I.J. Modern Education and Computer Science, **2014, 11, 11-24**
Published Online November 2014 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2014.11.02

A Novel Classification Method Using Hybridization of Fuzzy Clustering and Neural Networks for Intrusion Detection

Saeed Khazaee
Engineering Department, Islamic Azad University, Chalous Branch, Iran.
Email: khazaee@iauc.ac.ir

Karim Faez
Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran.
Email: kfaez@aut.ac.ir

Abstract—In this paper, a hybrid classifier using fuzzy clustering and several neural networks has been proposed. With using the fuzzy C-means algorithm, training samples will be clustered and the inappropriate data will be detected and moved to another dataset (Removed-Dataset) and used differently in the classification phase. Also, in the proposed method using the membership degree of samples to the clusters, the class of samples will be changed to the fuzzy class. Thus, for example in KDD cup99 dataset, any sample will have 5 membership degrees to classes DoS, Probe, Normal, U2R, and R2L. Afterwards, the neural networks will be trained by new labels then using a combination of regression and classification methods, the hybrid classifier will be created. Also to classify the outlier data, a fuzzy ARTMAP neural network is employed which is a part of the hybrid classifier.

Evaluation of the proposed method is performed by KDDCup99 dataset for intrusion detection and Cambridge datasets for traffic classification problems. Our experimental results indicate that the proposed system has performed better than the previous works in the case of precision, recall and f-value also detection and false alarm rate. Also, ROC curve analysis shows that the proposed hybrid classifier has been better than the famous non-hybrid classifiers.

Index Terms—Intrusion detection system, fuzzy clustering, neural network, classification, regression

I. Introduction

Ever-increasing growth of computer networks and the emergence of electronic commerce in recent years have led to the point that computer security has become a high priority. In general, it can be said that internet traffic monitoring is a very important tool for network management. It helps operators to premier predict future traffic matrices and demands, security personnel to detect abnormal behavior, and researchers to develop more pragmatic traffic models [1].

Firewalls are a key part of intrusion prevention but a firewall protection is not enough for computer network security. Intrusion detection is another way to protect computer networks. Undoubtedly, research about methods to design an intrusion detection system that can detect intrusions in the network with the appropriate rate is essential. Intrusion detection systems can be roughly categorized into two major groups: (1) misuse detection and (2) anomaly detection. Misuse-based detection systems identify intrusions that are matched with known attack patterns. However, anomaly detection is an attempt to find for malicious behavior that deviates from established normal patterns. [2] Some IDSs combine qualities from two categories that are known as hybrid IDSs [3]. In this paper our interest is in misuse detection.

Up to now, several researches and various methods have been developed for the intrusion detection problem. However, there is a growing interest in intrusion detection community toward the application of machine learning techniques in this case [4-8]. Large volume of the data in the problem of intrusion detection makes the classical classification methods not to achieve the targets on the issue easily. As regards, various techniques of dimension reduction reduce the size of data and the complexity of the problem greatly so that the dimension reduction will be used in preprocessing of data. KDD CUP99 dataset is a common benchmark for the evaluation of intrusion detection techniques. In many researches which have used the data mining methods such as [2-8]; KDD Cup 99 dataset, it has been used for the implementation and evaluation of IDSs. Using all samples and all available features may have a negative effect in the training phase. Therefore, the preprocessing methods such as sampling and feature selection are the most important ways for improving the performance of intrusion detection.

Different methods based on fuzzy logic have led to increase in performance of the intrusion detection problems [8-13]. In this research, a novel classification method using hybridization of fuzzy clustering and neural network will be proposed for intrusion detection. Fuzzy clustering will be used in the fuzzy splitting stage then a hybrid classifier will be created. Also, fuzzy ARTMAP neural network is employed to classify outlier data that is a part of proposed hybrid classifier. In this paper, fuzzy logic has been used to intelligent splitting data and the novel classification method for intrusion detection.

In general, the paper has four important innovations: (1) Outlier detection using membership degree and similarity between the clusters and classes that we call it the fuzzy splitting (2) Using outliers in2 the training phase differently (3) Fuzzified of the class of samples and (4) hybridization of neural networks for regression then an algorithm that will be created a novel hybrid classifier. These proposed methods are related to each other sequentially.

The rest of this paper is organized as follows: related work on IDS will be discussed in Section 2. Section 3 will be included a preliminary for the paper. In Section 4 the proposed method will be explained which is included the preprocessing, the fuzzy splitting and converting the ordinal class to the fuzzy class for any sample. In that section with proposed method, the fuzzy splitting will be done in two phases: (1) using membership degree for outlier detection and (2) similarity between the clusters and classes. In the next step, with using the previous step, the class of any sample will be fuzzified although the final classification is not a fuzzy classification. In section5 the proposed method is going to be compared with some other methods and its performance is evaluated and finally, section 6 draws on conclusions.

II. Related Work

The most well-known method to detect intrusions is using verification data generated by operating systems and network capture tools. Since almost all activities are logged on a system, it is possible that a manual checkup of these logs would allow intrusions to be detected [12]. In the early stage, rule-based expert systems and statistical approaches are two common ways to intrusion detection. A rule-based IDS can detect many well-known intrusions with high detection rate, but it is difficult to detect new intrusions, and its signature database needs to be updated manually and frequently Statistical-based IDS, employs various statistical methods including principal component analysis, cluster and multivariate analysis, Bayesian analysis, and frequency and simple significance tests. But this type of IDS needs to collect enough data to build a complex mathematical model, which is impossible in the case of complicated network traffic. To solve the limitations of the mentioned methods, a number of data mining techniques have been introduced [2]. Even, using data mining techniques such as methods based on probabilistic approach [10], feature selection [11], artificial neural networks [12], genetic algorithm [13], association rules [1] and [3], and e.t has been caused of improvements of the classification and the process of detection. Neural networks have been extensively used to detect both misuse and anomalous patterns [12] and [14]. Also, clustering and fuzzy clustering are methods that used to preprocessing or classification phase in the many recent works [2] and [8]. In [2], authors proposed a new approach, called FC-ANN. The general procedure of FC-ANN was as follows: firstly fuzzy clustering algorithm was used to generate different training subsets. Thereafter, based on different training subsets, different artificial neural network models were trained to formulate different base models. Eventually, a meta-learner, fuzzy aggregation module, was used to aggregate these results. Data mining-based IDSs have the two drawbacks, lower detection precision for low-frequent attacks and weaker detection stability that was considered in [2]. The research has been much succeeded to detection of U2R and Normal classes but not successful in detection probe attacks.

However, using a method with hybridization of fuzzy clustering and neural networks which to be able to separates inappropriate samples than appropriate samples that uses both the samples separately and finally, it has led to better classifying was not provided in the previous researches.

III. Preleminary Concepts

A. Classic and fuzzy clustering

According to a given similarity or distance measure, clustering techniques work by grouping the observed data into clusters. In classic clustering, any input sample belongs to one and only one cluster and they can’t be members of two or more clusters, but when the similarity of a sample with two or more clusters are equal, the classic clustering for determining that the sample to which clusters belongs will have a trouble. Here, the main difference between the classic clustering and the fuzzy clustering is here that in the fuzzy clustering, an instance may be belonged to more than one cluster. The membership degrees between zero and one are used in the fuzzy clustering instead of the crisp assignments of the samples to the clusters. The clustering techniques can be divided into hard clustering techniques and soft clustering techniques [2]. The results of fuzzy clustering will be used in the next phases. Therefore, one of the popular soft clustering techniques, fuzzy c-means clustering has been used for fuzzy clustering module.

FCM is a method of the clustering which allows one sample of the data to be belonged to more than one cluster. The method is based on fuzzy approach and minimization of the following objective function [15-16]:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^{m} \left\| x_i - c_j \right\|^2 \quad , 1 \leq m < \infty \qquad (1)$$

where m is any real number greater than 1, uij is the degree of membership of xi in the cluster j, xi is the ith of d-dimensional measured data, cj is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center .[15-16]

As already indicated, the data are limited to any cluster by means of a Membership Function, which exhibits the fuzzy behavior of this algorithm. For this purpose, an appropriate matrix was built that named U whose factors are numbers between 0 and 1, and show the degree of membership between data and centers of clusters. The values of the matrix U can be any value between 0 and 1 but the total of the degrees of membership to all clusters must be equal to 1.

B. *MLP Neural Network*

A multilayer perceptron (MLP) is a feed-forward artificial neural network model that maps input data onto a set of proper outputs. An MLP consists of multiple layers of nodes that each layer fully connected to the next one in a directed graph. Besides for the input nodes, each node is a neuron as processing element with a nonlinear activation function. MLP utilizes a supervised learning technique called back-propagation for training the network [17-18]. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable [19]. Generally, the multilayer perceptron has an input and an output layer with one or more hidden layers as each node in one layer connects with a certain weight to every node in the following layer.

In this paper a typical perceptron network with one hidden layer will be used. The first layer contains the inputs, which in this problem are the features will be described in Section 4-1-3. The final layer contains the outputs, and in this problem these relate to the 5 classes (KDD CUP99 dataset experiments) and the 8 classes (Cambridge dataset experiments) of membership to which a flow may belong. Intervening layers are described as hidden. There may be any number of hidden layers, comprising any number of nodes. More information about MLP is in references [17-19].

C. *Fuzzy-ARTMAP Neural Network*

The fuzzy ARTMAP neural network (FAMNN) has been introduced by [20-21]. The FAMNN has been successfully applied in many tasks such as data mining, remote sensing, and pattern recognition [22].

This network achieves a synthesis of fuzzy logic and adaptive resonance theory (ART) neural networks by exploiting a close formal similarity between the computations of fuzzy method and ART category choice, resonance and learning [20]. It is composed of two fuzzy ART modules, ART_a and ART_b, interconnected by an inter-ART using an associative memory module as illustrated in Fig. 1. The inter-ART module has a self-regulator mechanism, match tracking, whose objective is to maximize the generalization and minimize the network error. The F_2^a layer is connected to the inter-ART module by the weights w_{jk}^{ab}.

1. *Input data*: The input patterns of ART_a is represented by the vector $a = [a_1 ... a_{Ma}]$ and the input patterns of ART_b is represented by the vector $b = [b_1 ... b_{Mb}]$.
2. *Parameters*: There are three basic parameters for the performance and training of fuzzy ART neural network [21].
   - The choice parameter, ($\alpha > 0$): the parameter acts on the category selection. [21]
   - Learning rate, ($\beta \in [0, 1]$): the parameter controls the velocity of network matching. [21]
   - Vigilance parameter, ($\rho \in [0, 1]$): it controls the network resonance. The vigilance parameter is responsible for the number of formed categories. [21]
3. *Algorithm structure*: After the resonance is confirmed in each network, J is the active category for the ART_a network and K is the active category for the ART_b network. If the active category on ART_a communicates to the desirable output vector presented to ART_b, the next step is match tracking to verify. The vigilance criterion is given by [21]:

$$\rho_{ab} = \frac{\left|y^b \wedge w_{JK}^{ab}\right|}{\left|y_b\right|} \tag{2}$$

4. *Learning*: After the input has completed the resonance state by vigilance criterion, the weight adaptation is implemented. The adaptation of the ART_a and ART_b module weights is given by [21]:

$$w_J^{\,new} = \beta\left(I \wedge w_J^{\,old}\right) + \left(1 - \beta\right) w_J^{old} \tag{3}$$

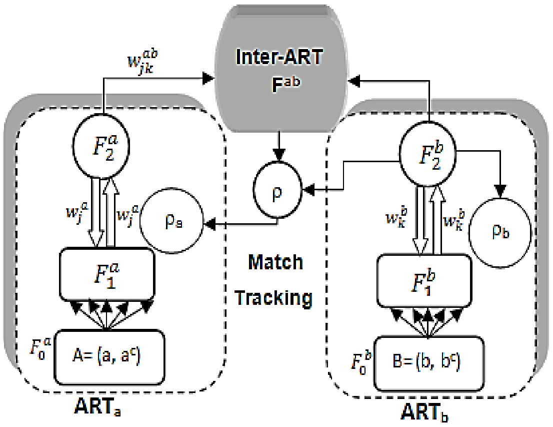


Fig. 1. Structure of the fuzzy ARTMAP [3]

IV. Proposed Framework

In this section, the proposed approach will be described for a new type of classification for intrusion detection.

A. *Preprocessing*

Traffic data generally consists of very high training samples and usually feature values in intrusion detection and traffic datasets are any type of discrete, continuous and symbolic. Range of values for some of these features is very large and diverse. Note that in these problems, training data has many different features; obviously, as a regular procedure the intrusion detection problem is a problem with high dimensions. Thus, the crude dataset is not very suitable for training an efficient system and training data must be preprocessed as well.

1. Primary Sampling

The sampling is the main method utilized for data selection. The sampling methods are classified as either probability or nonprobability. In probability samples, any member of the population has a known non-zero probability of being selected. Probability methods include the random sampling, stratified sampling, and systematic sampling. In nonprobability sampling, the data are selected from the dataset in some nonrandom manner [23]. Here, primary sampling is done with a probability method that is similar to the reference [2]. So, in the sampling phase 10% of KDD Cup99 in first experiment also 7 datasets for training and 3 datasets for test phase from Cambridge datasets in second experiment are selected. Table I shows number of the selected samples in KDD CUP99 and Table II shows sampled data from Cambridge datasets. In this paper, an intelligent sampling method will be proposed which is called the final sampling.

2. Normalization & Conversion

As regards, the features in KDD CUP99 and Cambridge datasets have various types: continuous, discrete, and symbolic with significantly varying resolutions and ranges. In most of the classification methods, the processing of data in the types is not possible. So, a preprocessing which can be applied in necessary conversions is required [24].

In conversion phase; symbolic-valued features are mapped to integer values ranging from 0 to S-1, where S is the number of symbols and range of the values is very different although continuous features having smaller integer value ranges. So, each of the mapped features are linearly scaled to the range by min-max normalization [0, 1]. [20]

3. Feature selection

Massive dataset which contains irrelevant and redundant features has a long time training or testing process, higher resource use as well as unsuitable detection rate[9] and [24]. So the performance of a pattern recognition system depends strongly on the employing of a feature-selection method [20]. Since the computational cost of system increases with rising number of features, feature selection seem to be necessary for reducing of complexities.

In most previous work, feature selection has been used to increase performance and reduce dimension of data in the classification problem [20] and [25-26]. KDD Cup99 data includes 41 different features and one label as a class. Also, each object in Cambridge datasets has 247 features that feature selection will be inevitable. In this study, as for necessity of feature selection, features with higher rank are selected by a method based on feature ranking. This is done by "Chi-Squared Feature Evaluation" that is one of feature ranking methods in *Weka* and 11 features is selected for KDD CUP99 and 7 features for Cambridge datasets. This method used in [9] and its performance has been appropriate.

Table 1. Sampled object in KDD CUP99

Class Number	ClassName	Number of Samples	
		Train	Test
1	DOS	10000	229853
2	Probe	4107	4166
3	R2L	1126	16189
4	U2R	52	288
5	Normal	3000	60593

Table 2. Sampled Object In The Cambridge Datasets

	Training Set							Test Set		
Flow Classes	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5	Data set 6	Data set 7	Data set 8	Data set 9	Data set 10
WWW	18211	18559	18065	19641	18618	16892	51982	51695	59993	54436
MAIL	4146	2726	1448	1429	1651	1618	2771	2508	3678	6592
ATTACK	122	19	41	324	122	134	89	129	367	446
P2P	339	94	100	114	75	94	116	289	249	624
DATABASE(DB)	238	329	206	8	0	0	36	43	15	1773
BULK	1511	1701	2736	600	928	521	484	551	1577	930
MULTIMEDIA(MM)	87	150	136	54	38	42	36	33	0	0
SERVICES(SRV)	206	220	200	113	216	82	293	220	337	212

B. Outlier detection based on fuzzy clustering (The fuzzy splitting)

As respects in this stage, by using fuzzy approaches, training data will be split into outlier data (*Removed-dataset*) and appropriate data therefore we call it the fuzzy splitting. The fuzzy c-means algorithm will be used for fuzzy clustering. Here, the number of clusters is considered equal to the number of the classes. So after running the algorithm on the data, 5 clusters are achieved. It should be noted that the clustering in software MATLAB (R2010a) has been done and samples have been clustered with the following command:

$$[center,U,obj\_fcn] = \mathrm{fcm}(data,\ n\_clusters) \qquad (4)$$

Clustering of the samples will be done regardless of their class. The variable *center* contains the coordinates of the two cluster centers, matrix *U* contains the membership degrees for each of the samples, and *obj_fcn* contains a history of the objective function across the iterations, *data* is the data which will be clustered and *n_clusters* is the number of clusters.

Hereinafter, the proposed system will described with KDD CUP99 dataset for example. The fuzzy splitting is performed in two stages with two different approaches:

1. Using membership degree for outlier detection

In this approach, with using matrix *U*, the samples which don't have appropriate membership degrees than any of clusters will be removed from the main dataset. Table III shows matrix *U* for 5 samples. In this table, S_1 to S_4 are 4 samples and C_1 to C_5 are membership degrees of the samples to the clusters. Note that the problem of intrusion detection is a classification problem; the samples must be sampled to improving of the classification. Because in this problem the clustering has been done with 5 clusters so can be said: each cluster is similar to a class. Therefore, if a sample has highest membership degree to a cluster compared to other clusters, it can be concluded that label of the sample must be equal to the same class of the cluster. However a topic is discussed what if the maximum of membership degree of a sample in clusters (C_i) has short difference with other membership degrees ($\{C_k: 1\leq k\leq 5,\ k\neq i\}$), class of the sample is not very similar to the cluster C_i. Obviously the samples which have this property are not good representatives for none of the classes. So these samples will have a negative impact in the pattern recognition process in final classifier. In table II, sample S_4 has this property. Maximum membership degrees in this table are marked with gray color. Maximum membership degree for S_4 is in C_4 that is equal to 0.3808. In the proposed approach, given that this number is smaller than 0.5, the S_4 will be removed from the original data and will move to other dataset that is called *Removed-Dataset.* Samples S_1, S_2 and S_3 with appropriate membership degree in C_4, C_5 and C_3, they will remain at this stage. Data in the *Removed-Dataset* will be unsuitable for the training process but it should be noted that such examples may exist in the test data and with removing of the samples from the training data, any pattern of these outlier samples will not be made for the final classifier. Thus *Removed-Data* will be collected and used separately.

After the above steps and removing some instances from training set, little change occurs in the training data, Table IV shows remaining samples with the breakdown of classes so that the greatest percent of decreasing of the data is in samples with class U2R.

Table 3. Matrix U For 4 Samples

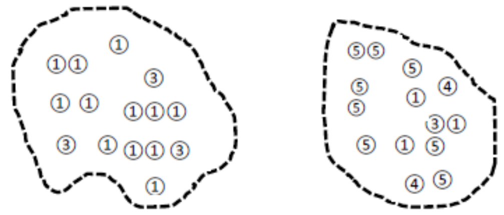
Clusters **Samples**	C_1	C_2	C_3	C_4	C_5
S_1	0.0025	0.0247	0.0140	0.9417	0.0171
S_2	5.2043×10^{-6}	5.6872×10^{-6}	4.4742×10^{-6}	7.5706×10^{-6}	0.9999
S_3	0.0411	0.0313	0.9019	0.0038	0.0219
S_4	0.1920	0.1851	0.2002	0.3808	0.0419

Table 4. Number Of The Moved Samples In First Step Of The Fuzzy Sampling

Class Number	ClassName	Number of Samples		Move to Removed-dataset
		Train	Test	
1	DOS	9957	229853	43
2	Probe	3868	4166	239
3	R2L	1075	16189	51
4	U2R	23	288	29
5	Normal	2928	60593	72

2. Using similarity between the clusters and classes

As also it was mentioned in the previous step, clustering has been done with 5 clusters (for KDD CUP99 dataset). It can be considered that any cluster would be similar to a class. At this stage, clusters are named; thus, the name of any cluster will be as the class of the same cluster. In this approach, if majority of samples in the cluster are the type of class *i* (Fig. 2) a cluster will be similar to class *i*. After the clusters were named, at this time, inappropriate samples should be removed. Any sample that its actual class is namesake with a cluster which sample is located in it will remain and other samples will be removed. For example, in Fig. 2 (a) samples with label "1" will remain and samples with labels "3" and "2" will be removed. Also in Fig 2 (b) samples with label "5" will remain only.



(a)The cluster is similar to class 1 (b)The cluster is similar to class 5

Fig. 2 Similarity between clusters and classes

Table V shows the number of final samples in the breakdown of their classes. Note that, in the previous step, the numbers of inappropriate data were excluded; data reduction from training set is lower. At this point, inappropriate data will be moved in to the *Removed -Dataset* also.

After the sampling was performed with the proposed approaches, in the next step of proposed method, the classes of instances will be converted to the fuzzy classes that these step is totally dependent on previous step.

Table 5. Number Of The Moved Samples After The Fuzzy Splitting

Class Number	**Class Name**	**Number of Samples**	**Move to Removed-dataset**	**Final Removed-dataset**
1	DOS	9944	13	56
2	Probe	3857	11	250
3	R2L	1064	9	60
4	U2R	17	6	35
5	Normal	2909	19	91

C. Fuzzification of Classes

Given that, the membership degree of each sample to the clusters is determined by matrix U also each cluster was labeled with DOS, probe, normal, r2l or u2r, each sample belong to DOS, probe, normal, r2l and u2r after clustering. In order that the samples bring up to fuzzified samples and these samples should belong to the all classes with specified membership degrees; after fuzzy clustering and final sampling, class of the sample will be replaced by membership degrees of the sample to the clusters as new features. For example, membership degree of a sample to cluster1 is membership degree of class *DoS*.

In this step and in the first training phase, instead of using the class of sample as a label and training of classifier, the membership degree of sample to different classes will be used to training of the neural networks for regression process. Thus according to Table VI, any sample in addition to the previous selected features, it has 5 new features which these features are the same membership degrees of the sample to different classes.

Table 6. New Features In The New-Dataset

ID	**Selected Features**			**Degrees of membership to classes**				
	Duration	. . .	dst_host_rerror_rate	μ_{DOS}	μ_{Probe}	μ_{R2L}	μ_{U2R}	μ_{Normal}
1	0.56	. . .	0.99	0.0025	0.0247	0.0140	0.9417	0.0171
2	0.12	. . .	0.34	5.2043×10^{-6}	5.6872×10^{-6}	4.4742×10^{-6}	7.5706×10^{-6}	0.9999
3	0	. . .	0.32	0.0411	0.0313	0.9019	0.0038	0.0219
4	0.41	. . .	0.87	0.0320	0.0353	0.8005	0.1307	0.0015

D. Using regression in the classification module

After creating of *new-dataset* as Table VI, training of proposed system will be done with a distinct approach. Fig. 3 shows how training by the samples *A* and *B*. Sample *A* is a sample that is remained in the fuzzy splitting and it is available in *new-dataset*. Sample *B* is a sample that is removed from training set and moved to *Removed-Dataset*. As mentioned before, the same samples of *A* and *B* will be used differently in training phase. So according to Fig. 3, sample *A* will be used to training of the 5 MLP neural networks. Inputs of neural networks will be 11 previous selected features and target will be one of the membership degrees to the 5 classes for every sample. For example, the first neural network that is named NN_{DOS} has 11 inputs and target value for this neural network is the membership degree of sample to class *DOS*.

Since, according to fuzzy c-means algorithm, sum of membership degrees of the sample to the clusters is equal with 1, so, it can be concluded:

$$\mu_{DOS}(A) + \mu_{Probe}(A) + \mu_{R2L}(A) + \mu_{U2R}(A) + \mu_{Normal}(A) = 1 \quad (5)$$

Therefore the target value for training of each MPL neural network is a number in the range [0, 1]. The regression output should have the value between 0 and 1 in the test phase, also. As Fig. 3 shows, the same samples of sample *B* will be given to a Fuzzy-ARTMAP neural network for training, separately. Inputs of this neural network are the same 11 features and target value is one of the classes *DOS*, *Probe*, *R2L*, *U2R*, *Normal*. So, the classification output will be one of the 5 classes.

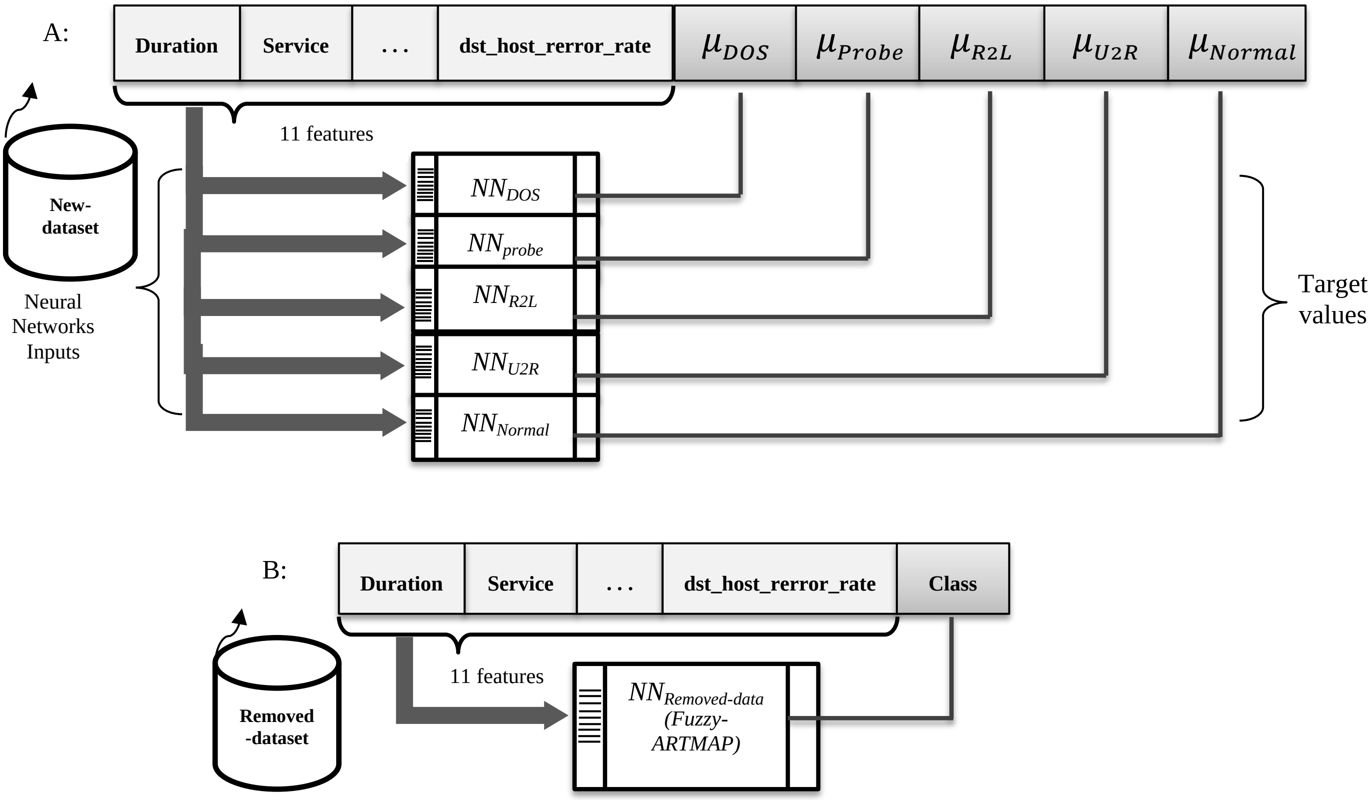
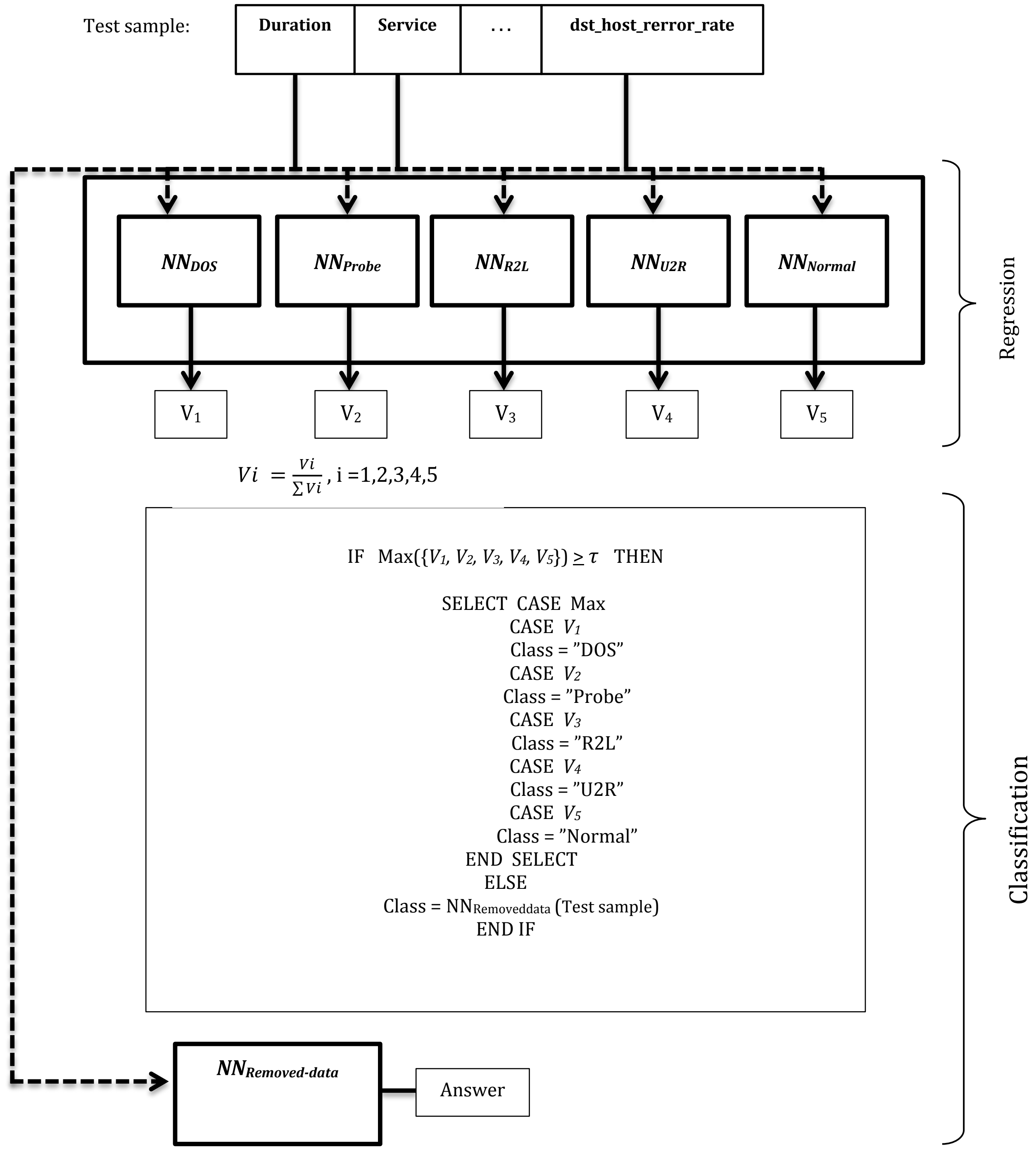


Fig. 3. Training of neural networks with New-Dataset and Removed-Dataset

After the training phase, the hybrid classifier uses regression for classification. In the regression step, any of training samples is given to 5 neural networks NN_{DOS}, NN_{Probe}, NN_{R2L}, NN_{U2R} and NN_{Normal} for calculating of membership degree of the sample to the 5 classes. Membership degree of every sample to 5 classes is a number in the range [0, 1] but here is not guarantee that the sum of answers of these 5 neural networks is equal to 1. This is Inconsistent with relation (5). To resolve this problem that is specified in Fig. 4, each output answer will be divided on the sum of the answers. In this case, sum of these membership degrees will be equal with 1. The obtained answers $\{v_1, v_2, ..., v_5\}$ will be sent to next step for classification phase. In the classification module, maximum value of v_1 to v_5 will be computed then it will be checked whether this value is greater than or equal to the τ or not.



```
IF  Max({$V_1, V_2, V_3, V_4, V_5$}) ≥ $\tau$  THEN
    SELECT CASE Max
        CASE $V_1$
            Class = "DOS"
        CASE $V_2$
            Class = "Probe"
        CASE $V_3$
            Class = "R2L"
        CASE $V_4$
            Class = "U2R"
        CASE $V_5$
            Class = "Normal"
    END SELECT
ELSE
    Class = $NN_{Removeddata}$ (Test sample)
END IF
```

Fig. 4. The proposed Hybrid method for intrusion detection

As it is clear in Fig.4:

- If i = {1, 2, 3, 4, 5} and Max (v_i) is greater than or equal to the τ then the class of sample is determined as follows (here the τ is obtained 0.6 experimentally):

  - If Max (v_i) = v_1 then the class will be *DoS*.
  - If Max (v_i) = v_2 then the class will be *Probe*.
  - If Max (v_i) = v_3 then the class will be *R2L*.
  - If Max (v_i) = v_4 then the class will be *U2R*.
  - If Max (v_i) = v_5 then the class will be *Normal*.

- If i = {1, 2, 3, 4, 5} and Max (v_i) is smaller than τ, the Class of sample is equal to the answer of Fuzzy-ARTMAP neural network and as described before, this neural network was trained by outliers data (*Removed-Dataset*).

Here it should be noted that each of the values v_1 to v_5 can be obtained by parallel processing. This shows that parallel processing will improve the speed of the proposed method because the value of each v_i has no effect on calculating of another v_i.

V. Experiments And Results

The experiments of this study were conducted in the environment of Microsoft Windows7 Ultimate using an IBM compatible computer with Intel(R), Core(TM) 2 Dou CPU 2.4 GHz and 2 GB of RAM. Also, the proposed method was implemented by MATLAB R2010a.

A. Datasets

1. KDD CUP99

In this paper, KDD CUP'99 dataset is used as the first experiment. The KDD CUP99 dataset, the most widely used dataset in the evaluation of intrusion detection, was selected. It was built based on the data produced from the 1998 DARPA Intrusion Detection Evaluation program ([10], [23] and [28-29]. This dataset is a set of network traffic data collected by the Information Systems Technology Group (IST) of MIT Lincoln Laboratory. In this work corrected KDD set is used because a dataset with different statistical distributions than either ''10% KDD'' or ''Whole KDD'' is provided by the ''Corrected KDD'' and is comprised of 14 additional attacks. Each connection contains 41 features and is labeled as either normal or an attack. Normal connections are created to profile that those expected in a military network and attacks fall into one of the following four categories namely Denial of Service (DoS), Remote to Local (R2L), User to Root (U2R) and Probe.

2. Cambridge datasets

For evaluation of the proposed classifier, another dataset aside KDD CUP99 is used. The Cambridge datasets are used that described originally in [30]. This data consists of descriptions of Internet traffic that have been manually classified. Hand-classification of two distinct days of data for an active Internet facility provides the input for sets of the training and testing phases [1] and [31]. The data was provided as a set of flows taken from two distinct days; each day consisted of ten sets of classified transport control protocol (TCP) traffic flows, with each object described by its membership class and a set of features. Each of the ten datasets covered the same length of time (approximately 24 min); these non-overlapping samples were spaced randomly throughout the 24-h period. The samples are intended to be representative of multiple times within the 24-h period. More information about of the classes, definitions of the sample of classification, and of the features of samples is explained later. Further details of the original hand-classification are given in [30-32], and the datasets themselves are described at length in [30]. In the present dataset, traffic was classified into common groups of network-based applications. Other approaches to classification may have fewer definitions, e.g., malicious versus non-malicious for an intrusion detection system, or may opt for protocol-specific definitions, e.g., the identification of specific applications or specific TCP implementations [1]. Table VII lists the classes of the dataset. In this research, class GAMES is selected because number of samples with the class was very low.

Table 7. Examples Of Network Traffic Allocated To Each Category (W. Moore and D. Zuev, 2005)

Classification	Example Application
BULK	ftp
DATABASE	postgres, sqlnet oracle, ingres
INTERACTIVE	ssh, klogin, rlogin, telnet
MAIL	imap, pop2/3, smtp
SERVICES	X11, dns, idnet, ldap, ntp
WWW	www
P2P	KaZaA, BitTorrent, GnuTella
ATTACK	Internet worm and virus attacks
GAMES	Microsoft Direct Play
MULTIMEDIA	Windows Media Player, Real

B. Evaluation Criteria

1. Standard basic metrics

To rank the different results, there are standard metrics that have been developed for evaluating network intrusion detection. True positives (Detection rate), true negatives, false positives (False alarm), and false negatives are often proposed to evaluate the intrusion detection system. A true positive indicates that the intrusion detection system detects precisely a particular attack having occurred. A true negative shows that the intrusion detection system has not made a mistake in detecting a normal connection. A false positive illustrates that a particular attack has been detected by the intrusion detection system but that such an attack did not occur actually. A false negative illustrates that the system is unable to detect the intrusion after a specific attack has occurred. However as the number of sample for the Probe, U2R, and R2L attacks in the training set and test set is every low, these metrics is not sufficient as a standard performance measure [2]. Hence, if these quantities use as a measure for testing the performance of the systems, it could be biased. So, we give the *precision*, *recall*, and *F-value* which are not dependent on the size of the training and the testing samples. They are defined as (6), (7) and (8) where *TP*, *FP* and *FN* are the number of true positives, false positives, and false negatives, respectively, and β corresponds to the relative importance of precision versus recall and is usually set to 1[12].

$$Precision = \frac{TP}{TP+FP} \tag{6}$$

$$Recall = \frac{TP}{TP+FN} \tag{7}$$

$$F - value = \frac{(1+\beta^2)*Recall*Precision}{\beta^2*(Recall+Precision)} \tag{8}$$

2. Receiver Operating Characteristic (ROC) analysis

ROC is another metric for comparing predicted and actual target values in a classification model. An ROC curve is a two-dimensional depiction of the accuracy of a signal detector, as it arises on a given set of testing data. Two dimensions are required to show the whole story of how the true positive rate decreases as the false positive rate increases. ROC curve has been commonly used in the field of IDS in order to exhibit tradeoff between detection rate and false alarm rate according to the change of internal thresholds [10].

C. Results and discussions

Before implementation of the proposed method, the essential preprocessing on the data has been done. Also, Table VIII shows the parameters and properties of the Fuzzy-ARTMAP in this problem. The number of hidden nodes was determined by formula $\sqrt{I+O}+\alpha$, $(\alpha = 1\ to\ 10)$ for MLP neural networks where I is the number of input node, O is the number of output node and α is random number [12].Considering the complexity of intrusion detection that in here α is equal to 10. So, the number of hidden layer in KDD CUP99 dataset experiments is $\sqrt{11+5}+10$ it is $\sqrt{7+8}+10$ in Cambridge dataset experiments. Therefore the number of hidden layer for the both dataset experiments is 14.

In KDD CUP99 dataset experiments; according to Table IX, it is specified that the best value for τ is 0.6. For specifying of this control parameter, impact of the parameter has been reviewed in Detection rate, Accuracy and false alarm rate. Table IX shows that the proposed system with τ =0.6 has a very good effect than other values in improving especially to False Alarm Rate. In Table X performance of the proposed method with τ =0.6 has been reported. Given that in this paper, outlier data has been used in training of Fuzzy-ARTMAP for classifying of unusual samples therefore effect of this data in the training phase must be reviewed. As in Fig.4 is specified, a counter has been considered for calculating of the number of samples which are correctly and incorrectly classified by Fuzzy-ARTMAP neural network. In Table X, these counters have been reported. As is illustrated by this table, the ratio of correctly classified than incorrectly classified samples was much appropriate in Probe and U2R. However, the results have been compared with BPNN, and other well-known methods such as decision tree, na ïve Bayes. These three techniques were run with the help of the *Weka* Data Mining tool (Witten, I. H., & Frank, E., 2005) and also, FC-ANN that proposed in [12].Table XI shows that the proposed method in measures Precision, Recall and F-value is far more efficient than other methods. As is illustrated by Table XI, the proposed method gets the highest precision, recall and F-value than other methods especially in Probe and R2L although this isn’t very good in U2R verses FC-ANN.

After evaluating of the system by precision, recall, and f-value, the proposed system has been evaluated by *Detection Rate*, *Accuracy*, and *False Alarm Rate*. In Table XII, the proposed method has been compared with the three previous methods that these criteria are available in the papers. Accordingly, Table XII shows that the proposed method has been more successful than other methods although the method of reference [3] is better than the proposed method in *DR* in Normal only.

In Cambridge datasets experiments; the results in Table XIII indicate that the proposed hybrid classifier has been better than TCSA [30], Bayesian Analysis Technique [31] and Bayesian neural network [1] in measures *Detection Rate* and *Accuracy*. In these methods, *False Alarm Rate* not reported but it reported for the proposed method. Although Bayesian neural network [1] has been partially better than the proposed method in classify of ATTACK and P2P however, in general, reported results illustrate that the hybrid classifier has been the most successful in the classification.

Evaluating of the hybrid classifier using Cambridge datasets shows that the proposed method can be efficiently for massive datasets and traffic classification. Also, Cambridge dataset was chosen because of the similarity of an intrusion detection dataset therefore due to the similarity, It can be concluded the proposed hybrid classifier will be useful for other (in) famous datasets.

Table 8. Properties And Parameters Of The Fuzzy-ARTMAP Neural Network

Number of output layer units	**400**
Number of Epochs	100
Choice Parameter (α)	0.01
Learning Rate (β_a)	0.5
Learning Rate (β_b)	0.5
Vigilance Parameter (ρ_a)	0.97
Vigilance Parameter (ρ_b)	0.99

Table 9. Impact Of The Control Parameter τ In Performance Of The Proposed Method

Metric		τ = 0.4	τ = 0.5	τ = 0.6	τ = 0.7
Detection Rate (%)	Normal	99.5	99.5	99.5	99.1
	Probe	97.1	98.6	98.6	98.6
	DoS	99.9	100	100	100
	R2L	60.9	60.8	60.9	60.9
	U2R	39.3	28.6	39.3	25.0
Accuracy (%)		97.7	97.8	97.8	97.7
False Alarm Rate (%)		0.6	0.5	0.2	0.9

Table 10. Performance Reports Of The Proposed Method. (Experiments On KDD CUP99)

Class	TP Rate	FP Rate	Precision	Recall	F-Value	Counter (True)	Counter (False)
Normal	0.995	0.023	0.911	0.995	0.951	207	390
Probe	0.986	0.003	0.837	0.986	0.905	133	27
DoS	1	0	1	1	1	323	730
R2L	0.609	0.001	0.979	0.609	0.751	1200	60
U2R	0.393	0	1	0.393	0.564	11	146

Table 11. Comparing Of The Proposed Method With Other Methods In Terms Of Precision, Recall And F-Value. (Experiments On KDD CUP99)

Classes	Measurement	Methods				
		Decision Tree	Na ïve Bayes	BPNN	FC-ANN	Proposed
Normal	Precision	0.912	0.892	0.897	0.913	0.911
	Recall	0.994	0.977	0.982	0.991	0.995
	F-value	0.951	0.933	0.938	0.950	0.951
Probe	Precision	0.500	0.526	0.609	0.481	0.837
	Recall	0.781	0.881	0.887	0.800	0.986
	F-value	0.609	0.659	0.723	0.601	0.905
DoS	Precision	0.998	0.997	0.998	0.999	1
	Recall	0.972	0.996	0.972	0.967	1
	F-value	0.985	0.981	0.985	0.983	1
R2L	Precision	0.333	0.461	0.571	0.932	0.979
	Recall	0.014	0.086	0.057	0.586	0.609
	F-value	0.027	0.146	0.104	0.719	0.751
U2R	Precision	0.500	0.250	0.500	0.833	1
	Recall	0.154	0.077	0.231	0.769	0.393
	F-value	0.235	0.118	0.316	0.800	0.564

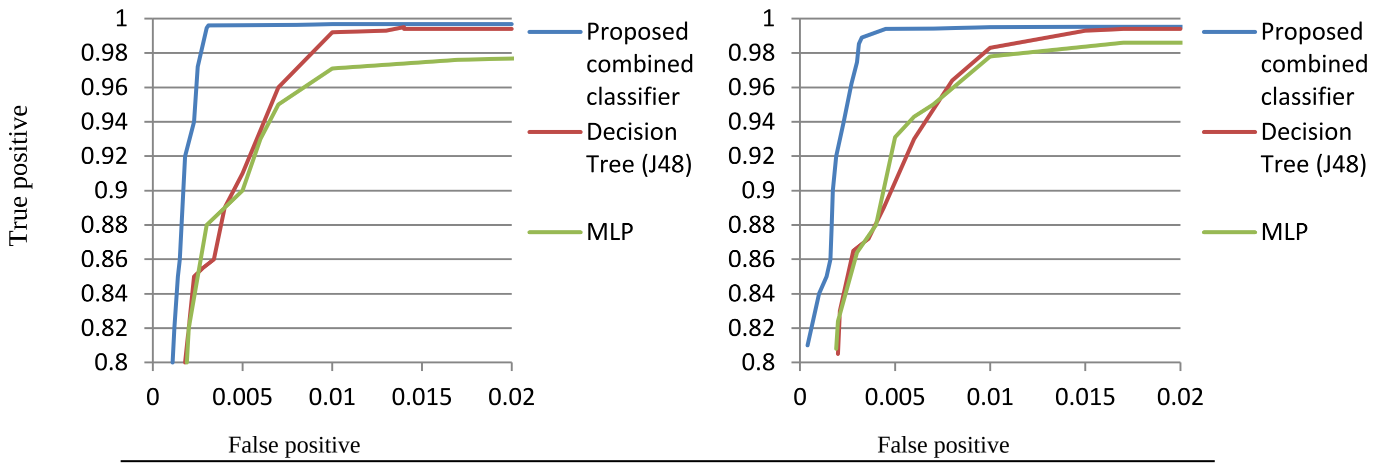
Table 12. Comparing Of The Proposed Method With Other Methods In Terms Of DR, Accuracy And FAR.(Experiments On KDD CUP99)

Method Metric		ESC-IDS	Hierarchical Clustering and support vector machines	GA-optimized FARM-based feature selector + GA-optimized Fuzzy ARTMAP	Proposed Method
Detection Rate (%)	Normal	98.2	99.3	99.9	99.5
	Probe	84.1	97.5	86.3	98.6
	DoS	99.5	99.5	99.8	100
	R2L	31.5	28.8	60.2	60.9
	U2R	14.1	19.7	17.6	39.3
Accuracy (%)		95.3	95.7	97.2	97.8
False Alarm Rate (%)		1.9	0.7	0.2	0.2

Table 13. Comparing Of The Proposed Method Previous Methods In Terms Of DR, Accuracy And FAR.(Experiments On Cambridge Dataset)

Method Metric		TCSA	Bayesian Analysis Technique	Bayesian neural network	Proposed Method
Detection Rate (%)	WWW	65.97	99.27	99.8	100
	MAIL	56.85	90.69	99.6	96.4
	BULK	89.26	89.76	97.4	95.67
	ATTACK	58.08	13.46	68.6	62.1
	P2P	45.59	36.45	62.0	46.8
	DATABASE	20.20	86.91	97.6	100
	MULTIMEDIA	59.45	80.75	67.0	92
	SEVICES	91.19	63.68	96.0	97.7
Accuracy (%)		83.98	93.73	99.3	99.5
False Alarm Rate (%)		Not reported	Not reported	Not reported	0.45

Finally, the proposed hybrid classifier compared with 2 famous classifiers on equal terms. So, the ROC curves are illustrated in Fig. 5. Fig. 5 shows that the proposed hybrid classifier has a better detection capability than MLP and J48 decision tree. It is clear in part (a) and (b); the proposed method has been succeeded in KDD CUP99 and Cambridge dataset.



a. Experiments on KDD CUP99

b. Experiments on Cambridge datasets

Fig. 5. ROC curve for proposed classifier, decision tree J48 and MLP.

VI. Conclusion

In this paper, the new approach was proposed to intrusion detection and traffic classification. The proposed classifier uses the hybridization of fuzzy clustering and neural networks. This method has high performance in terms of precision, recall, f-value, Detection Rate, False Alarm Rate, accuracy, and ROC curve analysis than the other works. According to, after preprocessing and the fuzzy splitting, the training dataset converted to two datasets: (1) the dataset which is contained of the suitable samples for training (2) the dataset which was contained outliers that were named *Removed-Dataset*. Outlier data weren't deleted and they were used differently in training phase with new approach. Also, 5 MLPs have been used as the 5 regression models in the proposed method which, finally, they were used in classification module. In classification module also, a Fuzzy-ARTMAP neural network was used and it trained by *Removed-Dataset*. In this paper as obtained results can clearly be seen that using of the outlier data was successfully able to generate a model with the desired characteristics for the unusual samples that their existence is inevitable in the test data. So, with generating of this model, the system performance has been increased. Also, parallel processing can be used to improving of speed of the proposed method.

Experimental results showed which the proposed method performed better in terms of *DR, False alarm, precision, recall* and *f-value* in most comparisons than other previous works. In future research, how to select the more appropriate classifiers instead of MPL for different classes remains an open problem.

References

[1] Tom Auld, Andrew W. Moore, Stephen F. Gull (2007). “Bayesian Neural Networks for Internet Traffic Classification”, IEEE Transactions On Neural Networks, volume 18, Issue 1, pp. 223-239.

[2] G. Wang, J. Hao, J. Ma and L. Huang (2010). "A new approach to intrusion detection using artificial neural networks and fuzzy clustering", Expert Systems with Applications, Volume 37, Issue 9, pp. 6225-6232.

[3] M. Sheikhan and M. Sharifi Rad,(2010). "Misuse detection based on feature selection by fuzzy association rule mining", World Applied Sciences Journal, 10 (Special Issue of Computer & Electrical Engineering), pp. 32- 40.

[4] S.Y. Wu and E. Yen, (2009). "Data mining-based intrusion detectors", Expert Systems with Applications, Volume 36, Issue 3, Part 1, pp. 5605-5612.

[5] E. Lundin, E. Jonsson, (2000). ” Anomaly-based intrusion detection: privacy concerns and other problems”, Computer Networks 34 (4), pp. 623–640.

[6] C.M. Chen, Y.L. Chen and H.C. Lin, (2010). "An efficient network intrusion detection", Computer Communications, Volume 33, Issue 4, pp. 477-484.

[7] Sheikhan M, Jadidi Z, Farrokhi A (2012) Intrusion detection using reduced-size RNN based on feature grouping. Neural Computing and Applications 21:1185–1190

[8] S.J. Horng, M.Y. Su, Y.H. Chen, T.W. Kao, R.J. Chen, J.L. Lai and C.D. Perkasa, (2011). "A novel intrusion detection system based on hierarchical clustering and support vector machines", Expert Systems with Applications, Volume 38, Issue 1, pp. 306-313.

[9] Saeed Khazaee, Mohammad Saniee Abadeh, (2011). “A Hybrid Model Based on Feature Extraction for Network Intrusion Detection”, Journal of computing, Volume 3, Issue 9, pp.65-72, New York.

[10] Seongjun Shin, Seungmin Lee, Hyunwoo Kim, Sehun Kim, (2013). ” Advanced probabilistic approach for network intrusion forecasting and detection”, Expert Systems with Applications, Volume 40, Issue 1, pp. 315–322.

[11] W. Li, J.L. Wang, Z.H. Tian, T.B. Lu and C. Young, (2009). "Building lightweight intrusion detection system using wrapper-based feature selection mechanisms", Computers & Security, Volume 28, Issue 6, pp. 466-475.

[12] D. Fisch, A. Hofmann and B. Sick, (2010). "On the versatility of radial basis function neural networks: A case study in the field of intrusion detection", Information Sciences, Volume 180, Issue 12, pp. 2421-2439.

[13] M. Saniee Abadeh, J. Habibi and C. Lucas, (2007). "Intrusion detection using a fuzzy genetics-based learning algorithm", Network and Computer Applications, Volume 30, Issue 1, pp. 414-428.

[14] Cannady J. "Artificial neural networks for misuse detection", (1998). National information systems security conference, p. 368–81.

[15] J. C. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithms, (1981)." Plenum, New York.

[16] J.C. Bezdek, R. Ehrlich, W. Full, (1984). "FCM: The fuzzy c-means clustering algorithm", Computers & Geosciences, Volume 10, pp. 191-203.

[17] Rosenblatt, Frank. x., (1961). Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington DC.

[18] Rumelhart, David E., Geoffrey E. Hinton, and R. J. Williams, (1986). "Learning Internal Representations by Error Propagation". David E. Rumelhart, James L. McClelland, and the PDP research group. (editors), Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1: Foundations. MIT Press.

[19] Cybenko, G., (1989)." Approximation by superpositions of a sigmoidal function", Mathematics of Control, Signals, and Systems, 2(4), 303–314.

[20] G.A. Carpenter, (2003). "Default ARTMAP", In Proceedings of the International Joint Conference on Neural Networks, Volume 2, pp. 1396–1401.

[21] G.A. Carpenter, S. Grossberg, N. Markuzon, J.H. Reynolds, D.B. Rosen, (1992). "Fuzzy ARTMAP: a neural network for incremental supervised learning of analog multidimensional maps", IEEE Transactions on Neural Network, Valume 3, Issue 5, pp. 689-713.

[22] Xu-sheng Gan, Jing-shun Duanmu, Jia-fu Wang, Wei Cong, (2013). "Anomaly intrusion detection based on PLS feature extraction and core vector machine", Knowledge-Based Systems, Volume 40, pp. 1–6.

[23] Lohr, Sharon L. Sampling, (1999): Design and analysis. Duxbury. ISBN 0-534-35361-4.

[24] H.T. Nguyen, K. Franke and S. Petrovi'c, (2010). "Towards a generic feature-selection measure for intrusion detection", International Conference on Pattern Recognition, ISSN: 1051-4651, pp. 1529-1532.

[25] Zainal, M.A. Maarof and S.M. Shamsuddin, (2007). "Feature selection using Rough-DPSO in anomaly intrusion detection", Lecture Notes in Computer Science, Computational Science and its Applications, Volume 4705, Part I, pp. 512–524.

[26] Porto-D'ıaz, D. Mart'ınez-Rego, A. Alonso-Betanzos and O. Fontenla-Romero, (2009). "Combining feature selection and local modelling in the KDD Cup 99 dataset", Lecture Notes in Computer Science, Artificial Neural Networks, Volume 5768,pp. 824–833.

[27] John Zhong Lei, Ali A. Ghorbani, (2012). "Improved competitive learning neural networks for network intrusion and fraud detection", Neurocomputing, Volume 75, Issue 1, Pages 135–145.

[28] MIT Lincoln Laboratory (2000). DARPA intrusion detection scenario specific datasets. <http://www.ll.mit.edu/mission/communications/ist/CST/index.html>.

[29] B. Kavitha, Dr. S. Karthikeyan, P. Sheeba Maybell, (2012). "An ensemble design of intrusion detection system for handling uncertainty using Neutrosophic Logic Classifier", Knowledge-Based Systems, Volume 28, pp.88- 96.

[30] W. Moore and D. Papagiannaki, (2005). "Toward the accurate identification of network applications," in Proc. 6th Passive Active Meas. Workshop (PAM), vol. 3431, pp. 41–54.

[31] W. Moore and D. Zuev, (2005). "Internet traffic classification using Bayesian analysis techniques," in Proc. ACM Sigmetrics, pp. 50–60.

[32] W. Moore and D. Zuev, (2005). Discriminators for use in flow-based classification, Intel Research Tech. Rep.

Saeed Khazaee Was born in Nowshahr, Mazandaran, Iran. He received his BSc. Degree in 2007 and MSc degree in 2011. He is a faculty member of the Computer Engineering department (science board) at Islamic Azad University of Chalous, Iran.

He is working on many projects in the field of data mining and he has built a good connection with industry and has applied the data mining techniques for real projects. In addition, he has a track record of producing research papers and has published them in journals and conferences that are directly related to Machine Learning and Data Mining approaches and he has held many expert seminars and workshops for faculty members of universities and other data mining fanatics. Generally, he really likes researching and working with special teams related to Data Mining, Pattern Recognition, Machine learning, Optimization.

Karim Faez Was born in Semnan, Iran. He received his BSc. degree in Electrical Engineering from Tehran Polytechnic University as the first rank in June 1973, and his MSc. and Ph.D. degrees in Computer Science from University of California at Los Angeles (UCLA) in 1977 and 1980 respectively.

Professor Faez was with Iran Telecommunication Research Center (1981-1983) before Joining Amirkabir University of Technology (Tehran Polytechnic) in Iran in March 1983, where he holds the rank of Professor in the Electrical Engineering Department.

He was the founder of the Computer Engineering Department of Amirkabir University in 1989 and he has served as the first chairman during April 1989-Sept. 1992.

Professor Faez was the chairman of planning committee for Computer Engineering and Computer Science of Ministry of Science, Research and Technology (during 1988-1996).

His research interests are in Biometrics Recognition and authentication, Pattern Recognition, Image Processing, Neural Networks, Signal Processing, Farsi Handwritten Processing, Earthquake Signal Processing, Fault Tolerance System Design, Computer Networks, and Hardware Design.

Dr. Faez coauthored a book in Logic Circuits published by Amirkabir University Press.

He also coauthored a chapter in the book: *Recent Advances in Simulated Evolution and Learning*, Advances in Natural Computation, Vol. 2, Aug.2004, and World Scientific.

He published about 300 articles in the above area. He is a member of IEEE, IEICE, and ACM, a member of Editorial Committee of Journal of Iranian Association of Electrical and Electronics Engineers, and International Journal of Communication Engineering. Emails: kfaez@aut.ac.ir, kfaez@ieee.org, kfaez@m.ieice.org.