ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

Intrusion Detection for Attaining Rapid Performance Using PK-Medoid-HHNN Technique.

Jabez J[1]* , MuthuKumar B[2], and Yovan Felix A[3].

[1]Research Scholar, Sathyabama University, Chennai -119.
[2]Professor, Sathyabama University, Chennai -119.
[3]Assistant Professor, Sathyabama University, Chennai -119.

ABSTRACT

Over past decade computer networks have a rapid growth, but computer security became a critical problem for computer systems. Thus in recent years various soft computing technique based methods were proposed to detect the growth of intrusion. Also many researchers have reported that large set of pattern classifications and machine learning algorithms are trained and tested for knowledge discovery in intrusion detection with dataset. These above mentioned mechanism are not more unsuccessful in finding client to root and remote to local attacks. Moreover Hyperbolic Hopfield Neural Network based IDS detection system has stability, detection ratio, particularly low-frequent attacks apart from this this mechanism still has concerns in its performance improvement. Thus this paper proposes a new method known as Potential K-Medoid-HHNN approach. This Potential K-Medoid-HHNN system achieves higher intrusion detection rate, detection stability and less false positive rate. At first the proposed system implements the PK-Medoid clustering technique on various training subsets. Finally mono HHNN model is trained using different training subsets to detect the intrusion. The experimental results show that PK-Medoid-HHNN approach achieves better result compared with other existing framework.

Keywords: Network Security, Intrusion Detection System, Hyperbolic Hopfield Neural Network, PK-Medoid Clustering.

**Corresponding author*

INTRODUCTION

In the last decades, the computer network systems are very fast in development and spreads all over the world. And also the intruders and hackers are highly connected into computing world and also prepared for a new ability as destructive cause. In the network, the cost of short-term or stable destruction rooted by illegal intruders' access urges to employ several schemes to observe data flow in networks. These schemes are known as Intrusion Detection Systems (IDSs).

Two types of Intrusion detection are there namely 1) Misuse intrusion detection and 2) Anomaly intrusion detection. Anomaly intrusion detection system is which identifies the intrusion detection by using the deviations of a normal pattern. The usage of normal pattern created from statistical measures system. For instance, CPU and I/O actions are extracted by client or programmer. Misuse intrusion detection is a well-structured pattern system that attacks the application software failure and identifies intrusions [2]. The patterns are used to enhance and also to compare against performance of user in order to detect the intrusions.

Many researches are used in the detection stability and detection precision systems [4]. In before stage, using statistical and rule based expert system method research focus deceit [5]. However, when utilizing bigger datasets, the effects of statistical approaches and rule-based expert systems are very poor. As a result, a number of data mining methods are designed to resolve this issue[7], [6] where the Neural Network (NN) is also a data mining technique that is a greatest technique and is utilized generally for solving the complex problem and also it is effectively applied in intrusion detection system[8].

Alternatively, Neural Network-based Intrusion Detection System is still having disadvantages that are listed by two phases: 1) Weaker detection stability and 2) Lower detection precision. The lower detection precision is the small frequent attacks. For instance, Root user and remote to local. The weaker detection stability is the poor attacks [9]. These two aspects mainly distribute the different attacks and imbalance. The low-frequent attacks are small in size as well as the high-frequent attacks. It directs the Neural Network that could not study characters of the attacks in simple manner and thus the detection accuracy is very low. In practical, there is no more important for low-frequent attacks. These attacks are succeeded by a serious consequence caused. Examples for these attacks are succeeded in the root user and it may be do anything in the network device or computer systems. Moreover, the intrusion detection systems are also used as low-frequent attacks. So the Neural Network is an unbalanced converges for the local minimum user. The previous research proposed a few approaches but when encountering huge datasets, these approaches ineffective [10] [4]. Using Time probability approach the network dataset identify non-anomaly and anomaly pattern dataset and also generate the rules. Hyperbolic Hopfield neural network train the non-anomaly and anomaly patterns [29]. To resolve these two problems, the proposed framework presents a novel approach for Neural Network-based on Intrusion Detection System using PK-Medoid-Hyperbolic Hopfield Neural Network, to improve detection for detection stability and less-frequent attacks. General purpose of PK-Medoid-HHNN approaches three stages. First stage, PK-Medoid cluster technique which is utilized to create various training datasets. Second stage, HHNN Mono model which is designed into the clustering dataset and in the final stage the dataset is tested by well-structured model. These models obtain detection rate and maximum accuracy with minimum time.

In this paper work structure is like follows. In Section two related work of Intrusion Detection Systems is discussed. In Section III it elaborates the PK-Medoid-Hyperbolic Hopfield Neural Network framework of approach briefly, and explains its working procedures and principles. To evaluate PK-Medoid-Hyperbolic Hopfield Neural Network approach, the Section 4 illustrates the evaluation criteria, data preparation, results and discussions of conducted experiments. Lastly, Section 5 depicts conclusions and future work of proposed framework.

RELATED WORK

Over last decade, different approaches have been proposed and developed to detect the intrusion analysis in data mining concept [4][17]. In early day's statistical methods and Rule based expert systems were two methods to identify the intrusion analysis.

Whereas the rule based expert system could detect the intrusion in the more detection rate but isn't easy to identify the new intrusions and the signature database that require to be modernized frequently and manually. And statistical based intrusion detection system establishes the different statistical methods such as Bayesian analysis [20], multivariate analysis [19], principal component analysis [18], simple significance tests and frequency [21] and cluster. But this kind of intrusion detection system (IDS) required gather sufficient data to establish complicated mathematical model. Also it is not practical in the case of difficult network traffic [22].

Li [11] introduced a method using the genetic algorithm to find out the anomalies in the network intrusion [12] [13]. The method also has incorporated with the categorical and quantitative features of network so as to take classification rules. Alternatively with addition of quantitative features, the detection rate is increased in IDS but there are no results in the experimental setup. In [14] Goyal and Kumar explain Genetic Algorithm support to categorize all kind of smurf attack with training dataset. The experimental detection rate is 100% and rate of false positive is too low at 0.2% [13].

Using GP Historical network dataset to get a set of classification in Lu and Traore [15][19]. Also the methodology uses the sustain confidence framework like a fitness function with exactly classified the several network intrusions. However the genetic programming advantage of creating implementation procedure, which is extremely hard also training method requires extra time and data.

It [16] uses the genetic algorithm to identify the anomalous using the information theory [12][13]. A few network characteristics could be recognized by network attacks using the data like kind of intrusion and mutual information among network features. Further features like a genetic algorithm and a linear structure is obtained. The combination of the linear rule result and mutual information look extremely efficient due to higher detection rate and reduced complexity. The main issue is it takes only discrete features. Gong et al [12] implemented the Genetic Algorithm based method to Intrusion Detection Network and shows the software implementations. Methods develop the classification rules utilized as sustain confidence framework to evaluate the fitness function. Time based intrusion detection system capable to identify penetration break and described computer mistreatment. Time based intrusion model depends on hypothesis and using security violate identify abnormal patterns from system usage. For representing subject performance through metrics and statistical objects this model has profile [29].

Neural Network is most popular techniques among the above discussed and it has been effectively used in the intrusion detection process [23]. Three types of neural network techniques are available they are as follow

- **i)** Supervised NN Intrusion detection
- **ii)** Unsupervised NN Intrusion detection
- **iii)** Hybrid NN Intrusion detection

When the managed Neural Network is implemented for the Intrusion Detection system, it mainly consist of MLFF (multi-layer-feed-forward) recurrent neural networks and neural networks [24] to detect anomalies based on the user performances. In practical amount of training set is huge. Thus the training set distribution is imbalanced. MLFF neural networks are simple to accomplish local minimum value and its stability very low.

Unsupervised Neural Network is used to categorize the input information and divide the normal performances from the intrusive and abnormal ones [8]. While using the unsupervised the intrusion detection system Neural Network analysis, there exist many advantages that could improve their new data investigation without retraining. Moreover detection of lower precision is achieved for less frequent attacks through this method [9] Hybrid Neural Network is the third Category that combines both unsupervised NN and Supervised NN used to detect the intrusion with the other data mining techniques [25]. The main objective of using hybrid NN is to reduce the disadvantage of individual Neural Network [26]

These approaches are using maximum size of KDD datasets to train the NN approaches. But, it is not feasible to train the NN with whole dataset and unordered way. This problem is motivated our research in order to improve the NN IDS performance.

PROPOSED WORK

Overview

In this work, our main aim is to model and develop IIDS. This IID system would be less in false alarm, being real time, more flexible, hard to cheat by small changes in patterns, and more accurate. The above structured diagram shows the structure of our proposed work to find the abnormal dataset.

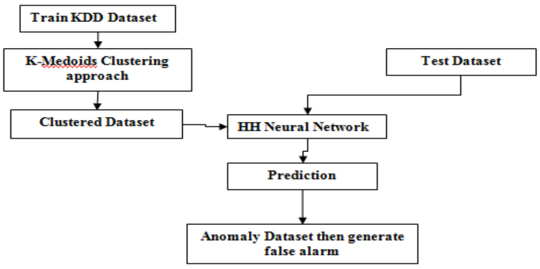


Figure 1.Proposed framework structure

INTRUSION DETECTION SYSTEM

This system has examined from large batch oriented system to dispersed networks as revealed in Fig.2. In present systems general building functional blocks collections are differentiated.

1. *Probe, Sensor:* These modules create components (primordial data gathering component) Intrusion detection, these components are utilized in highly system-specific manner, they check traffic of network, and behavior of system (translating more amounts of data into n number IDS monitor usable events).
2. *Monitor*: Analyze the components, performing segment of an ID, obtains measures from the sensor. Then these actions are properly correlated against the behavior of intrusion detection models, producing updates and alerts of models. Actions and alerts in it represent the major occurrences to the system security and these occurrences may be forwarded to resolver units, High-level monitors.
3. *Resolver*: This component obtains report from monitor and finds proper response – reconfiguring the firewall rules, logging, changing the activities of lower level components, and reporting operators.
4. *Controller:* In this intrusion detection system, providing component coordination and configuration, controller components are more important where manually configuring, initiating and upgrading components are impossible. Further, controller units provide examination point and Intrusion Detection administration, restarting components which are failed, and also act as supervisory capacity.

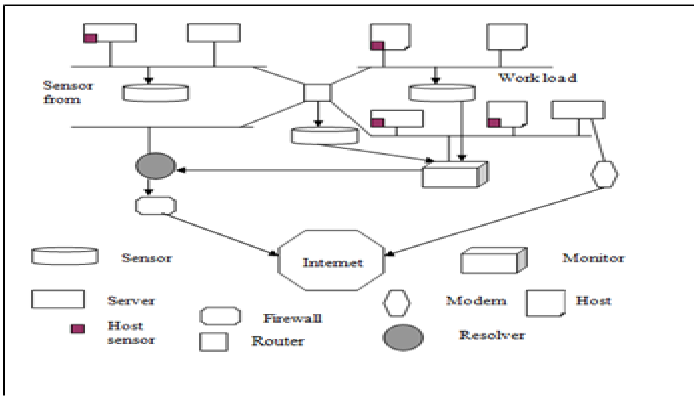


Figure 2: Intrusion Detection Network Structure

In more number of Intrusion Detection architecture these above mentioned parts are different. In large ID system, the entire components create a single unit or divided as tools and more number of processes.

NEURAL NETWORK APPROACH

The collections of neurons are called as Neural Network. A neuron is nothing but sequence automation, observes addition of more inputs as per more number of weights, and then determines a sigmoid or tangent function to receive the output value. The Transfer function option finds whether neuron is frequently valued or binary. To create NN, these Neurons are connected together as per the provided topology. This provided topology explains the output layer where the values of neuron activations offer net answer. The input layer, when set of neuron activations are kept to the input values. Thus, the NN is considered as non sequence transfer function. Parameters of transfer function are weights of neurons. Providing input to network generates output.

Programming concept by example is a difficult part of Neural Network in that n number of weights makes much complex to get the required result. Instead, the neural network is programmed by repetition as well as example. It is well trained repeatedly using input and output pairs. So, every time the input is given to Neural Network and the NN estimates an output. Output part from input-output pair is utilized to find whether network is correct or wrong. While the network is wrong, the network is corrected using learning algorithm. After performing every modification, the networks get near to the desired function.

The Neural Network is used for intrusion detection purpose in our proposed paper. In this field, T.F Lunt offers more number of benefits of neural networks [27]. The evaluation of these benefits and differentiation along with proposed advantages in this part is dedicated to the analyzed results.

The usage of neural network explained at the project for Harris Corporation to find viruses on computer systems and this is called as kohonen map [28]. Our approach supports Hyperbolic Neural Network intrusion detection. Hyperbolic Hopfield Neural Network IDS is already introduced in prior research work [29] and that shows that the malicious data are determined effectively. but, due to the dataset size the performance of HHNN is reduced. Here we have planned to introduce PK-Mediod-HHNN to resolve these type of problem very effectively. Also this proposed framework includes training, clustering, and testing from datasets. Initially tested and training datasets are been clustered with PK-Mediod, and then HHNN Model is trained using clustered result datasets that are obtained from PK-Mediod results . Finally the datasets are tested with help of trained datasets.

PK-MEDOID ALGORITHM

This algorithm is one of the clustering algorithms such as k-means. A mediod is one of the data point and act as the other entire data points. The k-means algorithm is more effective to outliers, because if an object exists with a large value, the distribution of data may be destroyed [13]. In this situation the PK-Medoid are strong to outliers as well as noise because in this, the partitioning method is done as per the concept of minimizing the total of dissimilarities between every object in cluster.
Procedure of PK-Medoid is defined below

```
Input:
Required cluster k
No of given input data n
Objects in dataset d

Output:
Set of cluster k
Function cluster()
Select data randomly
Data = rand (num_points, 2);
Iteration =0
Do clustering with the given cluster k
[Groups, centroids]=cluster (data, k);
Function [groups, centroids] =cluster (data, k);
Select initial centroids
K=1
```

```
Do
For k=1 to n
$K_i = 0$ ;
$n_i = 0$
End for
For i = 1 to n
Num_points= size (data, 1);
Group=ceil(k.*rand (num_point,1));
Computation of input data distance between each data point in D
Distance = d;
d= distance ($D_i, D_j$ , centroid (1,i), centropid (2,i)
if d< min, then
min =d
cluster = i
end if
d = DistMatrix (X,Y);
d(X, Y) = $\sqrt{(x1-y1)^2+(x2-y2)^2+\cdots\ \ldots\ldots\ldots..(xn-yn)^2}$
d(X, D) = Min(d(X,Y), where Y∈D)
Define distance of every data point,
$d_i(1 \leftarrow i \leftarrow n) to\ centroid\ c_i(1 \leftarrow i \leftarrow k)$
d ( $d_i, c_i$ );
Assign data to nearest centroid d ( $d_i, c_i$ )
Set cluster [i] =j
Set nearesrt_dist [i] = d ( $d_i, c_i$ )
End for;
For every cluster K$(1 \leftarrow j \leftarrow k)$;
Recalculate the centroid
End;
//UInitialize the attack on every centroid data
If protocol_type =”tcp”;

Step: 1
If flag value (SO, RSTO);
Src_byte=0;
While,
Dst_host_same_srv_rate = $d_i$ (0, 0.1)
Dst_host_ serror_rate =1;
Dst_host_srv_serror_rate=1;
Dst_host_reerror_rate=0
Disp(“DDOS Attack”);
end for;
end;
Step: 2
If
Service = gopher
Flag value (“SH”);
SRS_BYTES=0;
Disp (“PROBE Attack”);
end for ;
end;
Step: 3
If
flag value (RSTR)
While;
```

```
lnum_comromised = 0;
lroot_shell = 0;
lnum_root=0;
lnum_file_creation=0;
lnum_shell = 0;
lnum_access_files=0;
is_host_login=0;
hot=0;
disp("U2RAttack");
end for
end
Step: 4
If,
flag value (RSTR) = flag value (REJ);
Except;
Service = ("private, netbios_dgm, imap4, idap");
src_bytes=0;
and,
is_guest_login = 0;
disp("R2C Attack");
end for;
end;
Step: 5
If,
Flag value(RSTR)=flag value(REJ)=flag value(SF)
except;
land=0;
wrong_fragment=0;
num_failed_login=0;
logged_in=1;
dst_host_srv_count=255;
dst_host_same_src_port_rate("0, 0.1");
disp("Normal");
end for
end;
```

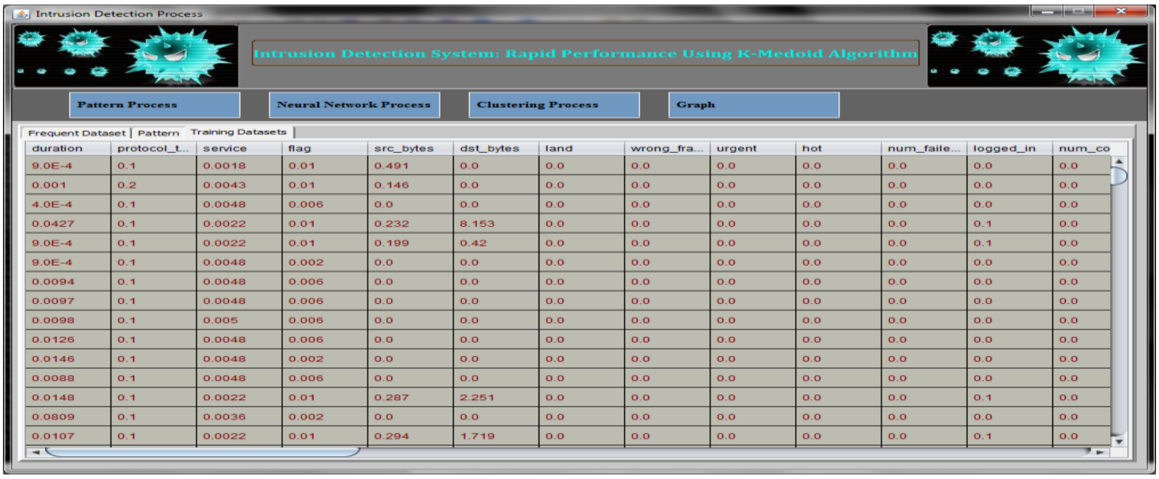
RESULTS AND DISCUSSION

Several experiments were performed on KDD dataset to analyze the proper function of PK-medoid-HHNN approach. By using following configuration Intel Pentium(R), Windows 7, processor speed 2.90 GHz and CPU G2020 our proposed methods that are implemented are analyzed and results are shown.

DATASET PREPARATION AND TESTING RESULTS

In the dataset nearly 2 thousand connection records are of training data and 5 thousand connection records are of test data. The KDD Cup „99 attack dataset is a public repository dataset which is used in promoting research works in the field of intrusion detection. Further a set of 41 features and 3, 11,029 records are derived from each connection records and a tag which specifies the position of connection records as general or unique attack type is also added in the dataset. All these features contain forms of discrete symbolic and continuous variables by extensively changeable series declining in four types: (1) intrinsic characteristics of a link, which contain essential TCP connections features. In connection time, TCP, UDP Protocol and telnet, http network services are various features. (2) Domain knowledge recommended the content features within a connection used to evaluate original TCP packets payload like no of failed login attempts. (3) Previous two seconds similar host features established connections that have similar host destination current connection, compute statistics related protocol service, behavior, etc.(4) in last two seconds similar service features examine inspect the connections that have similar service current connection.

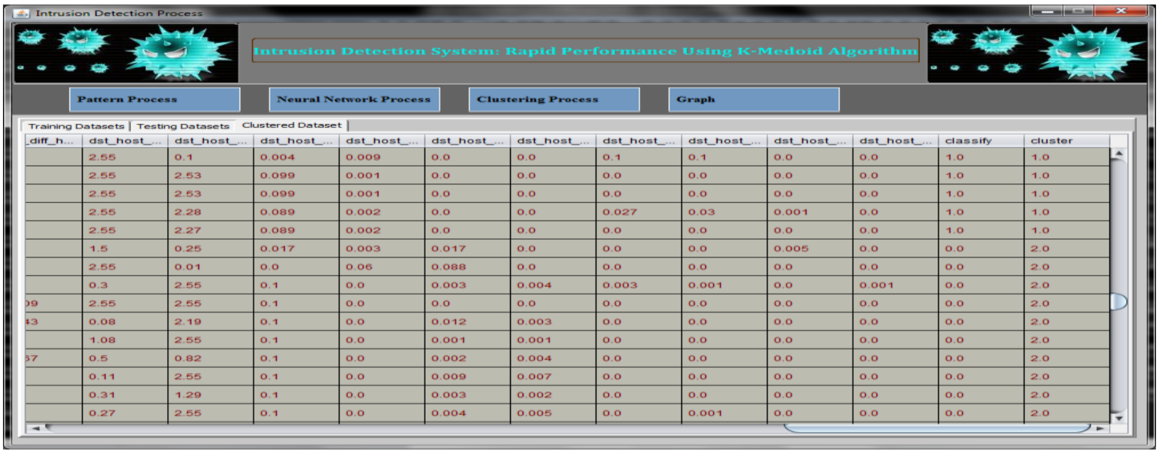
4.2 IMPLEMENTATION RESULTS



duration	protocol_t...	service	flag	src_bytes	dst_bytes	land	wrong_fra...	urgent	hot	num_faile...	logged_in	num_co
9.0E-4	0.1	0.0018	0.01	0.491	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.001	0.2	0.0043	0.01	0.146	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.0E-4	0.1	0.0048	0.006	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0427	0.1	0.0022	0.01	0.232	8.153	0.0	0.0	0.0	0.0	0.0	0.1	0.0
9.0E-4	0.1	0.0022	0.01	0.199	0.42	0.0	0.0	0.0	0.0	0.0	0.1	0.0
9.0E-4	0.1	0.0048	0.002	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0094	0.1	0.0048	0.006	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0097	0.1	0.0048	0.006	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0098	0.1	0.005	0.006	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0126	0.1	0.0048	0.006	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0146	0.1	0.0048	0.002	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0088	0.1	0.0048	0.006	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0148	0.1	0.0022	0.01	0.287	2.251	0.0	0.0	0.0	0.0	0.0	0.1	0.0
0.0809	0.1	0.0036	0.002	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0107	0.1	0.0022	0.01	0.294	1.719	0.0	0.0	0.0	0.0	0.0	0.1	0.0

Figure 3.Training Datasets

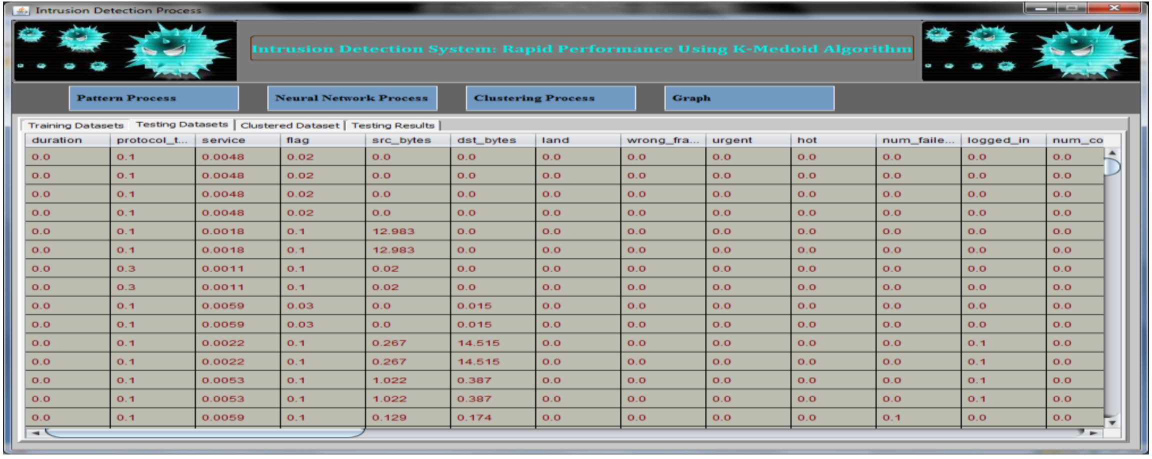
The figure 3 show output result of training datasets that are taken from KDD datasets. The dataset which has 41 feature attributes which represents the network communication structure are shown in figure3.



_diff_h...	dst_host_...	dst_host_...	dst_host_...	dst_host_...	dst_host_...	dst_host_...	dst_host_...	dst_host_...	dst_host_...	dst_host_...	classify	cluster
	2.55	0.1	0.004	0.009	0.0	0.0	0.1	0.1	0.0	0.0	1.0	1.0
	2.55	2.53	0.099	0.001	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0
	2.55	2.53	0.099	0.001	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0
	2.55	2.28	0.089	0.002	0.0	0.0	0.027	0.03	0.001	0.0	1.0	1.0
	2.55	2.27	0.089	0.002	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0
	1.5	0.25	0.017	0.003	0.017	0.0	0.0	0.0	0.005	0.0	0.0	2.0
	2.55	0.01	0.0	0.06	0.088	0.0	0.0	0.0	0.0	0.0	0.0	2.0
	0.3	2.55	0.1	0.0	0.003	0.004	0.003	0.001	0.0	0.001	0.0	2.0
09	2.55	2.55	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.0
43	0.08	2.19	0.1	0.0	0.012	0.003	0.0	0.0	0.0	0.0	0.0	2.0
	1.08	2.55	0.1	0.0	0.001	0.001	0.0	0.0	0.0	0.0	0.0	2.0
57	0.5	0.82	0.1	0.0	0.002	0.004	0.0	0.0	0.0	0.0	0.0	2.0
	0.11	2.55	0.1	0.0	0.009	0.007	0.0	0.0	0.0	0.0	0.0	2.0
	0.31	1.29	0.1	0.0	0.003	0.002	0.0	0.0	0.0	0.0	0.0	2.0
	0.27	2.55	0.1	0.0	0.004	0.005	0.0	0.001	0.0	0.0	0.0	2.0

Figure 4 Clustering Results

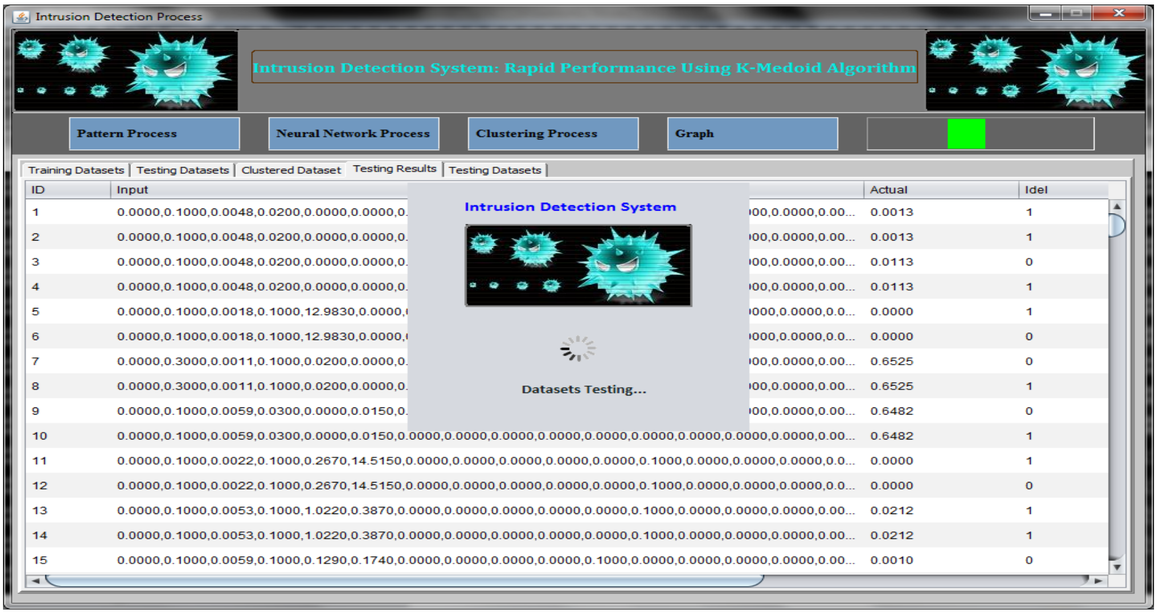
The figure 4 shows clustering results of proposed Pk-medoid method. The total number of un-clustered dataset size is 10000 and these records took very lesser time duration in clustered the dataset.



duration	protocol_t...	service	flag	src_bytes	dst_bytes	land	wrong_fra...	urgent	hot	num_faile...	logged_in	num_co
0.0	0.1	0.0048	0.02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.1	0.0048	0.02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.1	0.0048	0.02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.1	0.0048	0.02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.1	0.0018	0.1	12.983	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.1	0.0018	0.1	12.983	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.3	0.0011	0.1	0.02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.3	0.0011	0.1	0.02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.1	0.0059	0.03	0.0	0.015	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.1	0.0059	0.03	0.0	0.015	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.1	0.0022	0.1	0.267	14.515	0.0	0.0	0.0	0.0	0.0	0.1	0.0
0.0	0.1	0.0022	0.1	0.267	14.515	0.0	0.0	0.0	0.0	0.0	0.1	0.0
0.0	0.1	0.0053	0.1	1.022	0.387	0.0	0.0	0.0	0.0	0.0	0.1	0.0
0.0	0.1	0.0053	0.1	1.022	0.387	0.0	0.0	0.0	0.0	0.0	0.1	0.0
0.0	0.1	0.0059	0.1	0.129	0.174	0.0	0.0	0.0	0.0	0.1	0.0	0.0

Figure 5 Test Datasets

The figure 5 shows test datasets that are taken from the KDD dataset. The total number of record that are used in processing is 11,029 records. These records are tested with minimal time when compared with other frameworks.

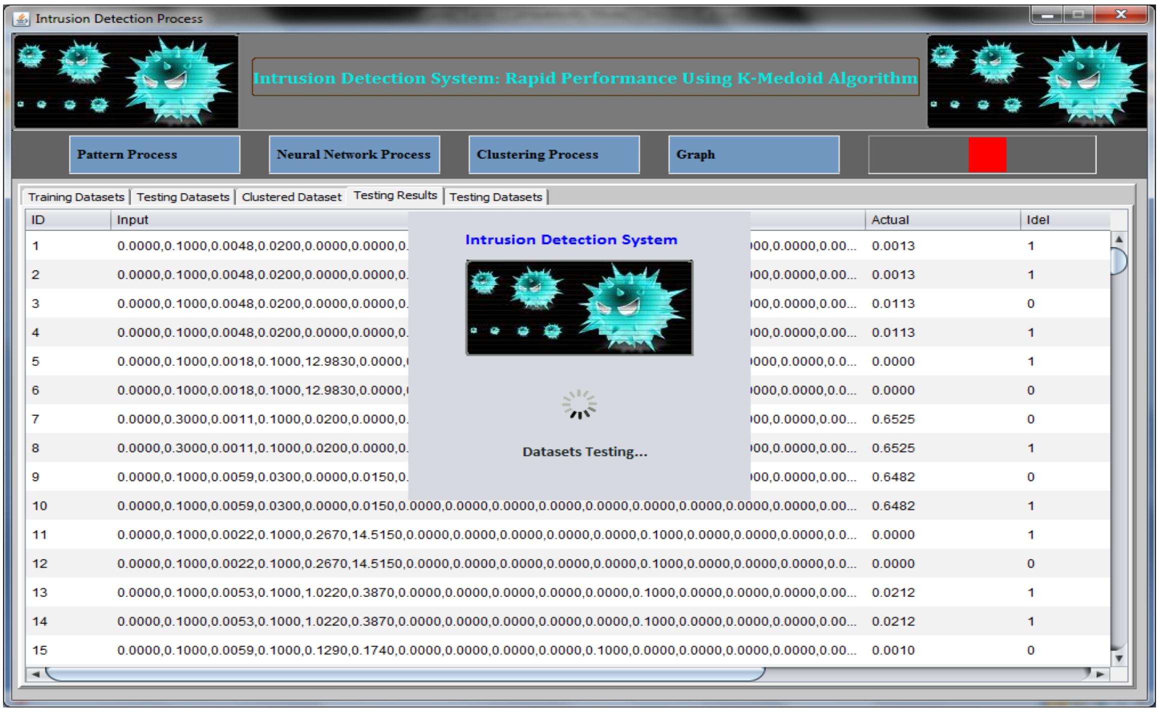


Intrusion Detection System

Datasets Testing...

ID	Input	Actual	Idel
1	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0. 00,0.0000,0.00...	0.0013	1
2	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0. 00,0.0000,0.00...	0.0013	1
3	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0. 00,0.0000,0.00...	0.0113	0
4	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0. 00,0.0000,0.00...	0.0113	1
5	0.0000,0.1000,0.0018,0.1000,12.9830,0.0000, 000,0.0000,0.0...	0.0000	1
6	0.0000,0.1000,0.0018,0.1000,12.9830,0.0000, 000,0.0000,0.0...	0.0000	0
7	0.0000,0.3000,0.0011,0.1000,0.0200,0.0000,0. 00,0.0000,0.00...	0.6525	0
8	0.0000,0.3000,0.0011,0.1000,0.0200,0.0000,0. 00,0.0000,0.00...	0.6525	1
9	0.0000,0.1000,0.0059,0.0300,0.0000,0.0150,0. 00,0.0000,0.00...	0.6482	0
10	0.0000,0.1000,0.0059,0.0300,0.0000,0.0150,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.6482	1
11	0.0000,0.1000,0.0022,0.1000,0.2670,14.5150,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0...	0.0000	1
12	0.0000,0.1000,0.0022,0.1000,0.2670,14.5150,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0...	0.0000	0
13	0.0000,0.1000,0.0053,0.1000,1.0220,0.3870,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.00...	0.0212	1
14	0.0000,0.1000,0.0053,0.1000,1.0220,0.3870,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.00...	0.0212	1
15	0.0000,0.1000,0.0059,0.1000,0.1290,0.1740,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0000,0.00...	0.0010	0

Figure 5 a) Testing Datasets



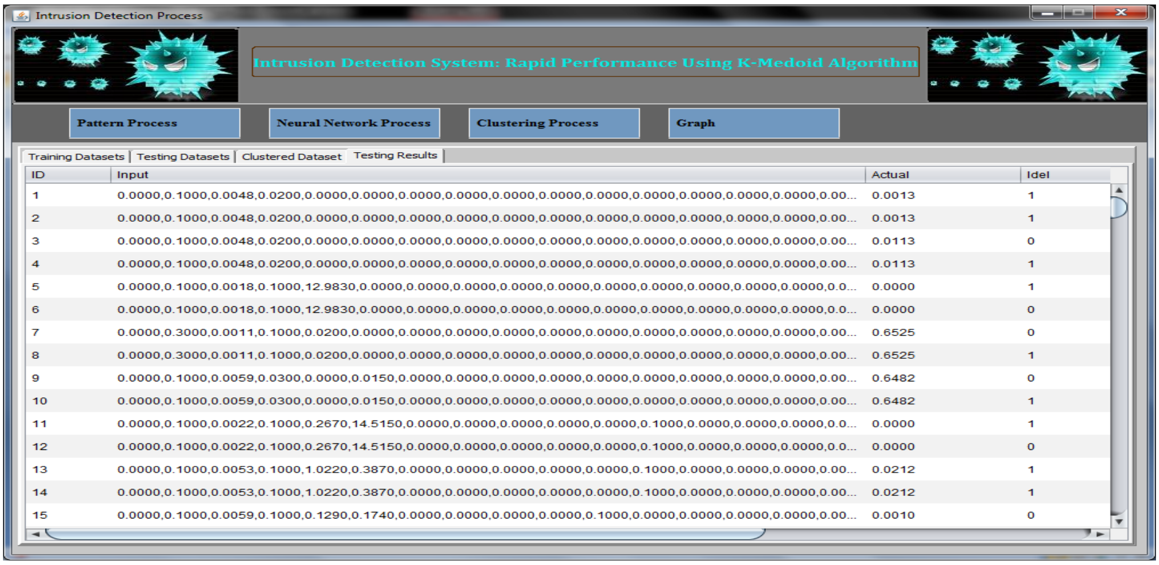
Intrusion Detection System

Datasets Testing...

ID	Input	Actual	Idel
1	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0. 00,0.0000,0.00...	0.0013	1
2	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0. 00,0.0000,0.00...	0.0013	1
3	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0. 00,0.0000,0.00...	0.0113	0
4	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0. 00,0.0000,0.00...	0.0113	1
5	0.0000,0.1000,0.0018,0.1000,12.9830,0.0000, 000,0.0000,0.0...	0.0000	1
6	0.0000,0.1000,0.0018,0.1000,12.9830,0.0000, 000,0.0000,0.0...	0.0000	0
7	0.0000,0.3000,0.0011,0.1000,0.0200,0.0000,0. 00,0.0000,0.00...	0.6525	0
8	0.0000,0.3000,0.0011,0.1000,0.0200,0.0000,0. 00,0.0000,0.00...	0.6525	1
9	0.0000,0.1000,0.0059,0.0300,0.0000,0.0150,0. 00,0.0000,0.00...	0.6482	0
10	0.0000,0.1000,0.0059,0.0300,0.0000,0.0150,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.6482	1
11	0.0000,0.1000,0.0022,0.1000,0.2670,14.5150,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0...	0.0000	1
12	0.0000,0.1000,0.0022,0.1000,0.2670,14.5150,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0...	0.0000	0
13	0.0000,0.1000,0.0053,0.1000,1.0220,0.3870,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.00...	0.0212	1
14	0.0000,0.1000,0.0053,0.1000,1.0220,0.3870,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.00...	0.0212	1
15	0.0000,0.1000,0.0059,0.1000,0.1290,0.1740,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0000,0.00...	0.0010	0

Figure 5 b) Testing Datasets

The figure 5 a & 5 b shows the intrusion dataset testing process where the green color shows the normal dataset which are been detected and the red color represents particularly about intrusion dataset.



ID	Input	Actual	Idel
1	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.0013	1
2	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.0013	1
3	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.0113	0
4	0.0000,0.1000,0.0048,0.0200,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.0113	1
5	0.0000,0.1000,0.0018,0.1000,12.9830,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0...	0.0000	1
6	0.0000,0.1000,0.0018,0.1000,12.9830,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0...	0.0000	0
7	0.0000,0.3000,0.0011,0.1000,0.0200,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.6525	0
8	0.0000,0.3000,0.0011,0.1000,0.0200,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.6525	1
9	0.0000,0.1000,0.0059,0.0300,0.0000,0.0150,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.6482	0
10	0.0000,0.1000,0.0059,0.0300,0.0000,0.0150,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00...	0.6482	1
11	0.0000,0.1000,0.0022,0.1000,0.2670,14.5150,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0...	0.0000	1
12	0.0000,0.1000,0.0022,0.1000,0.2670,14.5150,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0...	0.0000	0
13	0.0000,0.1000,0.0053,0.1000,1.0220,0.3870,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.00...	0.0212	1
14	0.0000,0.1000,0.0053,0.1000,1.0220,0.3870,0.0000,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.00...	0.0212	1
15	0.0000,0.1000,0.0059,0.1000,0.1290,0.1740,0.0000,0.0000,0.0000,0.0000,0.1000,0.0000,0.0000,0.0000,0.0000,0.00...	0.0010	0

Figure 6 Tested Results

The figure 6 shows the proposed framework tested results where the results are predicted with high accurately rate and the time taken for processing the entire dataset takes very minimal time compared with other existing techniques.

PK-MEDOID-HHNN FRAMEWORK IDS PERFORMANCE

Table 1: Memory Utilization for PK-Medoid-HHNN framework and other Existing approach

	Memory Utilization (MB)			
No.Of Records	Genetic Algorithm	Fuzzy Clustering Algorithm	Time Based-HHNN Approach	PK-Mediod - HNN Approach
2000	50	48	43	40
3000	52	48	42	40
5000	56	49	48	41
10000	71	50	48	43

The figure 7 shows the comparison of Memory Utilization for PK-Medoid-HHNN framework and other Existing approach such as genetic and fuzzy clustering for various sizes of datasets. From performance graph it is explored that proposed framework Memory Utilization is very less compared with other existing approaches.

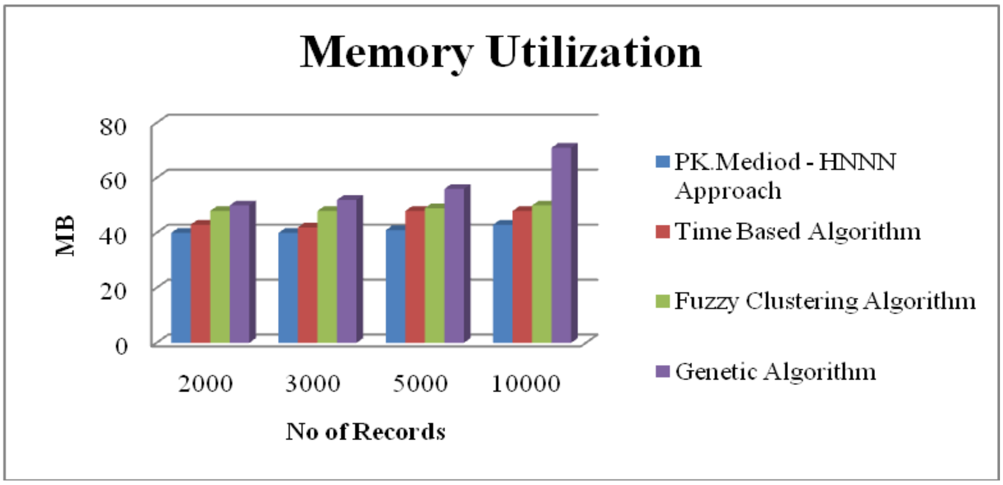


Figure 7 Comparison of PK-Medoid-HHNN framework and other Existing approach (Memory Utilization in MB)

The figure 8 shows Execution Time for Processing Different Record Set size by PK-Medoid-HNN approach and other Existing approaches. The proposed technique process data sets with less time duration rather than other Existing approach.

Table 2: Execution Time for Processing Different Record Set by PK-Medoid-HNN approach and other Existing approach

Execution time (in seconds)				
No.Of Records	Genetic Algorithm	Fuzzy Clustering	Time Based-HHNN Approach	PK.Mediod - HNN Approach
2000	3.984	1.967	1.767	1.467
3000	4.125	2.963	2.155	1.963
5000	6.324	4.734	4.111	3.734
10000	11.871	9.576	8.576	7.576

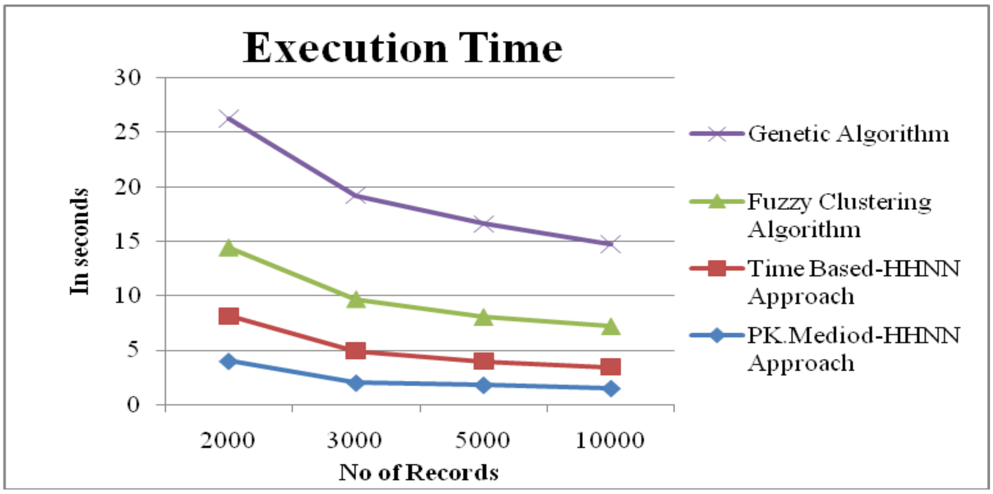


Figure 8 Comparison for Execution Time in Processing Different Record Set by PK-Medoid-HNN approach and other Existing approach

The figure 9 shows the accuracy of proposed work compared with other existing works like genetic algorithm and fuzzy clustering based approach. The PK-Medoid-HHNN framework accuracy is increased in linear order when training dataset size is also been increased.

The table 3 shows the various experimental results that are shown with proposed framework achieves better results rather than the other two frameworks whereas intrusion prediction accuracy is also increased when training dataset size is also increased. Finally prediction rates are measured by the following equation

Prediction rate = [(Number of normal or abnormal datasets)/Total Number of Tested datasets] X 100

Table 3: Intrusion Detection Rate

S. No	Training Dataset	Tested Dataset	Normal Data Prediction Rate	Anomaly Data Prediction Rate
1	2000	5000	46%	60%
2	7000	5000	76%	80%

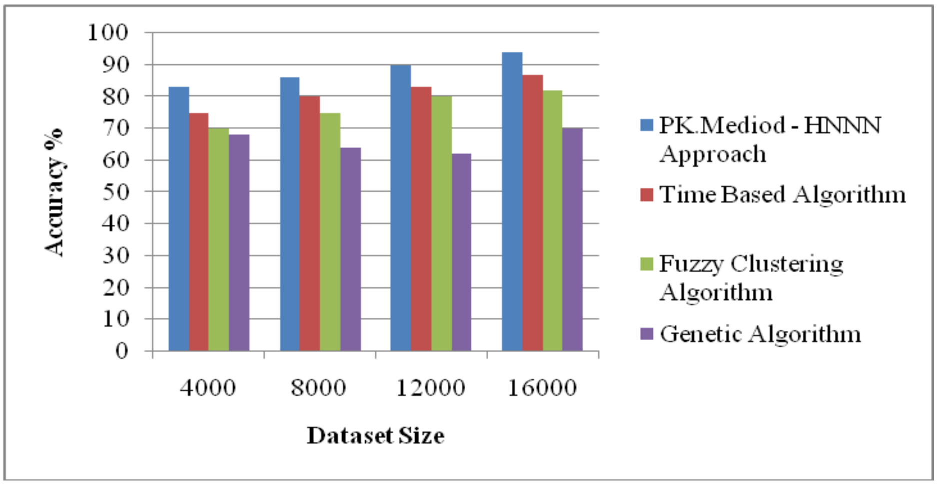


Figure 9 Comparison of PK-Medoid-HHNN framework and other Existing approach (Accuracy)

The different types of metrics are used for the purpose of checking performance and observing experimental results. Thus Detection Rate, Accuracy and False Alarm Rate are calculated to check the performance of proposed approach with other existing algorithms.

- The sensitivity or TPR (true positive rate) is described as positive fraction examples expected correctly through the model, i.e, TPR = TP / (TP + FN)
- The TNR (true negative rate) is described as the negative fraction examples expected correctly through the model, ie, TNR = TN / (TN + FP)
- FPR (False positive rate) is described as the negative fraction examples expected positive class model, ie, FPR = FP / (TN + FP)
- The FNR (false negative rate) is positive fraction examples expected as negative class. i.e, FNR = FN / (TP + FN

Table 4: Prediction of True positive Rate against various attacks

	True Positive Rate Prediction Ratio (%)			
Attack Types	Genetic Algorithm	Fuzzy Clustering Algorithm	Time Based-HHNN Approach	PK-Mediod - HHNN Approach
DoS	80	85	84	92
R2L	73	76	78	85
U2R	64	67	72	74
Probe	62	65	68	70

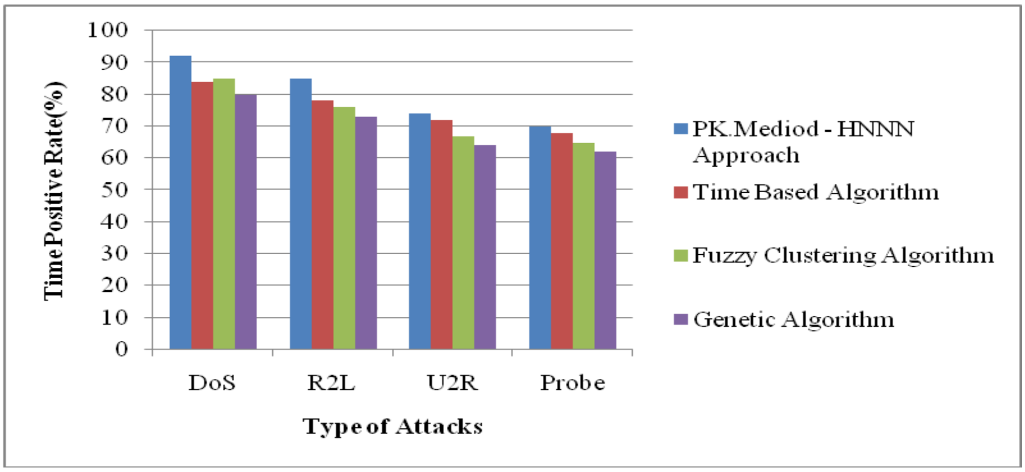


Figure 10 Comparison for Prediction of True positive Rate against various attacks

The figure 10 shows the comparison for Prediction of True positive Rate against various attacks. X – Axis parameters in this graph are for various attack prediction and Y – axis obtains the Rate of True Positive respectively for 41 attributes.

CONCLUSION

The existing security technologies lack more in security concerns. For the above reason intrusion detection system becomes a major part in network security. The main advantage of IDS is to reduce manpower monitoring.Also the system has an information security community which makes the system learns about new vulnerabilities and provide legal security solution for the system.

In this paper we have proposed a new approach for intrusion detection which is known as PK-Medoid-HNN. The Potential K-Medoid clustering method has various training set that splited into a number of homogenous subsets. Therefore the complexity of every sub training set is decreased with HHNN therefore detection performance improved. Experimental results used here is KDD dataset which shows efficiency of the new technique particularly for less-frequent attack detection, system on accuracy and detection stability. Inspired of many works that have been done till date, there are some unsolved problems that are as follows:

The selection of best algorithm for the analyzing the number of clusters because the set of clusters could be different in each dataset, so the selection of best method to find the specific number of clusters is an essential thing.

Classification of different type attacks is again a major problem. Despite defining the type of data as normal or anomaly it would be very useful if we can identify the type of attack represented by the data elements.

REFERENCES

[1] Ajith Abraham, Sandhya Peddabachigari, Johnson Thomas, Crina Grosan, (2007) “Modeling intrusion detection system using hybrid intelligent systems”, Journal of Network and Computer Applications volume 30, pages 114–132

[2] Kumar S. PhD thesis, (August 1995) “Classification and detection of computer intrusions” Department of Computer Science, Purdue University.

[3] Wang, G., et al. (2010), “A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering” Expert Systems with Applications doi:10.1016/j.eswa.2010.02.102

[4] Park, & Patcha, A., J. M. (2007) “An overview of anomaly detection techniques: Existing solutions and latest technological trends”, Computer Networks, volume 51(12),ages 3448–3470.

[5] Papavassiliou, S. & Manikopoulos, C., (2002). “Network intrusion and fault detection: A statistical anomaly approach”, IEEE Communications Magazine, volume 40(10), pages 76–82.

[6] Yen, E. & Wu, S., (2009). “Data mining-based intrusion detectors. Expert Systems with applications”, volume 36(3), pages 5605–5612.

[7] Ertoz, L., Dokas, P., Lazarevic, Tan, P. N. & A., Srivastava, J., (2002). “Data mining for network intrusion detection”. Proceeding of NGDM, pages 21–30.

[8] Schultz, E., Mellander, & J.Endorf, C., (2004). “Intrusion detection and prevention”, California: McGraw-Hill.

[9] Beghdad, R. (2008). “Critical study of neural networks in detecting intrusions”, Computers and Security, volume 27(5-6), pages 168–175.

[10] Hong, T., Han, I. & Joo, D., (2003). “The neural network models for IDS based on the asymmetric costs of false negative errors and false positive errors”, Expert Systems with Applications, volume 25(1), pages 69–75

[11] W. Li, (2004) “Using Genetic Algorithm for Network Intrusion Detection”. “A Genetic Algorithm Approach to Network Intrusion Detection”. SANS Institute, USA.

[12] R. H. Gong, P. Abolmaesumi, M. Zulkernine,(March-2012) “A Software Implementation of a Genetic Algorithm Based Approach to Network Intrusion Detection”, 2005. International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.2, page 119

[13] I. Abd-alghafar, B. Abdullah, A. Abd-alhafez, Gouda I. Salama, (2009) “Performance Evaluation of a Genetic Algorithm Based Approach to Network Intrusion Detection System”.

[14] Chetan Kumar, Anup Goyal, (2008) “GA-NIDS: A Genetic Algorithm based Network Intrusion detection System”.

[15] I. Traore, W. Lu, (2004) “Detecting New Forms of Network Intrusion Using Genetic Programming”. Computational Intelligence, volume. 20, page. 3, Blackwell Publishing, Malden, pages. 475-494.

[16] G. Qu,T. Xia, M. Yousif, S. Hariri, (2005) “An Efficient Network Intrusion Detection Method Based on Information Theory and Genetic Algorithm”, Proceedings of the 24th IEEE International Performance Computing and Communications Conference (IPCCC ‘05), Phoenix, AZ, USA..

[17] Topallar, M., Depren, O., Ciliz, M. K. & Anarim, E.,(2005). “An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks”, Expert Systems with Applications, volume.29(4), pages 713–722.

[18] Shyu, M. L., Chen, S. C., Sarinnapakorn, K., & Chang, L. (2003). “A novel anomaly detection scheme based on principal component classifier”, In: Proceedings of the IEEE foundations and new directions of data mining workshop (pages. 172–179).

[19] Alves-Foss, J. & Taylor, C.,(2001). ‘‘Low cost” network intrusion detection”, In: Proceedings of the new security paradigms workshop (pages. 1–15).

[20] Barbard, D., Jajodia, S. & Wu, N., (2001). “Detecting novel network intrusions using Bayes estimators”, In: Proceedings of the first SIAM international conference on data mining (pages. 1–17).

[21] Hwang, K. & Qin, M.,(2004). “Frequent rules for intrusive anomaly detection with Internet data mining” In: Proceedings of the 13th USENIX security symposium (pages.456–462).

[22] Gordeev, M. (2000), “Intrusion detection: Techniques and approaches”, <http:// www.gosecure.ca/SecInfo/library/IDS/ids2.pdf> (accessed March 2009).

[23] Horeis, T. (2003), “Intrusion detection with neural network – Combination of self organizing maps and redial basis function networks for human expert integration”, <http://ieee-cis.org/_files/EAC_Research_2003_Report_Horeis.pdf> (accessed March 2009).

[24] Mukkamala, S., Sung, A. & Janoski, G., (2002). “Intrusion detection using neural networks and support vector machines”,In: Proceedings of the IEEE international joint conference on neural networks (pages. 1702–1707).

[25] Cho, S. B. & Han, S. J., (2005), “Evolutionary neural networks for anomaly detection based on the behavior of a program” IEEE Transactions on Systems, Man and Cybernetics (Part B), volume 36(3), pages 559–570.

[26] Cho, S. B. & Han, S. J., (2005), “Evolutionary neural networks for anomaly detection based on the behavior of a program”, IEEE Transactions on Systems, Man and Cybernetics (Part B), volume 36(3), pages 559–570.

[27] Teresa F. Lunt, (Nov 1990), “IDES: An Intelligent System for Detecting Intruders”, Proceedings of the symposium: Computer Security, Threat and Countermeasures, Rome, Italy.

[28] Ronda R. Henning, Kevin L. Fox, Richard P. Sitnonian, Jonathan H. Reed, (July 1990) “A Neural Network Approach Towards Intrusion Detection”, Harris Corporation, Government Information Systems Division, P.O. Box 98000, Melbourne, FL 32902.

[29] Dr.B.Muthukumar, Jabez J,(September 2014) “Intrusion Detection System: Time Probability Method And Hyperbolic Hopfield Neural Network”, Journal of Theoretical and Applied Information Technology, Volume. 67 No.1