# Comparison study of machine learning classifiers to detect anomalies

**Nisha P. Shetty[1], Jayashree Shetty[2], Rohil Narula[3], Kushagra Tandona[4]**
[1,2,4]Department of Information and Communication Technology, Manipal Institute of Technology,
Manipal Academy of Higher Education, India
[3]Department of Computer Science and Engineering, Manipal Institute of Technology,
Manipal Academy of Higher Education, India

**Article Info**

***Article history:***

Received Nov 29, 2019
Revised Apr 3, 2020
Accepted Apr 13, 2020

***Keywords:***

Bagging
IDS
Network attacks
Neural network
Random forest
SVM

**ABSTRACT**

In this era of Internet ensuring the confidentiality, authentication and integrity of any resource exchanged over the net is the imperative. Presence of intrusion prevention techniques like strong password, firewalls etc. are not sufficient to monitor such voluminous network traffic as they can be breached easily. Existing signature based detection techniques like antivirus only offers protection against known attacks whose signatures are stored in the database.Thus, the need for real-time detection of aberrations is observed. Existing signature based detection techniques like antivirus only offers protection against known attacks whose signatures are stored in the database. Machine learning classifiers are implemented here to learn how the values of various fields like source bytes, destination bytes etc. in a network packet decides if the packet is compromised or not . Finally the accuracy of their detection is compared to choose the best suited classifier for this purpose. The outcome thus produced may be useful to offer real time detection while exchanging sensitive information such as credit card details.

***Corresponding Author:***

Jayashree Shetty,
Department of Computer Science and Engineering,
Manipal Institute of Technology, Manipal Academy of Higher Education,
Manipal-576104 India.
Email: jayashree.sshetty@manipal.edu

## 1. INTRODUCTION

Increasing rate of cybercrimes is a grave concern nowadays. Owing to the increased usage of Internet in all zones of life privacy and security has become the need of the hour. Any manipulation done to resource by an unauthorized entity with the intension of causing harm is termed as intrusion. an intrusion detection system (IDS) is a defense system which screens the activities in a computer system or a network automatically to detect breaches and subsequently notifies the user about any violations [1].

There are mainly four catagories of attacks [2]. In DoS Attack, attackers prevent other users to use a legitimate service for a period of time by preventing access to others. Banks websites, VTU sites etc. are prone to this kind of attacks. In remote to user (R2L), the threat caused by a secluded person to gain control of a target resource. Social Engineering is one such attack. In user to root (U2R), person with local privileges abuses the system's vulnerabilities to get super user rights. Buffer overflow errors and errors caused by irregularities in environmental assumptions are some common examples.In Probing, Attacker examines the system to find all its liabilities. By using these vulnerabilities the system is abused.

Two commonly used IDS based on the location are [3] network based intrusion detection system (NIDS) (Traffic flowing in the network is examined) and host based intrusion detection system (HIDS) (Traffic originated from or is destined to a particular host is scrutinized). Based on detection techniques IDS can be categorized as [4] Misuse detection and Anomaly detection. In misuse detection signatures of all known attacks are documented. Signatures of every new packet encountered are compared with the database to check whether it is an attack or not. Although this technique provides a high detection rate it is very time consuming and only is effective for known attacks only. In Anomaly detection, any variation from the normal expected behavior is flagged as attacks without any prior mastery on attacks. A higher false alarm rate is obtained by this method.

Problem Definition. Increase in Internet crimes nowadays exemplifies the need for a competent intrusion detection system. Every sector in the society is computerized, thus a large volume of important information such as personal profiles and credit card information are entered, edited and transferred across the network daily. This shift from centralized computing to networked environment has invoked a need to improve the security of the networks. Faulty packet filtering technology of firewalls, generality problem of antivirus, huge cost and performance bottleneck of application gateway which slows down the network etc does not allow them to evade all attackers and are not completely efficient. Machine learning classifiers are implemented here to learn how the values of various fields like source bytes, destination bytes etc. in a network packet decides if the packet is compromised or not .This research points out the need for a proficient intrusion detection system (IDS) which exposes malicious packets effectively even if a broad range of intrusions are encountered and cannot be tampered.

Significance of Proposed Research. The first reason for choosing the research is that Internet is a part of everyday routine nowadays for most of the people encompassing all aspects from online shopping to social media. Hence ensuring that only sanctioned people should have access to private information while preserving its integrity is quite necessary. Secondly, Signature based methods despite of having low false positive rates is ineffective in providing defense against unknown attacks. Statistical anomaly based detection explores on discrepancy of traffic characteristics from normal in terms of volume. It fails when attacker is crafty enough to keep the incongruity below certain levels. Finally, machine learning algorithms are chosen as they have proven to be an effective solution in identifying abnormalities immediately without being susceptible to any sort of manipulations from attackers.

## 2. LITERATURE REVIEW

U. Cavusoglu [5] employed various machine learning algorithms to evaluate which classifier gave better detection for each attack type. Data preprocessing and new feature reduction methods CfsSubsetEval and WrapperSubsetEval were used. The method can be further extended to find one optimum classifier which gives the optimum detection for all categories of attacks.

Kang et al [6] have demonstrated intrusion detection at the cluster head by employing SNORT and MYSQL data bases. Cluster head receives aggregated information from entire network making detection quicker. The presented research is most suitable for organization having large amount of data. However the technique is implemented only on static network and SNORT although offers good detection for known attacks fails for anomaly detection.

Baykara and Das [7] incorporated a honeypot based approach for real time intrusion detection. The proposed system reduces false positive level and provides protection against attacks such as zero day attack. However this approach is costly in terms of configuration, installation and management of honeypots when compared to machine learning classifiers. If regularly the attack signatures collected from log file of honeypots are not updated in the database then the detection rate suffers.

Zhao et al. [8] used Principal Component Analysis to reduce the dimensions for large dataset to make it suitable IOT devices. Accuracy of Softmax and KNN Classifiers is compared, where softmax regression shows better time performance. Unsupervised learning algorithms can be used so that many broader range of attacks can be discovered. Also since the algorithm is to be deployed on IOT memory saving techniques should be applied.

Singh et al. [9] proposed a four tier architecture having data preprocessing in first tier, feature extraction in second tier, classification in third tier and user interface in fourth tier. Generalized discriminant analysis was used for extracting features from KDD Cup 99 data set. C4.5 offered better detection for normal and probe classes, iSVM detected normal and DoS attacks and hybrid C4.5-iSVM perceived U2R and R2L attacks. Although the individual classifiers offered good accuracy there is a room for improvement in detection of U2R and R2L attacks.

Hoque et al. [10] used genetic algorithm to detect various types of attacks. Fitness of chromosome was realized using standard deviation method which can be made better by using heuristic approaches. Lin et al. [11] developed an approach which combines log file analysis technology and BP neural network technology. Even though this technique detected both misuse and anomaly data, log files used are monitored by daemons making it less trustworthy. Leu and Lin [12] employed Chi-Square method to detect variation in packet statistics which happens usually in case of attacks. On the contrary to clearly establish normal distribution huge amounts of data must be forked through which is time consuming.

Seo [13] implemented Multiple Support Vector Machines in which every hyperplane is trained to detect specific attack, thereby decreasing the false positive rate. But, since MSVM has bigger margin than classical SVM, sometimes even the normal packets are classified as attack packets. Mukkamala et al. [14] compared the accuracy of SVM and neural network on DARPA dataset. SVM was observed to be performing better than NN for the selected 13 features. However SVM was limited only in making binary classifications and the method could be extended to detect more variants of attacks.

## 3. PROPOSED METHODOLOGY

Figure 1 highlights the methodology followed in the paper where each field is described below.

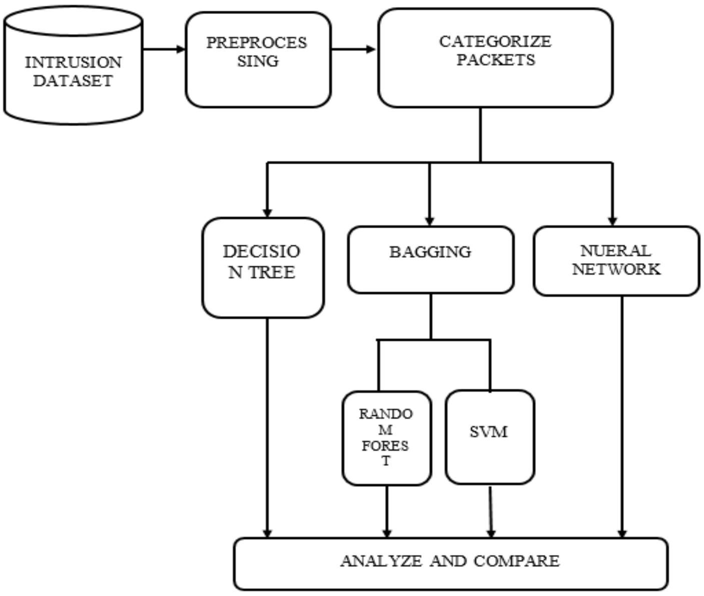


Figure 1. Overall methodology

a. Data set

The KDD99 dataset [15] embodies 41 attributes and the ‘class’ attributes [16] which specifies whether a given case is a normal or an attack as shown in Figure 2.

b. Pre-processing

Noisy, redundant, incomplete and data having different data types is observed. Without standardization the process of classification will be hampered. Various R preprocessing packages are applied to eliminate missing records having incomplete data and to get data in uniform form.

c. Principal component analysis

Most of the features in the NSL-KDD dataset largely do not account for most of the variance in the results. Therefore, a method called PCA [17, 18] is used to get a more concise dataset with less features that account for most of the variance in the data. The PCA method, developed by Karl Person in 1901, uses an orthogonal transformation and converts the possibly correlated data to linearly uncorrelated data sorted in terms of varying degrees of contribution of variance to the final result, such that, the first component explains more variance than the next and so on. The variances explained are calculated by squaring the Eigen values. In this way, the first k components can be selected in such a way that these k components explain most of the variance in the data. In this way, only k features are obtained as a result, without much change in the variance explained. This method of reducing the dimensions of the data helps data visualization and also mediates some of the high variance problems occurring due to excess features having little or no contribution to the results.

d. Categorize the packets

1) Neural network: Pre-processed data is divided into 3 sets - training, validation and test sets in 60:20:20 ratio. Model is trained using the above method [19] for different values of hidden layer data from

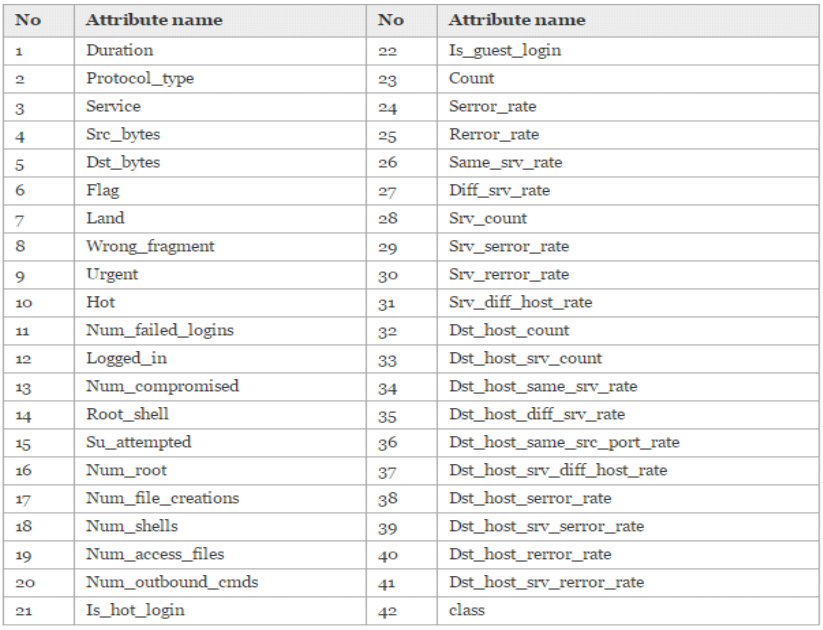
the training set and accuracy was tested subsequently using the validation set. Classification error E=M-Y is calculated using validation set where M is the expected output vector taken from the validation set and Y is the computed output resulting from the classification (Y=W*X) having weight W and input X. When the error observed is low the training phase ends. The entire process is repeated k times (k fold cross validation) [20] for different randomly selected data samples to find the most optimum value for hidden layer ensuring that the model wont over fit the network. Models observed with the highest validation accuracy are taken and tested with the test set. Again k- fold cross validation is applied to find the optimum value.

2) Bagging: When the model is bagged [21-23], several resamples of the data are taken in iteration and the model is trained on these samples. Then the predictions are averaged over the samples. This method is particularly useful when the model has a low variance as it helps increase the variance of the model and having little effect on the bias. This is done with the random forests as well as with the linear SVM model to compare results. Cross validation is included in order to find the most optimum value [24-26].

3) Additionally, simple decision trees with different complexity parameters (cp) (cross validation) are also used to compare results with the above models.

e. Accuracy calculation and comparison:

The most optimum classifier is selected in this step.



| No | Attribute name | No | Attribute name |
|---|---|---|---|
| 1 | Duration | 22 | Is_guest_login |
| 2 | Protocol_type | 23 | Count |
| 3 | Service | 24 | Serror_rate |
| 4 | Src_bytes | 25 | Rerror_rate |
| 5 | Dst_bytes | 26 | Same_srv_rate |
| 6 | Flag | 27 | Diff_srv_rate |
| 7 | Land | 28 | Srv_count |
| 8 | Wrong_fragment | 29 | Srv_serror_rate |
| 9 | Urgent | 30 | Srv_rerror_rate |
| 10 | Hot | 31 | Srv_diff_host_rate |
| 11 | Num_failed_logins | 32 | Dst_host_count |
| 12 | Logged_in | 33 | Dst_host_srv_count |
| 13 | Num_compromised | 34 | Dst_host_same_srv_rate |
| 14 | Root_shell | 35 | Dst_host_diff_srv_rate |
| 15 | Su_attempted | 36 | Dst_host_same_src_port_rate |
| 16 | Num_root | 37 | Dst_host_srv_diff_host_rate |
| 17 | Num_file_creations | 38 | Dst_host_serror_rate |
| 18 | Num_shells | 39 | Dst_host_srv_serror_rate |
| 19 | Num_access_files | 40 | Dst_host_rerror_rate |
| 20 | Num_outbound_cmds | 41 | Dst_host_srv_rerror_rate |
| 21 | Is_hot_login | 42 | class |

Figure 2. Fields in data set

## 4. RESULTS AND ANALYSIS

Table 1 to Table 11 show the results obtained for the classifiers used.

a. Neural network

The Table 1 show hidden layer = 2 and 6 was chosen as the parameter for having the highest cross validation accuracy and the model was tested again, with the test set. The Table 2 show hidden layer = 2 is found to be ideal for this dataset. Final Accuracy (test set): 96.26 % using hidden layer = 2.

Table 1. Neural network results (cross validation)

| Hidden Layers | Accuracy ( Cross Validation) |
|---|---|
| 1 | 90.6% |
| 2 | 97.67% |
| 3 | 96.34% |
| 4 | 97.01% |
| 5 | 97.34% |
| 6 | 97.67% |
| 7 | 97.00% |

Table 2. Neural network results (test data set)

| Hidden Layers | Test Accuracy |
|---|---|
| 2 | 96.26% |
| 6 | 95.97% |

b. Principle component analysis

38 features are taken after removing the factors. First 6 components are selected, since they have the highest variance.

Table 3. PCA I

| | Standard deviation | Proportion of Variance | Cumulative Proportion |
|---|---|---|---|
| PC1 | 1.543e+04 | 7.783e-01 | 7.783e-01 |
| PC2 | 8025.9533 | 0.2106 | 0.9888 |
| PC3 | 1.833e+03 | 1.098e-02 | 9.998e-01 |
| PC4 | 180.03313 | 0.00011 | 0.99992 |
| PC5 | 121.56628 | 0.00005 | 0.99997 |
| PC6 | 94.51151 | 0.00003 | 1.00000 |
| PC7 | 26.01 | 0.00 | 1.00 |
| PC8 | 3.973 | 0.000 | 1.000 |
| PC9 | 0.8027 | 0.0000 | 1.0000 |
| PC10 | 0.5096 | 0.0000 | 1.0000 |
| PC11 | 0.4524 | 0.0000 | 1.0000 |
| PC12 | 0.4335 | 0.0000 | 1.0000 |

Table 4. PCA II

| | Standard deviation | Proportion of Variance | Cumulative Proportion |
|---|---|---|---|
| PC13 | PC26 | 0.0000 | 1.0000 |
| PC14 | 0.3952 | 0.0000 | 1.0000 |
| PC15 | 0.353 | 0.0000 | 1.0000 |
| PC16 | 0.311 | 0.0000 | 1.0000 |
| PC17 | 0.2634 | 0.0000 | 1.0000 |
| PC18 | 0.2271 | 0.0000 | 1.0000 |
| PC19 | 0.2257 | 0.0000 | 1.0000 |
| PC20 | 0.1611 | 0.0000 | 1.0000 |
| PC21 | 0.1471 | 0.0000 | 1.0000 |
| PC22 | 0.1375 | 0.0000 | 1.0000 |
| PC23 | 0.1354 | 0.0000 | 1.0000 |
| PC24 | 0.1119 | 0.0000 | 1.0000 |
| PC25 | 0.09667 | 0.0000 | 1.0000 |
| PC26 | 0.0893 | 0.0000 | 1.0000 |

Table 5. PCA III

| | Standard deviation | Proportion of Variance | Cumulative Proportion |
|---|---|---|---|
| PC27 | 0.08696 | 0.0000 | 1.0000 |
| PC28 | 0.07945 | 0.0000 | 1.0000 |
| PC29 | 0.06688 | 0.0000 | 1.0000 |
| PC30 | 0.05377 | 0.0000 | 1.0000 |
| PC31 | 0.04769 | 0.0000 | 1.0000 |
| PC32 | 0.03988 | 0.0000 | 1.0000 |
| PC33 | 0.0371 | 0.0000 | 1.0000 |
| PC34 | 0.02908 | 0.0000 | 1.0000 |
| PC35 | 0.02908 | 0.0000 | 1.0000 |
| PC36 | 0.01832 | 0.0000 | 1.0000 |
| PC37 | 1.506e-12 | 0. 000e+00 | 1. 000e+00 |
| PC38 | 1.506e-12 | 0. 000e+00 | 1. 000e+00 |

c. For decision tree with PCA

The final value used for the model cp = 0.05830165. Accuracy against the test data set was obtained to be 0.6923.

Table 6. Decision tree

| For Cross Validation Data Set | | |
|---|---|---|
| cp | Accuracy | Kappa |
| 0.05830165 | 0.6851404 | 0.4989253 |
| 0.07224335 | 0.5828391 | 0.2845251 |
| 0.07477820 | 0.5730804 | 0.2561089 |

d. Bagging with random forest with PCA

nbagg = 30 is selected. The final accuracy: 0.9717.

Table 7. Random forest

| For Cross Validation Data Set | |
|---|---|
| nbagg | Accuracy |
| 10 | 0.9497 |
| 15 | 0.9560 |
| 20 | 0.9560 |
| 25 | 0.9623 |
| 30 | 0.9717 |

e. Bagging with SVM with PCA

The final value used for the model is C = 1 and nbagg = 25. The final accuracy for test data = 0.8641922 for C = 1 and nbagg = 25.

Table 8. SVM I

| For Cross Validation Data Set nbagg=20 | |
|---|---|
| c | Accuracy |
| 1 | 0.8381538 |
| 10 | 0.8375385 |
| 100 | 0.8369231 |
| 1000 | 0.8338462 |

Table 9. SVM II

| For Cross Validation Data Set nbagg=10 | |
|---|---|
| c | Accuracy |
| 1 | 0.8387692 |
| 10 | 0.8382534 |
| 100 | 0.8357934 |
| 1000 | 0.8302583 |

Table 10. SVM III

| For Cross Validation Data Set nbagg=25 | |
|---|---|
| c | Accuracy |
| 1.0 | 0.8642978 |
| 10.0 | 0.8658341 |
| 100.0 | 0.8650090 |
| 1000.0 | 0.8630626 |

f. Final accuracy

Table 11. Final accuracy comparison

| Classifier | Accuracy (in percent) |
|---|---|
| Nueral Network (Hidden Layer =2) | 96.26 |
| Random Forest (nbagg=30) | 97.17 |
| SVM (nbag=25 and c=1) | 86.41 |
| Decision Tree (Cp = 0.05830165) | 69.23 |

g. Significance of research

1) Research on intrusion detection system makes significant contributions to the society. Business organizations, banking sectors, defense system etc. can deploy such techniques to safe guard their vital data.
2) Secondly, this research makes a significant contribution to the body of knowledge by establishing methods which on learning ideal system behavior from past data discovers the patterns and deviations automatically, which is otherwise difficult to detect.

3) Finally, comparing and analyzing the accuracy of various classifiers and finding the most suitable classifier (subjective to the environment in which the system is deployed, the cost and computation precincts and the security level necessary) contributes considerably to the theory building for upgraded system design.

## 5. CONCLUSION

With the profusion in the usage of Internet for applications such as e-commerce web sites, online banking etc. protection of crucial information travelling over the network or residing in host machines becomes crucial. Effectiveness of any detection technique depends on the type and behavior of the data in the system, the environment in which the system is deployed, the type of anomalies and attacks that the system encounters, the cost and computation limitations assigned for the particular operation and the security level required. Firewalls act as a fence around the organization's network but do not provide protection from insider attacks. User authentication methods are costlier in terms of equipment and fails if the secret key which authenticates the person is leaked. Thus, there is a mammoth need for a detection system which can categorize any packet accurately as normal or intrusive in real time without having to rely on any database and being meddled by any attacker. Hybrid methods encompassing a combination of signature based and anomaly based detection can be implemented in future to offer real time detection with good detection rate.

## REFERENCES

[1] N. P. Shetty, "A Survey on Areas in Data Mining for Intrusion Detection," *International Journal of Data Mining and Knowledge Engineering*, vol. 7, no. 1, pp. 1-3, 2015.

[2] N. P. Shetty, "Using clustering to capture attackers," in *2016 International Conference on Inventive Computation Technologies (ICICT)*, pp. 1-5, 2016.

[3] A. Shenfield, et al., "Intelligent intrusion detection systems using artificial neural networks," *ICT Express*, vol. 4, no. 2, pp. 95-99, 2018.

[4] O. Depren, et al., "An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks," *Expert Systems with Applications*, vol. 29, no. 4, pp. 713-722, 2005.

[5] U. Cavusoglu, "A new hybrid approach for intrusion detection using machine learning methods," *Applied Intelligence*, vol. 49, no. 2, 2019.

[6] S. K. Kang, et al., "An implementation of hierarchical intrusion detection systems using snort and federated databases," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (Trust-Com/BigDataSE)*, pp. 1521-1525, 2018.

[7] M. Baykara and R. Das, "A novel honeypot based security approach for real-time intrusion detection and prevention systems," *Journal of Information Security and Applications*, vol. 41, pp. 103-116, 2018.

[8] S. Zhao, et al., "A dimension reduction model and classifier for anomaly-based intrusion detection in internet of things," in *2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*, pp. 836-843, 2017.

[9] S. Singh and S. Silakari, "An ensemble approach for cyber-attack detection system: A generic framework," in *2013 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pp. 79-84, 2013.

[10] M. S. Hoque, et al., "An implementation of intrusion detection system using genetic algorithm," *International Journal of Network Security and Its Applications*, vol. 4, no. 2, pp. 109-120, 2012.

[11] Y. Lin, et al., "The design and implementation of host-based intrusion detection system," in *2010 Third International Symposium on Intelligent Information Technology and Security Informatics*, pp. 595-598, 2010.

[12] F. Y. Leu and I. L. Lin, "A DOS / DDOS attack detection system using chi-square statistic approach," *Systemics, Cybernetics and Informatics*, vol. 8, no. 2, pp. 41-51, 2010.

[13] J. Seo, "An attack classification mechanism based on multiple support vector machines," in *International Conference on Computational Science and Its Applications*, pp. 94-103, 2007.

[14] S. Mukkamala, et al., "Intrusion detection using neural networks and support vector machines," in *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02*, vol. 2, pp. 1702-1707, 2002.

[15] "KDD Cup 1999 Data," 1999. [Online], Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[16] M. Tavallaee, et al., "A detailed analysis of the KDD CUP 99 data set," in *Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA'09)*, pp. 53-58, 2009.

[17] M. Brems, "A One-Stop Shop for Principal Component Analysis," *Medium*, 2017. [Online], Available: https://towardsdatascience.com/a-one-stop-shop-for-principal-component-analysis-5582fb7e0a9c.

[18] S. M. Almansob and S. S. Lomte, "Addressing challenges for intrusion detection system using naive Bayes and PCA algorithm," *2017 2nd International Conference for Convergence in Technology (I2CT)*, Mumbai, pp. 565-568, 2017.

[19] G. Karatas and O. K. Sahingoz, "Neural network based intrusion detection systems with different training functions," *2018 6th International Symposium on Digital Forensic and Security (ISDFS)*, Antalya, pp. 1-6, 2018.

[20] J. Brownlee, "A Gentle Introduction to k-fold Cross-Validation," *Machine Learning Mastery*, 2019. [Online], Available: https://machinelearningmastery.com/k-fold-cross-validation/.

[21] Udacity, "Bootstrap aggregating bagging," [online]. Available:https://www.youtube.com/watch?v=2Mg8QD0F1dQ

[22] M. Zareapoor and P. Shamsolmoali, "Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier," *Procedia Computer Science*, vol. 48, pp. 679-685, 2015.

[23] M. H. D. M. Ribeiro and L. dos S. Coelho, "Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series," *Applied Soft Computing*, vol. 86, pp. 105837, 2020.

[24] M. Belouch, et al., "Performance evaluation of intrusion detection based on machine learning using Apache Spark," *Procedia Computer Science*, vol. 127, pp. 1-6, 2018.

[25] F. A. Khan and A. Gumaei, "A Comparative Study of Machine Learning Classifiers for Network Intrusion Detection," *International Cnference on Artificial Intelligence and Security*, pp. 75-86, 2019.

[26] S. Zwane, et al., "Performance Analysis of Machine Learning Classifiers for Intrusion Detection," *2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC)*, Plaine Magnien, pp. 1-5, 2018.

## BIOGRAPHIES OF AUTHORS

**Nisha P. Shetty** has published in the areas network security and machine learning. Currently she is working in the area of social network security.

**Jayashree** is currently working in machine learning domains. She is interested in exploring the domain of machine learning and analytics in medical realm.

**Rohil Narula** is interested in areas like machine learning and analytics.

**Kushagra Tandon** is interested in areas like machine learning and analytics.