Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.025893
Article

Tech Science Press

Association Rule Mining Frequent-Pattern-Based Intrusion Detection in Network

S. Sivanantham[1,*], V. Mohanraj[2], Y. Suresh[2] and J. Senthilkumar[2]

[1]Department of Information Technology, Adhiyamaan College of Engineering, Hosur, Tamilnadu, India
[2]Department of Information Technology, Sona College of Technology, Salem, Tamilnadu, India
*Corresponding Author: S. Sivanantham. Email: ssivanantham21@outlook.com
Received: 08 December 2021; Accepted: 16 February 2022

Abstract: In the network security system, intrusion detection plays a significant role. The network security system detects the malicious actions in the network and also conforms the availability, integrity and confidentiality of data information resources. Intrusion identification system can easily detect the false positive alerts. If large number of false positive alerts are created then it makes intrusion detection system as difficult to differentiate the false positive alerts from genuine attacks. Many research works have been done. The issues in the existing algorithms are more memory space and need more time to execute the transactions of records. This paper proposes a novel framework of network security Intrusion Detection System (IDS) using Modified Frequent Pattern (MFP-Tree) via K-means algorithm. The accuracy rate of Modified Frequent Pattern Tree (MFPT)-K means method in finding the various attacks are Normal 94.89%, for DoS based attack 98.34%, for User to Root (U2R) attacks got 96.73%, Remote to Local (R2L) got 95.89% and Probe attack got 92.67% and is optimal when it is compared with other existing algorithms of K-Means and APRIORI.

Keywords: IDS; K-means; frequent pattern tree; false alert; mining; L1-norm

1 Introduction

The hacking and exploiting technologies develops rapidly and identification of new techniques for detection of intrusion system has become a great challenge in the globe of providing network security [1]. An IDS in the network is an essential of all system using Internet connection, because of internal and external attack [2,3] possibilities. The technique used in the concept of intrusion detection combines data mining technology and becomes popular in recent days in particular associative mining rule is used for classification. In the implementation of advanced security concept, sharing of information through internet always is affected by unknown threats. The most common threats are Denial of Service, Remote to Local (R2L), Probing, users to the Root (U2R) etc. The aim of intrusion detection system (IDS) is an automatic detection of attacks in continuous flow stream of information in the traffic network and generates alert signal.

The infrastructure of IDS is attack driven in the network during data transmission. Each transaction by the attacker has to raise an alert signal possibly, as well as alerting the administrator about the malicious activity in the network. In the network security, the anomaly based intrusion detection framework is used

for identifying intruders and monitoring the activity of the system as to categorize the transactions as normal and anomalous. This type of classification is done by applying rules or heuristics.

This paper uses associative rule mining which classify the attack types and generates the alarm signal. In the data mining frequent patterns tree method (FPT) is used for detecting the intruders and thus ensuring the network security. The advantage of proposed work is it needs less memory for storage and less time to identify the frequent item sets. The frequent pattern (FP) related distributed mining techniques are FP Forest [4], Parallel-FP (PFP) trees [5], machine learning pattern tree (MLPT) [6] and layered FP (LFP)-tree [7].

By applying modified FPT via K-Means algorithm, intrusion detection system can be widely used to verify the data obtained from network traffic and compare it is as normal data and abnormal or attack. This proposed work effectively detects the malicious attack and can separate the normal data from the huge amount of data in the network. To implement this, association rule mining with FPT algorithm via K-means is used in this paper. The main research contribution of this work is:

1. The input raw IDS data set is pre-processed with L1-Norm Kernel principal component analysis (PCA) for removing the outlier in the data set.
2. Implementing IDS for network security using Associative rule mining in an efficient way.
3. Applying associative rule in modified frequent pattern tree via K-means algorithm in the network security IDS.

The article has been arranged as follows: Section 2 discusses about the review of the existing methods, Section 3 give introduction on the in-network based IDS using MFPT-K Means, Section 4 concentrates on experimented result and Section 5 gives final conclusion with future direction of this research.

2 Review of Literature

Network security is major issue in providing security and sharing information in an efficient way. In the network system, detecting intrusion is a severe issue because single intrusion attack can modify, delete or steal the information. For implementing the intrusion detection data mining technique i.e., associative rule mining is used for the purpose of auditing data in the network traffic with accurate building blocks of IDS [8,9]. To protect the computer system and networks from the unknown threat, network intrusion detection plays an significant role in implementing the security defense. Deep learning techniques are successfully implemented in solving the problems for detecting intrusion detection system [10,11]. The outcome of implementing the network-based intrusion detection system uses deep learning algorithm to classify the attack as normal and abnormal attack. In the mining techniques, association rule mining is a classical method in the incorporation of IDS [12].

The steps involved in the association rules mining is identifying the frequent item sets which has support transaction in the network traffic. Similarly for generating the association rules in the database by using confidence. In the association rule mining, the knowledge base of rules is built using minimum confidence. The FP-growth and APRIORI algorithm is used in the association rule mining for Network IDS (NIDS) [13–15]. This paper was proposed for implementing the IDS datasets such as knowledge discovery dataset (KDD-CUP'99) and NSL-KDD. It performs the comparative analysis between different machine learning algorithms like Decision Tree, J48, Navie Bayes Tree, Random Forest, multi layer perceptron (MLP) and support vector machine (SVM) in those two datasets. In the paper [16] the authors have implemented stacked SVM for the IDS concept. It is the combination of the genetic algorithm with SVM for increasing the accuracy in NIDS. This paper [17–22] proposed hybrid classifiers by fusion of different algorithms like Random Tree, reduced error pattern (REP) Tree, Naïve Bayes and J48 and by using both the supervised and unsupervised learning process. The proposed association rule mining technique for network traffic data in the web services such as file transfer protocol (FTP), hyper text transfer protocol (HTTP) and HTTPS traffic in the identifying the unknown attacks of DoS attacks.

3 Proposed MFPT-K Means Methodology

A network intrusion detection system for detecting malicious attack by applying association rule in modified frequent Pattern Tree via K-means algorithm (MFPT-K means) is proposed. This modified algorithm consists of three stages. They are pre-processing, mining association rules and applying FPT via K-Means algorithm. The proposed architecture of this research is given in Fig. 1.

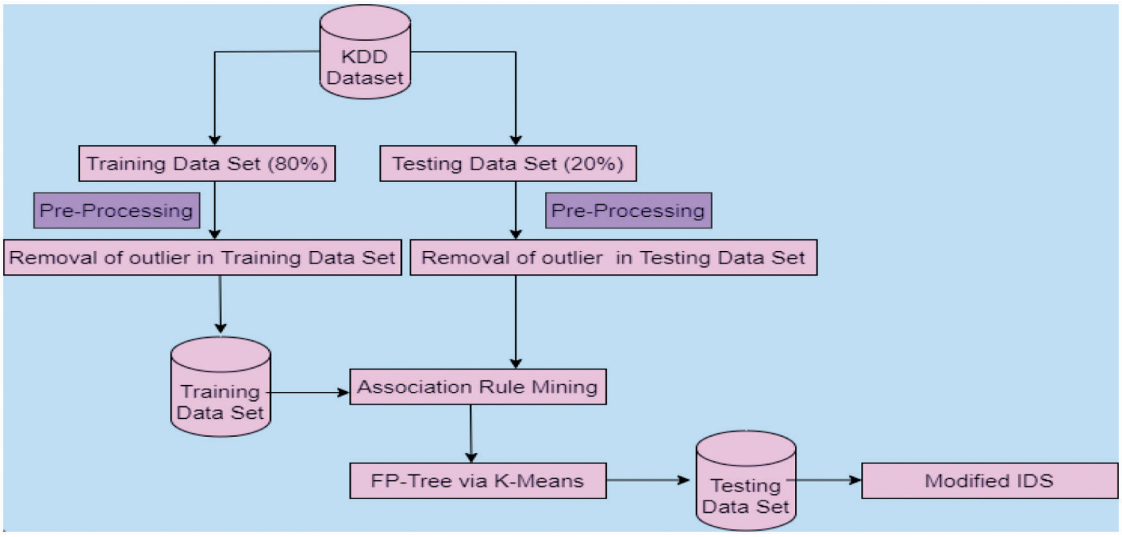

Figure 1: Architecture of MFPT-K means

In the Fig. 1, this modified algorithm is implemented based on the datasets of KDD Cup 1999 and Network Security Laboratory (NSL), considering the size for testing data and training data in the ratio of 20:80. For the pre-processing stage in this work, initially wiener filter is applied. Then association rule mining is implemented. Then training and testing the model is done using FP-Tree via K-means algorithm.

3.1 Pre-Processing

The anomaly detection in any network uses IDS in efficient manner if and only if the removal of outlier is done before mining the association rules. To remove the outlier, L1-Norm Kernel Principal Component Analysis (PCA) algorithm is used. This L1-Norm is a robust approach for detecting the outlier in the network-based IDS. L1-Norm is a minimized solution of linear optimization problem. It defines a variable of non-negative to unknown parameters and residuals of real numbers. The implementation of L1-Norm Kernel PCA algorithm is given below.

Algorithm 1: Removal of outlier using L1-Norm based Kernel PCA

Input: Read kernel matrix km

Output: Vector Sign value sc^*

Step 1: Identify $i^* = \arg max_{1\le i\le n} \sum_{j=1}^{n} |km_{ij}|/\sqrt{km_{ij}}$

Step 2: Initialize the vector sign $sc^0 with sc_i^0 = sign(km_{ij*})$

Step 3: $km \leftarrow -1$

Step 4: Repeat

Step 5: $km \leftarrow km + 1$

Step 6: Evaluate $sc^{km+1} = sign(kmsc^{km})$

Step 7: until $(sc^{km} - sc^{km+1})\ ^T km(sc^{km} - sc^{km+1}) = 0$

In the Algorithm 1 describes the removal of outlier in which the output of sign vector *sc* then compute the sores of principal component without mapping explicitly. The PCA for the i^{th} component can be evaluated as:

$$\widetilde{km} = km - \frac{(kmsc^*)(kmsc^*)^T}{(sc^*)^T kmsc^*} \tag{1}$$

3.2 Association Rule Mining

After removing the outliers found during detection, FP algorithm is implemented with association rule mining. In IDS, within the stipulated time interval, there are situations where more than one alert i.e., a group is raised for a single transaction. At the same time, it can discard the malicious or duplicate alerts. After the association rule mining is applied with support and confidence threshold value, obtaining association rules is the next step. It uses the data from the data base for transaction and represents the set of correlated items. Association rule mining identifies the correlation between the parameters/attributes within the data in the network. In association mining, the network data achieves good flexibility in forming the various relations. The mining outputs are defined as rules [23].

The association rule contains group of items item = {it1, it2, …, itn} and transaction of data items in the database db = {tr1, tr2, …,trn}. The implication of association rule is G → H. Network-based intrusion detection is G and H are itemsets of {fi1 = val1, fi2 = val2, fi3 = val3, …,fin = valn}, where f is field name or item name and val is a item value. The association rule G → H satisfy the minimum confidence and support i.e., G belongs to item and H belongs to item [24].

The association rule support is computed in percentage of transaction in both G and H. It measures the correlation between significance of items to be found together as itemsets. The measure of support is majorly the frequency of the item in the database.

3.3 Proposed Network-Based IDS Using Modified FP-Tree via K-Means Algorithm

The network-based intrusion detection system based on associate rule mining using modified frequent Pattern Tree via K-means algorithm (MFPT-K means) is implemented in this phase. Frequent Pattern Tree method helps to create frequent items occurring together assets from the database and it is called as frequent itemsets. This NIDS (Network-Based IDS) MFPT-K means algorithm uses the simple tree data structure which contains details about items that associates with one other. Modified FP-tree algorithm preserves the more space because it stores information in compressed format.

Each transaction in the database associates with single path of FP-tree in equal length of frequent itemsets. Therefore, size of modified FP-tree is bounded by the database's size. In the transaction, lot of frequent itemset are shared in the database then the tree size becomes smaller when is compared with database.

The Fig. 2 describes that network intrusion detection system which detect the data received in the network which has intrusion or not. It will work out in all layers of Transmission Control Protocol/ Internet Protocol (TCP/IP) protocol [25]. If intrusion is found in the received data came through network, then it is verified with the content of packets in the network. If no intrusion is found then the particular data is stored in dataset for future process. Generating rules for detecting intrusion in the NIDS is described below.

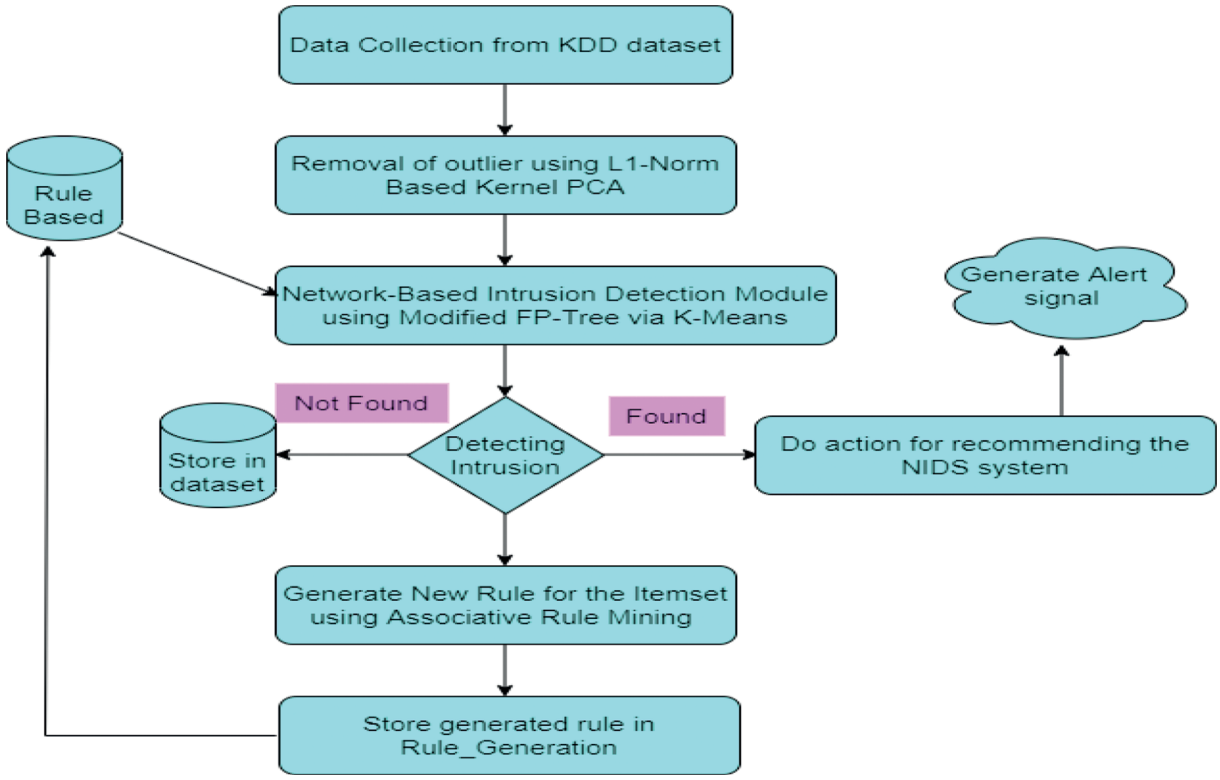

Figure 2: Architecture of MFPT-K-means

Algorithm 2: Generating of rules using associative rule mining

Input: Frequent item sets

Output: Intruders signatures

Step 1: Generate all possible frequent item sets and store it in database *FITEMDB*.

Step2: Evaluate Support (*SUPP*) using Eq. (2) and Confidence (*CONFI*)using Eq. (3).

Step 3: Count *SUPP* and *CONFI* value for each item set of *FITEMDB*.

Step 4: *IF* $(SUPP \geq Min\_Supp \&\& CONFI \geq Min\_CONFI)$ then

Step 5: Select that particular Itemset from database *FITEMDB* and stores in Rule_generation.

Step 6: Generate the rules for all frequent itemsets in the database and store in Rule_generation.

Step 7: ELSE discard the exact frequent itemset …

Step 8: Go To Next

Step 9: Return generated rules of rule_generation.

Step 10: End.

Algorithm 2 describes the collection of frequent item sets form the database and proposed work of NIDS using association mining rule to generate intruder signatures. The parameters are defined in Tab. 1.

Table 1: Parameters defined

Y	Frequent itemset
DB	Conditional database
fre_item	Frequent item set in the DB
intr_list	Interest of itesmset in DB
frec_item	Frequent closed itemset
FPT	Frequent pattern tree

Algorithm 3: Frequent Pattern Tree Algorithm (FPT)

Input: Frequent Itemset data from database *FITEMDB*

Output: Generating pattern-frequent Item sets of intruders

Step 1: Initialize frequent closed item set (*frec_item*) as ϕ.

Step 2: Scan the database *FITEMDB*and find frequent Itemset *fre_item* and obtain the interest item set*inter_item* from the *FITEMDB* based on the value of ϕ.

Step 3: Recursively execute the associate mining of frequent closed Item set.

Step 4: Call $Gen\_\phi(\phi,\ FITEMDB,\ \ fre\_item,\ \ intr\_list),\ fre\_item,\ frec\_item)$

Step 5: Subroutine $Gen\_\phi(Y,\ FITEMDB,\ \ fre\_item,\ \ intr\_list),\ frec\_item)$

Step 6: Let Z as ϕ-frequent Itemset in *intr_list*.

Step 7: In every transaction of database *FITEMDB* and $Y \cup Z$ is not proper subset of frequent itemset in *frec_item* is consider in the same support then include it in *frec_item*.

Step 8: Construct FP-Tree based on the values in database *intr_list*.

Step 9: if *FPT* has only one path then extract *frec_item*.

Step 10: For the remaining item sets in the *intr_list* construct the conditional database DB and evaluate frequent itemset in the DB.

Step 11: Randomly choose Item set in *intr_list*. If it is not a proper subset of same support recursively call Step 4 with respect to *frec_item*.

In the intrusion detection system the Algorithm 3, detects all intruders enter into the network and it is better than APRIORI algorithm [26,27]. It is a robust in nature because it detects the anomalous intruders of same item set in a various network. The clustering using K-means algorithm is described below:

Algorithm 4: K-Means Clustering Algorithm

Step 1: Initially, arbitrarily select the k points as clusters.

Step 2: In the database, item set is assigned to the close cluster by calculating Euclidean distance between each item set and cluster based Centroid value.

Step 3: For each cluster centroid value recalculate the average positions of the centroids.

Step 4: Steps 2 & 3 is repeated until it convergence.

Step 5: Applying Algorithm 2 for the associative rule mining

Step 6: End.

The k-means algorithm shown in Algorithm 4, is an evolutionary method which clusters the data into k groups of observations, where input parameter is set as K. The network-based intrusion detection system based on ϕ associate rule mining by using modified frequent Pattern Tree via K-means algorithm (MFPT-K means) is described below:

Describes the network-based IDS using Associative mining rule system for detecting the intruders. It is based on tree algorithm and here it contains Modified Frequent Pattern Tree via K-Means Clustering. It works based on FP association rule mining for detecting the intrusion detection using Algorithm 3. After applying the Algorithm 1 as pre-processing and it removes the outlier. Then the data sets are further optimized and it minimized by 80% of its average size of its original. It also reduces the memory space complexity and time complexity.

4 Result and Discussion

This section describes about the experimental results of the novel framework of network-based intrusion detection system using associative rule mining with modified frequent pattern tree via K-Means clustering algorithm.

4.1 Dataset

In this work, we are using two data sets like Knowledge Discovery in Databases (KDD) CUP 99 and NSL-KDD datasets [28] for implementing network intrusion detection system using Python Scikit learn. In the network traffic, the training details collected and saved as TCP format. Then this data undergoes pre-processing phase and transformed into connection record in a text file format. Each connection record contains sequence of Transmission control protocol (TCP) packets with IP address source to target's IP address. Each packet contains 150 bytes of information and labelled as normal or intruder data. In this work, various types of network intrusion classification tests are applied. Tab. 2 shows that KDD CUP 99 dataset with different data attack details.

Table 2: KDD CUP99 data details

Data type	Attack types	Training data set (instances)	Testing data set (instances)
Normal		85578	60475
Attack	Denial of service (DoS)	381556	224299
	Root to local (R2L)	1278	6993
	User to root (U2R)	78	45
	Probing attacks (Probe)	5129	3377

NSL-KDD data set contains 41 features which represents the network connection [28]. The data are in type of normal and attacked.

4.2 Receiver Operating Characteristics (ROC) Curve

ROC is plotted as true positive rate (TPR) on the y axis to false positive rate (FPR) on the x axis across various threshold values. Area under the ROC Curve (AUC) is the size of the area under the ROC curve used along with ROC.

$$AUC = \int_0^1 \frac{TP}{TP + FN} d \frac{FP}{TN + FP} \tag{2}$$

Fig. 3 shows that detection rate of KDD-CUP99 dataset. This detection rate is represented by using the various attack types of NIDS. The attack types are Normal, DoS, U2R, R2L and Probe.

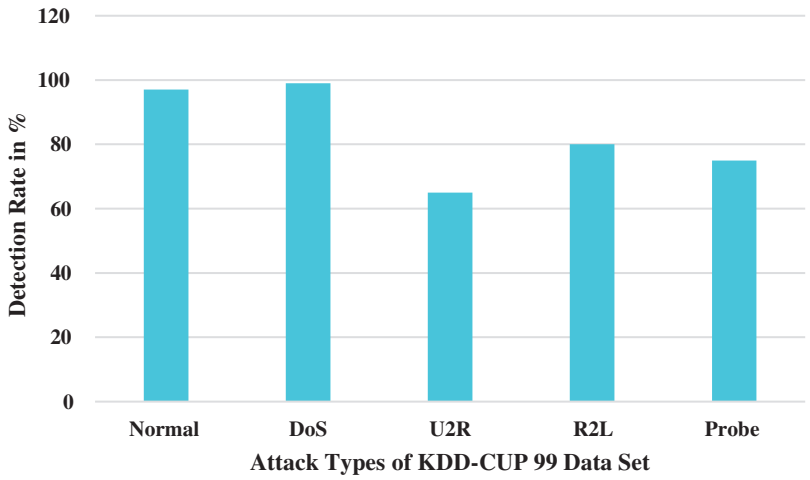

Figure 3: Detection rate of KDD-CUP 99 data set

From the Fig. 3 its shown that amount of data in the training data set is used to detect the type of attack. DoS is a unknown threat which appears higher in the data set KDD-CUP 99. Fig. 4 shows that the detection rate of unknown threat in the NSL-KDD data set.

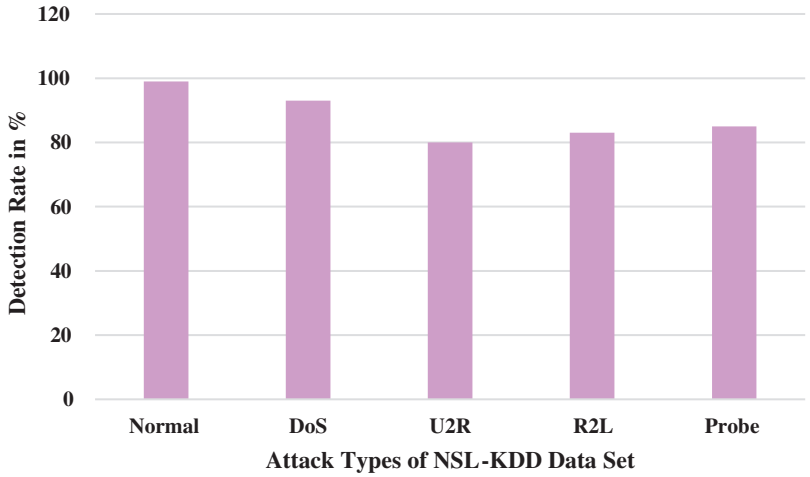

Figure 4: Detection rate of NSL-KDD data set

From the Fig. 4 its shown that amount of data in the training data set is used to detect the type of attack. DoS is a unknown threat which appears higher for the data set NSL-KDD.

4.3 Evaluation of FP-Tree Algorithm

In the FP-Tree algorithm, three parametric values are used they are support, confidence threshold value and time interval. If the value of time interval increased the time complexity of association rule mining is also increased. Because the possibility of alerts getting grouped together into single transaction also increases. Similarly, if the value of time interval decreases to a certain limit and its smaller value does not affect the creation of association rule. If the Support Threshold value is greater, then generation of association rule is decreased. Finally generating relevant association rule is treated as the final output. The alerts created will be in less number if the frequent associative rules are not successfully mined. Similarly when the confidence threshold value increases, the number of association rules generated also decreases. Tab. 3 shows that the parametric values which are used in this work.

Table 3: Parametric values

Parametric name	Value
Time interval	60 s
Confidence threshold value	0.8
Support threshold value	10

From the observation of Tab. 3 it is describes that, the proposed work's time interval is equal to one minute, confidence threshold value is 0.8, support threshold value is 10 and for this configuration our proposed work MFPT-K Means algorithm finds 55 association rules. In the analysis of each associative rule in the mining, we came to know that due to noise it contains irrelevant rules. Tab. 4 show that performance metric measures for various attack types in the data set of KDD-CUP 99.

Table 4: Performance metric measures of KDD-CUP 99 data set

K-means				
Attack types	FAR	Precision	Recall	F1-score
Normal	0.89	97.02	96.53	97.55
DoS	40.45	68.61	83.75	72.91
U2R	0.38	97.65	96.23	94.12
R2L	11.78	84.18	90.45	87.78
Probe	1.54	97.15	97.44	97.44
Apriori algorithm				
Attack types	FAR	Precision	Recall	F1-score
Normal	0.82	95.02	92.42	93.35
DoS	44.22	65.41	81.56	71.89
U2R	0.45	92.45	92.42	92.68
R2L	11.32	83.89	89.35	86.45
Probe	1.55	96.55	96.23	95.11

(Continued)

Table 4 (continued).

MFPT-K means				
Attack types	FAR	Precision	Recall	F1-score
Normal	0.79	99.02	98.12	98.35
DoS	39.22	70.61	84.56	73.89
U2R	0.35	99.45	99.42	99.68
R2L	10.74	85.89	91.45	88.45
Probe	1.45	98.55	98.23	98.45

Tab. 5 show that performance metric measures for various attack types in the data set of KDD-CUP 99. Our proposed work gives low false positive rate. From the analysis of Tab. 5 its shown that false alert rate of proposed work MFPT-K Means algorithm has low positive rate for Normal it is (0.79), for DoS attack it is 39.22, U2R got 0.35, R2L got 10.74 and probe attack got 1.45. Tab. 5 show that performance metric measures for various attack types in the data set of NSL KDD.

Table 5: Performance metric measures of NSL KDD data set

K-means				
Attack types	FAR	Precision	Recall	F1-score
Normal	0.79	96.45	94.53	96.77
DoS	39.54	67.16	84.57	71.91
U2R	0.48	96.56	96.23	92.11
R2L	13.23	86.78	89.15	88.87
Probe	1.63	95.15	95.44	96.89
Apriori algorithm				
Attack types	FAR	Precision	Recall	F1-score
Normal	0.88	92.21	90.11	92.78
DoS	40.56	63.51	79.23	76.89
U2R	0.55	91.45	89.67	91.88
R2L	14.32	88.89	86.52	85.45
Probe	1.76	96.55	96.23	95.11
MFPT-K means				
Attack types	FAR	Precision	Recall	F1-score
Normal	0.98	99.77	98.61	98.75
DoS	38.22	72.14	86.12	78.89
U2R	0.15	99.88	99.23	99.56
R2L	14.45	87.89	93.45	89.45
Probe	1.23	98.55	98.23	98.45

Tab. 5 show that performance metric measures for various attack types in the data set of NSL KDD. Our proposed work gives low false positive rate in NSL KDD dataset as well. Fig. 5 shows that memory consumption while using support in the associative rule mining.

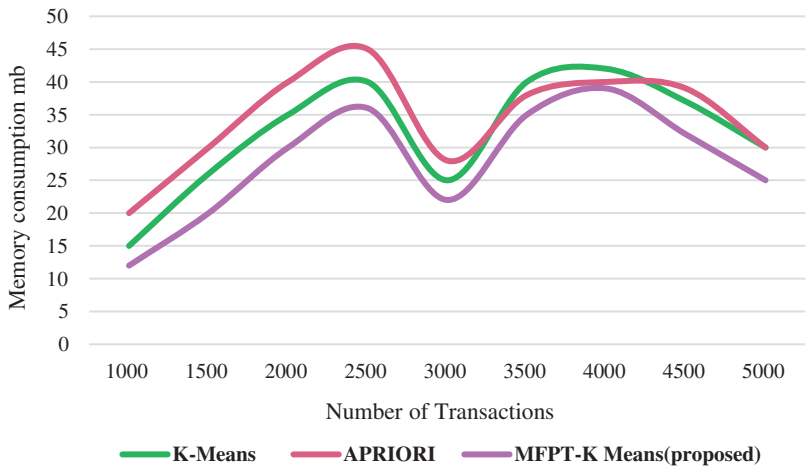

Figure 5: Memory consumption based on support

Fig. 5 describes that memory required by K-Means, APRIORI and proposed work (MFPT-K means). In the analysis of Fig. 5. We have varied the data set transactions from 1000 to 5000 with interval of 500 transactions and computed the memory consumption based on support threshold for frequent item set. Our proposed work needs less storage requirement. Fig. 6 shows that computation time with support in the associative rule mining.

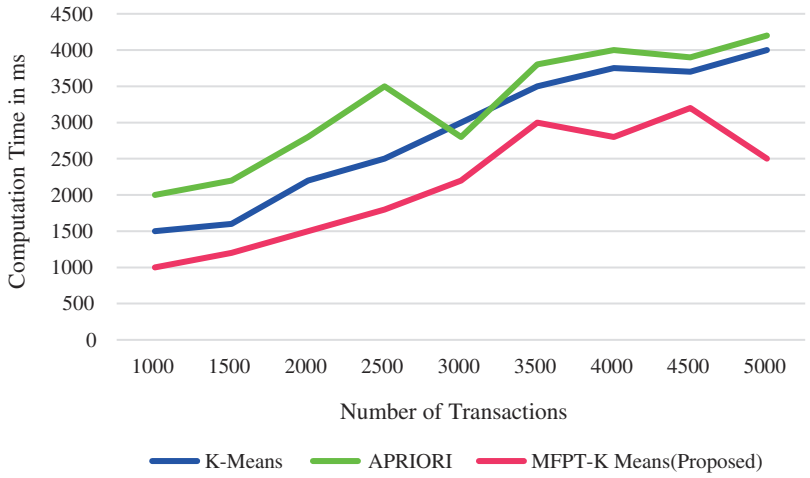

Figure 6: Computation time on support

Fig. 6 describes the time required by K-Means, APRIORI and proposed work (MFPT-K means). From the analysis of Fig. 6 it is shown that we have varied the transactions from 1000 to 5000 with interval of 500 transactions and computed the time consumption based on using support threshold for finding frequent itemset. Our proposed work needs less computation time. Fig. 6 shows that computation time on support in the associative rule mining. After applying the Algorithm 1 in the two datasets are KDD-CUP

99 and NSL KDD. Then the optimized data in the data set transactions are 1000, 1500, 2000, 2500, 3000, 3500, 4000, 4500, 5000 records and the time spent for the execution of transactions before applying L1-Norm Kernel based PCA and after applying L1-Norm Kernel based PCA is given in the Figs. 7 and 8.

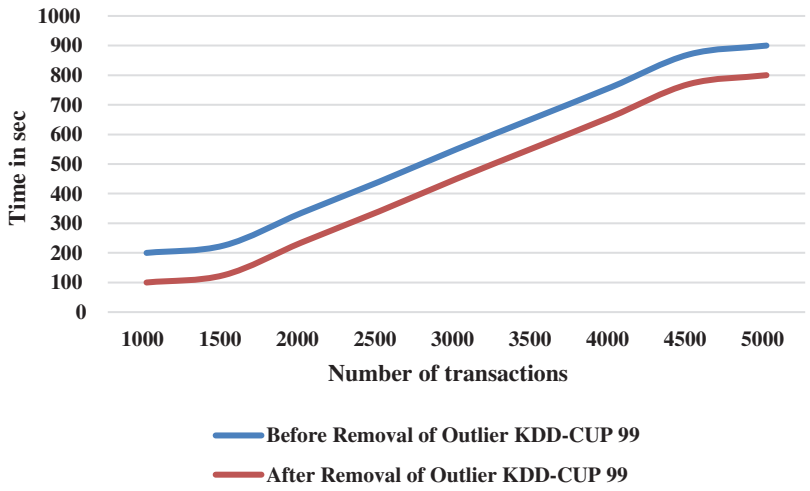

Figure 7: L1-norm kernel PCA algorithm comparison in KDD CUP 99

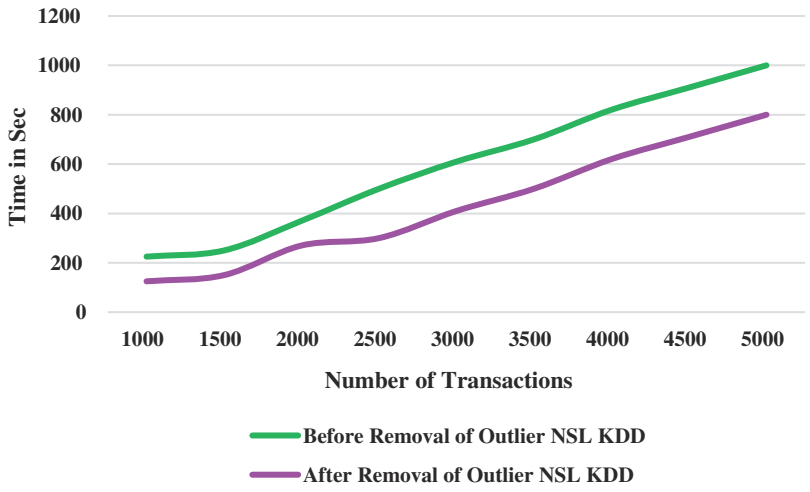

Figure 8: L1-norm kernel PCA algorithm comparison in NSL KDD

Fig. 7 shows that before applying L1-Norm Kernel PCA algorithm, the analysis of transactions of data in dataset KDD CUP 99 takes more time to process records. After applying L1-Norm Kernel PCA algorithm, the analysis of data in dataset KDD CUP 99 takes less time to process records.

Fig. 8 shows that before applying L1-Norm Kernel PCA algorithm, the analysis of transactions of data in dataset NSL KDD takes more time to process records. After applying L1-Norm Kernel PCA algorithm, analysis in dataset NSL KDD takes less time. Fig. 9 shows that accuracy in finding various attack types in the data set of KDD-CUP 99.

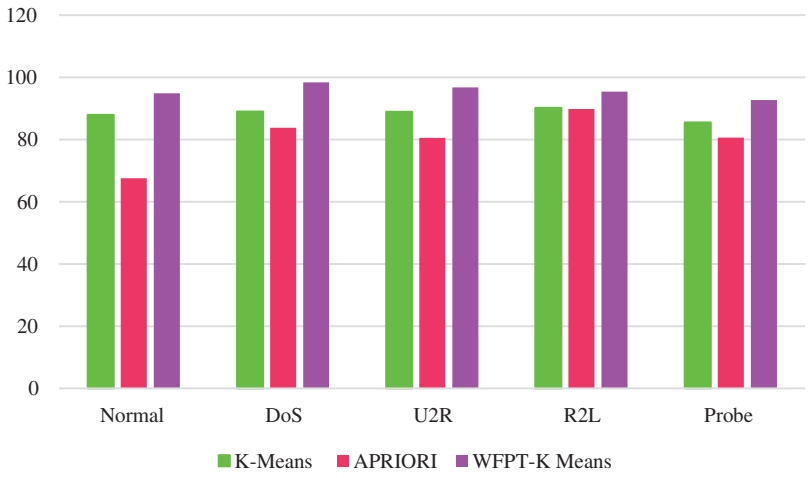

Figure 9: Accuracy rate

Fig. 9 shows that our proposed work (WFPT-K Means) is good in accuracy and it ranges from attack to attack like for Normal it is 94.89%, for DoS attack it is 98.34%, for U2R attack got 96.73%, for R2L got 95.89% and Probe attack got 92.67%.

When it is compared with other existing algorithms K-Means &APRIORI it is very optimal. Fig. 10 shows the receiver operating characteristics (ROC) curve for various attack types of K-means, APRIORI and WFPT-K Means for the datasets of KDD-CUP 99 and NSL-KDD algorithms. From ROC curves in Fig. 10, it is shown that for the training datasets of KDD-CUP 99 and NSL-KDD, our proposed method provide optimum results.

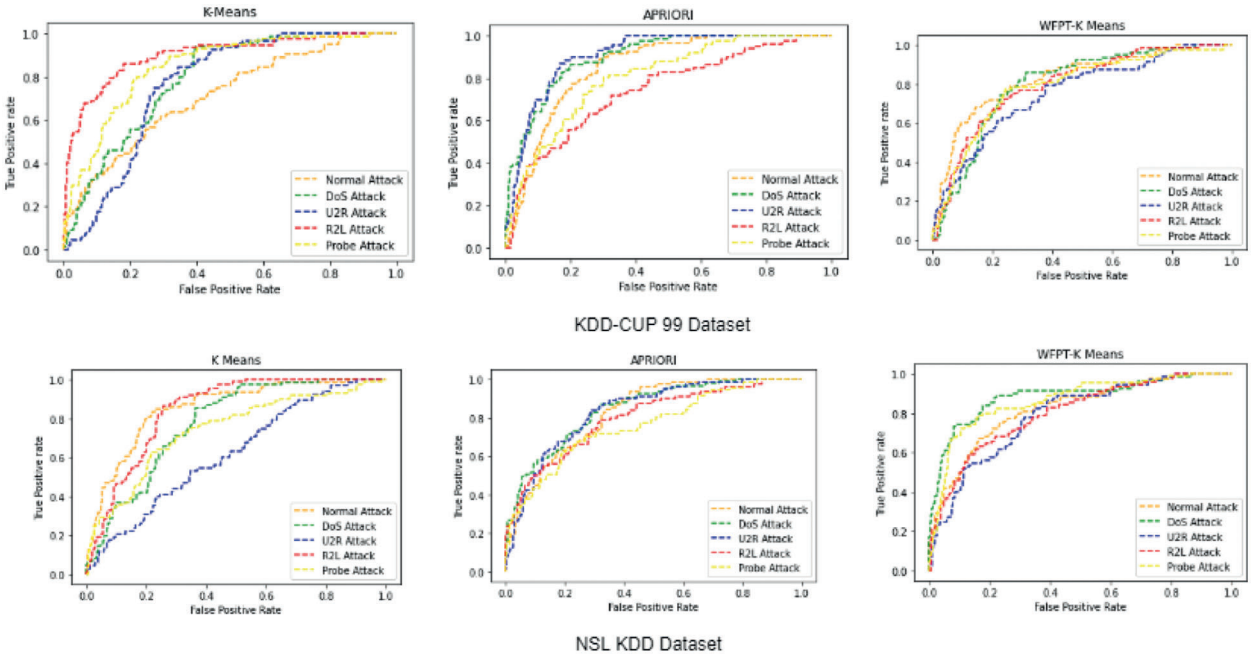

Figure 10: ROC for various classifier algorithms

5 Conclusion

A modified frequent pattern tree algorithm via K-Means algorithm has been proposed in this research work for the implementation of intrusion detection system for providing network security. This novel framework employed L1-Norm kernel PCA for the removal of outlier in the records of the datasets of KDD-CUP 99 and NSL-KDD. After applying L1-Norm kernel PCA, the data are optimized and evaluated for performance in the terms of complexity in time and space. In addition to that it is compared with K-Means, APRIORI associative algorithms. The experimented result of our proposed work shows it outperforms in terms of accuracy for Normal it is 94.89%, for DoS attack it is 98.34%, for U2R attack it got 96.73%, for R2L it got 95.89% and Probe attack it got 92.67% . In future, this research shall be evaluated with complex DNN architecture and also shall aim to improve the detection rate of the malware attack in various datasets. Since, the proposed detection system will minimize the generalization error, noise, time complexity and keeping classification accuracy in a higher rate.

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

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. M. Kamini 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, 2014.

[2] R. Kumari and J. Vashishtha, "Discovery of fuzzy hierarchical association rules," *International Journal of Computer Applications*, vol. 98, no. 19, pp. 20–26, 2014.

[3] K. Julisch, "Clustering intrusion detection alarms to support root cause analysis," *ACM Transactions on Information and System Security*, vol. 6, no. 4, pp. 443–471, 2003.

[4] J. Hu and X. Yang Li, "A fast parallel association rules mining algorithm based on FP-forest," in *Proc. 5th Int. Symp. on Neural Networks*, Cairo, Egypt, pp. 40–49, 2008.

[5] A. Javed and A. Khokhar, "Frequent pattern mining on message LFP-tree passing multiprocessor systems," *Distributed and Parallel Databases*, vol. 16, pp. 321–334, 2004.

[6] O. R. Zaane, M. El Hajj and P. Lu, "Fast parallel association rule mining without candidacy generation," in *Proc. Int. Conf. on Data Mining (ICDM)*, Maebashi, Japan, pp. 665–668, 2001.

[7] K. M. Yu, J. Zhou and W. C. Hsiao, "Load balancing approach parallel algorithm for frequent pattern mining," in *Proc. Int. Conf. on Parallel Architectures and Compilation Techniques (PaCT)*, Kaliningrad, Russia, pp. 623–631, 2007.

[8] M. Schultz, E. Eskin, E. Zadok and E. Stolfo, "Data mining methods for detection of new malicious executables," in *Proc. of IEEE Symp. on Security and Privacy*, CA, USA, pp. 38–49, 2001.

[9] S. Stolfo, W. Lee, P. Chan, W. Fan and E. Eskin, "Data mining-based intrusion detectors: An overview of the Columbia IDS project," *SIGMOD Record*, vol. 30, no. 4, pp. 5–14, 2001.

[10] E. Aminanto and K. Kim, "Deep learning in intrusion detection system: An overview," in *Proc. Int. Research Conf. on Engineering and Technology*, Kuta, Indonesia, pp. 1–12, 2016.

[11] R. Vani, "Towards efficient intrusion detection using deep learning techniques: A review," *International Journal of Advanced Research in Computer Science and Electronics Engineering*, vol. 6, no. 10, pp. 375–384, 2017.

[12] H. Wang, G. Zhang, H. Chen and X. Jiang, "Mining association rules for intrusion detection," in *Fourth Int. Conf. on Frontier of Computer Science and Technology*, Shanghai, China, pp. 978-0-7695, 2009.

[13] T. mielinski, A. Swami and R. Agarwal, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD Conf. on Management of Data*, Shanghai, China, pp. 207–216, 1993.

[14] J. Han, J. Pei, Y. Yin and R. Mao. "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *International Journal of Data Mining and Knowledge Discovery*, vol. 8, no. 1, pp. 53–87, 2004.

[15] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. on Computational Intelligence in Security and Defense Applications*, Honolulu, USA, pp. 1–6, 2009.

[16] N. Chand, P. Mishra, C. R. Krishna, E. S. Pilli and M. C. Govil, "A comparative analysis of SVM and its stacking with other classification algorithm for intrusion detection," in *Proc. Int. Conf. on Advances in Computing, Communication Control and Networking (ICACCA)*, United States, pp. 1–6, 2016.

[17] P. Tao, Z. Sun and Z. Sun, "An improved intrusion detection algorithm based on GA and SVM," *IEEE Access*, vol. 6, pp. 13624–13631, 2018.

[18] M. N. Kurt, Y. Yilmaz and X. Wang, "Real-time detection of hybrid and stealthy cyber-attacks in smart grid," *IEEE Transaction on Information Forensics Security*, vol. 14, no. 2, pp. 498–513, 2019.

[19] F. Padillo, J. Luna and S. Ventura, "Evaluating associative classification algorithms for big data," *Big Data Analytics*, vol. 4, no. 2, pp. 1–27, 2019.

[20] S. Elhag, A. Fernandez, A. Altalhi, S. Alshomrani and F. Herrera, "A multi-objective evolutionary fuzzy system to obtain a broad and accurate set of solutions in intrusion detection systems," *Soft Computing*, vol. 23, no. 4, pp. 1321–1336, 2019.

[21] Y. J. Wang, Q. Xin and F. Coenen, "Hybrid rule ordering in classification association rule mining," *Transactions on Machine Learning and Data Mining*, vol. 1, no. 1, pp. 1–15, 2018.

[22] W. C. Chen and C. Hsu, "An associative classification approach for enhancing prediction of imbalance data," in *Proc. Fifth Int. Conf. on Informatics and Applications*, Takamatsu, Japan, pp. 105–111, 2016.

[23] S. Flora, "Network intrusion detection using association rules," *International Journal of Recent Trends in Engineering*, vol. 2, no. 2, pp. 202–204, 2009.

[24] R. Agarwal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. 20th VLDB Conf.*, San Francisco, United States, pp. 487–499, 1994.

[25] V. Patil, R. Vasappanavara and T. Ghorpade, "Security in association rule mining using secure sum technique with FP growth algorithm in horizontally partitioned database," in *Proc. Int. Conf. on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, Chennai, India, pp. 2838–2843, 2017.

[26] W. Lee, S. Stolfo and K. Mok, "A data mining framework for building intrusion detection models," in *Proc. IEEE Symp. on Security and Privacy*, California, USA, pp. 120–132, 1999.

[27] D. Newman, "KDD cup 1999 data," in *The UCI KDD Archive*, Irvine: Information and Computer Science, University Of California. [Online]. Available: http://kdd.ics.uci.edu//databases/kddcup99/kddcup99.html*, 1999*

[28] S. Sivanantham, V. Mohanraj and Y. Suresh, "Rule precision index classifier: An associative classifier with a novel pruning measure for intrusion detection," *Personal Ubiquitous Computing*, Vol. Online first article, 2021.