AN INFORMATION GAIN BASED FEATURE SELECTION METHOD AND A NETWORK-BASED INTRUSION DETECTION SYSTEM FRAMEWORK UTILIZING ANOMALY DETECTION USING SELF ORGANIZING MAPS

by

Fatih Tiryakioğlu

B.S., Electrical and Electronics Engineering, İstanbul University, 2001

Submitted to the Institue for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Electrical and Electronics Engineering
Boğaziçi University

2008

ACKNOWLEDGEMENTS

First, I want to thank my advisors Prof. Emin ANARIM and Assist. Prof. Kerem HARMANCI for their guidance and support in development, completion of this thesis.

Secondly, I thank all my friends for their understanding. Special thanks go to my friends Ali EKŞİM, Fuat GELERİ, Murat BALABAN, Çağatay KARABAT, Sinan KAHRAMAN, and Fatih BİRİNCİ.

Finally, it is my pleasure to thank my family for their endless support and belief in me and my work.

This work is supported by the State Planning Organization of Turkey (DPT), project grant no. DPT-03K 120250.

ABSTRACT

AN INFORMATION GAIN BASED FEATURE SELECTION METHOD AND A NETWORK-BASED INTRUSION DETECTION SYSTEM FRAMEWORK UTILIZING ANOMALY DETECTION USING SELF ORGANIZING MAPS

In this work, an information gain based feature selection method and a network-based intrusion detection system utilizing anomaly detection using Self Organizing Maps (SOM) are proposed. KDD 99 (The International Knowledge Discovery and Data Mining Tools Competition 1999) is used for the feature selection and performance evaluation of the anomaly system. Feature selection method considers every combination of n feature groups as a unique feature and determines whether it is useful for the anomaly detection by calculating entropy of the each new feature. As the number of features in a group, namely n, goes up, both the number of the combinations and the time needed for calculating every new feature's information gain increases, and it becomes computationally infeasible. To overcome this problem, a quantization method, which is also information gain based, is proposed. The quantization of the basic features makes possible of the calculations of the information gains of the new combinational features as the n increases. In the anomaly detection part of the work, multi number of SOMs, every one is specialized to detect an attack group, is proposed. The useful features for each SOM is determined according to proposed feature selection process, and the performance of the SOMs are calculated.

ÖZET

BİLGİ KAZANÇ TABANLI ÖZELLİK SEÇME METODU VE KENDİ KENDİNİ EĞİTEN HARİTALAR KULLANILARAK OLAĞANDIŞILIK TESPİTİ YAPAN AĞ TABANLI GİRİŞİM TESPİT SİSTEMİ

Bu çalışmada bilgi kazanç tabanlı özellik seçme metodu ile SOM (kendi kendini eğiten harita) kullanılarak olağan dışılık tespiti yapan bir network tabanlı girişim tespit sistemi düşünülmüştür. Özellik seçme ve olağandışılık tabanlı sistemin performansını ölçmek için KDD 99 (Uluslararası bilgi keşif ve veri madenciliği araç yarışması 1999) kullanılmıştır. Özellik seçme metodu, n özelliğin her bir kombinasyonunu tek bir özellikmiş gibi kabul etmekte ve bu yeni özelliklerin entropilerini hesaplayarak olağandışılık tespiti için uygun olup olmadıklarına karar vermektedir. Grup içerisindeki özelliklerin sayısı, yani n sayısı, arttıkça hem kombinasyonların sayısı hem de her bir yeni özelliğin entropisini hesaplamak için gerekli zaman artmakta ve bu durum verimsiz bir hale dönüşmektedir. Bu problemi halletmek için yine bilgi kazanç tabanlı bir nicemleme metodu düşünülmüştür. Temel özelliklerin nicemlenmesi, n sayısı arttığında kombinasyonlarla elde edilen yeni özelliklerin bilgi kazançlarının hesaplanmasını mümkün hale getirmektedir. Çalışmanın olağandışılık tespit kısmında her biri ayrı bir saldırı grubu için özelleşmiş çok sayıda SOM tasarlanmıştır. Her SOM için faydalı özellikler, özellik seçme metodu ile bulunmuş ve SOM'ların performansı ölçülmüştür.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

c_j	Cluster center
$h_{ci}(t)$	Neighborhood kernel around the winner unit c
m_i	d-dimensional prototype (reference or codebook) vector
r_c	Location of unit c on the map grid
$r(t)$	Neighborhood radius
T	Training length
Th	Threshold
w_i	Normalized firing strength
$\sigma(t)$	Neighborhood radius at time t
$\alpha(t)$	Learning rate
α_0	Initial learning rate
α_T	Final learning rate
BMU	Best Matching Unit
DARPA	Defense Advanced Research Projects Agency
DoS	Denial of Service
DSS	Decision Support System
FAR	False Alarm Rate
FP	False Positives
FPR	False Positive Rate
HIDS	Host Based Intrusion Detection System
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IDSS	Intelligent Decision Support System
K-M	K-means clustering

KDD 99	The international Knowledge Discovery and Data Mining Tools Competition 1999
KNN	K-nearest Neighbor Algorithm
NIDS	Network Based Intrusion Detection System
NN	Neural Networks
PD	Probability of Detection
QV	Quantization Vector
R2L	Remote to Local
ROC	Receiver Operating Function
sMap	SOM Structure
SOM	Self Organizing Map
SVM	Support Vector Machine
TCP	Transmission Control Protocol
U2R	User to Root
UDP	User Datagram Protocol

1. INTRODUCTION

With the increasing rate of computer communications and internet, network security is becoming a major challenge. In order to keep the communication secure, static and dynamic solutions are implemented. Static solutions are general mechanisms such as software update, firewall programs etc. Because their implementation is apparent and static, they provide basic security but not complete security. Attackers exploiting these systems' vulnerabilities can access, modify, destroy information or prevent the clients to access. In this point, security needs dynamic mechanisms together with static mechanisms. Intrusion Detection Systems (IDS) are designed to overcome these security problems.

Attacks against networks target to stop availability of services, called Denial of Service attacks, or access some information, modify or delete it. Most widespread attacks are Denial of Service attacks which is a threat for the availability of the service. The other types are probing, remote to local, and user to root attacks. Probing attacks aim to get information about network topology, services etc for the future activities. Remote to local attacks is about getting access rights for a service. In user to root attacks, attacker is a user which has some rights, but seeks for more rights in the network. These hostile activities are all observed by Intrusion Detection Systems. Intrusion detection systems are the 'burglar alarms' (or rather 'intrusion alarms') of the computer security field. The aim is to defend a system by using a combination of an alarm that sounds whenever the site's security has been compromised, and an entity—most often a site security officer (SSO)—that can respond to the alarm and take the appropriate action, for instance by ousting the intruder, calling on the proper external authorities, and so on [1].

A major problem in the IDS is the guarantee for the intrusion detection. This is the reason why in many cases IDSs are used together with a human expert. In this way, IDS is actually helping the network security officer and it is not reliable enough to be trusted on its own [2].

The network based IDS is responsible to protect the entire environment of the network from the intrusion. This task asks for full knowledge of the system status and monitoring both the components of the network and the transactions between them. The host based IDS is only installed on a single host/terminal and is responsible for monitoring the status of that terminal/server only. This type of IDS is responsible for the security of its host and will monitor the entire network activities in that host [3]. One of the problems with the host based IDS is the high processing overhead that they impose on their host. These overheads will slowdown the host and therefore it is not welcomed. This approach is quite popular among the researchers [2].

In regard of the system, two approaches exist in the Intrusion Detection Systems: misuse and anomaly. Misuse or signature based systems are more frequent than the anomaly based systems. It simply has all the detected attack structures and tries to detect "known" attacks. Anomaly based systems focus on normal user activities, and detects abnormal activities. Although misuse detection systems are more effective in the known attack types, anomaly based systems can detect new attack types. Some systems make use of these two approaches. One disadvantage of anomaly based intrusion detection systems is to have high false alarm rates then signature based ones.

For the anomaly based systems, it is required to define normal traffic. However, it is not an easy task, and some statistical, mathematical models are used to define normal traffic pattern. Because normal deviates in time and system, adaptive approaches and learning algorithms are employed for the particular network traffic and user behavior. There is another Intrusion Detection (ID) approach that is called specification-based intrusion detection. In this approach, the normal behavior (expected behavior) of the host is specified and consequently modeled [2].

As for the last line of defense, and in order to reduce the number of undetected intrusions, heuristic methods such as Honey Pots (HP) can be deployed. HPs can be installed on any system and act as trap or decoy for a resource [2].

The information given to the Intrusion Detection Systems can be payload or connection based. Payload based systems monitors all the traffic in multiple layers and make use of

raw information. Connection based systems get summary of each connection. In the KDD dataset, which is widely used for connection based systems' performance, there are 41 features for every connection. 6 of them are basic features such as duration, sent and received bytes, protocol and service type. The rest of the features are content and host based. Because connection based Intrusion Detection Systems can not evaluate connection records before connections are ended, they are not real-time systems.

In the network based systems performance is very important because of the heavy weight of the traffic. Because efficiency is a requirement, using less information with high meaning is a parameter which increases the performance of the system. For this purpose the features in the connection based system is investigated and a feature selection process is defined. Feature selection process is executed in an attack type basis because of different characteristics of them, and a two level SOM architecture with different feature sets are proposed.

In the feature selection phase, information gain approach is employed. Because dataset is used for the calculations, it is a nonparametric approach. The decision about selecting a feature or not is made by calculating the entropy of the feature values in the dataset.

The performance of the system is measured by counting false positive and false negative signals in the test phase. False positive is false alarms such that the system signals a normal connection as to be an attack. False negative, on the other hand, is missed attacks which is the system could not detect it.

The organization of the thesis is as follows. In the next chapter, some anomaly detection approaches using artificial intelligence techniques, and some feature selection methods, including clustering are introduced. Chapter 3 provides feature selection methodology and proposed Intrusion Detection System. Results are presented in Chapter 4.

2. ARTIFICAL INTELLIGENCE OF ANOMALY INTRUSION DETECTION SYSTEMS

Artificial intelligence is widely used for intrusion detection purposes. Kabiri *et al.* groups intrusion detection approaches using artificial intelligence into rule-based methods, data mining using the association rule methods, fuzzy logic methods, multidisciplinary approaches (combining fuzzy logic, genetic algorithm, association rule, hidden markov model etc.), bayesian methodologies, and artificial neural networks [2].

Because artificial neural network methods are very sensitive to the dimension of the data, they use some methods to reduce the dimension of the input data. One of the most used methods is the Kohonen's Self Organizing Map (SOM). Self Organizing Maps are also used complementary with other methods.

The SOM is a neural network model proposed by Kohonen for analyzing and visualizing high dimensional data [4]. It belongs to the category of competitive learning models which are commonly used for various clustering problems successfully [4]. The SOM is based on unsupervised learning to map nonlinear statistical relationships between high-dimensional input data into two-dimensional lattice or grid which is also called the output space. In other words SOM provides topology preserving mapping from the input space to the two-dimensional lattice or grid of nodes [5].

A host based intrusion detection system is proposed by Lichodzijewski. Logged session information is reduced and preprocessed in real time and for the pattern discovery step, a hierachiacal two level Self Organizing Map is employed. In the first level, related session information is given some specific SOMs, and the outputs of these SOMs are inputted to the aggregate SOM for the final result [6]. In the host based intrusion detection system, proposed by S. Lee, Back Propogation Hierarchical Neural Networks is used as small detectors for anomaly detection. At the high level of the hierarchy, BP NNs are replaced with SOM detectors due to their some advantages [7]. Cho applied Hidden Markov Model (HMM) to the host based anomaly detection which is performed by using system call audit

data. Because additional system-call-related information is too large and various to apply to HMM directly, some data reduction techniques are needed in the preprocessing module of the system. Self Organizing Map is used to reduce the additional information [8].

Network based intrusion detection systems which employ Self Organizing Maps are presented much more than host based ones. Network based intrusion detection systems use either real time network traffic data composed of ip packets or offline connection information which is summary of the packets between a server and client. For the network based intrusion detection systems which use connection information, KDD dataset can be thought as a reference dataset.

In the system proposed by Depren *et al*., basic 5 features of a connection which are duration, service (ftp, mail, etc.), connection flag, received bytes, and sent bytes are selected, and given to the Self Organizing Map after two level of preprocessing. 3 independent SOM is employed; each one is allocated for one of three protocol type, namely TCP, UDP, and ICMP. It is assumed that basic features will give better results [9]. Kayacık proposed a hierarchical learning system using first 6 features which are features mentioned above and protocol type together [5]. Three level of SOM is trained; one SOM is allocated for each feature in the first level, the outputs of these SOMs are inputted to the aggregate SOM in the second level, and correction is performed for the undecided connections in the last level.

Some packet based real time intrusion detection systems also employ Self Organizing Map. Bolzoni *et al*. propose POSEIDON system which enhances former intrusion detection algorithm by combining SOM for clustering packet payload [10]. Zanero propose a two-tier network based intrusion detection architecture which observes rolling window of packets. Because packet's sizes are varying and large size of packets' analysis is difficult, payload information of the packets is classified by using Self Organizing Map. Following classification of the payload and decoding of the IP and TCP headers, second stage takes place for the anomaly detection [11]. Firewall architecture of the Yoo and Ultes-Nitsche also uses Self Organizing Map in their Smart Detection Engine for detecting virus infected files [12].

There are also some connection based intrusion detection systems which do not employ Self Organizing Map. Yang *et al*. presents an anomaly detection approach for the network intrusion detection based on Cellular Neural Network (CNN) model, and experiments with KDD 99 dataset exhibits better performance than back propagation neural networks [13]. Chen proposes a method based on Support Vector Machine (SVM) with a voting weight scheme to detect intrusion [14].

Most of the proposed connection based Intrusion Detection Systems using KDD 99 dataset selects basic features of the dataset. The basic features are first 6 features, duration, protocol type, service, status flag, source and destination bytes. Depren *et al.* selects first 5 features except protocol type, and gives them to the appropriate Self Organizing Map employed for each protocol type [9]. It is assumed that basic features will give better results. Kayacık proposed a hierarchical learning system using first 6 features [5].

It has been demonstrated that a large number of the (41) input features of KDD 99 dataset are unimportant and may be eliminated, without significantly lowering the performance of the IDS. In terms of the classification, Sung and Mukkamala found that by using only 19 of the most important features, instead of the entire 41-feature set, the change in accuracy of intrusion detection was statistically insignificant [15]. Kayacik explored valuable features among content, time, and host based features by using information gain [16]. In this work, information gain is calculated for every feature and found valuable features out of the basic features. Chebrolu *et al.* evaluated better performance by using feature sets which are evaluated from algorithms involving Bayesian networks (BN), Classification and Regression Trees (CART), and an ensemble of BN and CART [17].

Because clustering can be used for feature selection, some clustering methods are explored. K-means method is sensitive to noise and outlier data points since a small number of such data can substantially influence the mean value [18]. Li *et al*. propose Minimum Entropy criterion, which is the conditional entropy of clusters given the observations [19]. The minimum entropy criterion is equal to the nearest neighbor method when its α parameter is equal to 2.

3. METHODOLOGY

The basic objective of this work is to determine valuable different feature sets of KDD 99 dataset among 41 features for different attack type detection processes, to propose a network based IDS framework, and to benchmark the performance of the framework which is consisted of Preprocessing Module, Anomaly Detection Module and a Decision Support System. Based on the results of feature selection work, Preprocessing Module selects features and departs them from the others and enumerates. In the anomaly detection module, SOM classification algorithm is implemented for learning. For every protocol type (TCP, UDP and ICMP) different SOM analyzers with different feature sets are placed for better performance. Before predefined feature sets of connection records are inputted to the SOM analyzers, they encountered quantization in the Specific Quantization Module which has some look-up tables for the purpose of performance gain. For all the work, 10 % of KDD 99 dataset is used. In the following sections, feature selection and quantization methodologies are explained, and proposed network based IDS framework are described. Simulation results and the performance of the proposed system are presented in Chapter 4.

3.1. Feature Selection and Quantization

Feature selection and quantization is the starting point of this work. Because connection information has many features and computing power is limited, selection of the good features among them becomes a critical work. Quantization, on the other hand, decreases the needed computational power for some processes, and may increase the performance of the overall system by merging correlated feature values.

3.1.1. Feature Selection

The KDD 99 intrusion detection datasets are based on 1998 DARPA initiative, which provides designers of intrusion detection systems with a benchmark on which to evaluate different methodologies. KDD 99 dataset are composed of connection records which has 41 connection features and a label showing whether it is a normal or attack

connection. This dataset contains 22 attack types which fall into one of four categories: Denial of Service, Probe, User to Root, and Remote to Local. This work focuses on selecting good features related to these attack groups as well as all attacks as one group. The features for all attacks and every attack groups mentioned above are estimated separately for TCP, UDP, and ICMP protocol type connections. Thus, the feature set for Denial of Service attack group can be different in UDP protocol type from TCP protocol type.

To select discriminating feature sets of attack groups, information gain approach is employed. In the basic means, the information gain of a feature in related to an attack group is the information quality of the feature that shows if a connection is belong to the attack group or not. If we rearrange the labels 1 and 0 such that 1 indicates the connection is in the group and 1 does not, expected information which is the highest level of information gain of a feature can be calculated.

$$I(s_1, s_2) = -\sum_{i=1}^{2} \frac{s_i}{s} \log_2 \left(\frac{s_i}{s} \right) \tag{3.1}$$

where s_1 is the number of samples in the group, and s_2 is the number of samples out of the group. If all samples are in the group or out of the group, the expected information is zero. In this case, information gains of all features are also zero, simply because there is no need for any feature to analyze whether the connection is attack or not. On the other hand, if the half of the samples is in the group, and the other half is out of the group, the value of expected information equals to 1 which means the features are at the maximum importance level to classify connection records.

To calculate the information gain of a feature, entropy of the feature on the label class which is 0 or 1 in this case is subtracted from the expected information of the label class given above. If F feature has value set of $\{f_1, f_2, .. f_r\}$, entropy of the feature:

$$E(F) = \sum_{i=1}^{r} \frac{s_{1i} + s_{2i}}{s} I(s_{1i}, s_{2i}) \tag{3.2}$$

that is s_{1i} is the number of the samples whom feature value is f_i and in the group of the label. The same value subsets of an ideal discriminating feature match altogether into the group subset or out of the group subset of the label. Thus, a sample can be concluded to be in the label group or not by means of feature value. The information gain of the F feature:

$$Gain(F) = I(s_1, s_2) - E(F) \tag{3.3}$$

Looking for high level of information gain from only one feature seems to be unrealistic. Thus, it is needed a new approach calculating information gain of feature sets as if a feature set is one feature. Base upon that, of 41 features of KDD 99 dataset, feature sets containing 2 features, 3 features etc are constructed and information gain of these feature sets are calculated. The number of feature sets composed of n features among m features, 41 for our dataset, is:

$$C(m,n) = \frac{m!}{(m-n)!n!} \tag{3.4}$$

The entropy and information gain of features F_1 and F_2 which is a feature set of two features can be calculated as follows:

$$E(F_1, F_2) = \sum_{i=1}^{r} \sum_{j=1}^{t} \frac{s_{1ij} + s_{2ij}}{s} I(s_{1ij}, s_{2ij}) \tag{3.5}$$

$$Gain(F_1, F_2) = I(s_1, s_2) - E(F_1, F_2) \tag{3.6}$$

where F_1 and F_2 have feature value sets of $\{f_1, f_2, .. f_r\}$ and $\{f_1, f_2, .. f_t\}$, respectively.

The equations above can be generalized for feature sets composed of n features as below.

$$E(F_1, F_2, ..F_n) = \underbrace{\sum_{i=1}^{r} \sum_{j=1}^{t} ... \sum_{p=1}^{z}}_{n\ features} \frac{s_{1ij..p} + s_{2ij..p}}{s} I(s_{1ij..p}, s_{2ij..p}) \tag{3.7}$$

$$Gain(F_1, F_2, ..F_n) = I(s_1, s_2) - E(F_1, F_2, ..F_n) \quad (3.8)$$

such that, F_1, F_2, $..F_n$ have feature value sets of $\{f_1, f_2, .. f_r\}$, $\{f_1, f_2, .. f_t\}$, ... $\{f_1, f_2, .. f_z\}$, respectively. As the number of features is increased for the calculation of information gain, it is expected to evaluate better performance, because more features mean more information. On the other hand, increasing the number of features, considered, required more computational power. Two reasons aroused for this computational power need. First one is the increasing number of feature sets as the n goes up. For KDD 99 dataset, which has 41 features, the numbers of feature sets are shown in the Table 3.1.

Table 3.1. The number of feature sets composed of n feature in KDD 99 dataset

# of features in a feature set	# of feature sets
1	41
2	820
3	10660
4	101270
5	749398

The second reason for the computational power need is the increasing of the sum elements on the side in the equation 3.7. Because of these two reasons, it is not efficient to calculate the information gains of feature sets with many features. The solution for this problem could be decrease the number of feature sets, composed of n features as n goes up, is impossible since it means missing valuable feature sets. The only remaining solution stands out to be decreasing sum elements, which means quantizing feature values. However, a quantization that does not performed properly will result in information loss due to merging discriminating values of the feature. Thus, the records of a feature are quantized in a way that only similar values, causing the same results, should be merged. Because it is a difficult problem and it can not be performed simply, a new approach, described in the following section, is considered.

3.1.2. Entropy Based Quantization

In entropy based quantization, we consider the label entropy of the feature's neighbor values while deciding to quantize the values of the feature. In the simplest mean, the neighbor values which have same entropy values can be quantized. If $E(F/F_r)$ is the entropy of the f_r valued samples on the label which have 0 and 1 values:

$$E(F / F_r) = \frac{s_{1r} + s_{2r}}{s_r} I(s_{1r}, s_{2r}) \tag{3.9}$$

where $\{f_1, f_2, .. f_r\}$ is the feature values of F, and $I(s_{1r}, s_{2r}) = -\sum_{i=1}^{2} \frac{s_{ir}}{s_r} \log_2 \left(\frac{s_{ir}}{s_r} \right)$. We can rewrite the equation above such that,

$$E(F_r) = I(s_{1r}, s_{2r}) \tag{3.10}$$

because $\frac{s_{1r} + s_{2r}}{s_r} = 1$, and $E(F / F_r) = E(F_r)$ for one value of the feature.

This basic approach, however, has some shortcomings. Quantizing two neighbor values of a feature according to entropies of them does not give exact solution. Because entropy considers only ratios or probabilities of the labels, in our example, and it is not about what the labels are, it is needed to estimate a value like entropy, so that which label is weighted than the other one. In other words, when all fr valued samples maps 0 and all f_{r+1} valued samples maps 1 on the label, $E(F_r)$ and $E(F_{r+1})$ equals to 0, and they could be concluded to quantize them in the basic approach. However, this two feature values are completely distinct to each other, since fr valued samples is out of the group, call normal samples, and f_{r+1} valued samples is in the group which is mostly attack group. This situation can be shown in Figure 3.1, 1 bit entropy graph. To discriminate this symmetry, it is proposed a new calculation shown below. While 1 bit entropy ranges between 0 and 1, it ranges from 0 to 2 such that when the ratio of the samples in the group is greater than that of the samples out of the group, the entropy value is mapped to the value which is symmetric to 1. If we call the calculation $\hat{E}(F_r)$ for the fr values of the F feature:

$$\hat{E}(F_r) = \begin{cases} E(F_r) & s_{1r} \geq s_{2r} \\ 2 - E(F_r) & s_{1r} < s_{2r} \end{cases} \tag{3.11}$$

where s_{1r} is the the number of samples whose label is 0, and s_{2r} is 1 valued samples. Figure 3.2 shows the $\hat{E} - s_{2r}/(s_{1r}+s_{2r})$ graph assuming s_2 is the number of the attack labeled samples.

At that point, $\hat{E}(f_r)$ values are estimated for each fr valued label group to decide for quantizing F feature neighbor values. The main focus is on the only one F feature and f_r an f_{r+1} values whose $\hat{E}(f_r)$ and $\hat{E}(f_{r+1})$ equals to each other are tend to be quantized. If, for example, all f_r and f_{r+1} valued samples are in the 0 valued label group, they are quantizable. However, being one half of the f_r and f_{r+1} valued samples are in the 0 valued label group, and the other half is 1 valued label group condition requires some future considerations. Because the information evaluated from F feature is not enough; f_r and f_{r+1} valued samples may be similar values or distinct values equally. To overcome this problem, some other features which F feature generates low entropy together are used to conclude whether f_r and f_{r+1} valued samples are quantizable. The number of the other features is not important, but the entropy of the group is aimed to be low as much as possible. The feature selection method described in the previous section is proposed for the low entropy feature selection already. Let F_1 be the quantized feature, and F_1, F_2, ..F_n be the feature group such that gain (F_1, F_2, ..F_n) is higher from any other group. $\hat{E}(F_{1r})$ can be redefined with related to the group:

$$\hat{E}(F_1 / F_{1r}, F_2, ..F_n) = \underbrace{\sum_{j=1}^{t}\sum_{k=1}^{u}\cdots\sum_{p=1}^{z}}_{n-1 \ features} \frac{s_{1rjk..p} + s_{2rjk..p}}{s_r} \hat{I}(s_{1rjk..p}, s_{2rjk..p}) \tag{3.12}$$

where F_1, F_2, ..F_n have feature values {f_1, f_2, ..f_t}, {f_1, f_2, ..f_u}, {f_1, f_2, ..f_z}, respectively, and

$$\hat{I}(s_{1rjk..p}, s_{2rjk..p}) = \begin{cases} I(s_{1rjk..p}, s_{2rjk..p}) & s_{1rjk..p} \geq s_{2rjk..p} \\ 2 - I(s_{1rjk..p}, s_{2rjk..p}) & s_{1rjk..p} < s_{2rjk..p} \end{cases} \tag{3.13}$$

$$I(s_{1rjk..p}, s_{2rjk..p}) = -\sum_{i=1}^{2} \frac{s_{irjk..p}}{s_{rjk..p}} \log_2 \left(\frac{s_{irjk..p}}{s_{rjk..p}} \right) \tag{3.14}$$

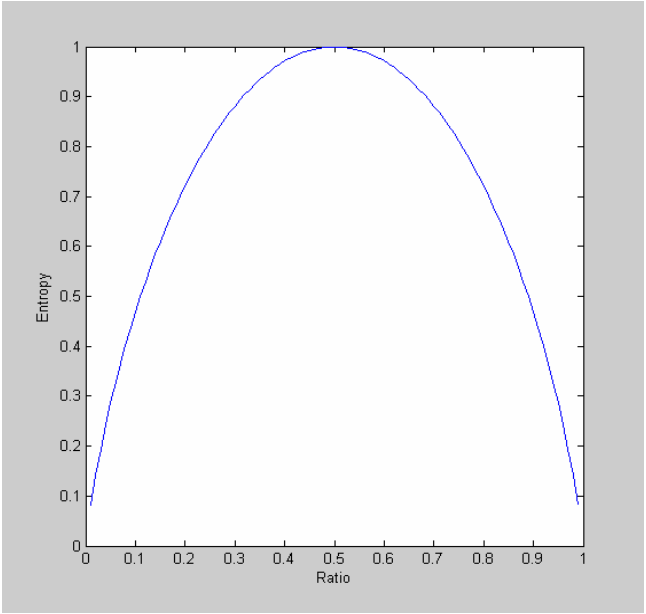

Figure 3.1. 1 bit entropy (ratio/entropy)

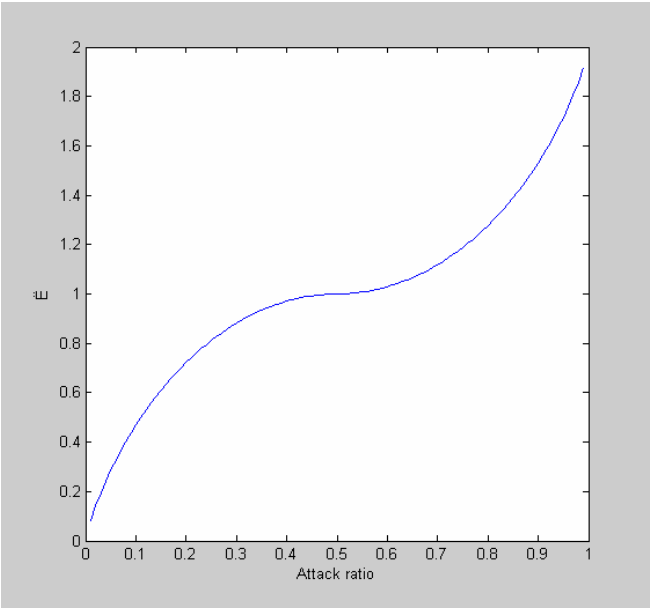

Figure 3.2. Ê-attack ratio graph

Although quantization is performed by means of not only quantized feature entropy on label, but entropy of feature on the features in the group, it is thought to be not enough for final decision. If f_r and f_{r+1} feature values are quantizable which means each one of them can be changed with the other one, f_r and f_{r+1} valued samples could correspond exactly the same feature values of the features in the group together with similar entropy like estimations shown above. With the last requirement, a correlation value is calculated between feature values neighbor to each other. For the same feature F_1 and feature group above, R_r is correlation between f_r and f_{r+1} feature values and calculated as:

$$R_r = \underbrace{\sum_{j=1}^{t}\sum_{k=1}^{u}\ldots\sum_{p=1}^{z}}_{n-1\ features} \min\left[\frac{s_{rjk..p}}{s_r}, \frac{s_{(r+1)jk..p}}{s_{r+1}}\right]\left(1-\hat{E}(F_{1r}, F_{2j}..F_{np})\right)\left(1-\hat{E}(F_{1(r+1)}, F_{2j}..F_{np})\right)$$

(3.15)

R_r value between f_r and f_{r+1} equals to 0 if f_r and f_{r+1} valued samples does not map the same values of the n-1 features among all samples. Thus, for relation not be 0, it is required such samples whose feature values are the same except F_1. If the f_r and f_{r+1} valued samples have common values in the n-1 features, the correlation can be positive or negative. Positive values point similar entropy like spreading on the label, and negative values show f_r and f_{r+1} valued features are distinct to each other. R_r values ranges between -1 and 1 meaning absolute negative and positive correlation, respectively.

Quantization vector of a feature is prepared iteratively by means of the equation above. For a feature whose values is $\{f_1, f_2,..f_r\}$, $\{R_1, R_2,..R_{r-1}\}$ correlation set is calculated in the first iteration. The feature values which has maximum Rr value among them is quantized and this time $\{R_1, R_2,..R_{r-2}\}$ correlation set is calculated in the second iteration since feature values are decreased by one. This iterative process continues before any R_r values are less than a threshold value which is determined prior to the all process.

```
set th;
for( ; ; )
        maxV = 0;
        for i =1:numberOfFeatureValues-1
                estimate Ri;
                if (Ri > maxV)
                        maxV = Ri;
                        maxF = i;
        end for
        if maxV > th
                fi+1 = fi;
                numberOfFeatureValues--;
        else
                break;
        end if
end for
```

At the end of this process, a quantization table is evaluated for the feature. The same process is implemented for the other features and quantization table process is finished. Because features have lower vector space, feature selection process defined in the previous section can be continued from the point it ended. Beside feature selection process, these quantization tables are used in the Specific Quantization Module of the proposed system for increasing performance of the anomaly detection rate.

3.2. The Proposed Network-Based IDS Framework Structure

The proposed network based IDS framework consists of three modules: Preprocessing module, Anomaly Detection Module and Decision Support System.

In the preprocessing module, connection records of network traffic are evaluated by a network analyzer. Network analyzer processes packets and logs connection records following to every connection closure. Because these records have many features, a set of these features which are selected by the methods mentioned above, and they are departed

from the other ones. Then these records are enumerated in the following preprocessing steps. Selected and enumerated features are not quantized here because this process is performed in the Anomaly Detection Module.

Anomaly Detection Module detects anomaly behavior from network traffic data. The components in the Anomaly Detection Module are Specific Quantization Module, Anomaly Analyzers, Management, and Communication Modules. Anomaly Analyzers and Specific Quantization Modules perform their task based on the protocol type of the connection record. For every protocol type, training and test processes are the same. In the training, SOM algorithm is used, and multiple SOM architectures are allocated for the sake of analyzing multiple aspects of the connection record.

Decision Support System collects results from Anomaly Detection Module and process decision-making algorithm. For decision-making, rule-based DSS is used.

The proposed framework has configuration, training and test phases. In the configuration phase, system administrator manages parameters of Anomaly Detection Module and DSS Module. Beside this, feature sets of every SOM Anomaly Analyzer and the values of Look-up Tables of Specific Quantization Module are entered to the system. Feature Selection and calculation of the look-up tables are performed according to the methods described in the previous section. In the training phase, Anomaly Analyzers of the system is trained by using normal traffic data. After training phase, the system is ready to use for intrusion detection which can be called test phase. In the test phase, all traffic data is processed through the proposed framework and anomaly activity is reported to the system administrator.

The detailed explanation of each module and the methods used prior to these modules explained in the following section. Preprocessing module is given in 3.2.1, Anomaly Detection Module is given in 3.2.2 and DSS Module is given in 3.2.3.

3.2.1. Preprocessing Module

Before Anomaly Detection Module, raw network traffic data is processed by two level preprocessing units, so that the data is ready for input to SOM anomaly analyzers.

3.2.1.1. First Level Preprocessing: Network traffic is evaluated by a network analyzer. Network analyzer processes packets and records connection data following to each connection closure. A connection is a sequence of packets between a source and a target in accordance with a communication protocol. This process includes network traffic data filtering, session information summarization and layer two – layer seven packet reconstructions. For each connection, extracted features can be categorized into three groups: basic features of individual TCP connections, content features within a connection suggested by domain knowledge, and traffic features computed using a two second time window.

Because KDD 99 dataset includes all connection features, this dataset is used in the training and test phases and not performed any effort for feature extraction in this work. Of the 41 features defined in KDD 99 dataset, all of them are given the same importance initially and they encountered a feature selection stage. The features, selected at the end of feature selection stage, are passed to the second level preprocessing. The feature categories which are basic features, content features, and host based features are listed in Table A.1, Table A.2 and Table A.3 in Appendix A.

Content based features are evaluated from payloads of the packets and they contains information such as number of logins, root access etc. Time based features gives information about last 2 seconds. Number of connections to the same host is one of them. Host base features uses connection window instead of time.

3.2.1.2. Second Level Preprocessing: Since some of connection features are alphanumeric, it is required to enumerate these ones which are protocol type, service, and flag features. Enumeration process is a mapping of every instance of alphanumeric character to integer values sequentially. For protocol type, the alphanumeric characters and corresponding integer values are listed in Table 3.5. Service and flag alphanumeric characters and integer values are not shown.

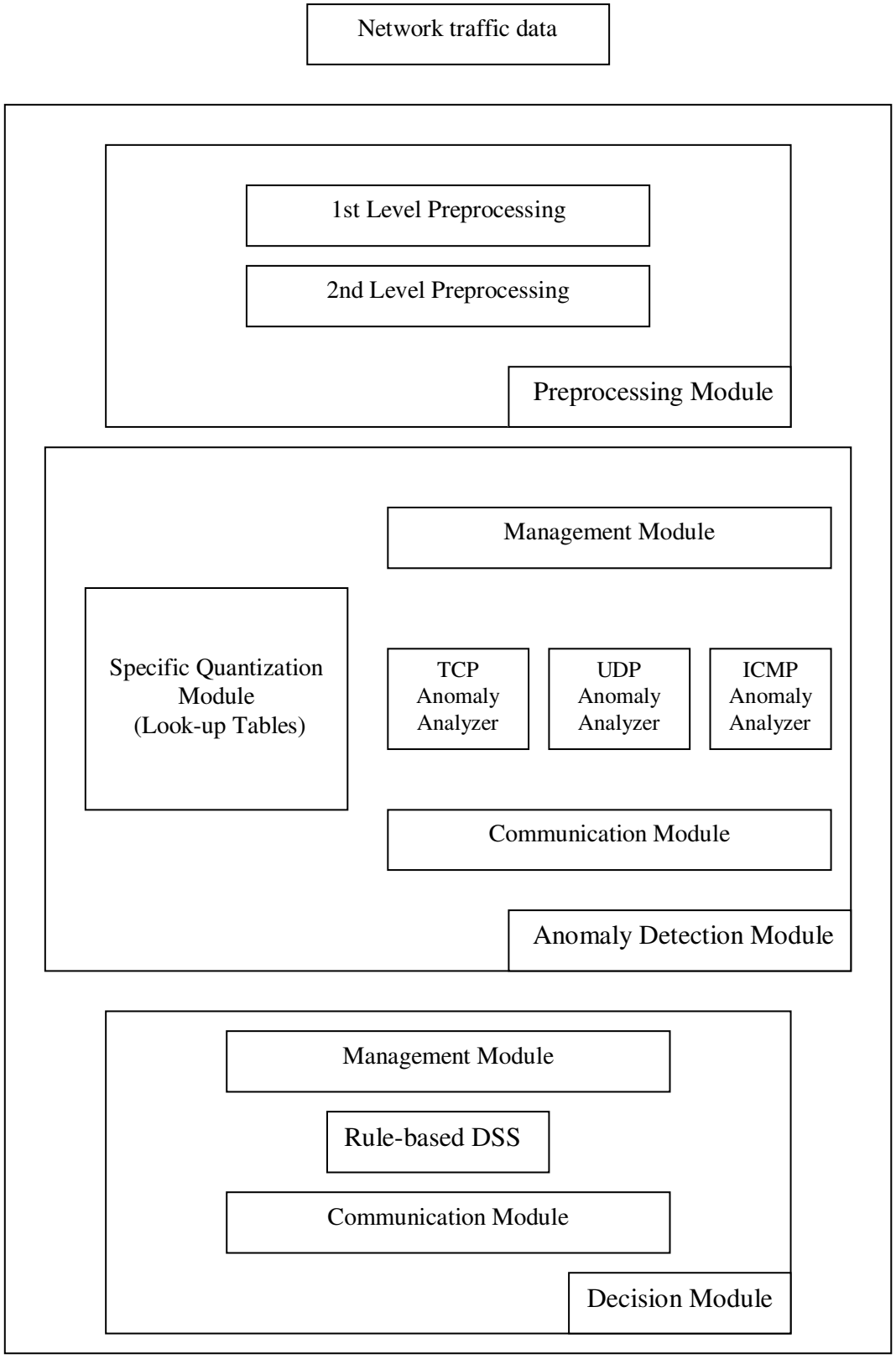

Figure 3.3. Proposed network-based IDS framework

Table 3.2. Enumeration of the alphanumeration of the alphanumeric protocol type features

Value	Assigned
TCP	1
UDP	2
ICMP	3

Enumeration process is not enough for SOM Analyzer to process, but they also must be normalized. However, feature data encounters quantization process in the Anomaly Detection Module, normalization can not performed in this stage. Connection feature data are normalized after the quantization process in the Anomaly Detection module just prior to SOM Analyzers.

3.2.2. Anomaly Detection Module

The duty of Intrusion Detection Systems is to detect attacks on the network infrastructure. For this purpose, it makes use of some techniques. One of these techniques is to detect anomaly behavior of the network traffic. Before detecting abnormal behavior which can be intentional or unintentional, the network traffic must be modeled by using normal traffic data. And based on the normal traffic, the solution may detect deviations from the normal data traffic. One advantage of the anomaly detection approach is to have the capability of finding out attack types which is not detected before.

To detect anomaly behavior, different feature sets are used in this work. After preselected features of a connection record are passed to the Anomaly Detection Module, the feature values are quantized and switch to the appropriate SOM Analyzer. SOM Analyzers processes data connection by connection based, so connection records do not affect the results of the other ones. In the training of the SOM Analyzers in which there are multiple SOMs, every SOM is inputted with preselected features compatible with the specific goal of the SOM. In the test or intrusion detection phase, the results of the hierarchical multiple SOMs are considered together in the Decision Support System.

There are four parts in the Anomaly Detection Module: Management Module, Specific Quantization Module, Anomaly Analyzer Modules, and Communication Module. Management Module is responsible for activating proper SOM Analyzer among TCP, UDP, and ICMP Anomaly Analyzers, passing connection feature data to the Specific Quantization Module, and coordination with Communication Module. It gets connection feature data one by one from the Preprocessing Module, passes it to the Specific Quantization Module together with the protocol type of the connection data and activates the proper SOM Analyzer allocated for the protocol type. It also signals Communication Module the connection number and the protocol type. Specific Quantization Module, upon receiving feature data and protocol type, prepares feature sets for the every SOM in the specified Analyzer Module and following quantization of the feature sets with the corresponding Look-up Tables, passes them to the Anomaly Analyzer Module activated by the Management Module. Every Anomaly Analyzer Module has five SOMs inside and any one of them processes different sets of feature data. Communication Module receives the results from the Anomaly Analyzer Modules and passes them to the Decision Support System.

System Administrator can interact with the Anomaly Detection Module through with Management Module. In the configuration phase, Management Module fills up Look-up Tables with their corresponding feature sets in the Specific Quantization Module, and configures the Anomaly Analyzers. The sub-modules of the Anomaly Detection Module are represented in the following sections.

3.2.2.1. Management Module: Management Module has configuration and operation duties. System administrator configures Specific Quantization Module and Anomaly Analyzers through Management Module. In the configuration phase, look-up tables for every protocol type are installed to the Specific Quantization Module and the parameters of the Anomaly Analyzers are entered. In the operation mode, Management Module is responsible for switching and activation works. Depending on received connection record, TCP, UDP, and ICMP, it activates proper Anomaly Analyzer and switches the predetermined feature set values of the protocol type to the Specific Quantization Module which performs quantization of the feature set and passes it to the activated Anomaly

Analyzer. Management Module also sends connection information to the Communication Module for decision process in accordance with connection record.

3.2.2.2. Specific Quantization Module: In the simplest means, it is composed of three look-up table groups; each group is allocated for one protocol, TCP, UDP, and ICMP. Each group has five look-up tables for five SOM architectures, namely Normal, DoS, r2l, u2r, in the specified protocol in the Anomaly Analyzer Modules. When activated one of these groups by Management Module, Specific Quantization Module firstly evaluates feature sets of every SOM Architecture, and then quantizes them according to quantization table of each feature set. Quantization process also implements normalization of the connection records, so that weights of the features are equal to the each other, a requirement to have better performance in the SOM Algorithm.

3.2.2.3. Anomaly Analyzer Modules: Anomaly Analyzer is composed of three Anomaly Analyzer Modules which are identical to each other: TCP Anomaly Analyzer, UDP Anomaly Analyzer, and ICMP Anomaly Analyzer. TCP, UDP, and ICMP connections are switched to the corresponding modules, thus they are completely independent in regard of training, test etc. Each module has five SOM Structures: Normal SOM, DoS SOM, Probe SOM, r2l SOM, and u2r SOM. These SOM structures use SOM algorithm to build normal behavior and they are trained independently. For example, TCP Normal SOM is trained with TCP connections labeled as normal, and TCP DoS SOM is trained with TCP connections not labeled as one of the DoS attack types because its duty is to detect only DoS attacks. Every SOM is trained and tested by using different feature set of the connection records as described in the feature selection section. At the end of the training phase, every SOM would have a lattice such that training data corresponds to the some clusters on it, and when it received an anomaly connection record which means to have a different characteristics from the trained data, it would correspond to the out of the clusters or to the clusters with high error. After having the results of the five SOMs as being normal or anomaly, these results are passed to the Decision Support System, and decision process is performed. Anomaly Analyzer Modules are given in the Figure 3.4.

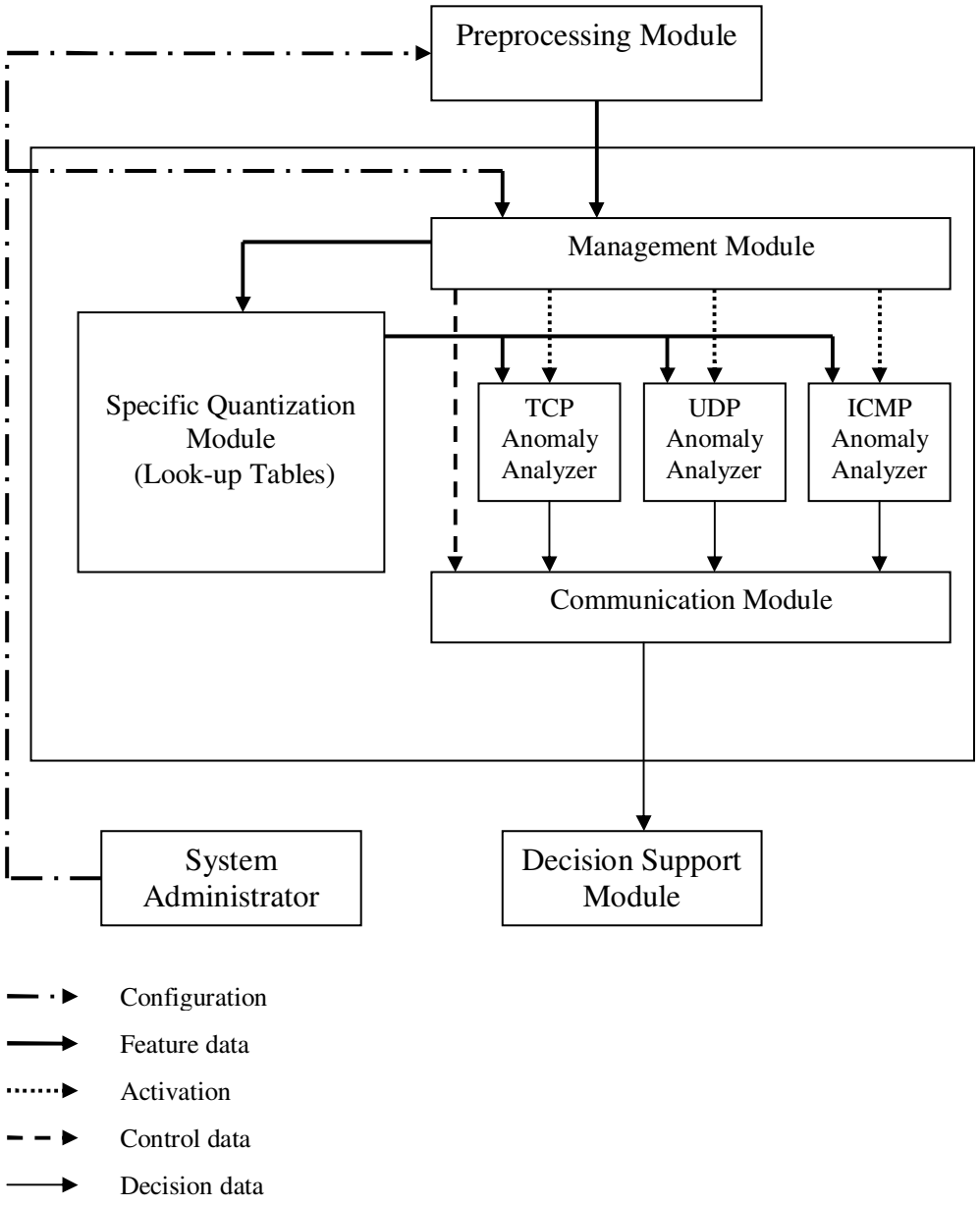

Figure 3.4. Anomaly Detection Module architecture

SOM Algorithm is an unsupervised clustering algorithm and it reduces high dimensional data to 2 dimensional grid which are composed of neurons. Neurons can be replaced hexagonal, rectangular, sheet etc. in the 2 dimensional space. Each neuron is represented by a d-dimensional weight vector $m=[m_1,..,m_d]$, where d is equal to the dimension of the input vectors. The neurons are connected to adjacent neurons by a neighborhood relation, which dictates the topology, or structure of the map.

Before the training process the weight vectors of each neuron are initialized randomly, or linearly. Then SOM is trained iteratively. In each training step, one sample vector x from the input dataset is extracted. The euclidian distance between x sample and each neuron is calculated and the nearest neuron which is called Best Matching Unit (BMU) is found such that:

$$||\mathbf{x} - \mathbf{m}_c|| = \min_i \{||\mathbf{x} - \mathbf{m}_i||\}, \tag{4.16}$$

After finding BMU, the weight vectors of the SOM are updated so that the BMU is moved closer to the input vector in the input space. The topological neighbors of the BMU are treated similarly. This adaptation procedure stretches the BMU and its topological neighbors towards the sample vector as shown in the Figure 3.5. The SOM update rule for the weight vector of the unit vector i is:

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + \alpha(t)h_{ci}(t)[\mathbf{x}(t) - \mathbf{m}_i(t)], \tag{4.17}$$

where t denotes time. The x(t) is the input vector drawn from the input dataset at time t, $h_{ci}(t)$ the neighborhood vector around the best matching unit c, and α(t) the learning rate at time t. The neighborhood kernel is a non-increasing function of time and of the distance of unit i from the winner unit c.

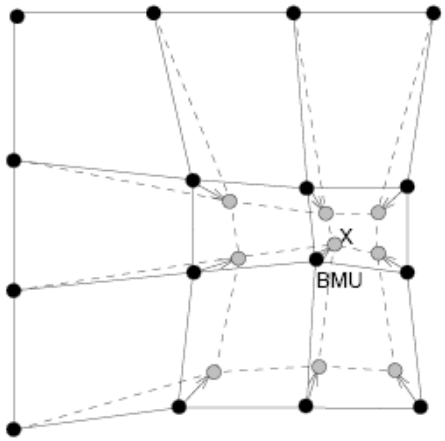

Figure 3.5. Training of SOM Neurons

The training is performed in 2 phases. In the first phase, relatively large initial learning rate is α_0 and neighborhood radius σ_0 are used. In the second phase, both learning rate and neighborhood radius are small right from the beginning [20].

After the training phase, SOM Classification phase take place. Before test data is inputted to the SOM, Quantization Vector (QV) is calculated. Quantization vector shows the error or threshold value of each neuron. In other words, when an input sample matches a neuron, its acceptable distance from the neuron is the threshold value of the neuron. The samples which are far away from the value of the threshold value is seen as an anomaly. As threshold values can be constant values, they can be adjusted by using varying values of the training data. For example, max distance of the training samples, which matches a specific neuron, is the threshold of this neuron. The threshold values can be limited by a global value. So that, the thresholds of the neurons which exceed the global values is decreased to the global value.

In the test phase, the BMU and the distance from the BMU is found for each test sample. If the distance is greater than the threshold value of BMU, then the sample is denoted as an anomaly. The samples whose distances are in the boundary of the threshold values of their BMUs are accepted as normal samples.

3.2.3. Decision Support System

Decision Support System interprets the results coming from Anomaly Detection Module. It uses a rule-based decision support system and sends the result to the system administrator through Communication Module.

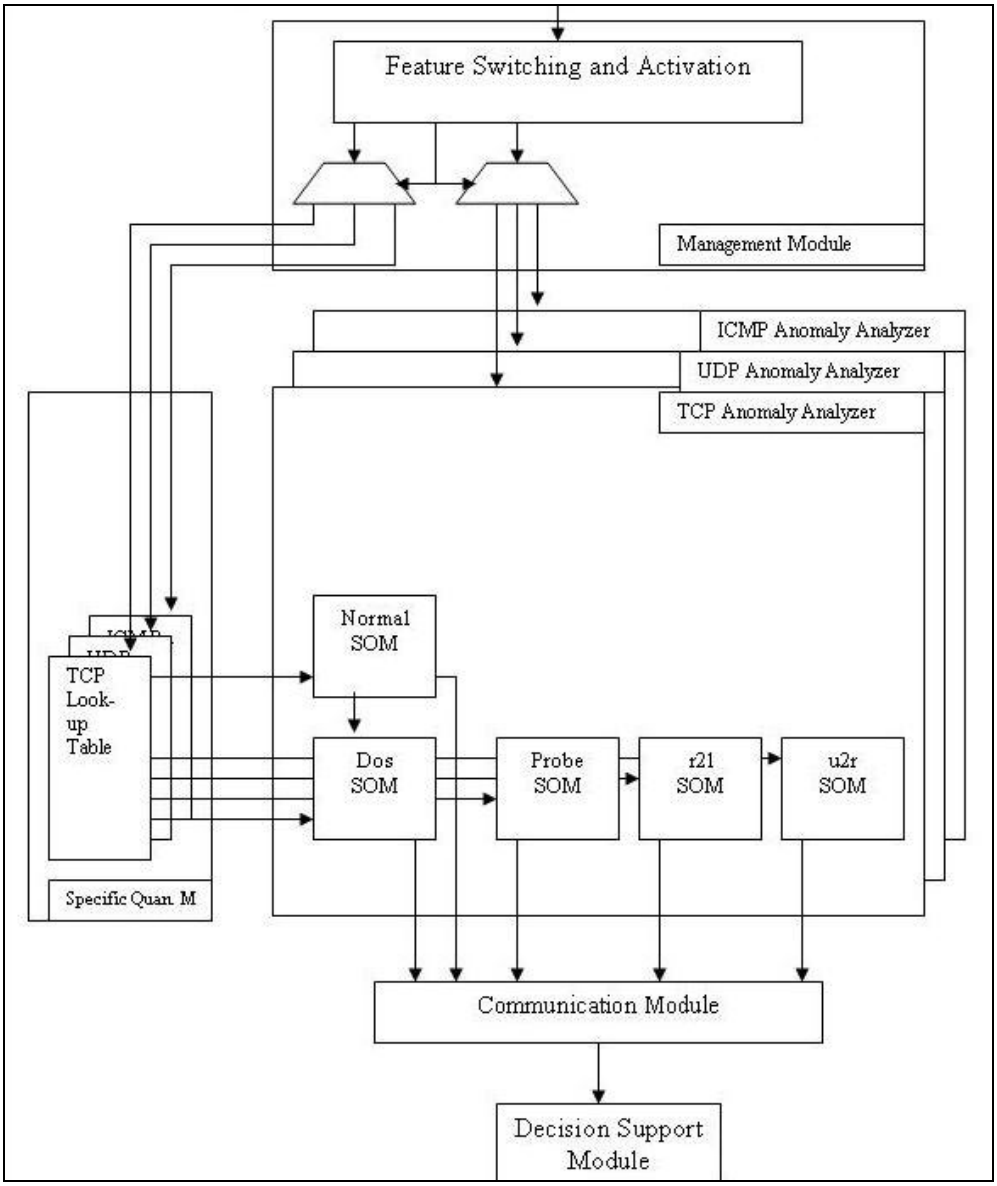

Figure 3.6. Anomaly Analyzer with three modules

3.2.3.1. Rule-based Decision Support System: Rule-based decision support system uses simple rules to make decision. The rules are common for all communication protocols, thus the same rules are in use for the results of TCP, UDP, and ICMP Anomaly Analyzers.

Rule set is composed of two rules. For any connection record, if the corresponding Normal SOM does not indicate an attack, then it is a normal connection. If the Normal SOM does indicate an attack, and at least one of the DoS, Probe, r2l, and u2r SOMs indicates an attack, then it is an attack.

3.2.3.2. Communication Module: Communication Module is responsible for the indicating the results of the Decision Support System to the system administrator.

4. SIMULATION RESULTS

Matlab is used for the simulation environment. The SOMs in the Anomaly Analyzer Modules are trained and tested by using SOM Toolbox. Firstly, the dataset and its usage are explained in the Section 4.1. Feature selection results are presented in the Section 4.2. Anomaly Detection results are presented in the Section 4.3.

4.1. KDD 99 Dataset

The KDD 99 dataset has 41 features and a label for each connection. Some information about features is given in the 3rd part. In all the simulation and calculations 10 % of the dataset is used. This portion of the dataset has 22 attack types and normal connection records. Most of these attacks are protocol specific, and mainly TCP protocol attacks. They are categorized as DoS, probing, r2l, and u2r attack groups. The number of samples in each group and dataset is presented in Table 4.1. Because DoS attacks' goal is to overload the target device by requesting connections, the number of them is more than the other attack groups. For the training, normal data connections are extracted from the dataset, and test is performed by using normal connections and the attack connections for which trained SOM architecture is responsible for detecting.

Table 4.1. KDD 99 dataset parts in terms of number of samples

Dataset	DoS	Probe	U2r	R2l	Normal
10% Dataset	391458	4107	52	1126	97277
Corrected KDD	229853	4166	70	16347	60593
Whole KDD	3883370	41102	52	1126	972780

Because 10 % KDD is employed as the training set in the International Knowledge Discovery and Data Mining Tools competion, this part of the dataset is used in the training and test processes. The number of training and test connection records in each protocol

type is given in Table 4.2, and the attacks with their numbers and categories is given in Table 4.3.

Table 4.2. Total number of connection records of KDD 99 dataset in each communication protocol

Dataset	**TCP**	**UDP**	**ICMP**	**Total**
Training (Normal) Dataset	72934	18644	1249	92827
Test (Attacks) Dataset	113229	1077	278661	392967

Table 4.3. Attack names, their categories and numbers with respect to protocol type

Attack name	**Attack category**	**TCP**	**UDP**	**ICMP**	**Total**
Smurf	DoS	-	-	277137	277137
Neptune	DoS	107178	-	-	107178
Back	DoS	2203	-	-	2203
Teardrop	DoS	-	879	-	879
Pod	DoS	-	-	264	264
Land	DoS	21	-	-	21
Normal	Normal	72934	18644	1249	92827
Satan	Probe	1416	170	3	1589
Ipsweep	Probe	94	-	1153	1247
Portsweep	Probe	1039	-	1	1040
Nmap	Probe	103	25	103	231
Warezclient	r2l	1020	-	-	1020
Guess_passwd	R2l	53	-	-	53
Warezmaster	R2l	20	-	-	20
Imap	R2l	12	-	-	12
Ftp_write	R2l	8	-	-	8
Multihop	R2l	7	-	-	7
Phf	R2l	4	-	-	4

Spy	R2l	2	-	-	2
Buffer_overflow	U2r	30	-	-	30
Rootkit	U2r	7	3	-	10
Loadmodule	U2r	9	-	-	9
Perl	U2r	3	-	-	3

4.2. Feature Selection Results

Feature selection process aims to find discriminating feature sets for a SOM in the Anomaly Detection Module. There are 5 SOMs in each of TCP, UDP, and ICMP Anomaly Analyzers: Normal, DoS, probe, u2r, and r2l SOMs. Feature sets, composed of 2 and 3 features, are determined by using information gain which is explained in the 3rd part. Information gains of all combinations of these sets are calculated and the ones that have high information gains are also used for quantization process. This quantization resulted in less difficult feature selection process of feature sets composed of 5 features, and the high gain feature sets of Normal and Dos SOMs are revealed. Feature selection is not applied for the other TCP SOMs, because main idea is to show whether the feature selection algorithm works properly.

4.2.1. TCP Results

Although total number of features is 41 for the whole dataset, it decreases because of having constant value. When TCP connection records are extracted from the dataset, it is seen to have 39 features. Features 8, 20, and 21 have one feature value. Thus, these features are omitted in the calculations for the sake of performance gain.

4.2.1.1. Normal SOM Features: Information required for classifying a sample is 0.9659. In the unique features, maximum information is about 0.80. In the two features and three features groups, information gain increases to 0.96 levels. Information gains of feature(s) are presented in Table 4.4, Table 4.5. and Table 4.6.

Table 4.4. Information gains of top features for normal connections in TCP protocol

Feature #	**Information gain (Max: 0.9659)**
3	0.8489
29	0.8256
30	0.8228
34	0.8077
23	0.80
5	0.7868
33	0.7815
4	0.7542
35	0.7458
6	0.7123

Table 4.5. Top feature groups composed of 2 features for normal connections in TCP protocol

Feature #	**Information gain (Max: 0.9659)**
3, 5	0.9526
5, 34	0.9522
5, 35	0.9471
5, 32	0.9442
5, 33	0.9392
3, 6	0.9352
5, 36	0.9321
5, 37	0.9305
5, 23	0.9258
5, 29	0.9226

Table 4.6. Top feature groups composed of 3 features for normal connections in TCP protocol

Feature #	**Information gain (Max: 0.9659)**
3, 5, 34	0.9645
5, 32, 33	0.9640
3, 5, 32	0.9636
5, 33, 35	0.9631
3, 5, 35	0.9631
5, 33, 34	0.9629
5, 6, 34	0.9627
5, 33, 36	0.9625
3, 5, 23	0.9616
3, 5, 36	0.9610

Information gains of 3 feature groups are used to quantize dataset as explained in the 3rd section. So that, calculation of informatin gains of 5 feature groups becomes possible. The feature groups which have top information gain are shown in Table 4.7. In the last column of the table, vector ranges of the merged features are also exist.

Table 4.7. Top feature groups composed of 5 features for normal connections in TCP protocol

Feature #	**Information gain (Max: 0.9659)**	**Vector range of the merged feature**
1, 3, 4, 5, 34	0.95132	2574
1, 3, 4, 5, 35	0.94769	1894
1, 3, 4, 5, 32	0.94551	7166
1, 3, 4, 5, 33	0.93734	5699
1, 3, 4, 5, 36	0.93378	2323
1, 3, 4, 5, 37	0.93238	1049
1, 3, 4, 5, 23	0.9273	857

1, 3, 4, 5, 29	0.92541	923
1, 3, 4, 5, 30	0.92527	893
1, 3, 5, 6, 36	0.92247	2156

However, these high information feature groups does not perform well, when they are tested by using Self Organizing Map algorithm. The reason for this is concluded to be the high value of the merged feature's vector range. 7th feature group "1, 3, 4, 5, 23" and 27th feature group "1, 3, 4, 6, 30" of the table above have 893 and 866 vector ranges, respectively. The vector ranges of these features are quite smaller than the other high gain feature groups. The performance of them is shown in the Figure 4.1.

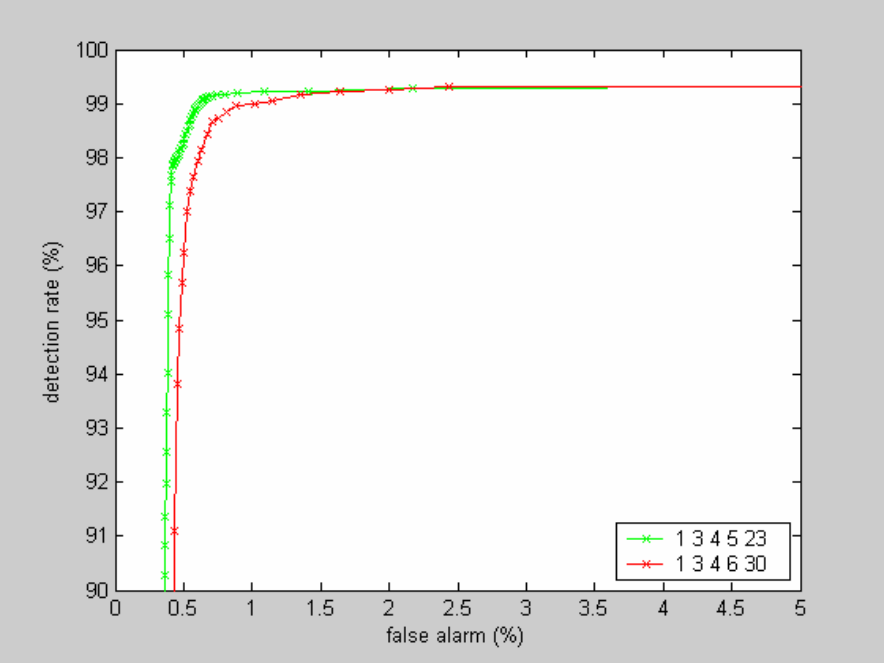

Figure 4.1. Detection-False Alarm rates of the feature groups "1 3 4 5 23" and "1 3 4 6 30"

These features give good results, and also it is noticed that the feature group "1, 3, 4, 5, 23" which has lower vector range permorfs better than the other feature group. These results emphasize the importance of the low level of the vector ranges. It may be even more important than information gain. To understand this, 2 feature groups, one of which has lower information gain and vector range, are tested. These features and performance are given in the Table 4.8, and in the Figure 4.2.

Table 4.8. Information gains and vector ranges of 2 features for Normal SOM

The rank in the table	Feature #	Information gain	Vector range of the merged feature
31	3, 4, 5, 6, 29	0.90971	434
59	3, 4, 9, 10, 29	0.88789	170

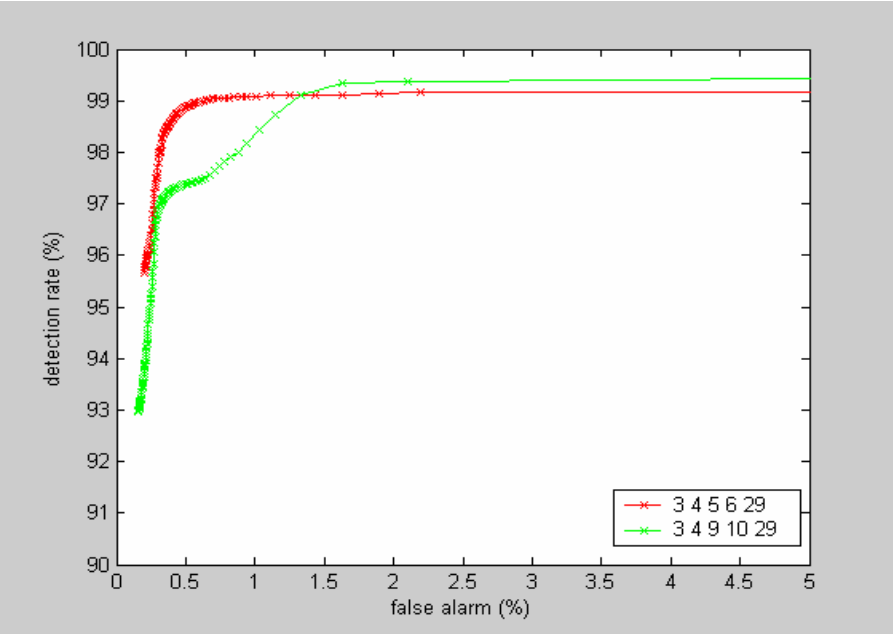

Figure 4.2. Detection-False Alarm rates of the feature groups “3 4 5 6 29” and “3 4 9 10 29”

As shown in the figure, the false-alarm performance of the lower vector range feature group “3, 4, 9, 10, 30” is better in spite of its lower information gain. Thus, it can be concluded that vector range is more important than information gain in some means. In fact, to have high information gain with a low level of vector range is much more difficult because it requires large connection samples of the same characteristics.

Following these results, a new table, which sorts feature groups according to vector ranges, is arranged. The top feature groups are shown in the Table 4.9. The performance of the 2 features “3, 14, 15, 18, 22” and “3, 9, 11, 12, 14” are given in the Figure 4.3.

Table 4.9. Information gains and vector ranges of 5 features with low level of vector ranges for the Normal SOM

Feature #	**Information gain (Max: 0.9659)**	**Vector range of the merged feature**
3, 14, 15, 18, 22	0.00056	26
3, 9, 15, 18, 22	0.00056	26
3, 9, 14, 15, 22	0.00040	27
3, 9, 14, 18, 22	0.00056	27
3, 9, 11, 12, 14	0.6647	28

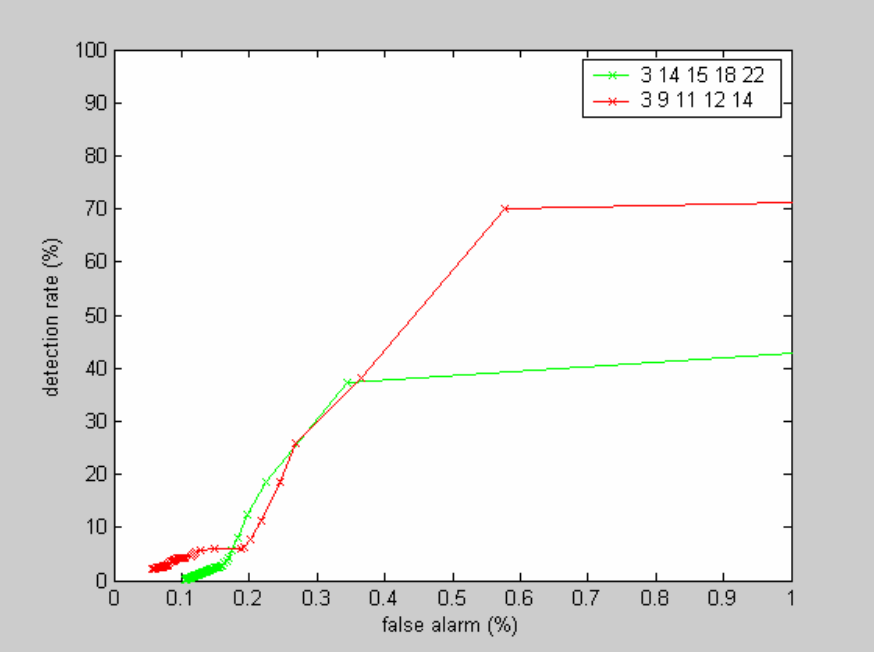

Figure 4.3. Detection-False Alarm rates of the feature groups “3 14 15 18 22” and “3 9 11 12 14”

These feature groups do not have good results because of low detection rate. The reason for that can be low information gain of the feature groups. The information gain of the feature group “3, 9, 11, 12, 14” is quite high in contrast to nearly zero information gain of the other group and it has better performance. Thus, it can be concluded that information gain is not a trivial parameter. To understand the degree of importance of the information gain, 4 feature sets are selected as the Table 4.10. These feature groups have nearly same vector ranges but varying information gains. The performance of the feature groups are given in the Figure 4.4.

Table 4.10. Information gains and vector ranges of 4 feature groups which have near vector ranges for Normal SOM

Feature #	**Information gain (Max: 0.9659)**	**Vector range of the merged feature**
3, 4, 9, 14, 15	0.00050	91
3, 10, 18, 22, 26	0.5232	97
3, 15, 17, 22, 29	0.8290	90
3, 9, 11, 13, 29	0.8765	94

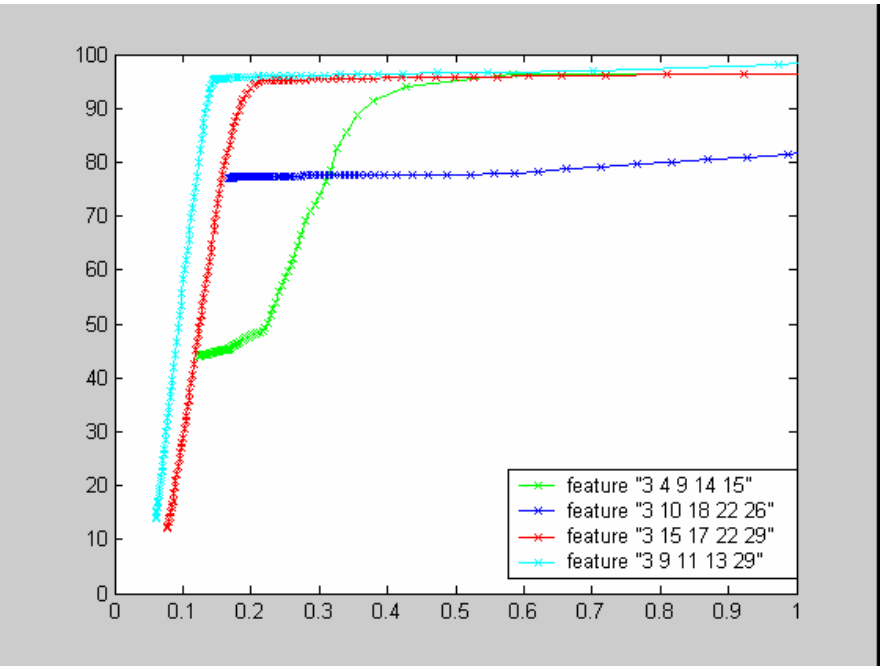

Figure 4.4. Detection-False Alarm rates of the 4 feature groups

As it can be seen in the figure, the feature group that has high information gain results in lower false alarm.

Overall results imply that both information gain and vector range are important parameters, vector range is more important than information gain, but only these two parameters do not give us enough information to find out best feature groups. Because any new parameter does not exist, the feature groups that have high information gains and low vector ranges are trialed. Although this approach does not give exact solution because there may be better feature groups, it shows that good feauture groups that perform well exist. 2 outstanding feature groups among the others and their performance are presented in the Table 4.11. and Figure 4.5 below.

Table 4.11. Information gains and vector ranges of 2 outstanding feature groups

Feature #	**Information gain (Max: 0.9659)**	**Vector range of the merged feature**
3, 9, 12, 13, 30	0.8720	87
3, 11, 12, 13, 30	0.8721	91

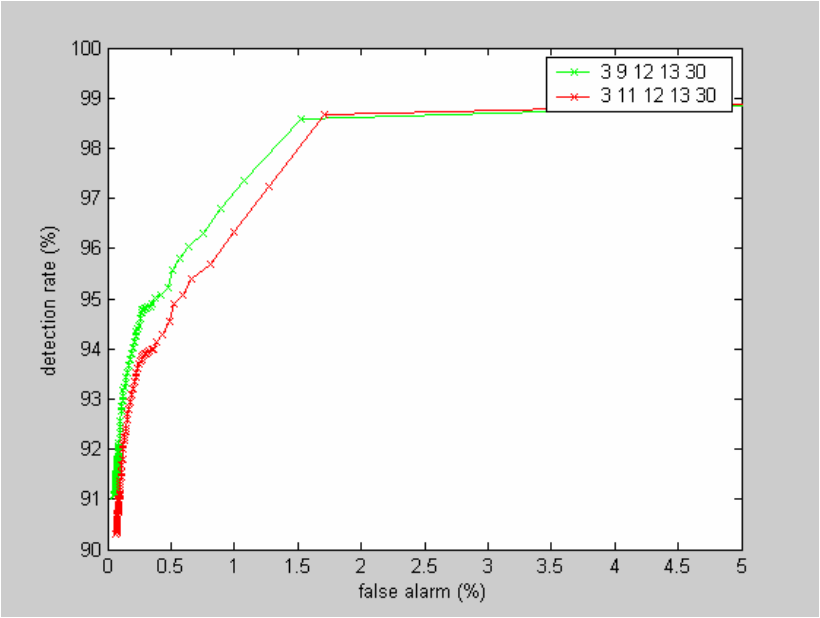

Figure 4.5. Detection-FP rates of the feature groups “3 9 12 13 30” and “3 11 12 13 30”

These feature groups have high detection rates, above 90 %, and low false alarm, below 0.1 %.

4.2.1.2. DoS SOM Features: DoS attacks are more than the other attack groups, and normal connection records. There are 3 DoS attack types as given in Table 4.4. Information gain required to classify a connection record as DoS or not is higher than the other groups. Maximum information for DoS in TCP communication protocol is 0.977. Beside these, information gains of features are also high compared to the other attack groups. The reason for that seems that DoS attacks are easier to detect because of their behavior. Dos attacks have very long or short connection times. And the number of connections connected to hosts is increased. The information gain of every feature is presented in Table 4.12.

Table 4.12. Information gains of top features for DoS attacks in TCP protocol

Feature #	**Information gain (Max: 0.977)**
30	0.8915
29	0.8647
23	0.8583
3	0.8263
34	0.7998
35	0.7892
4	0.7684
33	0.7525
5	0.7502
6	0.6665

Information gains of the 2 feature sets are high with compared to features. Feature set composed of 5 and 35 features has 0.9670 information gain. In the 3 feature groups, information gain improves and reaches nearly maximum information gain rates. The top groups composed of 2 and 3 features are given in Table 4.13 and 4.14.

Table 4.13. Top feature groups composed of 2 features for DoS attacks in TCP protocol

Feature #	**Information gain (Max: 0.977)**
5, 35	0.9670
5, 30	0.9669
6, 30	0.9633
5, 23	0.9612
10, 30	0.9607
13, 30	0.9602
6, 23	0.9488
5, 29	0.9422
30, 40	0.9412
30, 41	0.9411

Although 5th feature does not have high information gain itself, it takes place in the top groups of 3 and 2 feature sets. 5th feature is source bytes feature which is number of bytes sent from source to destination. Feature group 5, 35, and 40 has almost maximum information gain. 35th feature is “dest host diffsrv rate” which is % of different services on the current host. 40th feature, on the other hand, is “dst host rerror rate” meaning % of connections to the current host that have an RST error. In Figure 4.6, vector spaces of DoS attack connections and other connections are presented in three dimensional space whom dimensions are features 5, 35, 40. Blue points, indicating DoS attack vector spaces, place end regions.

Table 4.14. Top feature groups composed of 3 features for DoS attacks in TCP protocol

Feature #	**Information gain (Max: 0.977)**
5, 35, 40	0.9759
5, 23, 35	0.9755
5, 30, 40	0.9754
4, 5, 30	0.9752
5, 30, 35	0.9752

4, 6, 30	0.9751
5, 27, 30	0.9749
5, 23, 40	0.9748
5, 30, 38	0.9747
5, 35, 38	0.9742

In the feature sets composed of 4 features, information gain is lower than that of 3 feature groups. Before calculation of the information gains of the feature sets, quantization of the features are performed according to current knowledge as explained in the Section 3.1.2. The 4 feature sets which are on the top of the list are presented in Table 4.15.

Feature sets composed of 5 features have better information gains compared to 4 feature groups, but lower than 3 feature groups. In 5 feature sets, best group is features 1, 3, 4, 5, and 35 as shown in Table 4.16.

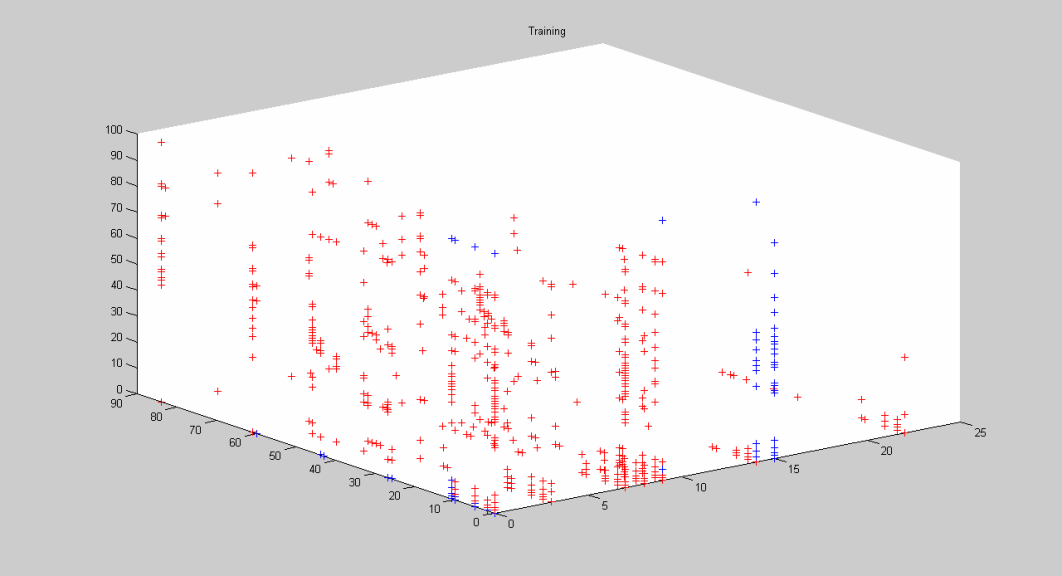

Figure 4.6. Vector spaces of 5, 35, and 40th features showing DoS attacks and others

Table 4.15. Top feature groups composed of 4 features for DoS attacks in TCP protocol

Feature #	**Information gain (Max: 0.977)**
5, 27, 29, 30	0.8918
5, 28, 29, 30	0.8914
3, 5, 29, 30	0.8912
6, 27, 29, 30	0.8911
5, 6, 29, 30	0.8910
5, 25, 29, 30	0.8910
3, 27, 29, 30	0.8910
4, 27, 29, 30	0.8910
5, 24, 29, 30	0.8909
5, 27, 28, 30	0.8909

Table 4.16. Top feature groups composed of 5 features for DoS attacks in TCP protocol

Feature #	**Information gain (Max: 0.977)**
1, 3, 4, 5, 35	0.9701
1, 3, 4, 5, 30	0.9694
1, 3, 4, 5, 23	0.9625
1, 4, 5, 6, 30	0.9619
1, 3, 5, 6, 30	0.9619
1, 3, 4, 6, 30	0.9618
3, 4, 5, 6, 30	0.9617
1, 3, 5, 10, 30	0.9592
1, 3, 4, 10, 30	0.9592
1, 4, 6, 10, 30	0.9591

Because we consider vector ranges of the merged features, a new table, sorting information gains of feature groups which have the lowest vector ranges, is arranged. Following this, the performances of the feature groups are measured in a trial and error approach. Some of the outstanding feature groups are given in the Table 4.17.

Table 4.17. Information gains and vector ranges of outstanding features for DoS SOM

Feature #	**Information gain (Max: 0.9659)**	**Vector range of the merged feature**
4, 9, 12, 22, 30	0.8889	56
4, 9, 12, 13, 30	0.9579	72
4, 9, 11, 12, 30	0.8938	67
4, 6, 9, 13, 30	0.958	80
4, 6, 9, 13, 29	0.9324	88

4.2.1.3. Probe SOM Features: Information required for classifying a sample as normal or probe is 0.2193. In the unique features, maximum information is about 0.20. In the two features and three features groups, information gain increases to 0.21 levels. Information gains of feature(s) are presented in Table 4.18, Table 4.19. and Table 4.20.

Table 4.18. Information gains of top features for probe attacks in TCP protocol

Feature #	**Information gain (Max: 0.2193)**
3	0.2085
34	0.203
33	0.20
35	0.196
27	0.142
4	0.140
40	0.126
5	0.123
23	0.120
29	0.112

Table 4.19. Top feature groups composed of 2 features for probe attacks in TCP protocol

Feature #	**Information gain (Max: 0.2193)**
3, 35	0.21776
4, 34	0.21655
5, 34	0.21634
3, 40	0.21632
4, 35	0.21590
5, 35	0.21588
3, 36	0.21586
34, 35	0.21578
3, 32	0.21546
3, 34	0.21546

Table 4.20. Top feature groups composed of 3 features for probe attacks in TCP protocol

Feature #	**Information gain (Max: 0.2193)**
3, 32, 35	0.21906
3, 34, 40	0.21902
3, 5, 35	0.21895
3, 35, 40	0.21895
3, 34, 35	0.21895
3, 33, 35	0.21895
3, 5, 34	0.21895
3, 32, 40	0.21895
3, 4, 35	0.2188
3, 35, 36	0.2187

4.2.1.4. u2r SOM Features: Information required for classifying a sample as normal or u2r attack is 0.008 because of small numbers of u2r attacks. In the unique features, maximum information is 0.71 in the 6^{th} feature. In the two feature groups, maximum information is of

features 6 and 33 by 0.00735. In the three feature groups, maximum information is at nearly top level. Information gains of feature(s) are presented in Table 4.21, Table 4.22. and Table 4.23.

Table 4.21. Information gains of top features for u2r attacks in TCP protocol

Feature #	**Information gain (Max: 0.0080)**
6	0.0071
5	0.0062
3	0.0041
33	0.0039
14	0.0032
10	0.0030
13	0.0026
17	0.0021
36	0.0017
32	0.0015

Table 4.22. Top feature groups composed of 2 features for u2r attacks in TCP protocol

Feature #	**Information gain (Max: 0.0080)**
6, 33	0.00735
3, 6	0.00718
5, 6	0.00711
3, 5	0.00675
3, 34	0.00671
1, 33	0.00666
3, 35	0.00661
3, 32	0.00658
1, 6	0.00653
5, 33	0.00642

Table 4.23. Top feature groups composed of 3 features for u2r attacks in TCP protocol

Feature #	**Information gain (Max: 0.0080)**
1, 3, 6	0.00799
1, 6, 33	0.00799
5, 6, 33	0.00799
1, 5, 6	0.00796
6, 32, 33	0.00796
3, 6, 34	0.00794
6, 33, 35	0.00793
6, 33, 36	0.00793
3, 6, 35	0.00791
6, 33, 34	0.00790

4.2.1.5. r2l SOM Features: Information required for classifying a sample as normal or r2l attack is 0.1136. As previous attack groups, the information gains of the groups enhance as the number of the features goes up. In the unique features, maximum information is about 0.91. In the two features maximum information gain is 0.1052, and 0.1121 at the three features groups. Information gains of feature(s) are presented in Table 4.24, Table 4.25. and Table 4.26.

Table 4.24. Information gains of top features for r2l attacks in TCP protocol

Feature #	**Information gain (Max: 0.1136)**
5	0.091
3	0.062
33	0.045
36	0.037
10	0.031
37	0.027
24	0.023

23	0.019
22	0.017
32	0.015

Table 4.25. Top feature groups composed of 2 features for r2l attacks in TCP protocol

Feature #	**Information gain (Max: 0.1136)**
5, 6	0.1052
3, 5	0.1047
3, 35	0.0987
3, 36	0.09834
3, 32	0.09740
3, 34	0.09739
5, 33	0.09730
5, 36	0.0940
5, 32	0.0918
5, 10	0.0909

Table 4.26. Top feature groups composed of 3 features for r2l attacks in TCP protocol

Feature #	**Information gain (Max: 0.1136)**
1, 3, 5	0.1121
3, 5, 32	0.1120
3, 5, 34	0.1120
3, 5, 35	0.1118
5, 6, 36	0.1113
3, 5, 36	0.1111
5, 6, 35	0.1110
5, 6, 34	0.1109
5, 32, 33	0.1109
5, 6, 32	0.1109

4.3. Performance Evaluation of the Anomaly Detection Module

Since feature selection of 5 feature sets is completed for only Normal and DoS SOM on TCP protocol, the performance of these SOMs are given. Each SOM is trained by using 50 % of the normal samples by using bootstrap method. Test, on the other hand, is performed by inputting normal samples and attack samples for which SOM is allocated. Thus, DoS SOM, for example, is not tested for the other attack groups. Only normal SOM is tested for all attack groups because of responsibility of detecting all of them.

Before training phase, feature sets and quantization table specified for each SOM are determined according to methods explained in Section 3.1 and 3.2.

50 % of the normal samples of 10 % KDD dataset are used for the training. SOM training parameters are given in Table 4.27. First of all, specified protocol type connection samples, TCP, UDP, or ICMP, and preselected features of them are extracted in the preprocessing module. In the Specific Quantization Module, they are quanted according to SOM type, normalized and given to the Anomaly Detection Module. At the end of the training phase, trained sMaps are evaluated. Beside these, maximum error rate for any neuron, which indicates maximum error rate of training samples matching that specific neuron, is found. If the maximum error rate is high than global threshold, maximum error rate is decreased to the global threshold value. In the test phase, best matching neuron and its error rate are determined for every sample. If a test sample has a high error rate than matching neuron's maximum error rate, it is labeled as anomaly. Based on all results, false positive and false negative rates are calculated.

$$false\ positive\ rate(f_p) = \frac{false\ positives}{normal\ samples} \tag{4.1}$$

$$false\ negative\ rate(f_n) = \frac{false\ negatives}{attack\ samples} \tag{4.2}$$

Table 4.27. Training parameters of the SOM structures

SOM Structure		
Map Size	20 x 20	
Map Shape	Rectangular	
Initialization type	Random	
Training Parameters	Ordering Phase	Fine Tuning Phase
Neighbourhood function	Gaussian	Gaussian
Radius[initial final]	[15 1]	[1 1]
Learning type	Sequential	Sequential
Learning rate	0.6	0.05
Learning function	Linear	Linear
Mask	[1 1 ... 1]	[1 1 ... 1]
Epochs	1	1

4.3.1. TCP Anomaly Analyzer

There are 5 SOMs for each protocol type. The performance of only Normal and DoS SOM is given because their feature sets are determined. For normal, probe, u2r, and r2l SOMs, only information gains of 3 feature groups are calculated in the feature selection step.

All results are compared to the basic feature set which is 1, 3, 4, 5, 6. For some modules more than one feature group are preferred for performance reasons.

4.3.1.1. Normal SOM Results: The best feature group among experimented ones is found to be “3 9 12 13 30” in the feature selection step. The comparison with basic feature group “1 3 4 5 6” is shown in the Figure 4.7. Feature group “3 9 12 13 30” performs better in the low false alarm rate region. The open ended line points out clustering of the connection samples in the two dimensinal space of the Self Organizing Map. At the tip point of the line, SOM can be used without threshold value. The clustering capability of the SOM is shown in a histogram in the Figure 4.8 such that green and red corresponds to the normal and attack samples, respectively.

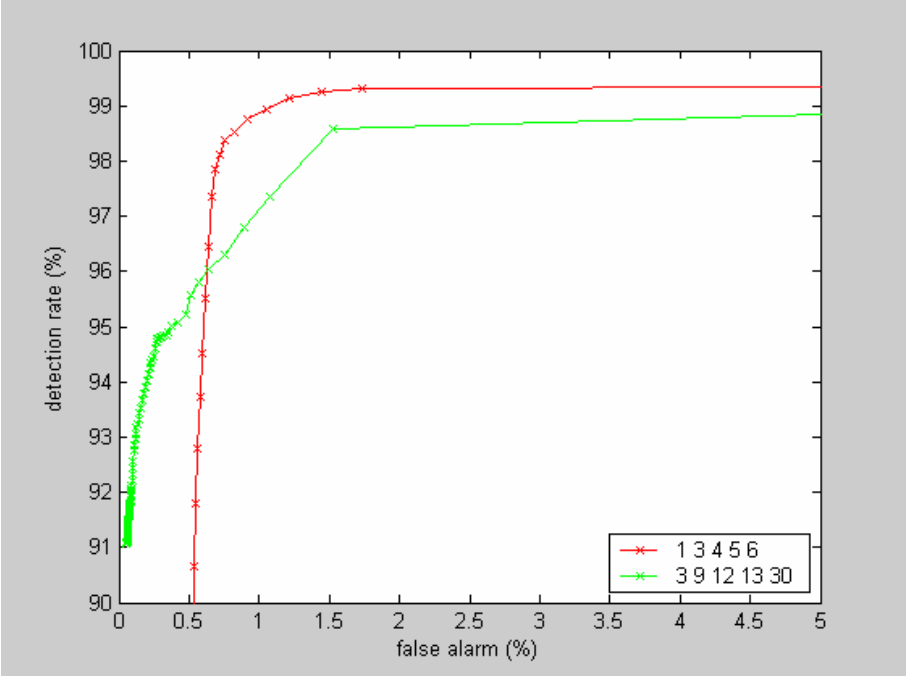

Figure 4.7. Detection-false alarm rate of feature groups “1 3 4 5 6” and “3 9 12 13 30”

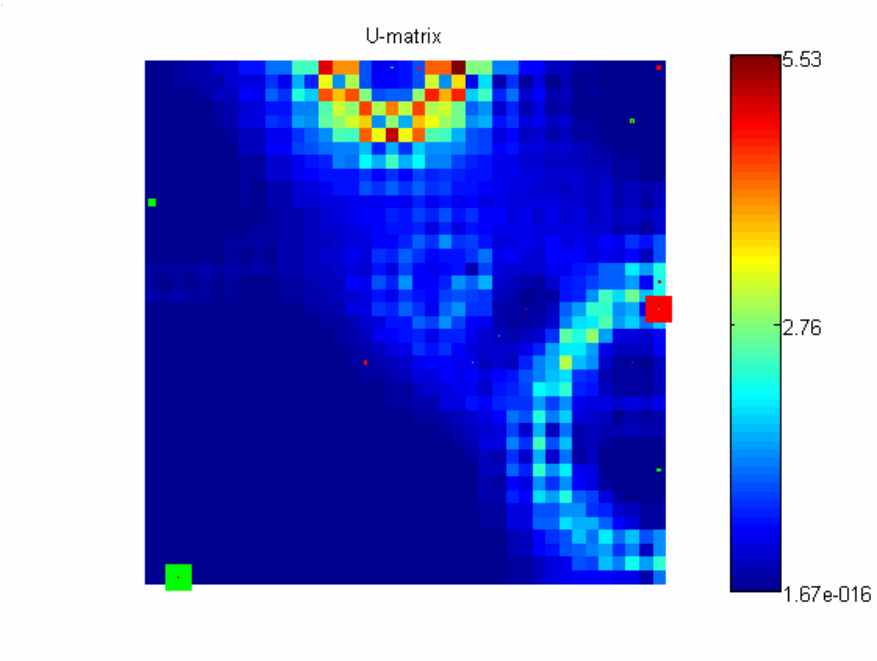

Figure 4.8. Histogram of training and all attack samples in sMap of the Normal SOM trained by feature group “3 9 12 13 30” in TCP protocol

4.3.1.2. DoS SOM Results: Some outstanding feature groups are given in the Table 4.17 in feature selection step. The ROC curves of these features are shown in the Figure 4.9. The

clustering capability of the SOM for the feature “4 6 9 13 29” is shown in a histogram in the Figure 4.10.

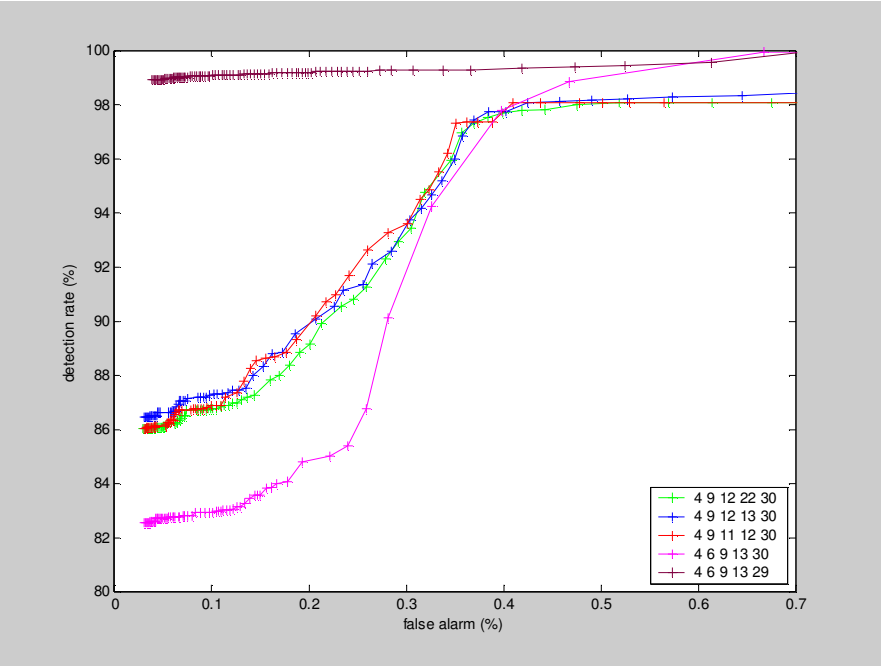

Figure 4.9. Detection-false alarm rate of 5 feature groups for the DoS attack detection

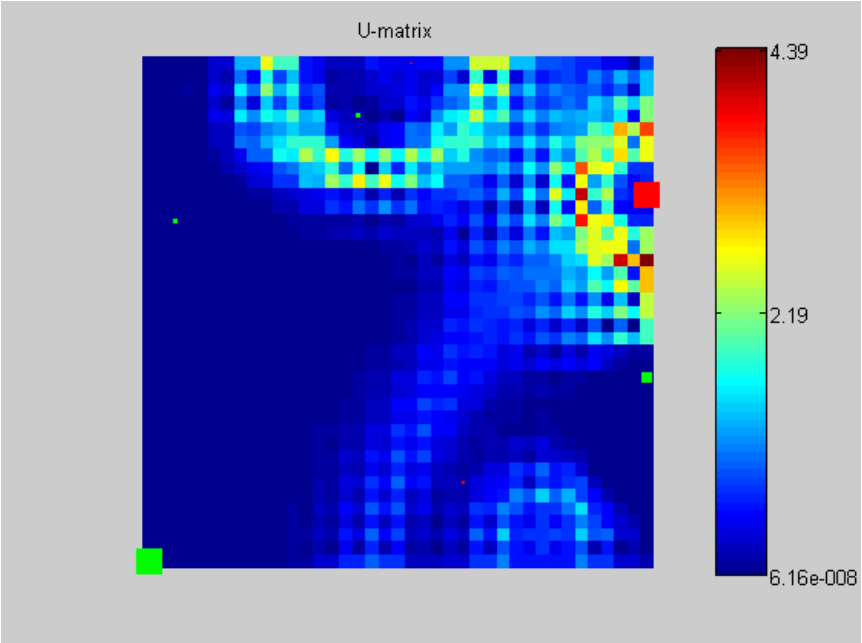

Figure 4.10. Histogram of training and all attack samples in sMap of the DoS SOM trained by feature group “4 6 9 13 29” in TCP protocol

5. CONCLUSION AND FUTURE WORK

This work's main contribution is a novel feature selection method in means of information gain. By using information gain, useful feature sets are found, and their performance is measured. An entropy based quantization technique is also proposed. The results show that there are numerous number of feature groups which perform well. If the information gain of a merged feature, which is formed from a feature group, is high and vector range of it is low, the feature group is a candidate for a proper selection. However, the exact relation of being a useful feature group is unknown. Thus, more parameters are needed for the feature selection process beside information gain and vector range.

The usage of multi SOMs gives advantage of focusing on attack groups which show similar characteristics. It is seen that, useful feature groups can be found for the other attack groups, namely probe, u2r, and r2l attacks. However, this process requires computational power, and it is difficult to find n useful features as n goes up. Although a quantization process is defined for the work, its performance for the number of features in a group above 5 is not measured. Because quantization is realized according to a threshold value, the selection of the threshold value is important, and it is required to decrease this value as the n increase for the sake of long interval quantization. However, this may cause information loss if it is selected too low. Thus, the determination process of the threshold value should be explored. If it is properly defined, the selection of valuable feature sets, which are composed of 6, 7 or 8 features, can become possible.

On the other hand, the information acquired from SOMs can be increased by using the relationship and distance between neurons. Detection is performed by error value of the best matching neuron for an input test vector, but not the place of the neuron. If the error rate is above the threshold value, the test input is determined to be anomaly. Thus, beside the neuron itself, showing normal or anomaly behavior, the neuron groups can be thought as a unique neuron, and the relationship between neurons can be analyzed for better performance. Beside these, s better performing decision support system is required to decide according to information coming up from multiple SOMs. Otherwise, the false

alarm rate of system may increase due to parallel construction of the SOMs. For this purpose, a fuzzy logic or neural network approach can be researched.

APPENDIX A: KDD 99 DATASET CATEGORIES

Information about features of KDD 99 dataset is presented in Table A.1, A.2, and A.3.

Table A.1. Basic features of individual TCP connections

Feature	Description	Type
1. Duration	Duration of the connection	Cont.
2. Protocol Type	Connection protocol (e.g. tcp, udp)	Disc.
3. Service	Destination Service	Disc.
4. Flag	Status flag of the connection	Disc.
5. Source Bytes	Bytes sent from source to destination	Cont.
6. Destination Bytes	Bytes sent from destination to source	Cont.
7. Land	1 if connection is from/to the same host/port; 0 otherwise	Disc.
8. Wrong Fragment	Number of wrong fragments	Cont.
9. Urgent	Number of urgent packets	Cont.

Table A.2. Contents features within a connection suggested by domain knowledge

Feature	Description	Type
10. Hot	Number of “hot” indicators	Cont.
11. Failed logins	Number of failed logins	Cont.
12. Logged in	1 if successfully logged in; 0 otherwise	Disc.
13. # of compromised	Number of “compromised” conditions	Cont.
14. Root shell	1 if root shell is obtained; 0 otherwise	Cont.
15. Su attempted	1 if “su root” command attempted; 0 otherwise	Cont.

16. # of root	Number of root accesses	Cont.
17. # of file creations	Number of file creation operations	Cont.
18. # of shells	Number of shell prompts	Cont.
19. # of access files	Number of operations on access control files	Cont.
20. # of outbound cmds	Number of outbound commands in an ftp session	Cont.
21. is hot login	1 if the login belongs to the "hot" list; 0 otherwise	Disc.
22. is guest login	1 if the login is a "guest" login; 0 otherwise	Disc.

Table A.3. Content based features computed using a two second time window

Feature	**Description**	**Type**
23. count	Number of connections to the same host as the current connection in the past two seconds	Cont.
24. srv count	Number of connections to the same service as the current connection in the past two seconds	Cont.
25. serror rate	% of connections that have "SYN" errors	Cont.
26. srv serror rate	% of connections that have "SYN" errors	Cont.
27. rerror rate	% of connections that have "REJ" errors	Cont.
28. srv rerror rate	% of connections that have "REJ" errors	Cont.
29. same srv rate	% of connections to the same service	Cont.
30. diff srv rate	% of connections to the different services	Cont.
31. srv diff host rate	% of connections to the different hosts	Cont.
32. dst host count	count of connections having the same destination host	Cont.
33. dst host srv count	count of connections having the same	Cont.

	destination host and using the same service	
34. dst host same srv rate	% of connections having the same destination host and using the same service	Cont.
35. dst host diff srv rate	% of different services on the current host	Cont.
36. dst host same src port rate	% of connections to the current host having the same src port	Cont.
37. dst host srv diff host rate	% of connections to the same service coming from different hosts	Cont.
38. dst host serror rate	% of connections to the current host that have an S0 error	Cont.
39. dst host srv serror rate	% of connections to the current host and specified service that have an S0 error	Cont.
40. dst host rerror rate	% of connections to the current host that have an RST error	Cont.
41. dst host srv rerror rate	% of connections to the current host and specified service that have an RST error	Cont.

REFERENCES

1. Axelsson, S., *Intrusion Detection Systems: A Survey and Taxonomy*, Technical Report No:99-15, Dept. of Computer Engineering, Chalmers University of Technology, Sweden, 2000.
2. Kabiri P., and A. A. Ghorbani, "Research on Intrusion Detection and Response: A Survey", *International Journal of Network Security*, ISSN 1816-3548, vol 1. no. 2, pp. 84-102, 2005.
3. Kayacik H. G., A. N. Zincir-Heywood, and M. I. Heywood, "On the capability of an som based intrusion detection system", *Proceedings of the International Joint Conference on Neural Networks*, vol. 3, pp. 1808–1813. IEEE, 2003.
4. Kohonen, T. *Self-Organizing Maps (3rd ed.)*, Springer-Verlag, Berlin, 2001.
5. Kayacık, H.G., *Hierarchical Self Organizing Map Based IDS on KDD Benchmark*, M.S. Thesis, Dalhousie University, Faculty of Computer Science, 2003.
6. Lichodzijewski P., A. N. Zincir-Heywood, and M. I. Heywood, "Host-Based Intrusion Detection Using Self-Organizing Maps", *Proceedings of the IEEE Int. Joint Conf. Neural Netw.*, 2002, pp. 1714–1719, 2002.
7. Lee S. C., D. V. Heinbuch, "Training a Neural-Network Based Intrusion Detector to Recognize Novel Attacks", *IEEE Workshop Inform. Assurance Security*, West Point, New York, 2000.
8. Cho S., S. Hun, "Two Sophisticated Techniques to Improve HMM-Based Intrusion Detection Systems", *Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection (RAID),* 2003.
9. Depren M.Ö., M. Topallar, E. Anarım, K. Cılız, "An Intelligent Intrusion Detection System (IDS) for Anomaly and Misuse Detection in Computer Networks", *Expert Systems with Applications*, 2005.
10. Bolzoni D., S. Etalle, P. Hartel, and E. Zambon "POSEIDON: a 2-tier Anomaly-based Network Intrusion Detection System", *Proceedings of 4th IEEE International Workshop Information Assurance,* 2006.

11. Zanero S., S. M. Savaresi, "Unsupervised learning techniques for an intrusion detection system", *Proceedings of the ACM Symposium on Applied Computing*, ACM SAC, 2004.
12. Yoo I., U. Ultes-Nitsche, "Adaptive Detection of Worms/Viruses in Firewalls", *Proceedings of International Conference on Communication, Network and Information Security,* CNIS, 2003.
13. Yang Z., and A. Karahoca, "An Anomaly Intrusion Detection Approach Using Cellular Neural Networks", *ISCIS*, LNCS 4263, pp. 908 – 917, 2006.
14. Chen R., and S. Chen, "An Intrusion Detection Based on Support Vector Machines with a Voting Weight Schema", *IEA/AIE*, LNAI 4570, pp. 1148–1157, 2007.
15. Sung A. H., S. Mukkamala, "Identifying important features for intrusion detection using support vector machines and neural networks", *Proceedings of International Symposium on Applications and the Internet*, p.209-17, 2003.
16. Kayacık, H.G., A.N. Zincir, M.I. Heywood, "Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Dataset", *Proceedings of the Third Annual Conference on Privacy, Security and Trust*, St. Andrews, NB, Canada, 2005.
17. Chebrolu S., A. Abraham, J.P. Thomas, "Feature Deduction and Ensemble Design of Intrusion Detection Systems", *ELSEVIER Computers and Security,* 24, 295-307, 2005.
18. Han J. and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, 2000.
19. Li, H., K. Zhang, and T. Jiang, "Minimum Entropy Clustering and Applications to Gene Expression Analysis", *Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference*, 2004.
20. Vesanto J., J. Himberg, E. Alhoniemi, and J. Parhankangas, *SOM Toolbox for Matlab 5*, Tech. Rep. A57, Helsinki University of Technology, 2000.