Implementation of Fuzzy C-Means and Dempster-Shafer Theory for Anomaly Intrusion Detection

P.Srinivasu[1], P.S.Avadhani[2], and Tummala Pradeep[3]

[1] Associate Professor, Department of CSE, Anil Neerukonda Institute of Technology and Sciences
Visakhapatnam
[2] Professor, Department of CS&SE, Andhra University
Visakhapatnam
[3] IVth Year CSE Student, Department of CSE, Birla Institute of Technology, Ranchi
Jharkhand

Summary:
Fuzzy clustering technique and Dempster-Shafer theory both have merit of resolving the uncertainty problems raised by limited and ambiguous information or data during a decision process. Also, the *k*-NN technique is applied to speed up the detection process. Intrusion detection in fact is a classification task that classifies network traffics into normal usage category or attack category. In our work, the main goal is to identify *U2R* and *R2L* attacks from the *KDD99* intrusion detection benchmark data set. For successfully achieving the goal, we divide the development of an intrusion detection system into two phases: training phase and classification phase. In the training phase, decision rules are generated in accordance with the clustering result of provided training data. The rules are used for classifying future network traffic whether is a normal activity or an attack in the classification phase.

Key words:
Fuzzy clustering, Dempster-Shafer, KDD 99, Anomaly detection

1. Introduction

Intrusion Detection:

In an internet connected world, Systems and Networks are prone to different attacks. The increasingly frequent attacks on internet visible systems are attempts to breach information security requirements [1]. First step towards securing the system from these security breaches is to detect the attacks before it turns out to be severe. For this purpose we use intrusion detection systems which help the computers to prepare from and deal with attacks.
Intrusion detection helps in extending the security management capabilities of system administrators to include security audit, monitoring, attack detection and response.
Intrusion detection techniques are of two types namely

1. Anomaly based intrusion detection and
2. Misuse based intrusion detection.

Anomaly based intrusion detection technique is based on the assumption that any event/interaction which is not normal is an intrusion. An abnormal pattern is defined as any packet whose attributes are different from the expected values which occur normally.
Misuse detection techniques are those where known attempts can be detected or tracked. Initially all the known attack patterns are specified and any incoming packet matching with any of these predefined packets is considered to be an intrusion. Some of the popular techniques under this category are rule based system, which is represented using fact and rule bases.
Both Anomaly and Misuse based systems have their own limitations. Misuse systems fail to detect novel patterns as they have a predefined database. While Anomaly based systems can detect novel attacks that have abnormal patterns but number of false positives and false negatives i.e. the possibility of wrongly classifying an attack and normal pattern is high. In misuse there is very little chance of wrong classification except the incapability of detecting novel attacks. Hence our focus is on developing an anomaly based intrusion detection systems (IDS) using soft-computing techniques such as Neural networks and Fuzzy logic in order to reduce the number of false positives and false negatives and to improve the accuracy of the system in classifying the attacks and normal patterns. Intrusions are actions that attempt to bypass security mechanisms of computer systems and there are any set of actions that threatens the confidentiality, integrity and availability of network resource.

Manuscript received September 5, 2011
Manuscript revised September 20, 2011

2. Uncertainty

Uncertainty happens due to lack of knowledge or unpredictable factors. Uncertainty is classified into two categories basing on their fundamental differences in nature: aleatory uncertainty and epistemic uncertainty.
Aleatory uncertainty is also known as variability, random uncertainty, stochastic uncertainty, objective uncertainty, and irreducible uncertainty [6],[7]. It is caused by inherent random variations associated with the physical system or the environment under consideration. The occurrence of an event is not predicable even a large quantity of past data is collected.
Epistemic uncertainty is an uncertainty that is due to a lack of knowledge or information of processes of the system or the environment. Since it is not caused by the inherent random variations of the system but by the incomplete information or knowledge, the uncertainty is possible to be reduced by including new knowledge or information about the system or environmental factors. This uncertainty is also referred to as imprecision, reducible uncertainty, subjective uncertainty, parameter uncertainty, model form uncertainty, and state-of-knowledge uncertainty [6],[7]. Epistemic uncertainty does happen in intrusion detection tasks.

3. Dataset Description

Initially, KDD Dataset is used to train the system. This is given as input to the system where the attributes useful for detecting the behavior are selected in the feature selection module.

KDD Cup Data set:

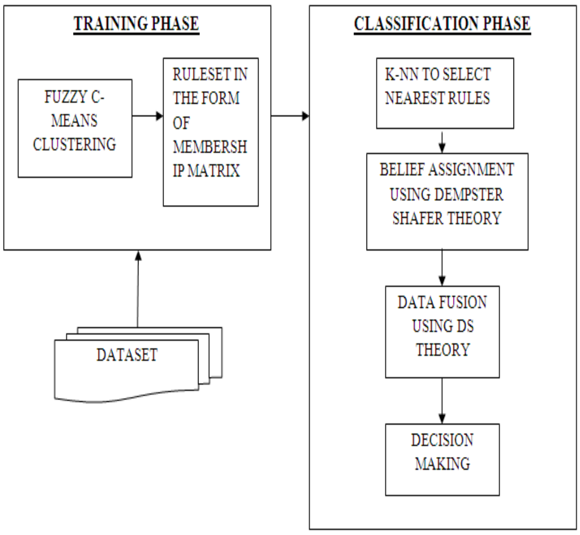
This is the data set used for The Third International Knowledge Discovery and Data Mining Tools Competition, which was held in conjunction with KDD-99, The Fifth International Conference on Knowledge Discovery and Data Mining[12]. Every record in the dataset represents an access to the system which has 41 attributes that represents the behavior of the access and classifies it as Normal or Attack. The 41 attributes used to describe the behavior of the access can be grouped into three broad categories as follows:

1. Basic features of Individual TCP connection
2. Content features within a connection suggested by domain knowledge
3. Traffic features computed using a two-second time window

In this module twelve attributes are selected and hence only the data regarding these attributes is enough for the system to function properly (reduced dataset).

4. Proposed Architecture

The proposed architecture contains various modules each defined with a specific purpose and connected together to yield an end result.


TRAINING PHASE
FUZZY C-MEANS CLUSTERING
RULESET IN THE FORM OF MEMBERSHIP MATRIX
DATASET
CLASSIFICATION PHASE
K-NN TO SELECT NEAREST RULES
BELIEF ASSIGNMENT USING DEMPSTER SHAFER THEORY
DATA FUSION USING DS THEORY
DECISION MAKING

Architecture of our proposed system

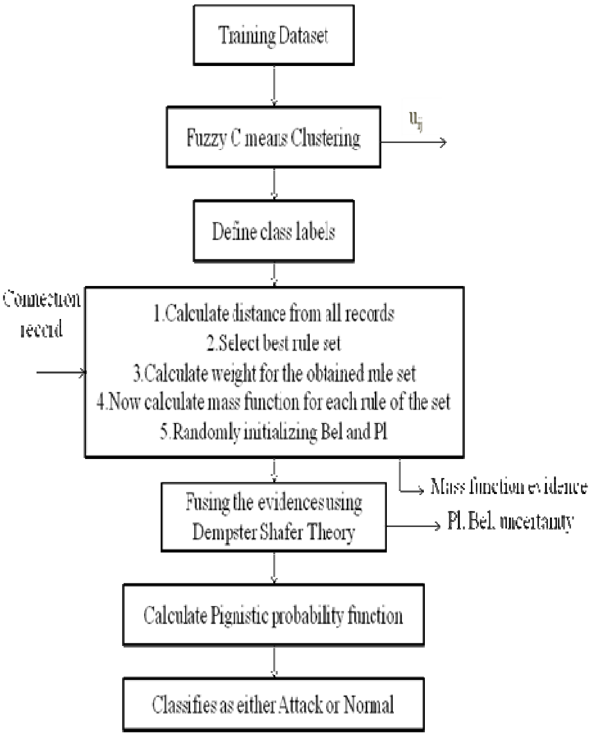
Intrusion detection in fact is a classification task that classifies network traffics into normal usage category or attack category. In our work, the main goal is to identify *U2R* and *R2L* attacks from the *KDD99* intrusion detection benchmark data set. For successfully achieving the goal, we divide the development of an intrusion detection system into two phases: training phase and classification phase. In the training phase, decision rules are generated in accordance with the clustering result of provided training data. The rules are used for classifying future network traffic whether is a normal activity or an attack in the classification phase. First and the foremost module is the Training Data Acceptance module wherein a set of records from previous interactions and knowledge are given to the system which acts as sample training data.

The second module is the Rule Set module. Here the decision rules are generated in accordance with the clustering result of provided training data. The rules are used for classifying future network traffic whether is a normal activity or an attack in the classification phase.

The third module is Belief Assignment module. Here the rules which are classified will act as pieces of evidence to assign beliefs to an incoming connection in the decision making stage. The weighted k-NN (k-Nearest Neighbor) rule is used to assign different weights to the selected rules.

The fourth and the important module is the Data Fusion module where Dempster Shafer Theory is used. It computes the probability that evidences support the attack or normal class. Here two independent evidences can be fused into a single belief function *Z* that expresses the support of the hypotheses in both evidences.
The fifth module is the Decision Making module where Pignistic probability function is applied to decide to which class an incoming record belongs to.

Flow of events:


Training Dataset
Fuzzy C means Clustering
u_{ij}
Define class labels
Connection record
1.Calculate distance from all records
2.Select best rule set
3.Calculate weight for the obtained mle set
4.Now calculate mass function for each rule of the set
5.Randomly initializing Bel and Pl
Mass function evidance
Fusing the evidences using Dempster Shafer Theory
Pl. Bel. uncertainty
Calculate Pignistic probability function
Classifies as either Attack or Normal

4.1 Training Phase[11]

In the training phase, decision rules are generated in accordance with the clustering result of provided training data. The rules are used for classifying future network traffic whether is a normal activity or an attack in the classification phase.

Let N be the number of traffic connections in the training, and each of them is composed of n distinct features with positive numeric values. Let T denote the training set, the training traffic connection be x, and the set of features in each connection be F. T and Fare are denoted as follows.

$$T = \{x_1, x_2 \ldots x_N\} \quad (1)$$

and

$$F = \{f_1, f_2 \ldots f_n\} \quad (2)$$

A training traffic connection sometimes could not be crisply defined as normality or abnormality. The boundary between normal activities and abnormal ones are always unclear. Crisp clustering algorithms cannot handle this ambiguity problem among network activities. Therefore, we decide to apply fuzzy c- Means (FCM) clustering technique [2],[3]. It allows one piece of data with gradual memberships to the clusters rather than completely assigning to just one cluster. By using this feature of FCM, the problem of ambiguity between attacks and normal activities can be solved. The connection could be assigned to diverse classes with different degrees of memberships. We denote the set L as a number of p possible classes.

$$L = \{l_1, l_2 \ldots l_p\} \quad (3)$$

The clustering procedure is done by using iterative optimization technique to minimize an objective function J.

$$J = \sum_{i=1}^{N} \sum_{j=1}^{p} u_{ij}^{\sigma} \left\| x_i - c_j \right\|^2 \quad (4)$$

where the parameter σ is a weighting exponent on each fuzzy membership and has a value in the range [1, ∞). This parameter determines the amount of fuzziness in the classification process. When it is set to 1, the FCM approaches a hard c-Means algorithm, i.e., the membership grade assigning to cluster is either 0 or 1. As this parameter becomes larger, the fuzzier are the membership assignments to the clusters. Also, convergence of the algorithm tends to be slower as the value of σ increases. Normally, its value is in the range of 1.25 to 2[8]. x_i is the i^{th} connection of the training set, c_j is the center of cluster j, and u_{ij} is the membership grade of xi in the cluster j with a value between 0 and 1. || || denotes norm expressing the distance between any measured data and the cluster center. The membership grades u_{ij} and cluster centers c_j are updated by the following expressions (5) and (6),

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{\sigma} x_i}{\sum_{i=1}^{N} u_{ij}^{\sigma}} \quad (5)$$

and

$$u_{ij} = \frac{1}{\sum_{k=1}^{p}\left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|}\right)^{\frac{2}{\sigma - 1}}} \qquad (6)$$

By iteratively updating the cluster centers and the membership grades for each training connection, FCM moves the cluster centers gradually to their correct values. Finally, the iteration stops when

$$\max_{ij}\left|u_{ij}^{(k+1)} - u_{ij}^{(k)}\right| < \varepsilon \qquad (7)$$

where ε is a selected threshold for terminating the iteration process and k denotes the number of iterations.

The connection that lies "closer" to the center of a class has a higher membership grade to that class. On the contrary, the connection that lies "farther" away from the center of a class has a lower membership grade to that class. Training connections are grouped into p classes such that each connection has a certain membership grade to every class. The set of cluster centers C and membership partition matrix U are shown below.

$C= \{c_1, c_{2,\ldots\ldots}\ c_p\}$ (8)

$U= \{u_{i1}, u_{i2,\ldots\ldots}\ u_{ip}\}$ (9)

where i is the connection number of the training set and p is the number of possible classes. For each cluster center, it has a number of n values.

Within each row of U, the p membership grades are treated intuitively to be our degrees of confidence on p classes that a connection can belong to. Consequently, we can build p decision rules from a connection and each consists of a number of feature values F, a class label l, and a confidence value α.

$$R_U = \{r_U\} \text{ where } r_U : \langle F, l\rangle, \alpha \qquad (10)$$

The confidence values are in proportion to the correspondent membership grades that a connection belongs to certain classes. For a training connection, only portion of our belief is devoted to a certain class in a rule whereas the rest of beliefs are committed to other classes in other rules. The summation of the degrees of confidence on rules that generated from a training connection must be equal to 1. It is not possible that the connection can belong to any other classes except these *p* classes.

$$\sum_{j=1}^{p} \alpha_{ij} = 1 \qquad (11)$$

where i is the connection number and j is the class number. Since the training set has N connections and each contains a number of p membership grades, totally N times p decision rules can therefore be generated.

In addition to the rules created from membership partition matrix U, a number of p rules are generated from the cluster centers. In each rule, the antecedent part includes n values of a cluster center and the corresponding class label. The degree of confidence is designated to 1 because we have full confidence that the cluster center should belong to that partitioned class without any doubt.

$$R_U = \{r_C\} \text{ where } r_C : \langle C, l\rangle, \alpha = 1 \qquad (12)$$

So totally (N+1)p rules are included in the decision rule set R. These rules will act as pieces of evidence to assign beliefs to an incoming connection in the decision making stage.

$$R = R_U \cup R_C \qquad (13)$$

4.2 Classification Phase

Dempster-Shafer theory is used in this phase. It computes the probability that evidences support the attack or normal class. It is suitable for anomaly detection on unseen network traffic by using limited information on the uncertainty. With the combination of accumulative evidences from an insufficient amount of information, it is capable of making decision on traffic whether it is normality or abnormality. In this phase, the pieces of evidences will be derived from the decision rules of the training phase. The k-NN technique is used to speed up the detection process.

k-Nearest Neighbor Rule

Let v be any incoming traffic connection. The decision rules generated in the Training phase are used for classifying future network traffic whether is a normal activity or an attack in this classification phase. These set of decision rules are considered as pieces of evidence that alters our degrees of belief to which class v should belong while classifying it into the correct class. If the distance is large between v and a decision rule, it represents that v is "far" from the rule, i.e., the rule only has a little influence on v. On the other hand, we have stronger belief that v should belong to the same class of the rule if v is "close" to it, which means the distance has a smaller value. Here, distances from v to all decision rules are computed and the most informative rules are selected. Additionally, the weighted k-NN rule [8],[9] is used to assign different weights to the selected rules.

$$w_i = \begin{cases} \frac{d(x_k,v)-d(x_i,v)}{d(x_k,v)-d(x_1,v)} & d(x_k,v) \neq d(x_1,v) \\ 1 & d(x_k,v) = d(x_1,v) \end{cases} \quad (14)$$

where xi is the ith rule, x_k and x_1 are the farthest and nearest rule of v, respectively, and d is the Euclidean distance between v and a rule. This weighting factor is used to give each decision rule a different amount of influence in a way that closer rule to v has larger influence. The factor is calculated such that the nearest neighbor of v has a weight value of 1 and the farthest kth neighbor has a value of 0. Since the range of this factor is from 0 to 1, the resulting weights possibly have very similar values.

Dempster Shafer Theory

Dempster Shafer Theory is also called as Evidence theory or Theory of Belief Functions [4][5].This theory defines a sample space named frame of discernment (or simply frame), which is a finite set of mutually exclusive and exhaustive hypotheses in a problem domain under consideration. we identify the set of class labels L as the frame of the problem domain. The possible subset A of L represent hypothesis that one could present evidence. The set of all possible subsets of L, including itself and the null set ∅, is called a power set and designated as 2^L. Assume v be an incoming traffic connection to be classified. To classify v means to assign it to one of the members in L, i.e., to assign v to a member of p classes: $v \in l_q$, q = 1, 2, …, p.

A piece of evidence that influences our degree of belief on a hypothesis can be quantified by a mass function which is denoted as m(·). It is a mapping function and defined as m: $2^L \rightarrow [0, 1]$ such that

$$\sum_{A \subseteq L} m(A) = 1 \quad (15)$$

And

$$m(\emptyset) = 0 \quad (16)$$

where A⊆L is called a focal element of m if m(A) > 0. The quantity m(A) is defined as the hypothesis A's basic probability assignment. It can be interpreted as the portion of total belief to hypothesis A given the available evidence.

For further differentiating the rules' degree of importance to v, the confidence value α is added to alter the degree of our belief on v.

$$m(l_q) = w_i . \alpha_i \quad (17)$$

where i is the rule number and q is the corresponding class number of the ith rule. Up to this stage, each rule creates a number of belief assignment indicating the degrees that v belongs to certain classes. If the value of m is large, it means that we have a strong belief that v belongs to the class of which m indicates. Otherwise v should belong to other classes if m is small. Nevertheless, we need to notice that a belief should also be designated to the frame (with every class labels). The reason is that only part of our beliefs is committed to single class for a given training connection, and the rest of our belief should be assigned to the frame. According to Dempster-Shafer theory, the summation of all mass functions inferred from one training connection is equal to 1. Thus, the belief belonged to the frame becomes one minus the summation of beliefs of all of the single class.

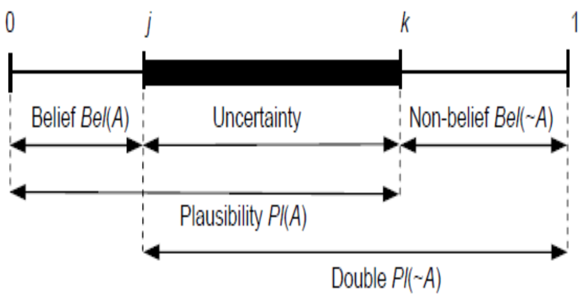
$$m(L) = 1 - \sum_{i=1}^{p} m_i(l_q) \quad (18)$$

From the mass function given in equation (18), the belief function Bel and plausibility function Pl can be derived to characterize certain hypotheses. They are shown in the following equations (19) and (20),

$$Bel(l_j) = m(l_j) \quad (19)$$

$$Pl(l_j) = 1 - Bel(\bar{l_j}) \quad (20)$$

where j is class number and $\bar{l_j}$ is the hypothesis "not lj" with value between 0 and 1. Belief function is a measure of the total amount of belief that directly supports for a given hypothesis. The greater the support assigns to a hypothesis, the higher belief that the hypothesis is true. It can be regarded as a lower bound that indicates the impact of evidence of the hypothesis. Plausibility quantifies the extent to which one doubts the hypothesis. It shows the belief on the given hypothesis can only up to this value, which is an upper bound on the belief. The gap between them indicates the uncertainty about the hypothesis. It is a good reference [10] in deciding whether more evidences are needed or not.



Functions of Belief and Plausibility

Generally speaking, the mass function is a piece of evidence that supports certain hypothesis concerning to the class member of a rule. When more evidences appear with same class label, those evidences can be integrated to generate a single belief function which represents the total support for the same class. Dempster Rule of Combination is applied here to combine all the beliefs induced from distinct pieces of information with same class label together. Using this combination rule, the final belief on every subset of class set can be obtained. In our case, a number of belief functions for single classes and one belief function for the class set will be generated.

Now assume that there are two mass functions m1 and m2 induced by distinct items of evidence X and Y. By using Dempster Rule of Combination, these two independent evidences can be fused into a single belief function Z that expresses the support of the hypotheses in both evidences. The combination result is called orthogonal sum of m1 and m2 and noted as in equation (21).

$$m = m1 \oplus m2 \qquad (21)$$

$$(Z) = \frac{\sum_{X\cap Y=Z} m_1(X).m_2(Y)}{\sum_{X\cap Y\neq\emptyset} m_1(X).m_2(Y)} = \left(\sum_{X\cap Y=Z} m_1(X).m_2(Y)\right).k^{-1} \qquad (22)$$

where

$$k^{-1} = \left(\sum_{X\cap Y\neq\emptyset} m_1(X).m_2(Y)\right)^{-1}$$

$$=\left(1 - \sum_{X\cap Y=\emptyset} m_1(X).m_2(Y)\right)^{-1} \qquad (23)$$

where the factor k^{-1} is the renormalization constant. Using the above equations, the final belief on single class and the frame are obtained. In an intrusion detection task, a number of p belief functions for single classes and one belief function for class set will be generated.

By using Dempster Rule of Combination, the above evidences can be aggregated into two fused belief functions Bel(N) and Bel(A). First, the renormalization constant factor k^{-1} is calculated then, individual fused mass functions can be obtained.

Decision Making

At the data fusing level, each piece of evidence initializes the finite amount of belief to hypotheses of the frame. Part of the belief is allocated to the single class and part of it is allocated to the frame. To decide which class v should belong to, the following equation (24) shows the **Pignistic probability function** and it is applied to make the final decision.

$$Bp(l_q) = m(l_q) + \frac{m(L)}{p} \qquad (24)$$

where q is the class number and p is the number of classes. The function quantifies our beliefs to individual classes with Pignistic probability distribution. These probabilities distributed from zero to one and the summation of them equals to one. For making an optimal decision, v is assigned to a class with the highest Pignistic probability.

Consider two pieces of evidences whose mass functions are 0.15vand 0.2 for normal class and attack class respectively.

EXAMPLE:

Let m_1(N)=0.15 and m_1 (A)=0.2

By using equations

Bel $(l_j) = m(l_j)$ and Pl $(l_j) = 1 - Bel(\bar{l}_j)$

We have

Bel (N) = m_1(N) = 0.15

Bel (A) = m_1(A) = 0.2

Pl (N) = 1- $Bel(\bar{N})$ = 1 - Bel (A) = 1 - 0.2 = 0.8

Pl (A) = 1- $Bel(\bar{A})$ = 1 - Bel (N) = 1 – 0.15 = 0.85

Un (N) = Bel (N) - Pl (N) = 0.8-0.15 = 0.65

Un (A) = Bel (A) - Pl (A) = 0.85-0.2 = 0.65

Let m_2(N)=0.25 and m_2(A) = 0.7

	m_1(N) = 0.15	**m_1(A) =0.2**	**m_1(N,A) = 0.65**
m_2(N) = 0.25	m(N) = 0.04	m_1(N∩A) = 0.05	m(N) = 0.16
m_2(A) = 0.7	m_1(N∩A) = 0.11	m(A) = 0.14	m(A) = 0.46
m_2(N,A) = 0.05	m(N) = 0.01	m(A) = 0.01	m_1(N∩A) = 0.03

$$k^{-1} = \left(\sum_{N\cap A\neq\emptyset} m_1(N).m_2(A)\right)^{-1} = \left(1 - \sum_{N\cap A=\emptyset} m_1(N).m_2(A)\right)^{-1}$$

$= [1 - \{ m_1(N).\ m_2(A) + m_1(A).\ m_2(N) \}]^{-1}$

$= [1- \{0.11 + 0.05\}]^{-1}$

$= (1 – 0.16)^{-1} \quad = (0.84)^{-1} = 1.19$

$m(l_q) = [m_1(l_q)\ m_2(l_q)+ m_1(l_q)\ m_2(L)+ m_1(L)\ m_2(l_q)]k^{-1}$

$m(N) = [m_1(N)\ m_2(N)+ m_1(N)\ m_2(N,A)+ m_1(N,A)\ m_2(N)]k^{-1}$

$= [0.04+0.01=0.16]1.19 = (0.21)1.19 = 0.25$

m(A) = $[m_1(A)\ m_2(A)+\ m_1(A)\ m_2(N,A)+\ m_1(N,A)\ m_2(A)]k^{-1}$
= [0.14+0.01+0.46]1.19 = (0.61)1.19 = 0.04
m (N,A) = $[m_1(N,A)m_2(N,A)]k^{-1}$
= (0.03)1.19 = 0.04

Bel (N) = m (N) = 0.25
Bel (A) = m (A) = 0.73
Pl (N) = 1- $Bel(\overline{N})$ = 1 - Bel (A) = 1 – 0.73 = 0.27
Pl (A) = 1- $Bel(\overline{A})$ = 1 - Bel (N) = 1 – 0.75 = 0.75
Un (N) = Bel (N) - Pl (N) = 0.27-0.25 = 0.02
Un (A) = Bel (A) - Pl (A) = 0.75-0.73 = 0.02

$$Bp(l_q) = m(l_q) + \frac{m(L)}{p}$$

Bp (N) = m (N) + $\frac{m(N,A)}{p}$ = 0.25 + $\frac{0.04}{2}$ = 0.25 +0.02 = 0.27

Bp (A) = m (A) + $\frac{m(N,A)}{p}$ = 0.73 + $\frac{0.04}{2}$ = 0.73 +0.02 = 0.75

	{N}	{A}	{N,A}
m_1	0.15	0.2	0.65
Bel_1	0.15	0.2	1
Pl_1	0.8	0.85	1
m_2	0.25	0.7	0.05
Bel_2	0.25	0.7	1
Pl_2	0.3	0.75	1
m	0.25	0.73	0.04
Bel	0.25	0.73	1
Pl	0.27	0.75	1
U	0.02	0.02	
Bp	0.27	0.75	

5. Experimental Results:

For testing we considered U2R and R2L attacks from KDD cup 99 dataset. A dataset of 1385 records consisting the blend of U2R and Normal Records and a dataset of 3300 records consisting the blend of R2L and Normal Records is taken for training and testing purpose. Three fold cross validation method of testing is implemented. The results are tabulated in the form of confusion matrix and efficiency tables.The efficiency are reported in the following tabular forms:

Efficiency for R2L:

Training Data Sets	Testing Data Set	Accuracy
2,3	1	89.4586
1,3	2	91.1182
1,2	3	91.8778

Efficiency for U2R:

Training Data Sets	Testing Data Set	Accuracy
2,3	1	96.4286
1,3	2	95.0549
1,2	3	95.3297

Average efficiency and False positive rates are as follows:

Attack Name	Efficiency	False positive rate
R2L	90.8182	8.24
U2R	95.6044	3.41

The confusion matrices for the datasets taken are reported as follows:

Confusion Matrix for R2L:

	Attack	Normal	Total
Attack	269	31	300
Normal	272	2728	3000
Total	541	2759	3300

Confusion Matrix for U2R:

	Attack	Normal	Total
Attack	42	10	52
Normal	37	996	1033
Total	79	1006	1085

6. Conclusion

The key idea is to imitate ambiguous of users activities by fuzzy clustering technique, and to simulate uncertainty caused by limited information by incorporating only a small amount of network traffic data for analysis.
With the use of Dempster-Shafer theory, we identify future network traffic by fusing evidences found in clustering development. Also, we employ *k*-NN technique to speed up the detection process.
We used Fuzzy Belief K-NN algorithm to detect intrusions in the KDD Cup 99 dataset. The results

obtained by implementing these algorithms are fairly good with appreciable accuracy and gives scope for future work.

References

[1] H. Debar, “An Introduction to Intrusion-Detection Systems,” Proceedings of Connect’2000, Doha, Qatar, April 2000.

[2] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.

[3] J. C. Dunn, “A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact Well-Separated Clusters,” Journal of Cybernetics, Volume 3, pp. 32-57, 1973.

[4] A. P. Dempster, “A Generalization of Bayesian Inference,” Journal of the Royal Statistical Society, Series B, Volume 30, pp. 205-247, 1968.

[5] G. Shafer, *A Mathematical Theory of Evidence, Princeton*, University Press, Princeton, NJ, 1976.

[6] J. M. Booker, M. C. Anderson, M. A. Meyer, “The Role of Expert Knowledge in Uncertainty Quantification (Are We Adding More Uncertainty or More Understanding?),” Seventh Army Conference on Applied Statistics, pp. 155-161, 2001.

[7] W. L Oberkampf, J. C. Helton, C. A. Jos lyn, S. F. Wojtkiewicz, and S. Ferson, “Challenge Problems: Uncertainty in System Response Given Uncertain Parameters,” Reliability Engineering and System Safety, Volume 85, pp. 11-19, 2004.

[8] T. J. Ross, *Fuzzy Logic with Engineering Applications*, 2nd Edition, John Wiley & Sons, Ltd., 2005.

[9] S. A. Dudani, “The Distance-Weighted k-NN Rule,” IEEE Transactions on Systems, Man and Cybernetics, Volume 6, Number 4, pp. 325-327, 1976.

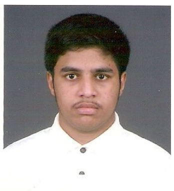
[10] R. Haralick and L. Shapiro, *Computer and Robot Vision*, Volume 2, Addison-Wesley, 1993.

[11] Te-Shun Chou, Kang K. Yen, Niki Pissinou, and Kia Makki, “Fuzzy Belief Reasoning for Intrusion Detection Design,” IEEE The third International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Kaohsiung, Taiwan, November 2007.

[12] KDD’99 archive: The Fifth International Conference on Knowledge Discovery and Data Mining. URL: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

Pakkurthi Srinivasu received his M.Tech(CST) from Andhra University , Visakhapatnam, Andhra Pradesh, India. Presently he is working as Associate Professor in Computer Science and Engineering in Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam Dist, AP, India. His research area includes Intrusion Detection, Network Security, Neural network, Data mining and fuzzy logic

Prof. P.S.Avadhani did his Masters Degree and PhD from IIT, Kanpur. He is presently working as Professor in Dept. of Computer Science and Systems Engineering in Andhra University college of Engg., in Visakhapatnam. He has more than 70 papers published in various National / International journals and conferences. His research areas include Cryptography, Data Security, Algorithms, and Computer Graphics etc..



Tummala Pradeep, pursuing his Final Year Computer Science and Engg., in Birla Institute of Technology, Ranchi, Jharkhand, INDIA. His interests include Network Security, Data Mining, Soft Computing and Crowd Computing. He is very active in fundamental research activities in the CSE Dept. of BIT.