Computers, Materials & Continua
DOI:10.32604/cmc.2021.014307
Article

Tech Science Press

Enhancing Network Intrusion Detection Model Using Machine Learning Algorithms

Nancy Awadallah Awad[*]

Department of Computer and Information Systems, Sadat Academy for Management Sciences, Cairo, 11742, Egypt
[*]Corresponding Author: Nancy Awadallah Awad. Email: rarecore2002@yahoo.com

Received: 12 September 2020; Accepted: 22 November 2020

Abstract: After the digital revolution, large quantities of data have been generated with time through various networks. The networks have made the process of data analysis very difficult by detecting attacks using suitable techniques. While Intrusion Detection Systems (IDSs) secure resources against threats, they still face challenges in improving detection accuracy, reducing false alarm rates, and detecting the unknown ones. This paper presents a framework to integrate data mining classification algorithms and association rules to implement network intrusion detection. Several experiments have been performed and evaluated to assess various machine learning classifiers based on the KDD99 intrusion dataset. Our study focuses on several data mining algorithms such as; naïve Bayes, decision trees, support vector machines, decision tables, k-nearest neighbor algorithms, and artificial neural networks. Moreover, this paper is concerned with the association process in creating attack rules to identify those in the network audit data, by utilizing a KDD99 dataset anomaly detection. The focus is on false negative and false positive performance metrics to enhance the detection rate of the intrusion detection system. The implemented experiments compare the results of each algorithm and demonstrate that the decision tree is the most powerful algorithm as it has the highest accuracy (0.992) and the lowest false positive rate (0.009).

Keywords: Intrusion detection; association rule; data mining algorithms; KDD99

1 Introduction

The IDS has 3 methods of identifying attacks; signature, anomaly, and hybrid-based detection. The first method, signature-based detection is formed by using signatures of those attacks to detect the known ones. This is an efficient means of identifying known attacks that are preloaded in the IDS database. Hence it should be much more accurate to identify a known attack intrusion attempt [1].

The disadvantage of this method is that the attack which has new forms cannot be identified, as their signatures are not displayed; as such the databases are regularly modified to improve their identification effectiveness [2].

Anomaly-based monitoring which matches actual user practices to predefined profiles is used to identify suspicious habits that may be intrusions. Anomaly-based identification is successful against unexpected threats without any system updates [3].

In sum, the availability, integrity, and confidentiality of computer networks are threatened by various types of attacks.

The Denial of Service attack (DoS) has been considered one of the most frequent and harmful ones.

DoS attacks aim to temporarily deny multiple end-user services.

It normally absorbs network bandwidth, which overloads the system with unnecessary demands.

Despite the above, DoS serves as a broad shield for all forms of threats targeted at accessing machine and network resources.

The researcher in this paper presents a framework to integrate data mining algorithms and association rules to implement network intrusion detection. Several experiments have been performed and evaluated to assess various machine learning classifiers based on the KDD intrusion dataset.

2 Literature Reviews

In this section, the researcher of this paper shows several previous studies concerned with machine learning techniques a deal with network intrusion detection.

Othman et al. [2] implemented the Spark-Chi-SVM model for intrusion detection by using the SVM classifier on Apache Spark Big Data platform using ChiSqSelector for feature selection and KDD99 to train and test the model. They proofed that the Spark-Chi-SVM model reduces the training time and is efficient for big data and it has high performance.

Peng et al. [4] suggested a decision tree-based IDS framework over Big Data in Fog Environment. The researchers implemented pre-processing algorithms to find the strings in the provided dataset and then standardize the results to ensure the accuracy of the input data to increase detection performance. The IDS decision tree approach with the Naïve Bayesian method and KNN system on the KDDCUP99 dataset. The results showed that this proposed method was effective and accurate.

Rupa et al. [5] said that using machine learning algorithms on the CIDDS-001 dataset provides better results than the existing research method. Deep learning algorithms were implemented on the latest datasets like CIDDS-001 and CIDDS-002 to improve the computational time and cost.

Diro et al. [6] adopted deep learning, to cybersecurity as a new approach, to enable the detection of attacks on the social internet of things. The experiments have shown that their distributed attack detection system is superior to centralized detection systems using a deep learning model. It has also been demonstrated that the deep model is more effective in attack detection than its shallow counterparts.

Liu et al. [7] introduced a survey that proposed a taxonomy of IDS that takes data objects as the main dimension to classify and summarize machine learning-based and deep learning-based IDS literature. The survey clarified the concept and taxonomy of IDSs and it fit for cybersecurity researchers. Researchers explained how to solve key IDS issues with machine learning and deep learning techniques.

Yavuz et al. [8] proposed a deep-learning-based machine learning method for the detection of routing attacks for IoT. The Cooja IoT simulator has been utilized for the generation of high-fidelity attack data, within IoT networks ranging from 10 to 1000 nodes. They proposed a highly scalable, deep-learning-based attack detection methodology for the detection of IoT routing attacks which are decreased rank, hello-flood, and version number modification attacks, with high accuracy and precision.

Panda et al. [9] compared the effectiveness of the classification algorithm, Naïve Bayes, with the decision tree algorithms namely, ID3 and J48. This helps one to construct an effective network intrusion detection system that the Naïve Bayes model is quite appealing because of its simplicity, elegance, robustness, and effectiveness.

Nalavade et al. [10] implemented a NID system by represented a model to integrate association rules to intrusion detection. They proofed that IDS using association rules can create attack rules that maintain a low false-positive rate.

Adebowale et al. [11] evaluated the performance of well-known classification algorithms for attack classification by applying the NSL-KDD dataset.

3 Data Mining Algorithms for Network Intrusion Detection

In this section, several data mining algorithms will be presented, including classification techniques, and association rule mining. Data mining algorithms Naïve Bayes, Decision trees, Support vector machines, Decision table, K-nearest neighbor algorithm and, Artificial neural network.

3.1 Classification Data Mining Techniques

3.1.1 Naïve Bayes (NB)

The advantages of the Naïve Bayesian technique are the ability to encode interdependencies between variables [12] and to forecast events, and the ability to integrate both prior information and data [13].

The most drawback of the naïve Bayesian method is the lack of available probability data. Another disadvantage that their results are comparable to those derived from threshold-based systems, although a considerably higher computational effort is required [14].

3.1.2 Decision Trees (DT)

The classification algorithm in this method is taught inductively to build a model from the pre-classified data set. Every data object is described by attribute values, and classification can be interpreted as a mapping from a set of attributes to a specified class [15].

Choosing a given division depends on the test outcome. The beginning at the root node and follow the assumptions down before reaching a terminal node, to identify a specific data object. A decision is taken when a terminal node is achieved [14]. The benefit of making use of this algorithm is that no domain knowledge is expected from its construction. This increases the appropriateness of DT algorithms for IDS in particular when taking into account the complexity and ever-growing scale of network communication results. Decision trees can process numerical and categorical data (This is in line with the alphanumeric existence of network link data) [15]. The lack of Decision tree algorithms is not unreliable, and computational schema trees can be complex.

3.1.3 K-Nearest Neighbor (K-NN)

It is a form of lazy learning that only approximates the function locally and delays all computations until classification. One of the easiest of all machine learning algorithms is the K-nearest neighbor algorithm: an object is categorized by a majority vote of its neighbors, assigning the object to the most common class of its nearest neighbors. Test instances are related to the instances stored, and the same class-mark is given to the most comparable instances stored in K.

3.1.4 Artificial Neural Networks (ANN)

Neural networks were used both in the detection of intrusion abnormalities and in the detection of intrusion abuse [13]. Neural networks were modeled to learn the typical characteristics of system users for the detection of anomaly intrusion and to identify statistically significant variations from the user's established behavior.

The neural network will lead to intrusion detection harassment will collect network stream data and evaluate the data for incidents of abuse. Another drawback of the algorithm for the neural network is its comparatively greater computing load [14].

3.1.5 Support Vector Machine (SVM)

SVM is introduced as a novel intrusion detection technique. By some nonlinear mapping, an input of SVM maps (really valued) features vectors into a higher-dimensional feature space.

SVMs were built based on the systemic risk minimization principle [14].

The goal of systemic risk minimization is to consider a theory (h) for which the lowest likelihood can be found error while conventional pattern recognition learning approaches are focused on mitigating empirical risk, which aims to maximize the learning set's efficiency.

3.1.6 Decision Table

Decision Table constructs a greater component classifier list of baseline decisions. It tests highlighted subsets using the best initial quest and can use cross-approval for evaluation. The algorithm traverses 4 stages: Pruning, optimization, and selection are the norm [16].

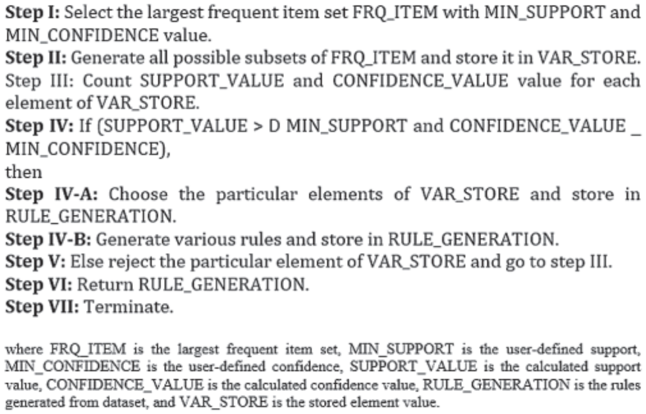
3.2 Association Rule Mining Algorithm

This algorithm is used to test the large audit data sets and to calculate support-confidence to discover the regular sets of objects. It creates rulesets based on the IF-THEN rules' frequent item-sets.

The ruleset is created with the help of values of support and trust. Generally, the rulesets are easier to understand When opposed to other algorithms. Every ruleset rule defines a specific context associated with a class or attribute. The goal of this approach is to improve performance based on the class level. The proposed algorithm is modified from the algorithm for the Apriori association rules [17,18].

In this paper, the KDD dataset includes 42 features, the protocol attribute could be "ICMP", "UDP," or "TCP", each protocol is described in terms of their services and flags. If the protocol is "ICMP" JJ, "UDP" JJ, "TCP" JJ then the class is declared to be Normal, DoS, Probe, R2L, or U2R.

This method is tested before there is a sufficient number of instances. For each group, the rules were created based on the importance of support and trust. Fig. 1 shows steps that are used to generate rules [19].



Step I: Select the largest frequent item set FRQ_ITEM with MIN_SUPPORT and MIN_CONFIDENCE value.
Step II: Generate all possible subsets of FRQ_ITEM and store it in VAR_STORE.
Step III: Count SUPPORT_VALUE and CONFIDENCE_VALUE value for each element of VAR_STORE.
Step IV: If (SUPPORT_VALUE > D MIN_SUPPORT and CONFIDENCE_VALUE _ MIN_CONFIDENCE),
then
Step IV-A: Choose the particular elements of VAR_STORE and store in RULE_GENERATION.
Step IV-B: Generate various rules and store in RULE_GENERATION.
Step V: Else reject the particular element of VAR_STORE and go to step III.
Step VI: Return RULE_GENERATION.
Step VII: Terminate.

where FRQ_ITEM is the largest frequent item set, MIN_SUPPORT is the user-defined support, MIN_CONFIDENCE is the user-defined confidence, SUPPORT_VALUE is the calculated support value, CONFIDENCE_VALUE is the calculated confidence value, RULE_GENERATION is the rules generated from dataset, and VAR_STORE is the stored element value.

Figure 1: Association rule mining algorithm

3.2.1 *Apriori Algorithm*

This algorithm is easy to apply and is very fast. Since the first implementation of the Apriori algorithm and the accumulation of knowledge, various attempts have been made to formulate more effective algorithms for regular item-set mining. This involves methodology focused on hash, partitioning, sampling. Apriori is a seminal algorithm that uses candidate generation to find frequent item-sets [17]. This algorithm uses item-sets anti-monotonicity," If an Items are not frequent set, none of their supersets is ever frequent.

4 Proposed a Framework to Classify Network Intrusion

This paper proposed a framework for NID that processing the KDD dataset, and apply classification techniques and association rule mining. In this framework, the researcher implements association rules to create rules to detect the attack. Fig. 2 illustrated the proposed framework.

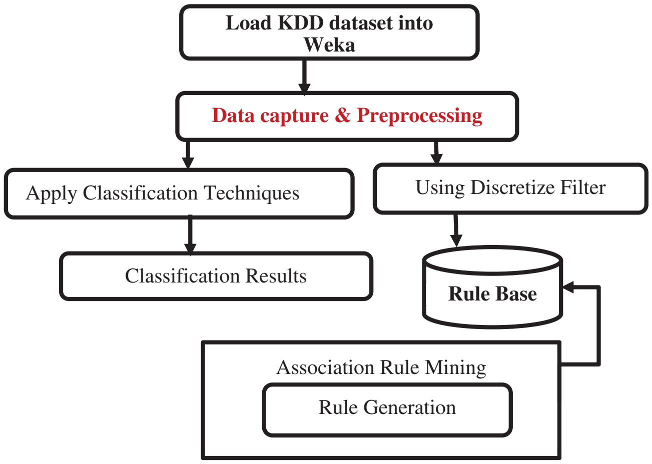


Figure 2: Proposed framework for network intrusion detection

4.1 KDD Dataset

The dataset KDD99 is used to test the proposed pattern. It has 42 attributes and the number of instances used is 494.021.

Label attribute contains 37 attacks which classified into 4 main attacks as the following:

- Denial of Service (DoS): Use of resources that deny legal users legitimate requests on the system.
- Remote to Local (R2L): A non-domain intruder achieves a legitimate user account on the victim's computer by transmitting packets through networks
- User to Root (U2R): Attacker attempts to access limited machine privileges.
- Probe: Attacks that can search a computer network to gather details or find known vulnerabilities [20].

Tab. 1 illustrated the relevant feature and class name for each attack.

Table 1: Relevant feature and class for each attack [21]

Class	Attack name	Relevant features	Class	Attack name	Relevant features
DoS	apache2	5, 6, 12, 15, 29, 32, 37, 38	DoS	pod	20, 28, 41
DoS	back	16, 24, 17	PROBE	portsweep	6, 12, 15, 29, 37
U2R	buffer_overflow	22, 26, 27, 30	DoS	processtable	9, 10, 12, 13, 14, 15, 20, 28, 33, 38
R2L	ftp_write	40	U2R	ps	18, 27
R2L	guess_passwd	23, 28, 33	U2R	rootkit	18, 27, 40
R2L	httptunnel	12, 27, 29	PROBE	saint	11, 12, 15, 31
R2L	imap	6, 12	PROBE	satan	5, 15, 31
PROBE	ipsweep	11	R2L	sendmail	23, 30
DOS	land	13, 38	DoS	smurf	7, 10, 20, 28
U2R	loadmodule	30	R2L	snmpgetattack	20, 28, 33, 35, 36
DoS	mailbomb	6, 8, 12	R2L	snmpguess	5, 10, 20, 33, 35, 36
PROBE	mscan	3, 5, 6, 9, 12, 13, 15, 29, 31, 32	U2R	sqlattack	30
R2L	multihop	10, 17, 18, 21, 23, 27	DoS	teardrop	41
R2L	named	14, 18	DoS	udpstorm	9
DoS	neptune	3, 5, 6, 12, 13, 15, 31, 32	R2L	warezmaster	7, 10, 17, 34
PROBE	nmap	13	R2L	worm	5, 8, 17, 20, 28, 36
NORMAL	normal	8, 9, 12, 13, 17, 18, 19, 28, 29, 40, 41	R2L	xlock	23, 34
U2R	perl	11, 30	R2L	phf	12, 15, 30

Fig. 3 Illustrated the classification of 37 attacks into 4 main classes (DoS, U2R, R2L, PROBE).

After data preprocessing via the Weka tool, the framework is divided into two parts:

- Applying classification data mining techniques
- Applying association rule: To implement this process, "Discretize Filter" should be used.

In the next section, the two algorithms will be implemented.

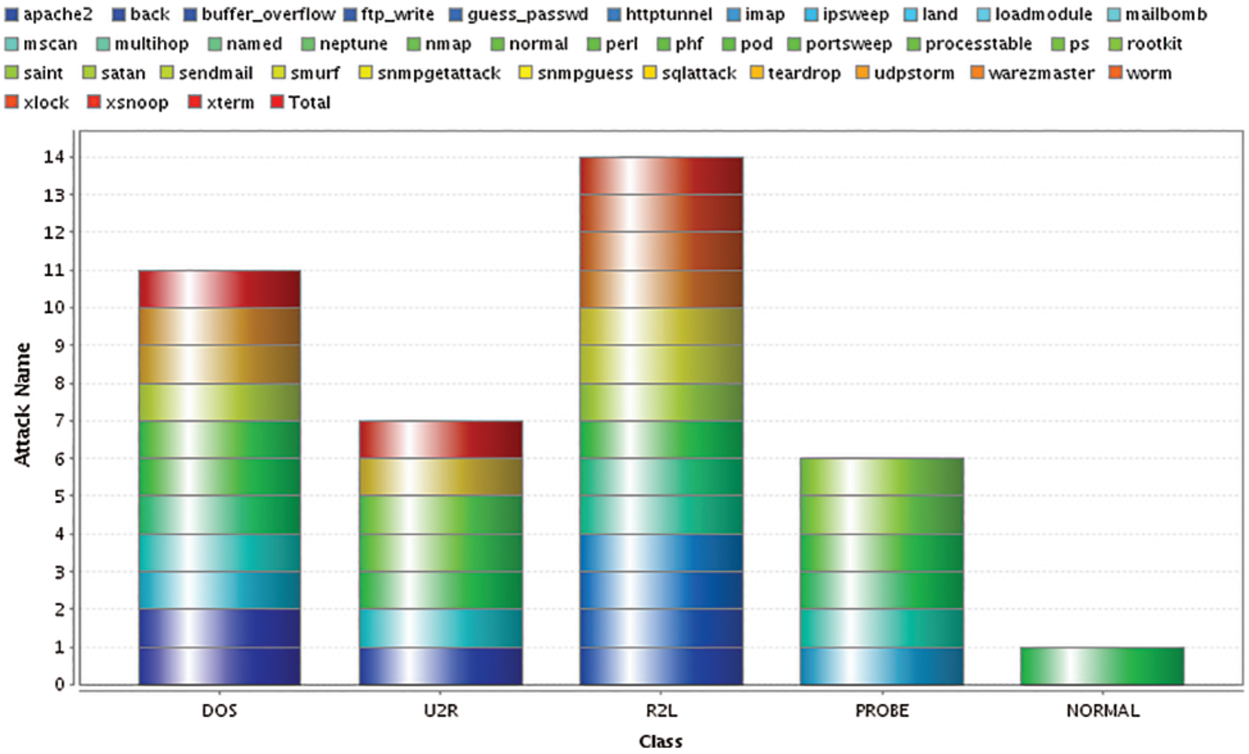


Figure 3: The main four categories of attacks (DoS, Probe, U2R, R2L)

5 Results

5.1 Performance Parameters

Many measures are available for assessing system performance. Typically tests after tests are used for assessing intrusion detection.

- True positive (TP): The number of corrected instances classified as an intrusion.
- True Negative (TN): The number of incorrect instances classified as an intrusion.
- False-positive (FP): The number of intrusion instances that were incorrectly classified as normal.
- False-negative (FN): The number of normal instances that were incorrectly classified as an intrusion.

To determine how many misclassifications are found we use the term Recall. Precision is how many records are correctly classified by the system [10].

$$\text{Precision} = \frac{\text{TP}}{\text{Total number of positive connections}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

5.2 *Experimental Results*

5.2.1 Applying Classification Techniques

This section demonstrates the result of implemented several classification techniques such as Naïve Bayes, Decision tree, Decision table, SVM, KNN, and ANN. Tab. 2 shows the comparison between these techniques from true, positive rates and precision for normal and anomaly classes. The results show that the precision of decision tree J48 is the best of all techniques.

Table 2: Comparison between true, false positive rates and precision ratio of classification algorithms

Method	Algorithm	TP (Normal)	TP (Anomaly)	FP (Normal)	FP (Anomaly)	Precision (Normal)	Precision (Anomaly)
Bayes.NaiveBayes	NB	0.908	0.892	0.108	0.092	0.906	0.894
J48	Decision tree	0.991	0.991	0.009	0.009	0.992	0.99
SMO	SVM	0.995	0.989	0.011	0.005	0.99	0.994
rules.DecisionTable	Decision table	0.992	0.978	0.022	0.008	0.981	0.991
Lazy.LWL	KNN	0.977	0.915	0.085	0.023	0.929	0.972
Functions.MultilayerPreceptron	ANN	0.985	0.952	0.048	0.015	0.96	0.982

Tab. 2 indicated the comparison between Naïve Bayes, Decision tree, Decision table, SVM, KNN, and ANN algorithms, and the results showed that the decision tree is the best algorithm used to classify network intrusion detection as the accuracy parameter is (0.992) the highest one and false-positive (0.009) is the lowest one, while Naïve Bayes achieve the lowest accuracy and the highest false-positive rate.

Fig. 4. described previous results.

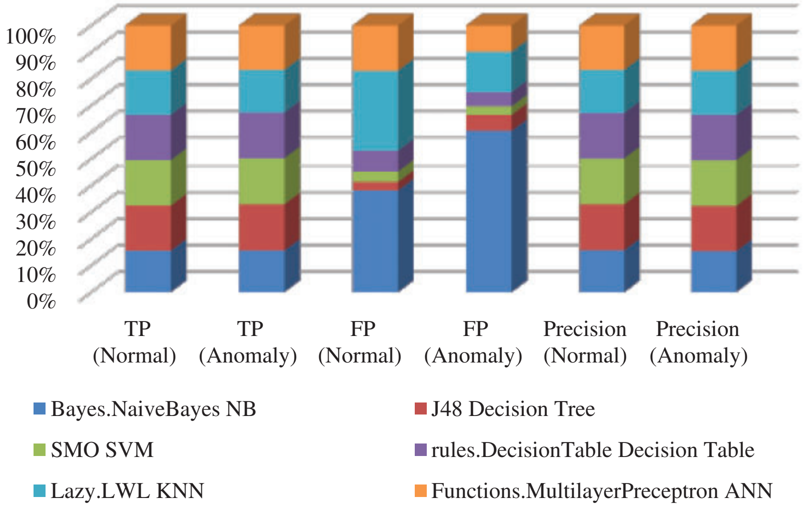


Figure 4: True and false-positive rates of classification algorithms

Tab. 3 indicated the comparison between previous algorithms in the ratio of correctly, incorrectly classified instances and time is taken to build a model. The results pointed to that correctly classified instances of the J48 technique is the best one (99.10%) and the Lazy.LWL has taken 0.01 s to build a model.

Table 3: Comparison between classification techniques in (correctly, incorrectly and time is taken to build a model)

Method	Correctly classified instances (%)	Incorrectly classified instances (%)	Time is taken (s)
Bayes.NaiveBayes	90.04	9.96	0.06
J48	99.10	0.90	0.35
SMO	99.22	0.78	2.6
rules.DecisionTable	98.58	1.42	1.87
Lazy.LWL	94.79	5.21	0.01
Functions.MultilayerPreceptron	96.98	3.02	0.043

Fig. 5 described the previous results.

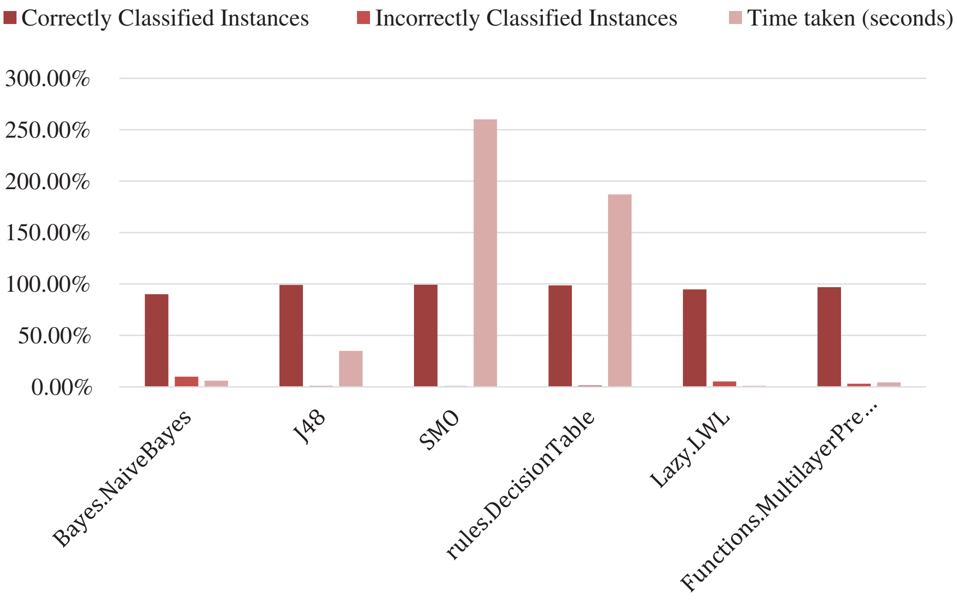


Figure 5: Comparison between comparison between classification techniques in (Correctly, incorrectly and time is taken to build a model)

5.2.2 Applying Association Rule Mining

This section discusses the experiment result for applying the association rule. The focus on finding association rules to detect DoS, R2L attacks.

Rule mining is implemented for attributes of the KDD99 dataset which are (service, duration, source bytes, protocol type, flag, and destination bytes).

Rules generated from creating rules for IDS who exceed minimum support 50% and confidence is 90% threshold.

Tab. 4 shows some of the rules created from frequent itemset and satisfying minimum support and confidence.

Table 4: The generated association rules with support and confidence values

Id	Antecedent	Consequent	Support	Confidence	Lift
1	protocol_type = icmp	flag = SF	0.5196	1	1.31
2	service = ecr_i, flag = SF	protocol_type = icmp	0.5260	1	1.74
3	protocol_type = icmp, flag = SF	hot > 3	0.5196	1	1.01
4	service = ecr_i, label = smurf	protocol_type = icmp	0.5260	1	1.74
5	label = smurf	flag = SF	0.5260	1	1.31
6	label = smurf	protocol_type = icmp	0.5196	1	1.74
7	dst_bytes, protocol_type = tcp	label = guess password	0.5196	0.94	1.15
8	protocol_type = tcp	label = guess password	0.5260	0.92	1.12

As indicated in Tab. 1 that smurf attack is categorized under DoS attack and guess password is categorized under R2L, the rules 4, 5, 6, 7, 8 indicated that the relation between:

- Service attribute when it's value "erc_i", label value "smurf" => protocol_type attribute value "icmp",
- Label "smurf" => flag attribute value "SF"
- Label name "smurf" => protocol_type value "icmp"
- Dst_bytes, protocol_type = "tcp" => label = guess password
- Protocol_type = "tcp" => label = guess password

6 Conclusion

To achieve any improvement in network intrusion detection, the researcher should focus to reduce false-positive rates and increase the accuracy rate. A framework was presented to integrate data mining classification techniques and association rules to implement network intrusion detection. This framework is used to detect the unknown attacks with a high accuracy rate and low false-positive rate and illustrated that the decision tree is the best classification. This paper demonstrated the association rules with the Apriori algorithm which is applied to the KDD cup 1999 dataset. The results showed that association rules detect DoS, R2L attacks.

Funding Statement: The author received no specific funding for this study.

Conflicts of Interest: The author declares that she has no conflicts of interest to report regarding the present study.

References

[1] A. Sahasrabuddhe, S. Naikade, A. Ramaswamy, B. Sadliwala and P. R. Futane, “Survey on intrusion detection system using data mining techniques,” *International Research Journal of Engineering and Technology*, vol. 4, no. 5, pp. 1780–1784, 2017.

[2] S. M. Othman, F. M. Ba-Alwi, N. T. Alsohybe and Y. A. Amal, “Intrusion detection model using machine learning algorithm on big data environment,” *Journal of Big Data*, vol. 5, no. 1, pp. 521, 2018.

[3] L. Dali, A. Bentajer, E. Abdelmajid, K. Abouelmehdi, H. Elsayed *et al.,* “A survey of intrusion detection system,” in *2nd World Sym. on Web Applications and Networking*, Tunisia, Piscataway: IEEE, pp. 1–6, 2015.

[4] K. Peng, V. C. M. Leung, L. Zheng, S. Wang, C. Huang *et al.,* “Intrusion detection system based on decision tree over big data in fog environment,” *Wireless Communications and Mobile Computing*, vol. 2018, no. 5, pp. 1–10, 2018.

[5] T. Rupa Devi and S. Badugu, “A review on network intrusion detection system using machine learning,” in *Int. Conf. on Emerging Trends in Engineering 2019, LAIS*, Switzerland: Springer Nature Switzerland, vol. 4, pp. 598–607, 2020.

[6] A. A. Diro and N. Chilamkurti, “Distributed attack detection scheme using deep learning approach for Internet of Things,” *Future Generation Computer Systems*, vol. 82, pp. 761–768, 2017.

[7] H. Liu. and B. Lang, “Machine learning and deep learning methods for intrusion detection systems: A survey,” *Applied Sciences*, vol. 9, no. 20, pp. 4396, 2019.

[8] F. Y. Yavuz, D. Ünal and E. Gul, “Deep learning for detection of routing attacks in the Internet of Things,” *International Journal of Computational Intelligence Systems*, vol. 12, no. 1, pp. 39–58, 2018.

[9] M. Panda and M. R. Patra, “A comparative study of data mining algorithms for network intrusion detection,” in *First Int. Conf. on Emerging Trends in Engineering and Technology*, Nagpur, Maharashtra, 2008.

[10] K. Nalavade and B. B. Meshram, “Mining association rules to evade network intrusion in network audit data,” *International Journal of Advanced Computer Research*, vol. 4, no. 2, pp. 560–567, 2014.

[11] A. Adebowale, S. A. Idowu and A. Amarachi, “Comparative study of selected data mining algorithms used for intrusion detection,” *International Journal of Soft Computing and Engineering*, vol. 3, no. 3, pp. 237–241, 2013.

[12] D. Barbara, N. Wu and S. Jajodia, “Detecting novel network intrusions using Bayes estimators,” in *Proc. of the First SIAM Int. Conf. on Data Mining (SDM 2001)*, Chicago, IL, 2001.

[13] T. D. Lane, “Machine learning techniques for the computer security domain of anomaly detection,” Ph.D. dissertation, Purdue University, Electrical and Computer Engineering, 2000.

[14] S. Brugger, “Data mining methods for network intrusion detection,” Ph.D. dissertation, University of California, Davis, pp. 1–65, 2011.

[15] K. Reddy, M. Iaeng, V. N. Reddy and P. G. Rajulu, “A study of intrusion detection in data mining,” in *Proc. of the World Congress on Engineering 2011*, London, UK, 2011.

[16] V. Veeralakshmi and D. Ramyachitra, “Ripple down rule learner (RIDOR) classifier for iris dataset,” *International Journal of Computer Science Engineering*, vol. 4, no. 3, pp. 79–85, 2015.

[17] L. Hanguang and N. Yu, “Intrusion detection technology research based on apriori algorithm,” *International Conference on Applied Physics and Industrial Engineering*, vol. 24, pp. 1615–1620, 2012.

[18] M. Jiang, X. Gan, C. Wang and Z. Wang, “Research of the intrusion detection model based on data mining,” *Elsevier Energy Procedia*, vol. 13, no. 4, pp. 855–863, 2011.

[19] S. Devaraju and S. Ramakrishnan, “Detection of attacks for ids using association rule mining algorithm,” *IETE Journal of Research*, vol. 61, no. 6, pp. 624–633, 2015.

[20] R. R. Chaudhari and S. P. Patil, "Intrusion detection system: Classification, techniques and datasets to implement," *International Research Journal of Engineering and Technology*, vol. 4, no. 2, pp. 1860–1866, 2017.

[21] A. A. Olusola, A. S. Oladele and D. O. Abosede, "Analysis of KDD '99 intrusion detection dataset for selection of relevance features," in *Proc. of the World Congress on Engineering and Computer Science 2010*, San Francisco, USA, vol. 1, 2010.