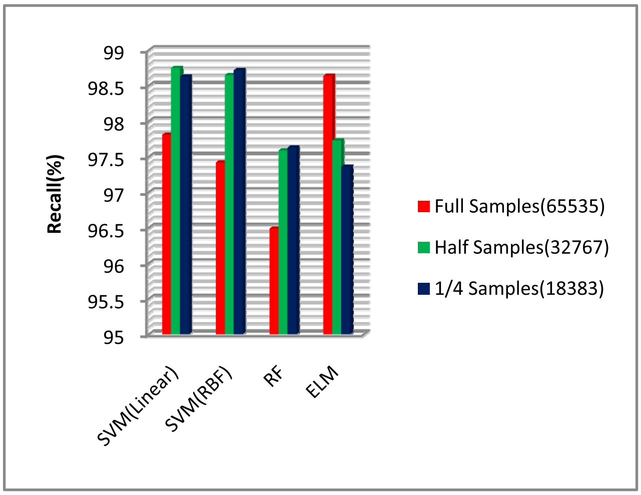


FIGURE 7. Recall of SVM, RF, and ELM (80% training and 20% testing).

The recall of SVM (Linear), SVM (RBF), RF, and ELM on 20% testing and 80% training data samples is shown in Figure 7. On full data samples, the recall of ELM performs better than those of SVM (Linear), SVM (RBF), and RF. The recall of SVM (Linear) is greater than those of SVM (RBF), ELM, and RF. The ranking of recall on 1/4 of data samples is as follows: first for SVM (RBF), second for SVM (Linear), third for RF, and fourth for ELM. The abovementioned discussion indicates that SVM performs better on a small dataset, whereas EML outperforms others approaches on large datasets.

The accuracy of SVM (Linear), SVM (RBF), RF, and ELM on 10% testing and 90% training data samples is shown in Figure 8. On the full data samples, the accuracy of ELM is better than that of SVM (linear), SVM (RBF), and RF. The SVM (RBF) outperforms SVM (Linear), ELM, and RF

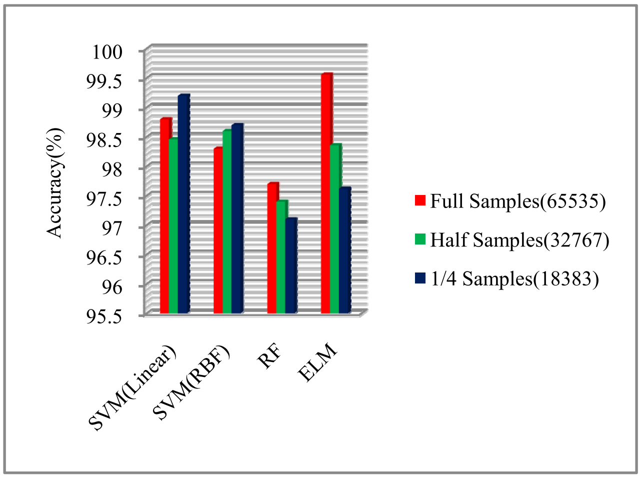


FIGURE 8. Accuracy of SVM, RF, and ELM (90% training and 10% testing).

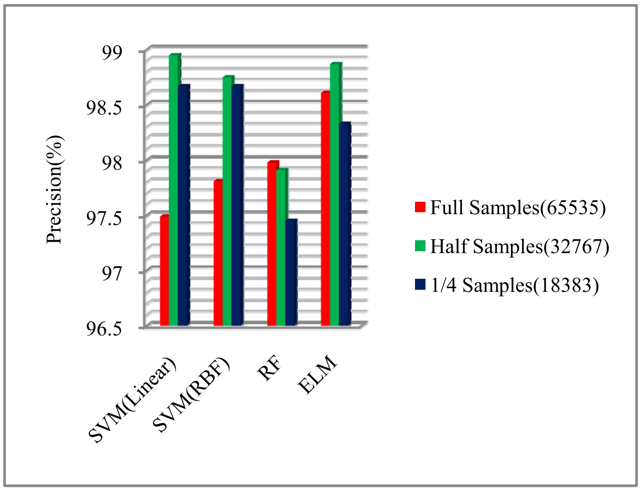


FIGURE 9. Recall of SVM, RF, and ELM (90% training and 10% testing).

on the half data samples. The SVM (linear) indicates better performance on 1/4th data samples as compared with SVM (RBF), RF, and ELM.

The precision of SVM (Linear), SVM (RBF), RF, and ELM on 10 % testing and 90% training data samples is shown in Figure 9. The results indicate that the ELM indicates better precision than RF, SVM (RBF), and SVM (Linear) on full data samples, whereas SVM (Linear) indicates better precision on the half data samples. Furthermore, SVM (Linear) performs better than ELM and RF on 1/4th dataset.

The recall of SVM (Linear), SVM (RBF), RF, and RLM on 10% testing and 90% training data samples is shown in Figure 10. On full data samples, the recall of ELM outperforms those of SVM (linear), SVM (RBF), and RF, whereas the recall of SVM (linear) is better than those of SVM (RBF), ELM, and RF on half data samples. On the 1/4th data samples, SVM (RBF) is almost equal to SVM (Linear), whereas it indicates better results over RF and ELM, as shown in Figure 10.

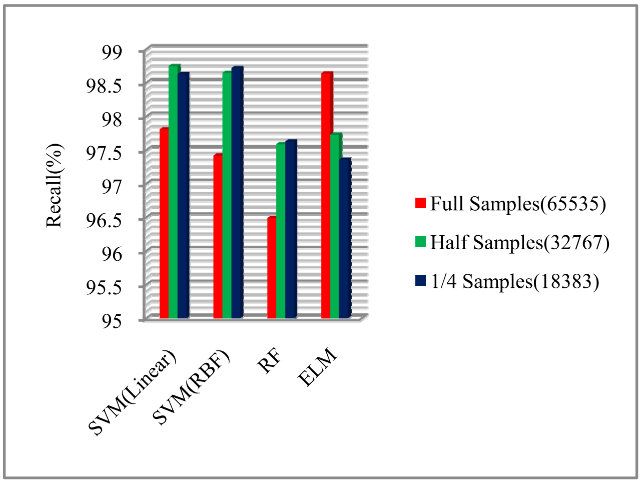


FIGURE 10. Recall of SVM, RF, and ELM (90% training and 10% testing).

V. CONCLUSION

Intrusion detection and prevention are essential to current and future networks and information systems, because our daily activities are heavily dependent on them. Furthermore, future challenges will become more daunting because of the Internet of Things. In this respect, intrusion detection systems have been important in the last few decades. Several techniques have been used in intrusion detection systems, but machine learning techniques are common in recent literature. Additionally, different machine learning techniques have been used, but some techniques are more suitable for analyzing huge data for intrusion detection of network and information systems. To address this problem, different machine learning techniques, namely, SVM, RF, and ELM are investigated and compared in this work. ELM outperforms other approaches in accuracy, precision, and recall on the full data samples that comprise 65,535 records of activities containing normal and intrusive activities. Furthermore, the SVM indicated better results than other datasets in half of the data samples and in 1/4 of the data samples. Therefore, ELM is a suitable technique for intrusion detection systems that are designed to analyze a huge amount of data. In future, ELM will be explored further to investigate its performance in feature selection and feature transformation techniques.

REFERENCES

[1] H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on SVM with feature augmentation," *Knowl.-Based Syst.*, vol. 136, pp. 130–139, Nov. 2017, doi: 10.1016/j.knosys.2017.09.014.

[2] F. Kuang, W. Xu, and S. Zhang, "A novel hybrid KPCA and SVM with GA model for intrusion detection," *Appl. Soft Comput.*, vol. 18, pp. 178–184, May 2014, doi: 10.1016/j.asoc.2014.01.028.

[3] A. A. Aburomman and M. B. I. Reaz, "A novel SVM-kNN-PSO ensemble method for intrusion detection system," *Appl. Soft Comput.*, vol. 38, pp. 360–372, Jan. 2016, doi: 10.1016/j.asoc.2015.10.011.

[4] M. R. G. Raman, N. Somu, K. Kirthivasan, R. Liscano, and V. S. S. Sriram, "An efficient intrusion detection system based on hypergraph—Genetic algorithm for parameter optimization and feature selection in support vector machine," *Knowl.-Based Syst.*, vol. 134, pp. 1–12, Oct. 2017, doi: 10.1016/j.knosys.2017.07.005.

[5] S. Teng, N. Wu, H. Zhu, L. Teng, and W. Zhang, "SVM-DT-based adaptive and collaborative intrusion detection," *IEEE/CAA J. Automatica Sinica*, vol. 5, no. 1, pp. 108–118, Jan. 2018, doi: 10.1109/JAS.2017.7510730.

[6] N. Farnaaz and M. A. Jabbar, "Random forest modeling for network intrusion detection system," *Proc. Comput. Sci.*, vol. 89, pp. 213–217, Jan. 2016, doi: 10.1016/j.procs.2016.06.047.

[7] R. M. Elbasiony, E. A. Sallam, T. E. Eltobely, and M. M. Fahmy, "A hybrid network intrusion detection framework based on random forests and weighted k-means," *Ain Shams Eng. J.*, vol. 4, no. 4, pp. 753–762, 2013, doi: 10.1016/j.asej.2013.01.003.

[8] I. Ahmad and F. e Amin, "Towards feature subset selection in intrusion detection," in *Proc. IEEE 7th Joint Int. Inf. Technol. Artif. Intell. Conf.*, Chongqing, China, Dec. 2014, pp. 68–73.

[9] J. Jha and L. Ragha, "Intrusion detection system using support vector machine," *Int. J. Appl. Inf. Syst.*, vol. ICWAC, no. 3, pp. 25–30, Jun. 2013.

[10] S. M. H. Bamakan, H. Wang, T. Yingjie, and Y. Shi, "An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization," *Neurocomputing*, vol. 199, pp. 90–102, Jul. 2016.

[11] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

[12] Y. Liu, Y. Wang, and J. Zhang, "New machine learning algorithm: Random forest," in *Information Computing and Applications*, B. Liu, M. Ma, and J. Chang, Eds. Berlin, Germany: Springer, 2012, pp. 246–252.

[13] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: A new learning scheme of feedforward neural networks," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, vol. 2, Jul. 2004, pp. 985–990, doi: 10.1109/IJCNN.2004.1380068.

[14] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012, doi: 10.1109/TSMCB.2011.2168604.

[15] A. Qayyum *et al.*, "Image classification based on sparse-coded features using sparse coding technique for aerial imagery: A hybrid dictionary approach," in *Neural Computing and Applications*. London, U.K.: Springer, 2017 pp. 1–21, doi: 10.1007/s00521-017-3300-5.

[16] A. Derhab, A. Bouras, M. R. Senouci, and M. Imran, "Fortifying intrusion detection systems in dynamic ad hoc and wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 12, p. 608162, 2014.

[17] I. Yaqoob *et al.*, "The rise of ransomware and emerging security challenges in the Internet of Things," *Comput. Netw.*, vol. 129, pp. 444–458, Dec. 2017.

[18] A. R. Sfar, E. Natalizio, Y. Challal, and Z. Chtourou, "A roadmap for security challenges in the Internet of Things," *Digit. Commun. Netw.*, vol. 4, no. 2, pp. 118–137, 2017, doi: 10.1016/j.dcan.2017.04.003.

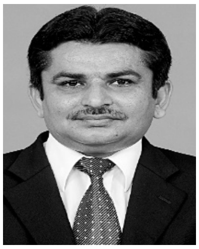
[19] A. A. Aziz, S. EL-Ola Hanafi, and A. E. Hassanien, "Comparison of classification techniques applied for network intrusion detection and classification," *J. Appl. Log.*, vol. 24, pp. 109–118, Nov. 2017, doi: 10.1016/j.jal.2016.11.018.

[20] I. Ahmad, "Feature selection using particle swarm optimization in intrusion detection," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 10, p. 806954, 2015, doi: 10.1155/2015/806954.

[21] *Basic Evaluation Measures From the Confusion Matrix*. Accessed: May 20, 2018. [Online]. Available: http://WordPress.com and https://classeval.wordpress.com

IFTIKHAR AHMAD received the B.Sc. degree from Islamia University, Bahawalpur, Pakistan, in 1999, the M.Sc. degree in computer science from the University of Agriculture, Faisalabad, Pakistan, in 2001, the M.S./M.Phil. degree in computer science from the COMSATS Institute of Information Technology, Abbottabad, Pakistan, in 2007, and the Ph.D. degree in information technology from Universiti Teknologi PETRONAS, Malaysia, 2011. He served as a faculty member and a research supervisor at various universities since 2001. He is currently a Faculty Member with the Information Technology Department, Faculty of Computing and Information Technology, King Abdulaziz University. He has been involved in several funded projects as PI and Co-PI. He has published several papers in reputed journals and conferences. He is also a member of several scientific and professional bodies.

MOHAMMAD BASHERI received the Ph.D. degree from Durham University, U.K., in 2013. He has served as a consultant for some institutes and organizations. He is currently the Chairman of the Information Technology Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. He has received several trainings and certifications from well reputed institutes and organizations. He has published several papers in conferences and journals of international repute.



MUHAMMAD JAVED IQBAL received the M.Sc. degree in computer science from the University of Agriculture, Faisalabad, Pakistan, in 2001, the M.S./M.Phil. degree in computer science from International Islamic University Islamabad, Pakistan, in 2008, and the Ph.D. degree in information technology from Universiti Teknologi PETRONAS, Malaysia, in 2015. He is currently a HEC approved Ph.D. Supervisor and an Assistant Professor with the Computer Science Department, University of Engineering and Technology Taxila, Pakistan. His work has been published in several international publications including journals, book chapters, and conferences. He is also a reviewer of renowned national and international journals and conferences.

ANEEL RAHIM received the Ph.D. degree in computer science from IIU, Pakistan, in 2011. He is currently an Assistant Lecturer with the Dublin Institute of Technology, Dublin, Ireland, since 2015. He was involved in two EU FP7 projects, i.e., Aniketos and FINESCE as a Post-Doctoral Security Researcher. In 2014, he was involved in EU FP7 Project Campus21 as a Research Support Officer. He has published over 30 papers in various international conferences and journals. He has been serving as a Guest Editor of *Multimedia Tools and Applications* (Springer), *Telecommunication System* (Springer), and *Information International Journal*.