International Journal of Database Theory and Application
Vol.6, No.5 (2013), pp.165-174
http://dx.doi.org/10.14257/ijdta.2013.6.5.15

Developing a hybrid method of Hidden Markov Models and C5.0 as a Intrusion Detection System

Mahsa Khosronejad, Elham Sharififar, Hasan Ahmadi Torshizi and Mehrdad Jalali

Department of Software Engineering, Islamic Azad University, Mashhad branch, Mashhad, Iran
Khosronejad.m3@gmail.com, Sharififar_el65@yahoo.com, Ahmadi_torshizi@yahoo.com, jalali@mshdiau.ac.ir

Abstract

In today's communication system computer and information security is a major concern as these are vulnerable to potential attackers. Because to increase the potential of advanced computer communication and distributed systems leads to attack on the data flow over the network which affects integrity and availability of information greatly. Therefore, the security of Web applications is a key topic in computer security. This paper presents two hybrid approaches for modeling IDS. C5.0 and HMM are combined as a hierarchical hybrid intelligent system model (C5.0-HMM). Empirical results with KDD Cup 99 Intrusion data illustrate that the proposed hybrid systems provide more accurate intrusion detection systems.

Keywords: *Intrusion Detection System (IDS), Hidden Markov Model (HMM), hybrid intelligent system and C5.0*

1. Introduction

Nowadays Web-based services and social networking platforms are quite common, and their number is still increasing as the Web-based architecture is the most frequently used in software deployments. The results of a recent study by the X-Force team show that approximately 50% of vulnerabilities discovered during 2009 affected Web applications. In consequence of this, intrusions into computer systems have grown enormously. Attacks are increased both in quantity. Therefore, the security of Web applications is a key topic in computer security [1]. Intrusion Detection System (IDS) is a tool that can detect attacks with practically reasonable accuracy.

IDS is a defense system which assumes that the attacker gained an authorized access. It tries to identify the attackers by scanning the behavior of active users over the network and computing system. If a user exhibits a different characteristic then the normal user profiles, then it is identified as an attacker [2].

The IDS is an identification system which can be characterized based on false acceptance and false rejection probability. False acceptance means the IDS allow the intruders to continue their activity whereas false rejection is termed as probability to stop the activity of a legitimate user. Generally, an IDS analyses information patterns of network and host activity. The IDS logs the network event and looks after the existing system logs. Then it analyses the event logs to determine if any suspicious activity is going on. The IDS event analyzer uses knowledge of previous attacks and system vulnerabilities to identify intrusion.

The IDS uses two techniques according to the type of information used for intrusion detection: misuse detection and anomaly detection.

Misuse detection uses knowledge about attacks. It attempts to model the attacks on a system as specific patterns and systematically scans the network and system events for each occurrence of the patterns. The advantage of this technique is that, the known attacks

ISSN: 2005-4270 IJDTA

can be detected efficiently with low false positive error and it is economical enough as it requires scanning of known attack patterns. The disadvantage of this technique is that it suffers from detecting the new kind of generated attacks.

Anomaly detection technique is able to detect novel or newly generated and unknown attack, because it attempts to detect intrusions that have a significant deviation from normal behavior of a legitimate user. But drawback of anomaly detection technique is that the nonintrusive behavior failing outside the normal range maybe identified as an intrusion which in turn results high false positive error. Also a large amount of data and audit trail is to analyzed to model normal behavior [3].

The remainder of this paper is organized as follows. Section 2 deals with the previous studies on Intrusion Detection System. The underlying methodology of the proposed approach, the hybrid method of Hidden Markov Models and C5.0, is briefly introduced in Section 3. The proposed approach is explained and illustrated with a case study in Section 4. The paper ends with conclusions in Section 5.

2. Related Works

In recent years a vast majority of research activities in the area of anomaly detection have been focused on studying the behaviour of programs and the creation of their profiles based on system call log files [4].

There are a number of methods for constructing IDS models such as machine learning [5-7], Hidden Markov Models [8, 9],statistical profiling [10], and data mining [11]. Among these approaches, Hidden Markov Models (HMM) have been shown to be very promising for anomaly detection over several other techniques because of their high accuracy in identifying intrusions[3]. However, the HMM-based algorithms suffer from long training time during the construction of the models, which hinders their efficiency [9].

In this paper, we improve the performance of Hidden Markov Model (HMM) by combining it with C5.0 method. The motivation for using the hybrid approach is to improve the accuracy of the intrusion detection system when compared to using individual HMM approach. To evaluate quality of our proposed method we use DARPA dataset.

We will survey past research concerning data mining, C5.0 and Hidden Markov Model in follow.

2.1. Data Mining

Due to the information technology improvement and the growth of internet, enterprises are able to collect and to store huge amount of data. These massive databases often contain a wealth of important data that traditional methods of analysis fail to transform into relevant knowledge. Specifically, meaningful knowledge is often hidden and unexpected, and hypothesis driven methods, such as on-line analytical processing (OLAP) and most statistical methods, will generally fail to uncover such knowledge. Inductive methods, which learn directly from the data without an a priori hypothesis, must therefore be used to uncover hidden patterns and knowledge [5].

As it is shown in **Error! Reference source not found.**ach data mining technique can perform one or more of the following types of data modelling:

(1) Association;
(2) Classification;
(3) Clustering;
(4) Forecasting;
(5) Regression;
(6) Sequence discovery;

(7) Visualization.

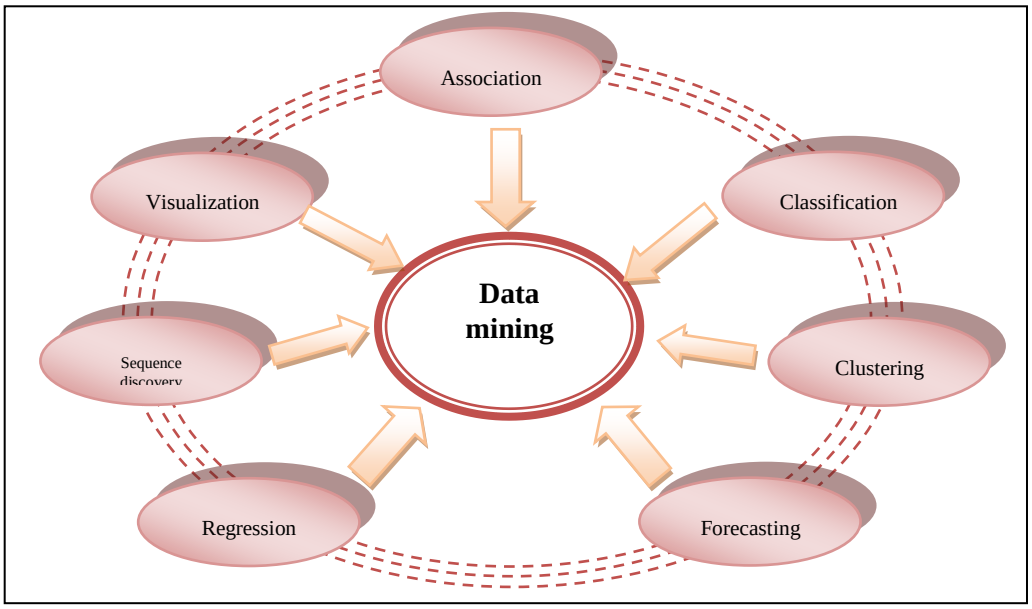

Figure .1 Data Mining Techniques

Here are a few specific things that data mining might contribute to an intrusion detection project:

- Remove normal activity from alarm data to allow analysts to focus on real attacks
- Identify false alarm generators and "bad" sensor signatures
- Find anomalous activity that uncovers a real attack
- Identify long, ongoing patterns (different IP address, same activity)

To accomplish the above tasks data miners employ one or more of the following techniques:

- Data summarization with statistics
- Visualization *i.e.*, presentation of a graphical summary of the data
- Clustering of the data into natural categories
- Association rule discovery: defining normal activity and enabling the discovery of anomalies
- Classification: prediction of the category to which a particular record belongs [3, 12]

2.2. C5.0

The C5.0 algorithm is a new generation of Machine Learning Algorithms (MLAs) based on decision trees. It means that the decision trees are built from list of possible attributes and set of training cases, and then the trees can be used to classify subsequent sets of test cases. C5.0 was developed as an improved version of well-known and widely used C4.5 classifier and it has several important advantages over its ancestor. The generated rules are more accurate and the time used to generate them is lower (even around 360 times on some data sets). In C5.0 several new techniques were introduced:

- boosting: several decision trees are generated and combined to improve the predictions.
- variable misclassification costs: it makes it possible to avoid errors which can result in a harm.
- new attributes: dates, times, timestamps, ordered discrete attributes.
- values can be marked as missing or not applicable for particular cases.

- supports sampling and cross-validation.

The C5.0 classifier contains a simple command-line interface, which was used by us to generate the decision trees, rules and finally test the classifier[13].

2.3. Hidden Markov Model

The Hidden Markov Model (HMM) starts with a finite set of states. Transitions among the states are governed by a set of probabilities (transition probabilities) associated with each state. In a particular state, an outcome or observation can be generated according to a separate probability distribution associated with the state. It is only the outcome, not the state, that is visible to an external observer. The states are "hidden" to the outside; hence the name Hidden Markov Model. The Markov Model used for the hidden layer is a first-order Markov Model, which means that the probability of being in a particular state depends only on the previous state. While in a particular state, the Markov Model is said to "emit" an observable corresponding to that state. One of the goals of using an HMM is to deduce from the set of emitted observables the most likely path in state space that was followed by the system.

Given the set of observables contained in an example corresponding to an attack, an HMM can also determine the likelihood of an attack of a specific type. In our case, the observables are the set of alerts contained in the example. Each example is constructed to contain all of the alerts that occurred between a specified source host and a specified target host, over a specified time period (in our case 24 hours). The goal for the HMM is therefore to determine the most likely attack type corresponding to a sequence of alerts.

To train the HMM, a set of training examples is created, with each example labeled by category. The maximum likelihood method is used to adjust the HMM parameters for optimal classification of the example set. The HMM parameters are the initial probability distribution for the HMM states, the state transition probability matrix, and the observable probability distribution.

The state transition probability matrix is a square matrix with size equal to the number of states. Each of the elements represents the probability of transitioning from a given state to another possible state. The matrix is not symmetric. For example, the likelihood of transitioning from the state corresponding to an *ip scan* into the state corresponding to a *port probe* is very high, since these two events are likely to occur in proximity to each other in the order given. However, the transition *port probe* to *ip scan* is much less likely, since port probes are usually conducted on hosts that have been identified by ip scans. The observable probability distribution is a non-square matrix, with dimensions number of states by number of observables. The observable probability distribution represents the probability that a given observable will be emitted by a given state. For example, in the late stages of an attack, the observable "root login" would be much more likely than "port probe," since port probes typically happen at the beginning of an attack and a root login typically happens towards the end [2].

3. Research Method

As previously mentioned, in our research, we aim to improve the accuracy of the original HMM algorithm. Therefore, this study proposes a new procedure, joining C5.0 and HMM algorithms to extract meaning knowledge from records about Intrusions to classify them. This section briefly introduces the research model of this study and the proposed procedure for classifying Intrusions.

Our research methodology consists of three major steps: data collection, data processing and model construction (Figure 3).

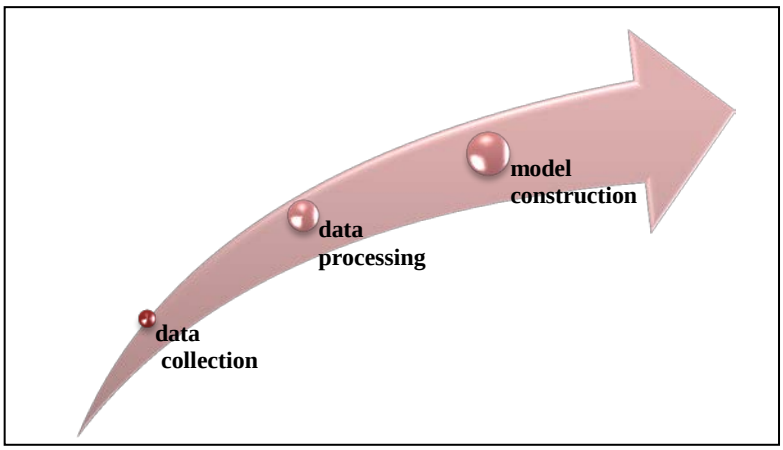

Figure 2. Steps of our Research Methodology

3. 1. Data Collection

To verify our proposed model for IDS problem, we used DARPA data set taken from [14]. The MIT Lincoln Laboratory under DARPA and AFRL sponsorship, has collected and distributed the first standard corpora for evaluation of computer network Intrusion Detection Systems (IDS). This DARPA evaluation dataset is used for the purpose of training as well as testing the intrusion detectors. These evaluations contributed significantly to the intrusion detection research by providing direction for research efforts and an objective calibration of the technical state-of-the-art. They are of interest to all researchers working on the general problem of workstation and network intrusion detection.

The raw data was processed into connection records, which consist of about 5 million connection records. The data set contains 24 attack types. These attacks fall into four main categories:

1. Denial of service (DOS): In this type of attack an attacker makes some computing or memory resources too busy or too full to handle legitimate requests, or denies legitimate users access to a machine. Examples are Apache2, Back, Land, Mailbomb, SYN Flood, Ping of death, Process table, Smurf, Teardrop.
2. Remote to user (R2L): In this type of attack an attacker who does not have an account on a remote machine sends packets to that machine over a network and exploits some vulnerability to gain local access as a user of that machine. Examples are Dictionary, Ftp_write, Guest, Imap, Named, Phf, Sendmail, Xlock.
3. User to root (U2R): In this type of attacks an attacker starts out with access to a normal user account on the system and is able to exploit system vulnerabilities to gain root access to the system. Examples are Eject, Loadmodule, Ps, Xterm, Perl, Fdformat.
4. Probing: In this type of attacks an attacker scans a network of computers to gather information or find known vulnerabilities. An attacker with a map of machines and services that are available on a network can use this information to look for exploits. Examples are Ipsweep, Mscan, Saint, Satan, Nmap.

3.2. Data Processing

The data preparation phase covers all activities to construct the final dataset (data that will be fed into the modeling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed multiple times and not in any prescribed order. Basic tasks in data preparation phase are as follows [15]:

Data table and record: The data sources are first located, accessed, and integrated. Next, selected data is put into a tabular format in which instances and variables take place in rows and columns, respectively.

Data cleaning involves techniques for filling in missing values, smoothing out noise, handling outliers, detecting and removing redundant data.

Data integration and transformation: Sometimes it is useful to transform the data into a new format in order to extract additional information. It is useful to be able to summarize a large data set of data and present it at a high conceptual level. Dates are a good example of data that you may want to handle in special ways. Any date or time can be represented as the number of days or seconds since a fixed point in time, allowing them to be mapped. In the data matrix, the month of the year is used instead of date for detecting seasonal knowledge [16].

Data reduction and projection: This includes finding useful features to represent the data (depending on the goal of the task) and using dimensionality reduction, feature discretization, and feature extraction (or transformation) methods. Application of the principles of data compression can play an important role in data reduction and is a possible area of future development, particularly in the area of knowledge discovery from multimedia data set [17].

Discretisation: This is a form of data reduction, reduces the number of levels of an attribute by collecting and replacing low-level concepts with high-level concepts [17].

3.3. Model Construction

Model construction, in software engineering, is the process of creating a data model by making descriptions of formal data models, using data modeling techniques [5]. In this phase, various modeling techniques are selected and applied and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type. Model construction is the critical part in Intrusion Detection Systems. Modeling technology can provide quantitative methods for the analysis of data, to represent, or acquire expert knowledge, using inductive logic programming, or algorithms, so that AI, cognitive science and other research fields are afforded broader platforms for the development of IDS [4, 6]. This paper presents a hybrid intelligent system for IDS problem.

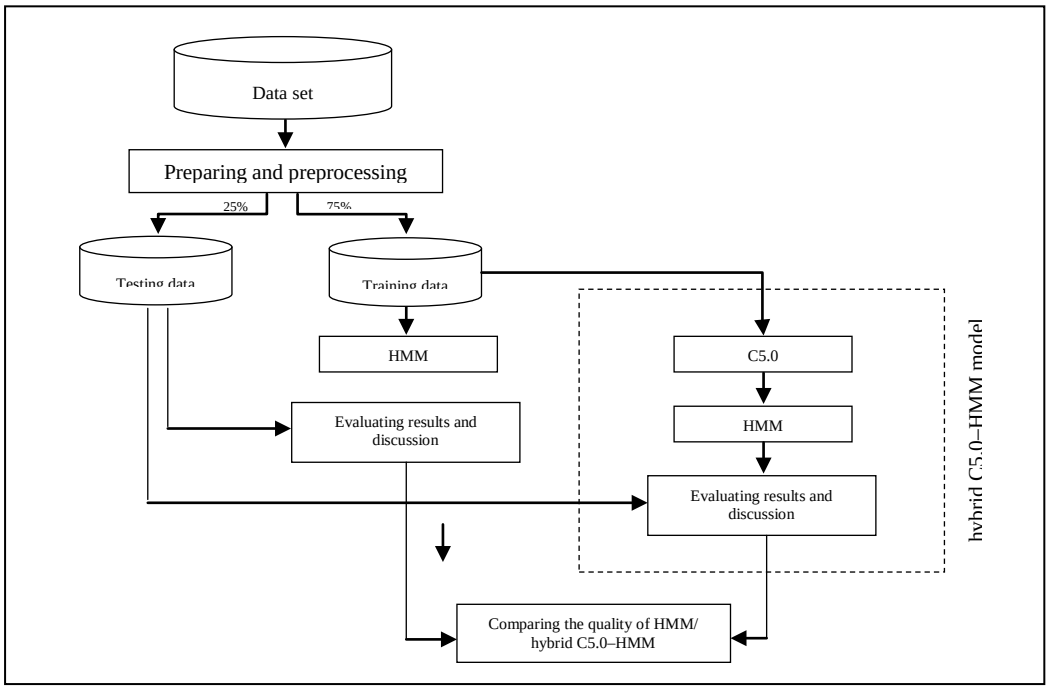

Figure 3. Information Flow of Proposed Model

A hybrid intelligent system uses the approach of integrating different learning or decision-making models. Each learning model works in a different manner and exploits different set of features. Integrating different learning models gives better performance than the individual learning or decision-making models by reducing their individual limitations and exploiting their different mechanisms. We combine HMM with C5.0 as Our proposed hybrid intelligent system to improve individual HMM, then investigate and evaluate the performance of HMM and hybrid C5.0–HMM. Figure 3 shows the architecture of the hybrid intelligent system with C5.0 and HMM.

The data set is first passed through the C5.0 and data classification is generated. This data classifier (as an additional attribute) along with the original set of attributes is passed through the HMM to obtain the final output. The key idea here is to investigate whether the data classification provided by the C5.0 will improve the performance of the HMM.

4. Implementation of Proposed Approach

The data set has 41 attributes for each connection record plus one class label. R2L and U2R attacks don't have any sequential patterns like DOS and Probe because the former attacks have the attacks embedded in the data packets whereas the later attacks have many connections in a short amount of time. Therefore, some features that look for suspicious behavior in the data packets like number of failed logins are constructed and these are called content features. Our experiments have two phases, namely, a training and a testing phase. In the training phase the system constructs a model using the training data to give maximum generalization accuracy (accuracy on unseen data). The test data is passed through the constructed model to detect the intrusion in the testing phase. Besides the four different types of attacks mentioned in Section 3-1, we also have to detect the normal class. The data set for our experiments contained 1048576 records [14]. 20 different data sets randomly generated from the MIT data set. These data sets include the 5 – 100% with step 5 of records. Each data set is again divided into training data and testing with 75 and 25% partition size respectively. All the intrusion detection models are trained and tested with the same set of data. Experiments were performed using an Intel Core 2 Duo, 2.1 GHz processor with 2 GB of RAM.

We have implemented a C5.0 approach on data sets with SPSS Clementine 13.0. Next, new data set (original data set with an additional attribute) is passed through the HMM to obtain the final output. Here, we also used the classifiers.bayes.HMM class of Weka 3.7.4 with follows parameters to implement the HMM model.

Covariance Type: Full matrix (unconstraint)
Iteration Cutoff: 0.01
LeftRight: False
NumState: 6
Random State Initializes: False
Seed: 1
Tied: False

After construction of each model, we verified all of them by applying the cross validation technique. Figure 4 and Table 1 compares the performance of the HMM algorithm against the hybrid C5.0–HMM for several data set that are randomly selected from original DARPA data set. Number size of data sets is 5-95 % of total records number (1048576).

Table 1. Accuracy of Original HMM Algorithm and Proposed Model for Data Sets with Different Number Size

Number size	Accuracy of HMM (%)	Accuracy of hybrid C5.0–HMM (%)	Improvement (%)
5	57.86	56.88	1.71
10	57.69	56.80	1.56
15	56.87	57.03	-0.28
20	57.61	56.76	1.51
25	56.88	56.89	-0.02
30	57.65	56.77	1.54
35	57.61	56.86	1.31
40	57.57	56.83	1.31
45	56.94	56.76	0.31
50	56.92	56.64	0.49
55	57.61	56.77	1.48
60	56.89	56.80	0.17
65	56.85	56.77	0.14
70	57.61	56.79	1.44
75	56.88	56.87	0.01
80	57.58	56.85	1.28
85	57.59	56.82	1.35
90	heap size Error	heap size Error	-
95	heap size Error	heap size Error	-
Average	**56.81799**	**57.3303**	0.9

As it is seen in Table 1 , The average accuracy of the hybrid C5.0–HMM is 57.3%, whereas it is noticeably low (56.8%) for the original HMM algorithm. However, hybrid C5.0–HMM is about 0.9% better than HMM in the later generations. This is because the hybrid C5.0–HMM reduce the individual limitations of original HMM algorithm and exploiting their different mechanisms. The accuracy graph, shown in Figure 4, also reflects that using a good hybrid intelligent system ensures better accuracy in classification and anomaly detection algorithms.

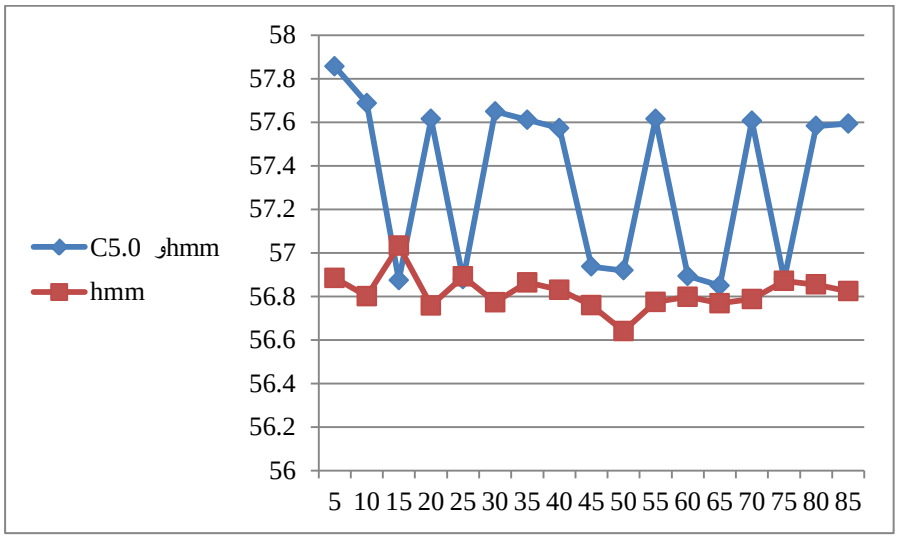

Figure 3. The Accuracy Graph of Original HMM Algorithm and Proposed Model for Data Sets with Different Number Size

5. Conclusion

Intrusion Detection System (IDS) is a defense system which assumes that the attacker gained an authorized access. It tries to identify the attackers by scanning the behavior of active users over the network and computing system. In this research, we have investigated some new techniques for intrusion detection and evaluated their performance based on the benchmark KDD Cup 99 Intrusion data. We have combined C5.0 and HMM as intrusion detection models and designed a hybrid C5.0–HMM model. The hybrid C5.0–HMM approach improves performance IDS for about 90% of classes when compared to individual HMM approach.

References

[1] H. Farhadi, M. AmirHaeri and M. Khansari, "Alert Correlation and Prediction Using Data Mining And HMM", The ISC Int'l Journal of Information Security, vol. 3, **(2011)**, pp. 77-101.
[2] D. Ourston, S. Matzner, W. Stump and B. Hopkins, "Applications of Hidden Markov Models to Detecting Multi-stage Network Attacks", International Conference on System Sciences, Hawaii, **(2003)**.
[3] J. C. Badajena and C. Rout, "Incorporating Hidden Markov Model into Anomaly Detection Technique for Network Intrusion Detection", International Journal of Computer Applications, vol. 53, **(2012)**, pp. 42-47.
[4] P. Dorogovs, A. Borisov and A. Romanovs, "Building an Intrusion Detection System for IT Security Based on Data Mining Techniques", Scientific Journal of Riga Technical University, vol. 49, **(2011)**, pp. 43-48.
[5] J. L. Seng and T. C. Chen, "An analytic approach to select data mining for business decision", Expert Systems with Applications, vol. 37, **(2010)**, pp. 8042-8057.
[6] X. Zhang, L. Jia, H. Shi, Z. Tang and X. Wang, "The Application of Machine Learning Methods to Intrusion Detection", Engineering and Technology (S-CET), 2012 Spring Congress on, **(2012)**, pp. 1-4.
[7] K. Satpute, S. Agrawal, J. Agrawal and S. Sharma, "A Survey on Anomaly Detection in Network Intrusion Detection System Using Particle Swarm Optimization Based Machine Learning Techniques", Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA), **(2013)**, pp. 441-452.
[8] X. Cheng and Y. Ni, "The Research on Dynamic Risk Assessment Based on Hidden Markov Models", Computer Science & Service System (CSSS), 2012 International Conference on, **(2012)**, pp. 1106-1109.
[9] A. Shameli Sendi, M. Dagenais, M. Jabbarifar and M. Couture, "Real Time Intrusion Prediction based on Optimized Alerts with Hidden Markov Model", Journal of Networks, vol. 7, **(2012)**, pp. 311-321.
[10] M. G. Saganowski and T. Andrysiak, "Anomaly Detection Preprocessor for SNORT IDS System", Image Processing and Communications Challenges 4: Springer, **(2013)**, pp. 225-232.
[11] Y. Jiao, "Base on Data Mining In Intrusion Detection System Study", International Journal of Advanced Computer Science, vol. 2, **(2012)**.
[12] S. H. Liao, P. H. Chu and P. Y. Hsiao, "Data mining techniques and applications; A decade review from 2000 to 2011", Expert Systems with Applications, vol. 39, **(2012)**, pp. 11303-11311.
[13] X. Y. Chen, L. Z. Ma, N. Chu, M. Zhou and Y. Hu, ""Classification and Progression Based on CFS-GA and C5. 0 Boost Decision Tree of TCM Zheng in Chronic Hepatitis B", Evidence-Based Complementary and Alternative Medicine, vol. 2013, **(2013)**.
[14] "http://www.ll.mit.edu/IST/ideval/data/data index.html".
[15] G. Koksal, Batmaz and M. C. Testik, "A review of data mining applications for quality improvement in manufacturing industry", Expert Systems with Applications, vol. 38, **(2011)**, pp. 13448-13467.
[16] C. Ciflikli and E. Kahya-Ozyirmidokuz, "Implementing a data mining solution for enhancing carpet manufacturing productivity", Knowledge-Based Systems, vol. 23, **(2010)**, pp. 783-788.
[17] F. Gürbüz, L. Özbakir and H. Yapici, "Data mining and preprocessing application on component reports of an airline company in Turkey", Expert Systems with Applications, vol. 38, **(2011)**, pp. 6618-6626.

