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 Prashantha, 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

![Diagram of IDS Architecture](image-url)

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; Subair and Anitha, 2014). The drawbacks (Hubbali 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 Mudunkumar (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. Ellag 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 (IIDDPS). 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 IIDDPS. 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. Dhamabal and Shantharajah (2015) investigated various classification models that includes decision tree, Neural Networks (NN), Naïve 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, it 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
- count
- rdatecount
- error_rate
- Srv_serror_rate
- error_rate
- rdateerror_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_srv_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
- Dtl
- Sload
- Dload
- loss
- loss
- inpkt
- Dirpkts
- Sjit
- Djit
- Swin
- Stcpb
- Dtcpb
- Dwin
- Tcprrt
- 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_flow_http_mthd
- t_src_ltm
- t_srv_dst
- s_sm_ips_ports
- stack_eat_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 be the original dataset, D_p be the preprocessed dataset, A_v 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: The dataset is preprocessed by separating the data into labels and other features

Step 4: Assign the binary and multi-class labels to the records

Step 5: The number of records utilized for training

n = b \times A_v

where, n = total number of records in the file

m = a \times A_v

where, m = total number of records to be training

Step 6: for i = 1 to n

Tr = D_p, j = i + 1

end for i

for j = 1 to m

T_r = D_p, 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_v maximum of A_v and the mean of A_v (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_v //Where, A_v-Number of attributes

Step 5: Compute $\text{Memb}_{\text{mat}_{ij}} = \min [A_v(j)]$

Step 6: $x = x+1$

Step 7: Compute $\text{Memb}_{\text{mat}} = \max [A_v(j)]$

Step 8: $x = x+1$

Step 9: Compute $\text{Memb}_{\text{mat}} = \sum [A_v(j)] / \text{Size}[A_v(j)]$

Step 10: $x = x+1$

Step 11: For k = 1 to Size($A_v$)

Step 12: Compute

$$T_{ei} = \frac{\sum [A_v(j)] - \text{Memb}_{\text{mat}_{ij}}}{\text{Size}[A_v(j)]}$$

Step 13: End for $k$

**Algorithm 3; Feature extraction:**

Input: Mem, Mat, Training data T

Output: Extracted Features

Let n = 1 to size(Mem, Mat), For i = 1 to size(T)

For j = 1 to A_v

For k = 1 to Mem, Mat

$\text{v}_i = (\text{Mem, Mat}_{ij}; \text{Memb}_{\text{mat}_{ij}})$

$\text{v}_i = (\text{Mem, Mat}_{ij}; \text{Memb}_{\text{mat}_{ij}})$

$\text{v}_i = (\text{Mem, Mat}_{ij}; \text{Memb}_{\text{mat}_{ij}})$

$\text{v}_i = (\text{Mem, Mat}_{ij}; \text{Memb}_{\text{mat}_{ij}})$

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 $v_1$, $v_2$, $v_3$, $v_4$ (Algorithm 3).
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 Memh_mat and Testing data (T) as input and n = Size of T, Dis_mem, dissimilarity output matrix
For k = 1 to Memh_mat:
   For i = 1 to n:
      For j = 1 to A:
         Let inp_val = (T)ij
         If inp_val > Memh_matij
            dis_val = Memh_matij - inp_val
            UpdateDis_mat = dis_val
         Else if inp_val < Memh_matij
            dis_val = Memh_matij - inp_val
            UpdateDis_mat = dis_val
         End if

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 efficacious of the DM-FCM 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:
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:

\[ F_{\text{AR}} = \frac{N_{\text{FA}}}{N_{1}} \]  

(1)

Where:
- \( N_{\text{FA}} \) = The Number of False Acceptances
- \( N_{1} \) = 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:

\[ \text{GAR} = 1 - \text{FRR} \]  

(3)
Figure 5 shows the GAR of existing and proposed intrusion detection mechanisms with respect to 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 with 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:

\[
\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FP}}
\]

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

\[
\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}
\]

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:

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

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

\[
\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}
\]
Recall is also termed as sensitivity which provides the most relevant results during intrusion detection:

\[ \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \]  

(J9)

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:

\[ \text{Jaccard} = \frac{|Y_{xi} \cap Y_{xj}|}{|Y_{xi}| + |Y_{xj}|} \]  

(D0)

Dice is also used to isolate the similarity between two different objects \(Y_i\) and \(Y_j\):

\[ D_x = \frac{2|Y_{xi} \cap Y_{xj}|}{|Y_{xi}| + |Y_{xj}|} \]  

(D1)

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

\[ \text{Kappa coefficient} = \frac{(\text{sum} \times \text{M}_n - \text{sum}(\text{M}_a \times \text{M}_a))}{\text{sum}^2 - \text{sum}(\text{M}_a \times \text{M}_a)} \]  

(D11)

Table 1 and 2 show the overall comparative analysis of the existing and proposed techniques for KDD cup dataset.

<table>
<thead>
<tr>
<th>Measures</th>
<th>DM-FMC</th>
<th>SVM</th>
<th>k-means</th>
<th>FCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>TP</td>
<td>663</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TN</td>
<td>2310</td>
<td>2225</td>
<td>2329</td>
<td>2195</td>
</tr>
<tr>
<td>FP</td>
<td>6</td>
<td>101</td>
<td>7</td>
<td>141</td>
</tr>
<tr>
<td>FN</td>
<td>1</td>
<td>664</td>
<td>664</td>
<td>664</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>0.99</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Specificity</td>
<td>0.77</td>
<td>0.67</td>
<td>0.99</td>
<td>0.96</td>
</tr>
<tr>
<td>Precision</td>
<td>0.99</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Recall</td>
<td>0.99</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Jaccard</td>
<td>0.76</td>
<td>0.5</td>
<td>77.63</td>
<td>73.16</td>
</tr>
<tr>
<td>Dice</td>
<td>0.88</td>
<td>0.38</td>
<td>87.40</td>
<td>84.50</td>
</tr>
<tr>
<td>Kappa</td>
<td>0.99</td>
<td>0.21</td>
<td>0.01</td>
<td>0.29</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.77</td>
<td>0.45</td>
<td>77.63</td>
<td>73.17</td>
</tr>
</tbody>
</table>

Table 2: Comparative analysis between existing and proposed techniques for A DFA dataset.

<table>
<thead>
<tr>
<th>Measures</th>
<th>DM-FMC</th>
<th>SVM</th>
<th>k-means</th>
<th>FCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>TP</td>
<td>1412</td>
<td>1381</td>
<td>376</td>
<td>372</td>
</tr>
<tr>
<td>TN</td>
<td>52</td>
<td>52</td>
<td>64</td>
<td>65</td>
</tr>
<tr>
<td>FP</td>
<td>21</td>
<td>21</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>FN</td>
<td>57</td>
<td>108</td>
<td>1113</td>
<td>1117</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>0.96</td>
<td>0.74</td>
<td>25.25</td>
<td>24.98</td>
</tr>
<tr>
<td>Specificity</td>
<td>0.73</td>
<td>0.73</td>
<td>87.67</td>
<td>89.04</td>
</tr>
<tr>
<td>Precision</td>
<td>0.98</td>
<td>0.58</td>
<td>97.66</td>
<td>97.89</td>
</tr>
<tr>
<td>Recall</td>
<td>0.96</td>
<td>0.74</td>
<td>25.25</td>
<td>24.98</td>
</tr>
<tr>
<td>Jaccard</td>
<td>0.97</td>
<td>0.74</td>
<td>28.16</td>
<td>27.97</td>
</tr>
<tr>
<td>Dice</td>
<td>0.74</td>
<td>0.69</td>
<td>43.95</td>
<td>43.72</td>
</tr>
<tr>
<td>Kappa</td>
<td>0.70</td>
<td>0.70</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.95</td>
<td>0.74</td>
<td>20.17</td>
<td>27.98</td>
</tr>
</tbody>
</table>

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.

**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**


