Journal of Engineering and Applied Sciences 14 (9): 2964-2974, 2019
ISSN: 1816-949X

A Density Maximization-Fuzzy Means Clustering Algorithm for Network Intrusion Detection

Ruby and Sandeep Chaurasia
Department of Computer Science Engineering, Manipal University Jaipur, Jaipur, India

Abstract: Detecting intrusions from the network traffic dataset is one of the demanding and critical task in recent days. This study aims to develop a Density Maximization-Fuzzy Means Clustering (DM-FMC) algorithm for identifying the intrusions from the network traffic datasets. In this process, the raw datasets are preprocessed at the initial stage for removing the irrelevant attributes and to normalize the data for further use. Based on the values of threshold, density and fuzziness index, the cluster is formed by using the DM-FMC technique. In the end, the cluster is categorized to efficiently identify the anomalies from the dataset.

Key words: DM-FMC, network, dataset, cluster, fuzziness index, Jaipur

INTRODUCTION

Data mining is the most widely used research field in current days, in which intrusion detection is the one of the critical application area. Normally, an intrusion (Naik and Prashsntha, 2014) is defined as a kind of unlawful activity that is carried out by the intruders for affecting the performance of the network. Intrusion detection is one of the efficient methodology for detecting the malicious activities against the attacks in network (Kumar *et al*., 2016). Typically, the data mining techniques that used for intrusion detection is split into the types of anomaly based detection and misuse based detection. In which each occurrence in a dataset is labeled as normal or intrusion during the detection of misuse activities (Kim *et al*., 2014). Then, the incoming intrusions are determined in the anomaly detection (Bhuyan *et al*., 2014) by the use of rule based methods which describes the category of attack. The Intrusion Detection System (IDS) as shown in Fig 1 is developed to detect the network intrusions by analyzing the features of the data. It monitors the system information and generates the alarm when it detects the malicious activities. The IDS is categorized into two types (Kenkre *et al*., 2014) such as:

- Network based Intrusion Detection System (NIDS)
- Host based Intrusion Detection System (HIDS)

Both IDS has the capability to detect all kinds of malicious network traffic which includes unauthorized access, permission escalation, access to sensitive files and attacks against sensitive services (Nadiammai and Hemalatha, 2014). Most of the data mining techniques are used for detecting the intrusions by analyzing the given datasets.

Problem identification: Traditionally, the IDS follows the data mining procedure for detecting and identifying the

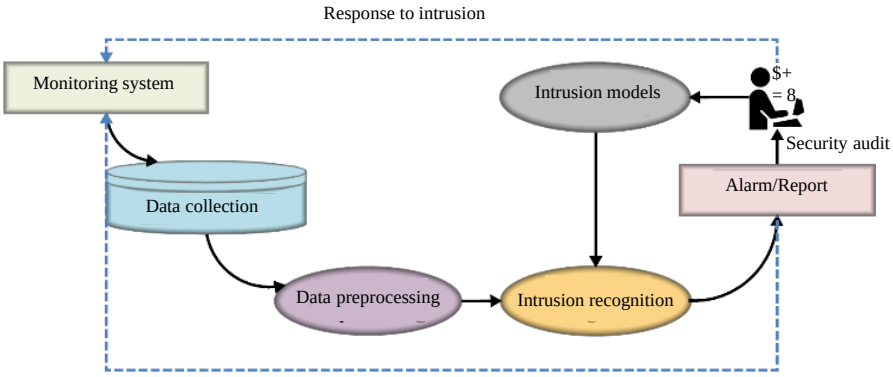

Fig. 1: Architecture of IDS

Corresponding Author: Ruby, Department of Computer Science Engineering, Manipal University Jaipur, Jaipur, India

intruders in networks. The anomaly based IDS investigates the ongoing traffic, activity, transactions and behavior order for identifying the anomalies. Normally, the behavior of intrusion is vary from the behavior of normal data, so, it can be easily identified based on its differences. Moreover, the anomaly based IDS creates increased processing overhead on the network (Ashfaq *et al*., 2017; Subaira and Anitha, 2014). The drawbacks (Hubballi and Suryanarayanan, 2014) of the traditional works are resource usage problem and reliability problem. The components of the IDS are running at all-time even there is no intrusion (Ravale *et al*., 2015). Then, the IDS (Zhao, 2016) required to infer the behavior of the system from the collected data which results some misinterpretations or missed events. Also, the intruder could modify the programs that running on a system which leads to the IDS as unreliable or useless.

Objectives: The major objectives of this study are as follows:

- To test the performance of intrusion detection, the widely used network traffic datasets are considered in the proposed IDS
- To normalize the original data by eliminating the irrelevant attributes, the data preprocessing is performed
- To classify the network data as normal or anomalous, a new Density Maximization-Fuzzy Means Clustering (DM-FMC) algorithm is proposed
- To compute the degree of membership for each data, a membership matrix is generated

Literature review: In this study, the existing techniques and approaches related to intrusion detection are surveyed with its advantages and disadvantages.

Li *et al*. (2016) implemented a Multi-View based false Positive reduction (MVPs) approach for automatically detecting the network intrusions. It aimed to exploit the unlabeled and labeled data without any human intervention. Here, the machine learning techniques such as k-Nearest Neighbor (k-NN), Neural Networks (NN) and Support Vector Machine (SVM) were investigated to perform an efficient intrusion detection. The key objectives that focused in this work were to extract the proper features for constructing a multi-view dataset. Moreover, the labeled and unlabeled data were automatically leveraged by using the semi-supervised learning algorithm. Yet, this study required to prove the efficacy of MVPs by comparing it with the active learning approaches. Saurabh and Verma (2016) developed a proactive Artificial Immune System (AIS) for detecting the unseen anomalies in the system. This research contains the three modules such as Repertoire Training Module (RTM), Vulnerability Assessment Module (VAM) and Response Module (RM). Here, the principles of immunity was integrated with the agents for detecting attacks as well as providing suitable measures to stop them. The benefit of this research was it reduced the required number of computations with increased detector selection rate and reduced detector rejection rate. However, the overall performance of this work was not highly efficient which was the major limitation of the suggested system. Jabez and Muthukumar (2015) employed an outlier detection approach for identifying the possible incidents and logging information to develop an Intrusion Detection and Prevention System (IDPS). Also, the Neighborhood Outlier Factor (NOF) was utilized to detect the anomalies based on the features that extracted from the internet packets. In this system, the false alarm rate was generated by checking the outlier value with the specified threshold. The merit of this study, it formed a cluster by the use of density connected objects. Still, this research required to compute the distance function between the training and testing models which leads to reduced efficiency.

Duque and bin Omar (2015) suggested a machine learning algorithm for developing an IDS with increased efficiency and lower false rate. The aim of this study was to detect the data as whether attack or normal by using the k-means clustering technique. Here, the data preprocessing was performed to eliminate the data inconsistency and to convert the attributes into a numerical data. Xie *et al*. (2016) implemented a Provenance Aware Intrusion Detection and Analysis (PIDAS) for detecting the online intrusions and analyzing the offline forensics in the following steps: collecting provenance, detecting intrusions and analyzing system vulnerabilities. Here, the provenance was utilized to record a series of activities for detecting the intrusions. Also, two types of search such as backward search and forward search were performed to analyze the intrusions. Though, the suggested technique has an increased computational complexity and reduced accuracy. Ashfaq *et al*. (2017) used a Single Hidden-Layer Feed Forward Neural Network (SLFN) for detecting the intrusions with increased efficiency. Here, the large amount of unlabeled samples were considered for building a better classification technique. Also, the generative models, graph based models and Transductive SVM (TSVM) were investigated in this paper for analyzing its

features. Moreover, the fuzziness of the cognitive uncertainty and transition of uncertainty were considered for intrusion detection. Still, this research failed to increase the efficiency of detection and reduce the time consumption.

Lin *et al*. (2015) introduced a Cluster Center and Nearest Neighbor (CANN) approach for representing the features during network intrusion detection. Here, two distances were measured and added at first the distance between each data sample and cluster center was estimated and the distance between the data and its nearest neighbor was computed. The drawback that observed from this research was, it has a reduced computational efficiency. Kaur and Singh (2016) investigated various anomaly detection techniques for providing security to the Online Social Networks (OSN). The models that surveyed in this study were supervised methods, unsupervised methods and semi-supervised methods. The researchers of this study stated that the classification methods were highly dependent on the training data, so, it does not offer the better performance during anomaly detection. Ha *et al*. (2016) suggested a traffic sampling strategy for analyzing the malicious traffic on the Software Defined Networks (SDN). In this system, certain number of packets were inspected to analyze the signatures from the known threats. The motive of this paper was to reduce the capture failure rate by analyzing the packets at switches based on an optimization process. Also, the throughput of each switch was estimated to distribute the current malicious traffic and flow path information.

Posenato *et al*. (2008) suggested two statistical models such as moving principle component analysis and moving correlation analysis for monitoring the structures in an uncertain environments. Here, the variability of the time period was estimated and its threshold was defined for detecting the anomalous behavior. Moreover, the Principal Component Analysis (PCA) and correlation analysis algorithm were utilized to detect the damages. The drawbacks of this study were, it has an increased computational complexity and increased time consumption. Ni *et al*. (2016) recommended some data mining techniques for detecting the practical network anomalies. The techniques that surveyed in this study related to clustering, classification, association rules and feature selection. Here, various data mining models were integrated to attain a better detection result with increased accuracy. Elhag *et al*. (2015) utilized the combination of genetic fuzzy systems and pairwise learning models for detecting intrusions in networks. Here, the divide and conquer learning model was used to obtain the better separability between the normal events and attack events. Also, the large set of candidate association rules were generated for reducing the search space. The benefit of this approach was, it maintained a low false alarm rate with increased detection accuracy.

Leu *et al*. (2017) recommended different data mining and forensic techniques for developing an Internal Intrusion Detection and Prevention System (IIDPS). The motive of this system was to detect the internal attacks at System Calls (SCs) by the use of data mining and forensic approaches. Here, the accuracy of attack detection was improved by identifying the user's forensic features. Yet, this research required to use the third party shell commands for improving the performance of IIDPS. Buczak and Guven (2016) utilized a Machine Learning (ML) and Data Mining (DM) models for cyber security intrusion detection. In this research, the class attributes and classes were identified from the training data and the trained model was utilized to classify the unknown data. In addition, three classes of rules were defined in this work that includes traffic flow rules, services rules and service usage rules. These rules were used to determine the patterns for further use. Wu and Shan (2015) utilized a web data mining technology for network information security. Here, the processes such as security audit, code detection, spam detection and virus notice were performed to increase the security level. Moreover, the data mining prevention model was employed to analyze the normal network traffic patterns. Patel and Panchal (2015) suggested a Classification and Regression Tree (CART) Model for efficiently detecting intrusions in networks. The alert was generated for the anomaly data for classifying the normal and abnormal activities. This research intended to improve the accuracy rate and to reduce the false alarm rate during intrusion detection. Also, the SNORT was used to detect the attacks based on the set of predefined rules. It contains the components of decode engine, preprocessor plug, detection engine and output plug-ins. Also, the merits and demerits of various approaches were investigated in this study. Dhakar and Tiwari (2014) developed a hybrid model for detecting network intrusions with the use of KDD Cup dataset. The network data from various networks were analyzed to detect the distributed attacks. Also, the signature based techniques were implemented to identify the unknown and novel attacks. However, this research required to prevent the network from cyber-attacks and to identify the new system vulnerabilities. Dhanabal and Shantharajah (2015) investigated various classification models that includes decision tree, Neural Networks (NN), Naive Bayes (NB), Support Vector Machine (SVM) and Apriori algorithm for intrusion detection. In this research, the NSL-KDD dataset was utilized to evaluate the results of

these techniques. Moreover, the testing accuracy of the techniques were analyzed based on the classes of DoS, probe, U2R and R2L. The researchers of this study stated that, the optimization techniques must be implemented for attaining a better accuracy rate.

From the survey, it is analyzed that the existing techniques have both advantages and disadvantages but it mainly lacks with the following limitations:

- Increased false alarm rate
- It has the ability to detect only the trained attacks
- Highly expensive
- Due to the insufficient data, it detected the intrusive behavior as normal behavior

To solve these issues, this study aims to develop a clustering IDS for classifying the attacks based on its density.

MATERIALS AND METHODS

In this sector, the detailed description about the proposed methodology is presented with its clear flow representation. The motive of this study is to efficiently detect the intrusions from the network traffic dataset based on the clustering process. For this purpose, the KDD Cup and Australian Defense Force Academy (ADFA) datasets are considered which contains both normal and anomalous data. In order to identify the anomalous data from these datasets, the Density Maximization-Fuzzy Means Clustering (DM-FMC) algorithm is proposed in this research. At first, the missing and inadequate data are identified and eliminated in the preprocessing stage. Then, the membership function is estimated to compute the degree of membership for each data, based on this the membership matrix is generated. Consequently, the centroid values of this matrix is computed and the dissimilarity function is applied on that data. It is helpful to identify the distance between the centroid and each data point.

After that, the degree of membership at each data point for all clusters is computed by finding the distance between the data point and nearest data point. Then, the threshold, density and fuzziness index values are also calculated. The cluster is formed by calculating the distance between the data point centroid and density centroid. If the computed distance less than the data point, tit is added into the cluster; otherwise, the mean value is compared with the threshold density value. Finally, the cluster is categorized based on its sensitivity which efficiently identifies the anomaly. The graphical representation of the proposed IDS is shown in Fig. 2.

Preprocessing: Initially, the input KDD Cup and ADFA datasets are preprocessed by performing the data cleaning, relevancy analysis and data transformation processes. Typically, the KDD Cup dataset contains four different types of attacks such as Denial of Service (DoS), User to Root (U2R), Remote to Local (R2L) and probing. Moreover, it contains various attributes in the form of symbolic, continuous and discrete. Then, it is not possible to process those attributes in its current form, thus, the preprocessing is required at the first stage. The ADFA dataset contains both normal and anomalous data and is widely used for HIDS. The attributes of these datasets are illustrated as:

Features of KDD Cup dataset

Feature name:

- duration
- protocol_type
- service
- Flag
- Src_bytes
- Dst_bytes
- land
- Wrong_fragment
- urgen
- hot
- Num_failed_logins
- Logged.in
- Num_compromised
- Root_shell
- Su_attempted
- Num_root
- Num_file_creations
- Num_shells
- Num_access_files
- um_outbound_cmds
- s_hot_login
- s_guest_login
- ount
- rv_count
- error_rate
- Srv_serror_rate
- error_rate
- rv_rerror_rate
- ame_srv_rate
- iff_srv_rate
- rv_diff_host_rate
- st_host_count
- st_host_srv_count
- st_host_same_srv_rate
- Dst_host _diff_srv_rate
- Dst_host_same_src_port_rate
- Dst_host_srv_diff_host_rate
- Dst_host_serror_rate
- Dst_host_srv_serror_rate
- Dst_host_rerror_rate
- Dst_host_srv_rerror_rate

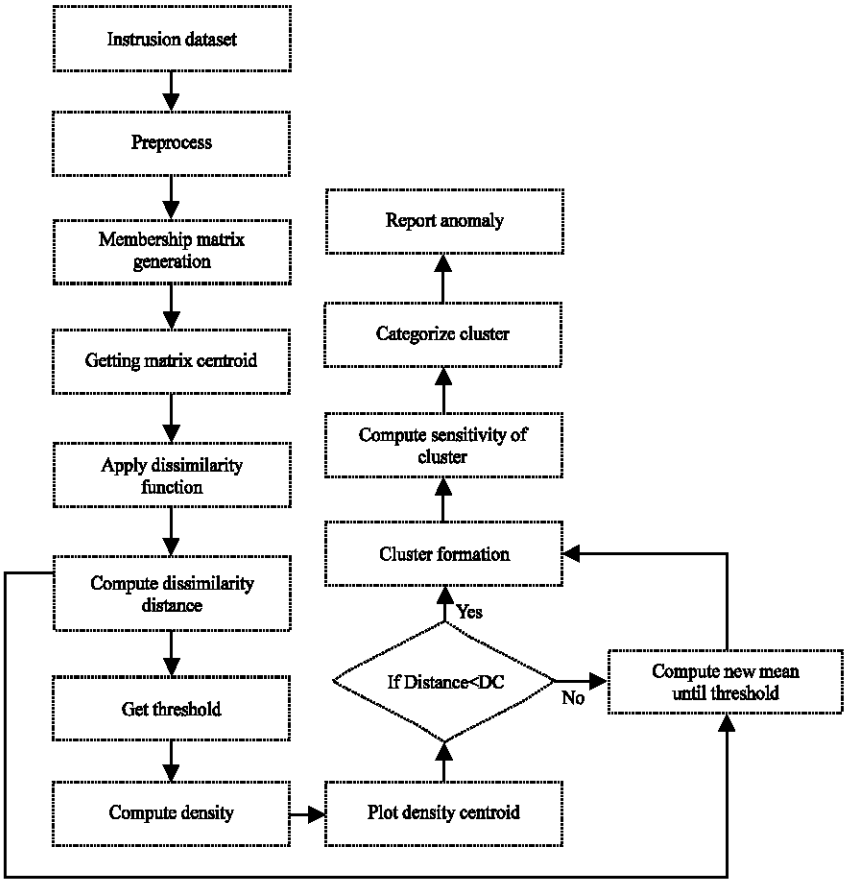

Fig. 2: Flow of the proposed clustering based IDS

Features of ADFA dataset

Features:

- Id
- Duration
- Protocol
- State
- Spkts
- Dpkts
- Sbytes
- Dbytes
- Rate
- Sttl
- Dttl
- Sload
- Dload
- loss
- loss
- inpkt
- Dinpkt
- Sjit
- Dijt
- Swin
- Stcpb
- Dtcpb
- Dwin
- Tcprtt
- Ackdat
- Dmean
- Trans_depth
- Response-body_len
- Synack
- Ct_srv_src
- Ct_state_ttl
- Ct_dst_ltm
- Ct_src_dport_ltm
- t_dst_sport_ltm
- s_ftp_login
- t_ftp_cmd
- t_flw_http_mthd
- t_src_ltm
- t_srv_dst
- s_sm_ips_ports
- ttack_cat Label

Here, the data cleaning is performed to eliminate the noise and to fill up the missing values. Then, the irrelevant and redundant attributes are removed during relevancy analysis and the resultant data is normalized for transformation. The data preprocessing is mainly performed to classify the data as normal or anomalous. In this algorithm, the given data is separated into labels and other features and then,the binary and multi-class labels are assigned to the records. Here, the split function is used to remove the ground truth from the raw dataset where the ground truth is defined as the last two columns of the dataset (i.e., attack category and label). So, it must be eliminated to process the dataset and to train the machine. During testing, the data is tested and the resultant data is compared with the ground truth. Consequently, the number of records used for the process training and testing are estimated. Finally, the separate preprocessed training and testing data are obtained (Algorithm 1).

Algorithm 1; Data preprocessing:

Input: ADFA and KDD Cup datasets
Output: Preprocessed datasets
Step 1: Let D_0 be the original dataset, D_p be the preprocessed dataset, A_n be the number of valid attributes after preprocessing
Step 2: The dataset is preprocessed by separating the data into labels and other features
Step 3: Extract the original data and separate the labels with ground truth
Step 4: Assign the binary and multi class labels to the records
$D_P \leftarrow$ Split (D_o, $D_0(D_L, D_{GT})$)
//Where, L is the label and GT is the ground truth
Step 5: Estimate the number of records utilized for training
n = b % (a) of D_P //Where, a-total number of records in the file
Estimate the number of records utilized for testing
m = a-(b% (a) of D_P)//Where, b-% of records to be training
Step 6: for i = 1 to n
$Tr_i \leftarrow D_{P(i)}$//Where, Tr_i-Training data
i = i+1
end for i
for j = n to m
$T_{sj} \leftarrow D_{P(j)}$ //Where, T_{sj}-Testing data
j = j + 1
end for j

Membership matrix generation: After preprocessing, the membership matrix is generated by loading the training data. For the number of available labels and attributes the minimum, maximum, mean and variance values are computed. Then, the distance between the data point and the cluster center is estimated for assigning the membership to each data point. In this technique, the objects that belong to various clusters are allowed with different degrees of membership where the partial membership is indicated by the value between 0 and 1. Moreover, the fuzzy membership value is assigned based on the approximate membership values of the neighboring objects. At each data point, it iteratively update the membership values with pre-defined number of clusters. Finally, the membership matrix is constructed with respect to the minimum of A_n maximum of A_n and the mean of A_n (Algorithm 2).

Algorithm 2; Membership matrix generation:

Step 1: Load training data to create the membership matrix
Step 2: Let, x = 1 and y = 1
Step 3: For i = 1 to Lb //Where, Lb-Number of available labels
Step 4: For j = 1 to A_n//Where, A_n-Number of attributes
Let $A_n(j)$ be the list of values in A_n
//Where selected attribute is j
Compute $Memb\_mat_{xy}$ = Min [$A_n(j)$]
x = x+1
Compute Membmatxy = Max [$A_n(j)$]
x = x+1
Compute $Memb\_mat_{xy}$ = Sum[$A_n(j)$]/Size [$A_n(j)$]
x = x+1
For k = 1 to Size[$A_n(j)$]
Compute

$$T_{val} = \sqrt{\frac{\sum[A_n(j)_k] - Memb_{matxy}}{Size[A_n(j)]}}$$

End for k
$Memb_{matxy} = T_{val}$, x = x+1
j = j+1
x = 1
y = y+1
Step 5: End for j
x = x+1
y = 1
Step 6: End for i

In this stage, the generated membership matrix and training data are given as the input, based on this, the features are extracted. It is used to reduce the data dimensionality by selecting the most representative features from the original feature subspace. Based on the size of training data, the difference and addition between the matrices are performed and stored in a separate variables. Then, the features are labeled as 0 and 1 by checking the condition between the training data and the estimated values of v_1-v_5 (Algorithm 3).

Algorithm 3; Feature extraction:

Input: $Memb\_mat_{xy}$ Training data T_r
Output: Extracted Features
Let n = 1 to size ($Memb\_mat_{xy}$)
For i = 1 to size (T_r)
For j = 1 to A_n
For k = 1 to $Memb_{matxy}$
$v_1 = (Memb\_mat_{j,2} - Memb_{matj,4})$
$v_2 = (Memb\_mat_{j,2} + Memb_{matj,4})$
$v_3 = (Memb\_mat_{j,1} \_ Memb_{matj,4})$
$v_4 = (Memb\_mat_{j,3} \_ Memb_{matj,4})$

v_5 = ($Memb\_mat_{j,2}$+$Memb_{matj,4}$)
If (Tr_i >= v_1 && Tr_i <= v_2)
Feat (i, j) = 1
If (Tr_i >= v_4 && Tr_i <= v_5)
Feat (i, j) = 1
If (Tr_i >= 0 && Tr_i <= v_5)
Feat (i, j) = 1
Else If (Tr_i >= v_5 && Tr_i <= v_4)
Feat (i, j) = 1
Else
Feat (i, j) = 0
End for k
j = j+1
End for j
End for i

Clustering: The density based clustering is the most extensively used technique for detecting the network intrusions which groups the similar patterns to retrieve the group of meaningful data. In this research, the Density Maximization-Fuzzy Means Clustering (DM-FCM) algorithm is proposed which identifies the arbitrarily shaped clusters based on the concept of density. Here, the standard Fuzzy C-Means (FCM) clustering technique is integrated with the density maximization approach for an efficient detection. It is a kind of non-parametric approach where the density is measured by using various number of objects that are nearest to the cluster. In this approach, the similar patterns are assigned to one cluster by grouping the set of unlabeled patterns into a set of clusters. The advantage behind this approach is it reduces the required amount of computation time by finding the distance. Also, the proposed DM-FCM is highly depends on the election of initial cluster center and its membership value from the preprocessed dataset. After performing numerous iterations, the final result meets to the actual cluster center. Here, the computational steps and the number of iterations are simplified for minimizing the time consumption. Algorithm 4 describes the working procedure of the proposed DM-FCM algorithm. At first, the dissimilarity function is estimated based on the generated membership matrix and testing data (Algorithm 4).

Algorithm 4; Clustering:

//Dissimilarity function estimation
Consider $Memb\_mat_{xy}$ and Testing data (T_s) as input and
n = Size of T_s Dis_mat_{xy} dissimilarity output matrix
For k = 1 to $Memb\_mat_{xy}$
For i = 1 to n
For j = 1 to A_n
Let inp_val = ($Ts_{i,j}$)
If inp_val <$Memb\_mat_{ij}$
dis_val = $Memb\_mat_{ij}$ _ inp_val
$UpdateDis\_mat_{ij}$ = dis_val
Else if inp_val >$Memb_{mati,1}$ && inp_val
< $Memb_{mat\,i,2}$
dis_val = $Memb\_mat_{i,2}$ inp_val
$UpdateDis\_mat_{ij}$ = dis_val
Else if inp_val >$Memb_{mati,2}$ && inp_val < $Memb_{mati,3}$
dis_val = $Memb\_mat_{i,3}$
inp_val
$UpdateDis\_mat_{ij}$ = dis_val
Else if inp_val >$Memb_{mati,2}$ && inp_val
<$Memb_{mati,3}$
dis_val = Memb_mati3
inp_val
$UpdateDis\_mat_{ij}$ = dis_vall
If loop >4
K = k+1
$Memb\_mat_{i,j}$ = $Memb\_mat_{i,j+4}$
End if
End for j
End for i
End for K
//Grouping
Compute Td_p←Preprocessed T_s//Where, T_s Testing data
Extract feat (Td_p)
Set Thres = 80%//Where, % cluster validity index, based on the % of the threshold set, the size of the group of values in the cluster varies
Let, M be the size of the Dis_mat:

$$Density = Density + \frac{\sum_{i=1}^{M} Dis_{mati}}{Thres\%}$$

Set V_1 = ($Memb_{matj,2}$-$Memb_{matj,4}$)
Set V_2 = ($Memb\_mat_{j,2}$-$Memb_{matj,4}$)
Group feat (Td_p) into low, medium, high
For i = 1 to size (feat (Td_p))
Inp = Feat (Td_p (i))
If (inp <= v_1)
Gr = 'Low'
Else if (inp <= v_2 && inp >v_1)
Gr = 'Medium'
Else if (inp >v_2)
Gr = 'High'
End for i

The major advantages of this technique are as follows:

- It observed the number of clusters at the beginning stage
- It provides the better results for the overlapped dataset
- In this technique, the data point is belongs to more than one cluster center

RESULTS AND DISCUSSION

Performance analysis: In this sector, the experimental results of existing and proposed techniques are evaluated by using various performance measures such as False Acceptance Rate (FAR), False Rejection Rate (FRR), Genuine Acceptance Rate (GAR), Receiver Operating Characteristics (ROC), sensitivity, specificity, accuracy, Jaccard, dice and Kappa. The datasets that used to evaluate the performance are KDD Cup and ADFA. To demonstrate the efficacy of the DM-FMC system, it is equated with the existing Support Vector Machine (SVM), k-Means and Fuzzy C-Means (FCM) techniques.

False acceptance rate: The False Acceptance Rate (FAR) is defined as the measure of probability that the IDS system will incorrectly detect the intruder data as normal data which is estimated a follows:

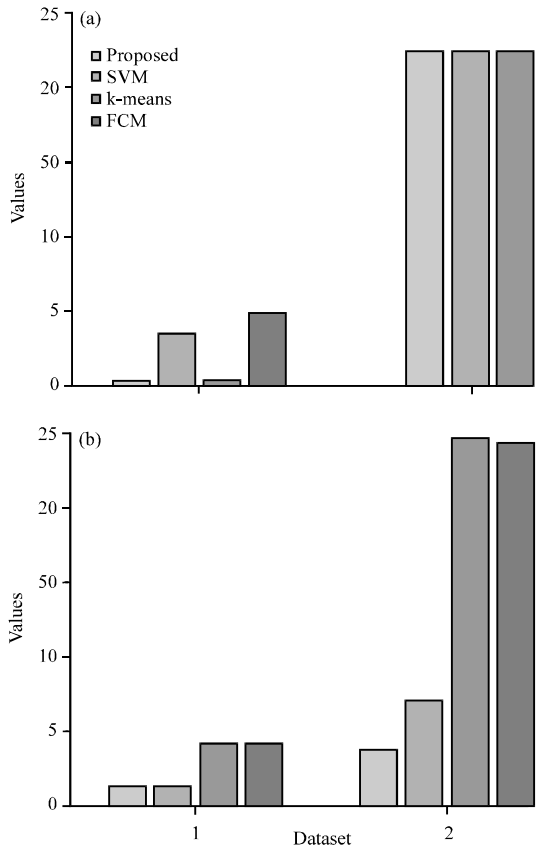

Fig. 3: a) FAR-KDD Cup dataset and b) FAR-ADFA dataset

$$F_{AR} = \frac{N_{FA}}{N_I} \quad (1)$$

Where:

N_{FA} = The Number of False Acceptances

N_I = The Number of Identification items

Figure 3 shows the FAR of existing and proposed intrusion detection mechanisms with respect two different classes for both KDD Cup and ADFA datasets. By using the DM-FMC technique, the reduced FAR for KDD Cup dataset is 0.2-class 1 and 0-class 2 and for ADFA dataset is 1.3-class 1 and 3.6-class 2. From the examination, it is perceived that the DM-FMC algorithm offers the minimized FAR for both datasets by efficiently clustering the data based on its density.

False rejection rate: The False Rejection Rate (FRR) is defined as the measure of probability that the IDS will incorrectly identify the normal data as intruder data which is estimated as follows:

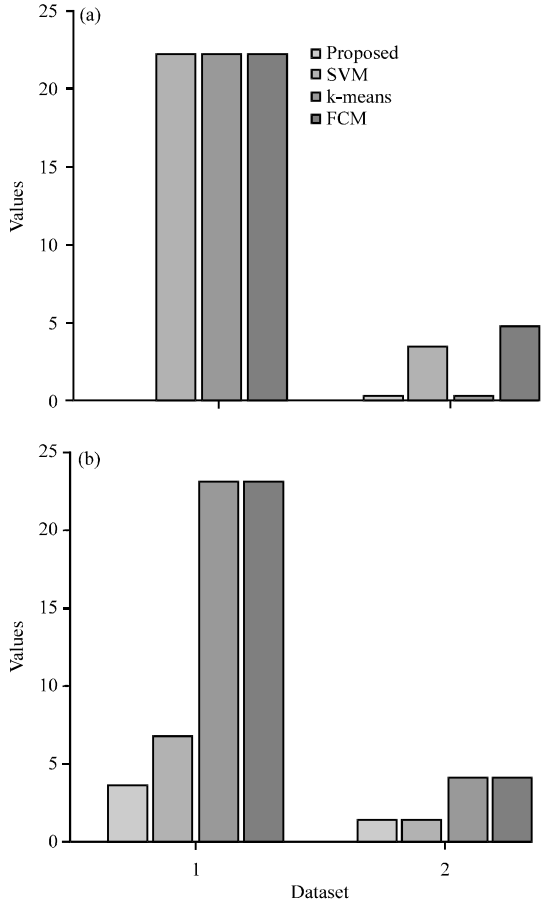

Fig. 4: a) FRR-KDD Cup dataset and b) FRR-ADFA Cup dataset

$$FRR = \frac{N_{FA}}{N_I} \quad (2)$$

Where:

N_{FA} = The Number of False rejections

N_I = The Number of Identification items

Figure 4 shows the FRR of existing and proposed intrusion detection mechanisms with respect two different classes for both KDD Cup and ADFA datasets. By using the DM-FMC technique, the reduced FRR for KDD Cup dataset is 0-class 1 and 0-class 2 and for ADFA dataset is 3.6-class 1 and 1.3-class 2. From the study, it is experiential that the DM-FMC algorithm delivers the minimized FRR for both datasets by efficiently clustering the data based on its density.

Genuine acceptance rate: The Genuine Acceptance Rate (GAR) is defined as the measure of truly matched intruder data over the total number of tests. It is evaluated as follows:

$$GAR = 1 - FRR \quad (3)$$

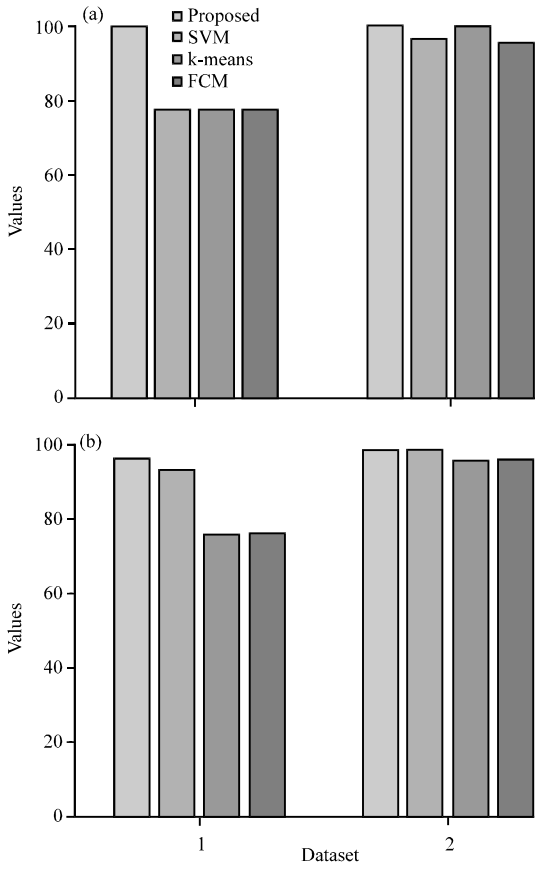

Fig. 5: a) GAR-KDD Cup dataset and b) GARADFA Cup dataset

Figure 5 shows the GAR of existing and proposed intrusion detection mechanisms with respect two different classes for both KDD Cup and ADFA datasets. In this analysis, the GAR of the KDD Cup dataset is increased as 99.96-class 1 and 99.80-class 2 and the GAR for ADFA dataset is 96.35-class 1 and 98.65-class 2. From that, it is perceived that the DM-FMC algorithm offers an increased GAR compared than the other methods. Because, it integrates the benefits of both density estimation and dissimilarity computation.

ROC for classification: Receiver Operating Characteristics (ROC) is plotted by the fraction of true positives out of the True Positive Rate (TPR) vs. the fraction of the false positives out of the False Positive Rate (FPR) at different thresholds. Figure 6 shows the ROC analysis of both existing and proposed techniques with respect to the TPR and FPR.

Overall performance: Sensitivity is defined as the ratio of the true positives and the sum of true positives and false negatives:

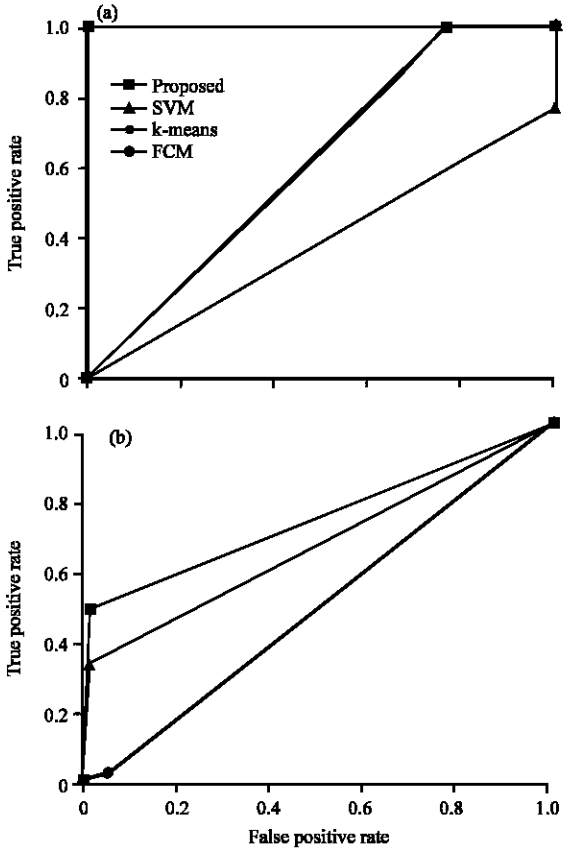

Fig. 6: a) Analysis of ROC for KDD Cup dataset and b) Analysis of ROC for ADFA cup dataset; ROC for classification

$$\text{Sensitivity} = \frac{TP}{TP+FN} \tag{4}$$

Similarly, specificity is defined as the number of true negative results that are divided by the sum of true negatives and false positives:

$$\text{specificity} = \frac{TN}{TN+FP} \tag{5}$$

The correctness and efficiency of the intrusion detection is validated based on the measure of accuracy. It is determined by the values of both sensitivity and specificity:

$$\text{Accuracy} = \frac{TN+TP}{(TN+TP+FN+FP)} \tag{6}$$

Precision is defined as the positive predictive value that provides the results relevant to an accurate intrusion detection:

$$\text{Precision} = \frac{TP}{(TP+FP)} \tag{7}$$

Recall is also termed as sensitivity which provides the most relevant results during intrusion detection:

$$\text{Recall} = \frac{TP}{(TP+FN)} \quad (8)$$

Jaccard is defined as the ratio of intersection and union of two objects which varies from 0 to 1. If the value is 1, the two objects are identical and their sets have no common regions:

$$\text{Jaccard} = \frac{|Y_{x1} \cap Y_{x2}|}{(|Y_{x1}|+|Y_{x2}|)} \quad (9)$$

Dice is also used to isolate the similarity between two different objects Y_1 and Y_2:

$$D_s = \frac{2.|Y_{x1} \cap Y_{x2}|}{(|Y_{x1}|+|Y_{x2}|)} \quad (10)$$

Then, the difference between the observed agreements are identified by using the kappa coefficient:

$$\text{Kappa coefficient} = \frac{(o * \text{sum } M_{ii}) - \text{sum}(M_{i+} * M_{+i})}{o^2 - \text{sum}(M_{i+} * M_{+i})} \quad (11)$$

Table 1 and 2 show the overall comparative analysis of the existing and proposed techniques for both KDD Cup and ADFA datasets. In this evaluation, it is verified that the DM-FMC yields better detection results associated with the other techniques by efficiently identifying the intrusion from the datasets.

Table 1: Comparative analysis between existing and proposed techniques for KDD cup dataset

Measures	DM-FMC	SVM	k-means	FCM
TP	663	0	0	0
TN	2330	2235	2329	2195
FP	6	101	7	141
FN	1	664	664	664
Sensitivity	99.84	0	0	0
Specificity	99.74	95.67	99.70	93.96
Precision	99.10	0	0	0
Recall	99.89	0	0	0
Jaccard	99.76	74.50	77.63	73.16
Dice	99.88	85.38	87.40	84.50
Kappa	0.99	0.21	0.01	0.29
Accuracy	99.77	74.5	77.63	73.17

Table 2: Comparative analysis between existing and proposed techniques for ADFA dataset

Measures	DM-FMC	SVM	k-means	FCM
TP	1432	1381	376	372
TN	52	52	64	65
FP	21	21	9	8
FN	57	108	1113	1117
Sensitivity	96.17	92.74	25.25	24.98
Specificity	71.23	71.23	87.67	89.04
Precision	98.55	98.50	97.66	97.89
Recall	96.17	92.74	25.25	24.98
Jaccard	95	91.74	28.16	27.97
Dice	97.43	95.69	43.95	43.72
Kappa	0.79	0.60	0.03	0.03
Accuracy	95	91.74	20.17	27.98

CONCLUSION

This study developed a DM-FMC methodology to detect the anomalous data from the network traffic datasets. The data preprocessing is performed to normalize the data by removing the irrelevant attributes. Then, the membership matrix is generated by computing the degree of membership for each data and it is classified by the membership function. The centroid is calculated to decide the number of clusters. After that, the distance between the centroid and data point is computed by applying the dissimilarity function on the data. Based on the distance, the threshold and density values are computed. Consequently, the cluster is formed by comparing the distance between the data point centroid and the density centroid, if the distance is less, it is added into the cluster. Finally, the cluster is categorized based on the sensitivity value which provides an increased detection efficiency. In experiments, the results of the existing SVM, k-means, FCM and proposed techniques are evaluated and compared by using various measures. From the investigation, it is examined that the DM-FMC outperforms the other approaches by estimating the density.

RECOMMENDATIONS

In future, this research can be enhanced by practically applying the proposed IDS in a dynamical network traffic capture environment. Once performing the misdetection, it is hard to reset the IDS, so, it can be focused on our next work. Also, the additional maintenance cost and false positive rate are minimized by using an automated reinforcement learning model.

REFERENCES

Ashfaq, R.A.R., X.Z. Wang, J.Z. Huang, H. Abbas and Y.L. He, 2017. Fuzziness based semi-supervised learning approach for intrusion detection system. Inf. Sci., 378: 484-497.

Bhuyan, M.H., D.K. Bhattacharyya and J.K. Kalita, 2014. Network anomaly detection: methods, systems and tools. IEEE. Commun. Surv. Tutorials, 16: 303-336.

Buczak, A.L. and E. Guven, 2016. A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE. Commun. Surv. Tutorials, 18: 1153-1176.

Dhakar, M. and A. Tiwari, 2014. A novel data mining based hybrid intrusion detection framework. J. Inf. Comput. Sci., 9: 037-048.

Dhanabal, L. and S.P. Shantharajah, 2015. A study on NSL-KDD dataset for intrusion detection system based on classification algorithms. Intl. J. Adv. Res. Comput. Commun. Eng., 4: 446-452.

Duque, S. and M.N. Bin Omar, 2015. Using data mining algorithms for developing a model for Intrusion Detection System (IDS). Procedia Comput. Sci., 61: 46-51.

Elhag, S., A. Fernandez, A. Bawakid, S. Alshomrani and F. Herrera, 2015. On the combination of genetic fuzzy systems and pairwise learning for improving detection rates on intrusion detection systems. Expert Syst. Applic., 42: 193-202.

Ha, T., S. Kim, N. An, J. Narantuya and C. Jeong *et al*., 2016. Suspicious traffic sampling for intrusion detection in software-defined networks. Comput. Networks, 109: 172-182.

Hubballi, N. and V. Suryanarayanan, 2014. False alarm minimization techniques in signature-based intrusion detection systems: A survey. Comput. Commun., 49: 1-17.

Jabez, J. and B. Muthukumar, 2015. Intrusion Detection System (IDS): Anomaly detection using outlier detection approach. Procedia Comput. Sci., 48: 338-346.

Kaur, R. and S. Singh, 2016. A survey of data mining and social network analysis based anomaly detection techniques. Egypt. Inf. J., 17: 199-216.

Kenkre, P.S., A. Pai and L. Colaco, 2014. Real time intrusion detection and prevention system. Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014, November 14-15, 2014, Springer, Cham, Switzerland, ISBN:978-3-319-11932-8, pp: 405-411.

Kim, G., S. Lee and S. Kim, 2014. A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. Expert Syst. Appl., 41: 1690-1700.

Kumar, G.R., N. Mangathayaru and G. Narsimha, 2016. A novel similarity measure for intrusion detection using Gaussian function. Rev. Tec. Ing. Univ. Zulia., 39: 173-183.

Leu, F.Y., K.L. Tsai, Y.T. Hsiao and C.T. Yang, 2017. An internal intrusion detection and protection system by using data mining and forensic techniques. IEEE. Syst. J., 11: 427-438.

Li, W., W. Meng, X. Luo and L.F. Kwok, 2016. MVPSys: Toward practical multi-view based false alarm reduction system in network intrusion detection. Comput. Secur., 60: 177-192.

Lin, W.C., S.W. Ke and C.F. Tsai, 2015. CANN: An intrusion detection system based on combining cluster centers and nearest neighbors. Knowl. Based Syst., 78: 13-21.

Nadiammai, G.V. and M. Hemalatha, 2014. Effective approach toward intrusion detection system using data mining techniques. Egypt. Inf. J., 15: 37-50.

Naik, P.P. and S.J. Prashsntha, 2014. An approach for building intrusion detection system by using data mining techniques. Intl. J. Emerging Eng. Res. Technol., 2: 112-118.

Ni, X., D. He and F. Ahmad, 2016. Practical network anomaly detection using data mining techniques. VFAST. Trans. Software Eng., 9: 1-6.

Patel, J. and K. Panchal, 2015. Effective intrusion detection system using data mining technique. J. Emerging Technol. Innovative Res., 2: 1869-1876.

Posenato, D., F. Lanata, D. Inaudi and I.F. Smith, 2008. Model-free data interpretation for continuous monitoring of complex structures. Adv. Eng. Inf., 22: 135-144.

Ravale, U., N. Marathe and P. Padiya, 2015. Feature selection based hybrid anomaly intrusion detection system using K means and RBF kernel function. Procedia Comput. Sci., 45: 428-435.

Saurabh, P. and B. Verma, 2016. An efficient proactive artificial immune system based anomaly detection and prevention system. Expert Syst. Appl., 60: 311-320.

Subaira, A.S. and P. Anitha, 2014. Efficient classification mechanism for network intrusion detection system based on data mining techniques: A survey. Proceedings of the 2014 IEEE 8th International Conference on Intelligent Systems and Control (ISCO), January 10-11, 2014, IEEE, Coimbatore, India, ISBN:978-1-4799-3837-7, pp: 274-280.

Wu, D. and S. Shan, 2015. Meta-analysis of network information security and Web data mining techniques. Proceedings of the 1st International Conference on Information Sciences, Machinery, Materials and Energy, July 7-18, 2015, Atlantis Press, Paris, France, pp: 1974-1977.

Xie, Y., D. Feng, Z. Tan and J. Zhou, 2016. Unifying intrusion detection and forensic analysis via. provenance awareness. Future Gener. Comput. Syst., 61: 26-36.

Zhao, Y., 2016. Network intrusion detection system model based on data mining. Proceedings of the 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), May 30-June 1, 2016, IEEE, Shanghai, China, ISBN:978-1-5090-0804-9, pp: 155-160.