Cascaded Neural Network based Data Mining Strategy for Cloud Intrusion Detection

Purniemaa.P[1], R. Jagadeesh Kannan[2]
[1]School of Computer Science and Engineering, VIT University, Vellore, Tamilnadu, India
[2]School of Computer Science and Engineering, VIT University, Chennai, Tamilnadu, India

Article Info

Article history:

Received May , 2018
Revised Jun 20, 2018
Accepted Jun 30, 2018

Keywords:

Cloud intrusion detection
Data mining
Cascaded neural network

ABSTRACT

In recent years data mining has acquired huge popularity in the field of knowledge discovery. Thus, this approach has inspired several researches for anomaly detection, fraud detection and intrusion detection with higher accuracy, all round generalization of the problem and its sub cases; all giving higher performance in conditions subjected to continuous alteration. Though there remain quite a few challenging problems in design and implementation of a data mining based cloud intrusion detection system, as deception tactics and modeling of behavior remains a daunting problem to compute for anomaly owing to massive size of data to process in reasonable time. In this study we present a cascaded neural network based data mining strategy for cloud intrusion detection systems (IDSs) and presents the comparison and performance results tested on DARPA Intrusion Detection (ID) Data Sets, Knowledge Discovery and Data Mining Cup, NSL-KDD dataset. The study exhibits numerous advantages offered by the presented method and give reliable results of anomaly detection in real time scenario.

Corresponding Author:

Purniemaa.P
School of Computer Science and Engineering,
Vellore Institute of Technology (VIT)
Vellore 632014, India
Email: purniemaa@yahoo.co.in

1. INTRODUCTION

In recent years, intrusion attacks are on the rise; wherein the perpetrators tried to hack in the access of system, install malicious programs or block the services provided by the system to its end users. According to a survey, among the overall cyber attacks in the year 2014 25% of them were non-cyber threats, 19% of it was attempted remote access to the systems and the remaining 17% were policy violation [1]. The study is further validated by survey conducted by FBI and CSI of USA; the survey reveals that the virus attacks for unauthorized access were more prominent and the denial of service attacks has evolved drastically and being used as a major form of blackmailing major companies by asking for ransom by disrupting the services provided by them. To state a few recent examples, the massive attacks suffered by PlayStation and Xbox live servers of Sony and Microsoft is the fresh events of such attacks to disrupt services or hammer down the brand name of the companies [2, 3]. Also in several cases the attackers cyber attacks tend to steal sensitive or personal data in an Endeavour to coerce organizations by debilitating to pitch it to outsiders [4, 5]; the events of digital assault on Code Spaces and JP Morgan had affected more than 70 million individually [6, 7].

Heady et al. [8] depicts cyber intrusion as an arrangement of cyber activities that endeavours to challenge the integrity and security of the digital asset or internet services. For the most part the act of recognizing such intrusion includes the chasing of essential anomalistic events which occur in an internet framework and investigating them keeping in mind the end goal to recognize the potential proximity of matched anomalies with that of intrusion activities [9]. Alessandria describes a more complete meaning of intrusion detection by portraying it as a gathering of practices and instruments used to distinguish flaws within the system that may prompt security threat with the utilization of oddity by diagnosing interruptions and assaults [8]. Correspondingly it might be included that a framework for intrusion detection is the reasonable usage of standards and components over a system [8]. This is blend of programming or potentially equipment segments that keep running on a host machine observing the cyber activities of clients and projects scanning for conceivable dangers on the host system and furthermore investigating system activity of modules that are associated with the host, searching for outcast dangers [8]. The goal of an ID is to prepare administrator of suspicious cyber activities and at times even endeavour to curb the cyber assaults. The practices utilized for intrusion detection do vary from other security strategies, for example, firewalls; get to control or encryption which expects to secure the system. With this being recognized anyway it is firmly prescribed that these security checkpoints are utilized as a part of conjunction with each other as this strengthens safeguard of a framework and guarantees that a significantly bigger extent of a secure framework is ensured [9].

Initially, Intrusion Detection (ID) requires manual supervision. They were entrusted with altogether observing every movement on a support recognizing any abnormalities. This early type of ID demonstrated insufficient because of the slip-up it caused. Latter, computerized log takes its place and intrusion events where at that point created permitting brisk hunting down abnormalities and unapproved work force based on sets of rules [8]. It is important that early forms of ID were deeply relying on couple of association's rules and its processing was not up for real time services [8]. The introduction of review logs helped curbing intrusions to some extent into a quantifiable method; whereby organization examined data and just distinguished issues after occurrences had just happened and not amid the procedure of a cyber assault [9].

Before the 90s" ID was a type of post investigation, examinations of interruptions and changes in framework structure were just distinguished long after the cyber attack. The procedures were repetitive, moderate tedious and introduced capability of human mistake because of substantial association [11]. Amid the 1980 to 1990s research was completed in an offered to reinforce existing ID programming. Some recommend that the leap forward came in the 1990s because of the Intrusion Detection System proposed by Denning [12]. Specialists built up IDS that checked on review information as it was delivered. This progression produced the main variant of constant IDSs taking into account cyber assault pre-emption through strategies for continuous checking of system status and being reactive to any anomalistic changes [10].

As the world entered the mechanical age, the market interest for IT security expanded and IDS were additionally created and made accessible to vast institute and organizations. New highlights were produced, for example, different new strategies, updates to cyber assault design definitions, devoted easy to use interfaces and anticipation mechanism that consequently ceased assaults when distinguished [11]. With the concentrations now moving toward upgrading safety efforts, more up to date assault strategies kept on producing from each side of the web; most outstandingly the Millennium bug and Morris worm. Because of this it ended up noticeably evident to system administrators that in a regularly changing condition one should dependably look to enhance and remain ahead as dangers turn out to be more differing in their techniques to discover better approaches to infiltrate frameworks.

By additionally breaking down Axelssons scientific classification of intrusion detection, two strategies for IDS might be found when seen from another point of view as showed in Table 1 [12]. The first is building a scientific classification mechanism utilizing principals of IDSs, where categorisation depends on the accompanying identification strategies; Anomaly recognition, Signature discovery and Hybrid/compound location. Following on from this the researchers can build up IDS framework, in light of attributes such as, time of location and reaction to identifying interruptions as introduced in Table 2. In [13] worked on the Hopfield Neural network approach in detecting the rogue access points automatically in the wireless type of networks. For the Authentic devices it stores the password in weight matrix design that matches those patterns in the login time.

In [14] explored adaptive type of detection rule supporting to particular artificial neuron and developed a technique to detect harmonics by utilizing artificial neural network method. By this method and processing methodologically the acquired harmonics data using those Lab-VIEW software development environment in the virtual instruments, there the harmonic waves were detected and finally analyzed. Thus it was verified that designed system was effective which was mentioned from the analysis of current ball crusher type of harmonics. In [15] explained the back propagation neural network that will not incur limitations because of the impedance range setting. If, the provided output have with the wrong result, then the correct of the weights are been minimized with the response of galat. The value of back propagation neural network was expected to obtain a closer value to the correct value. In this work it shown that back propagation neural network modelling was used to detect the fault location and it identifies the operating result that current circuit breaker is been tripped it.

Table 1. General form of IDS Taxonomy

Anomaly	Self learning	Non-time series
		Time series
	Programmed	Descriptive stats
		Default deny
Signature	Programmed	State modelling
		Expert system
		String matching
		Simple rule-based
Signature inspired	Self learning	automatic feature selection

2. RESEARCH METHOD

2.1. Database Used

The proposed model is prototyped over MATLAB R2012a under Windows platform. The experiments are conducted over the machine with hardware configurations of Intel's seventh generation 8-core microprocessor, 8GB RAM giving a fine clocking speed of 2.7 GHz. As the emphasis, our work is over the recognition of anomalistic states in intrusion detection setting and its correlation for over a larger group of attributes. The consolidated database available online are used as test data sets for the algorithm are:

(i) DARPA datasets,
(ii) KDD cup datasets and
(iii) NSL – KDD datasets.

Table 2: Database used for experimentation and its attributes

Attributes	DARPA Datasets	KDD Datasets	NSL-KDD dataset
Total sets of attributes or levels	-	41	19
Time for detection	-	270	>51
Security Level	Private (Military)	Public	Public
Type of Attacks	-	DoS, U2R, R2L, Probe	DoS, U2R, R2L, Probe
Purpose	Experimental	Host and Network	Host and Network

(i) In DARPA datasets the cyber assault situation is completed over different system and review sessions. These sessions have been assembled into 5 assault stages through the span of which the enemy tests, softens up, introduces Trojan stream DDoS programming, and dispatches a DDoS assault against an off-site server. the five periods of the assault situation are:
1. I sweep of the AFB from a remote site
2. Probe of experience IP's to search for the sad mind daemon running on Solaris has.
3. Breakings through the sad mind defencelessness, both fruitful and unsuccessful on those hosts
4. Installation of the Trojan stream DDoS programming on three hosts at the AFB
5. Launching the DDoS

(ii) KDD cup dataset is used for recognizing ``bad" associations, and ``good" ordinary associations this database contains a standard arrangement of information to be examined, which incorporates a wide assortment of interruptions mimicked in a military system condition.

(iii) NSL – KDD datasets does exclude repetitive records in the training dataset, so the classifiers won't be one-sided towards more incessant records. The quantity of chose records from each difficulty level gather is contrarily corresponding to the level of records in the first KDD informational index. Therefore, the order rates of particular machine learning strategies fluctuate in a more extensive territory, which makes it more proficient to have a precise assessment of various learning methods. The quantity of records in the training and test sets are sensible, which makes it reasonable to run the trials on the total set without the need to haphazardly choose a little part. Thusly, assessment consequences of various research works will be reliable and tantamount.

2.2. Model

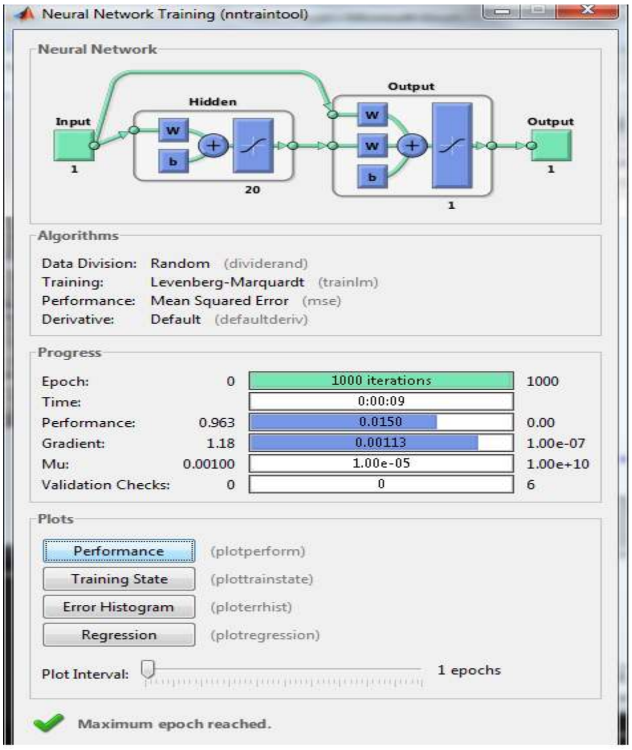
In this section a cascaded neural network based model is used to data mining the state of the system to detection the intrusion attempts. For this we examine the time, place for anIP datagram exchange progressively immediately which gives anomalistic circumstance where reliance of the few parameters can't be displayed for the master framework to shape a versatile system to perform astute operations. In this manner, the two fundamental properties of a specialist framework or computerized reasoning exists in two parts i.e., input and swarm conduct for extending cooperative parametric assessment at each moment in the due procedure. This is where our proposed algorithm comes in to play which rebuild the Cascaded Neural Network (CNN) in Figure 1 for expanding its adequacy against multivariable conditions of time subordinate examining IP datagram exchanges with a few states to mine the IP communication history data in an intelligent sequencing. The engineering of the proposed data mining based IDS method is exhibited in the Figure 2 underneath. Likewise, the steps for computing the anomalistic behavior are shown below:

Algorithm: Cascaded IDS Classifier (UCPC)

Input: List of t IP communications& $CC(i,j)$ i.e., template classes of the trained vectors for the t-1 IP datagram exchange.
Output: $CC'(i,j)$, final state of the IP datagram exchange (1 or 0 for validation).

```
For1: $b_{jq}$//for each IP datagram exchange instances
//Run the conventional waterfilling algorithmfor set CC(i,j), get water level W
        $W = P(CC(i,j) = P_{t+1} | P_{t+n} = S_i), 1 \le i < N, N \le j \le 1$
Where P is the probability of states and S the number of states S= $\{S_1,S_2, \dots S_n\}$ & N is the number of water channels.
For t = 1 to X:
        $H_n = \tanh(w_{HX} X_N + w_{HH} H_{N-1} + B_N)$
        $k_i = w_{HH}.X_N + B_N + \sum_{i+1}^{N} \tanh(\delta_H .(1 - \mathrm{y}_{\mathrm{N}-1})$
Move sub channels to temporal set CC'' (i, j) = $\{CC \left| \sum_1^N H_N{}^{-1} \le w_{HH} - \frac{\Delta W}{N} \right\}$ //for each hidden layer
While j<X
        $k_j = w_{HO}.X_N + \sum_{i+1}^{N} \tanh(\delta_O .\mathrm{y}_{\mathrm{N}-1})$
        $O_N = w_{OH} H_N + B_O$
Update CC' (i, j) (attributeclasses) with the channel N & $\Delta w_{XX}$ as weighted sub channels to buffer the sate sequences:
        $\Delta w_{XX} = \eta. \delta_O. k_i. O_N + \frac{\Delta W}{N}$
        $CC''(i,j) = \sum_{i=1}^{m} \sum_{j=1}^{n} sgn\left(\Delta w_{XX} .\ O(k_i) + \Delta W .\ Y(t', k_j)\right) + \sum_{i,j}^{m,n} P(i,j)$ // attribute classes
Where, m,n belongs to runtime IP datagram exchange made in real time.
End loop
End while loop

IfP$(i,j) = \frac{\| CC''(i,j) - CC(i,j) \|}{|W|} < |N|$
        $CC'(i,j) = \prod_{i=1}^{N} \Delta w_{XX}\, t_{f_t}(R_i, L_i)$
Else
        Print "IP datagram exchange Blocked"
End //if
End loop
```



Algorithms

Data Division: Random (dividerand)
Training: Levenberg-Marquardt (trainlm)
Performance: Mean Squared Error (mse)
Derivative: Default (defaultderiv)

Progress

Epoch:	0	1000 iterations	1000
Time:		0:00:09	
Performance:	0.963	0.0150	0.00
Gradient:	1.18	0.00113	1.00e-07
Mu:	0.00100	1.00e-05	1.00e+10
Validation Checks:	0	0	6

Plots

Performance (plotperform)
Training State (plottrainstate)
Error Histogram (ploterrhist)
Regression (plotregression)

Plot Interval: 1 epochs

✓ Maximum epoch reached.

Figure 1. Architecture of the cascaded neural network used in the study.

Given a sequence of input vectors (X_1,X_2,X_3, ...,X_n) from CC(i,j), a sequence of hidden states (H_1,H_2,H_3, ...,H_n), and a sequence of outputs (O_1,O_2,O_3, ...,O_n) are generated in due process. Notions in this equation are namely, w_{HX} is the input-to-hidden weight matrix, w_{HH} is the hidden-to-hidden (or recurrent) weight matrix, w_{OH} is the hidden-to-output weight matrix, and the vectors B_{N} and B_{O} are the biases. The expression replaces the inputs received form feedback loops with a special initial bias vector checked for nonlinearity while ensuring that the training is done coordinate wise. Ƞ is the learning rate, t' is the time of the next frame and k_i is the local induced field of activation potential for the ith neuron, k_j is the co-activation neuron field for the next sequence of activation units,δ_H&δ_O are the pointer variable for the field & sub-field trace of an emotion, respectively. Table 3 represents the status of labels and the output cascaded sequence processed from the presented algorithm.

Table 3. Classification of IP datagram exchange based on attributes divided into classes for water filling estimation

IP Datagram exchange Id	Class 1 (Source of Exchange)	Class 3 (Destination of Exchange)	Class 3 (Time of established connection)	Class 4 (IP exchange Acknowledgment)	Class 5 (frequency of connection termination)	Class 6 (Cascaded Sequence)
Datagram exchange Profile 1	A	V	P	V	X	1010101010
Datagram exchange Profile 2	B	X	Q	Y	X	1010010001
Datagram exchange Profile 3	A	Y	R	X	V	1011010110
Datagram exchange Profile 4	C	Z	S	X	Y	1110101010

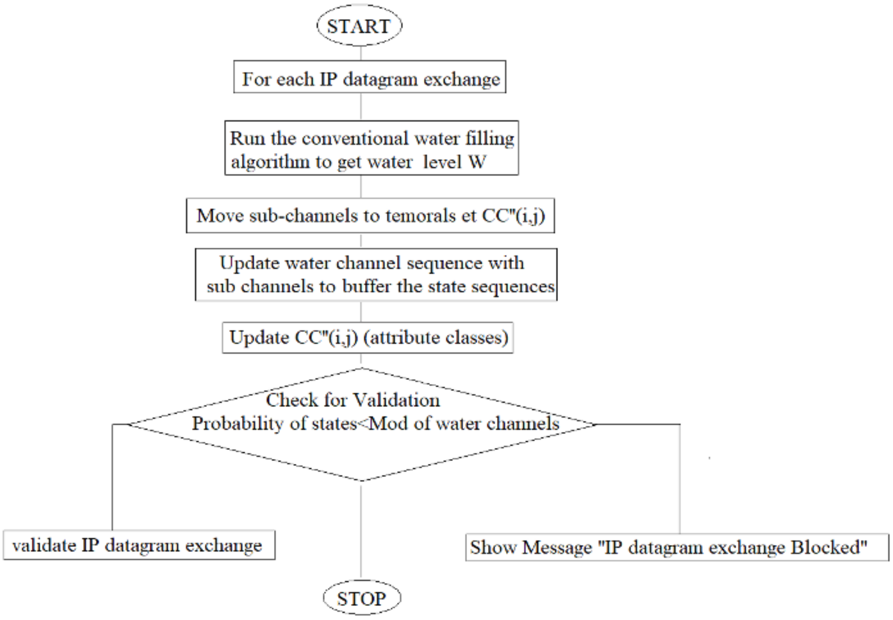

START
For each IP datagram exchange
Run the conventional water filling algorithm to get water level W
Move sub-channels to temorals et CC"(i,j)
Update water channel sequence with sub channels to buffer the state sequences
Update CC"(i,j) (attribute classes)
Check for Validation
Probability of states<Mod of water channels
validate IP datagram exchange
Show Message "IP datagram exchange Blocked"
STOP

Figure 2. Flowchart of the proposed algorithm describing the workflow process

3. RESULTS AND ANALYSIS

As can be inferred from the table 4 above the presented method outperforms the existing data mining based IDSs. Clearly, the various data mining techniques currently employed for IDS isn't superior of confronting several issues and parameters at the same instance. Therefore, there is still a wide room of improvement in the existing techniques. As shown in the snippets of transaction at table 3 of the sample attributes of the IP exchange, 5 classes has been defined as the principal attributes of the parameter which are encoded through the cascaded logical network into a sequence.

Table 4. Performance comparison of the presented method on the database used.

Method	DARPA Datasets	KDD Datasets	NSL-KDD dataset
Decision Tree based Technique	94.80%	95.80%	96.6%
Genetic Algorithms based Technique	94.30%	93.33%	95.7%
Clustering based Techniques	96.30%	96.96%	98.5%
Neural Network based Techniques	97.50%	96.81%	97.2%
Proposed Method	99.32%	97.87%	98.8%

Here, place of IP datagram exchange made for represents the locality of the company/mall/hotel etc. for whom the datagram is issued; this in turn may be the subset of the region. Destination represents the final end region of the IP datagram, similarly other attributes like time of established connection; IP exchange Acknowledgment, frequency of connection termination is used as an attribute to classify the anomalistic behavior. The plot representation of this data is shown in the figure 3(A) which shows the total cascaded length of the sequence in bits required to be made for the KDD Dataset. This shows the length of rise in cascaded sequence is ideally increases in linear fashion. The figure 3(B) shows the Plot for testing time v/s sequence generated for the cascaded neural network. Here, the time required to train the network is comparatively low than the previous methods cited in the literature, thereby giving high computational processing for large streaming information with a promise for real-time application. Figure 3(C) shows the datagram exchange profiles of the four exchanges ids portrayed in Table 3. The nodes in the transaction profiles show problematic exchanges having high chances of being malicious. Also, the curve shown in the plot of transactions Profile 3 & 4 are instantly made at the same time with the same machines and for the same reasons, thereby giving an overlap but since the distribution of the transactions are concurred thereby didn't show much variance in water filling channels and hence are considered to legitimate whereas upon comparison Table 4 with that of transaction profile 2&1 the trough observed in the curve shown in Figure 2 represents high chances of transactions being malicious but with the similar pattern and thus corresponding to the activities perused by the same criminal in the associated region/locality.

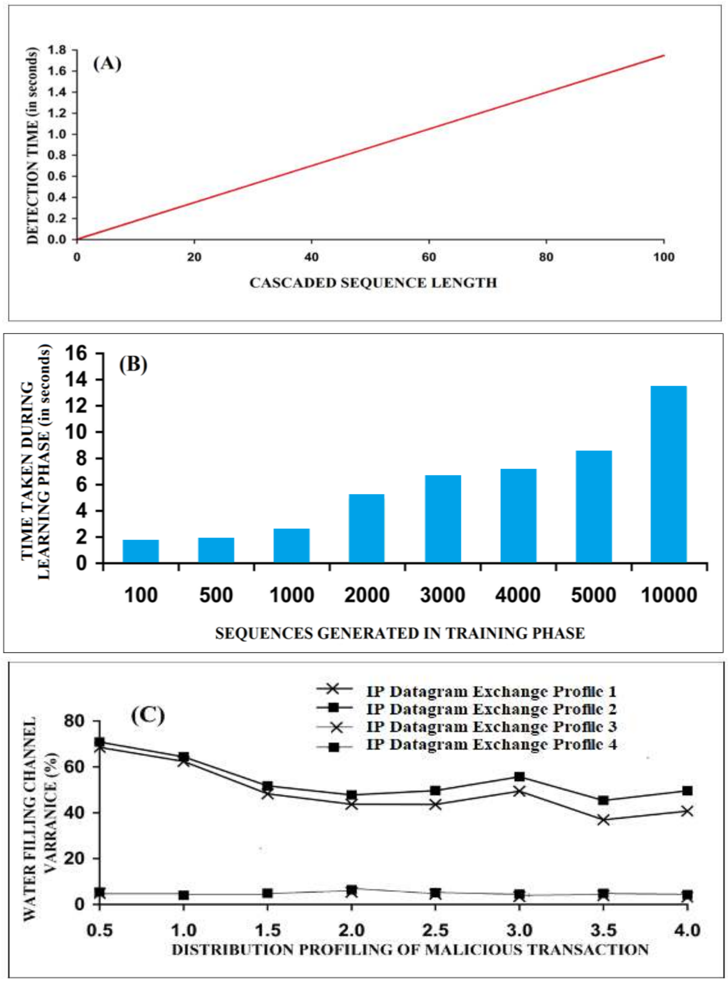


Figure 3. (A) Fraud detection time with the help of proposed algorithm with the rise in complexity of the cascaded sequence. (B) Plot for testing time v/s sequence generated for the cascaded neural network (C) Plot of variance in water filling channel for the consequent transaction made by same person and the distribution of the malicious transaction represented by nodes

Here, the time required to train the network is comparatively low than the previous methods cited in the literature, thereby giving high computational processing for large credit card transactions database with a promise for real-time application. The GUI implemented based on the proposed framework and the architecture of the CNN is shown in Figure 4. The CNN performance is plotted with two parameters in focus i.e., the number of epochs and the Mean Square Error. As can be inferred from the below visualization that the trained plot of the CNN during its training phase and the testing of CNN at the simulation during the phase of classification operations exactly coincide over one another. Thereby, it is giving the best performance without the intrusion of any sort of randomness. Also, the plot is asymptotic to the x and y axis revealing that the convergence of processing is already achieved at the 200th epoch. Thus, the saturation in computing the result is achieved fast enough to determine the result in real time processing environment with clock ticking for 49 Million floating point operations per second. The error contribution in the neural network is kept nominal, as shown in the above figure the MSE drops quickly to the range of 10^{-1} to 10^{-2} in short strides of iteration.

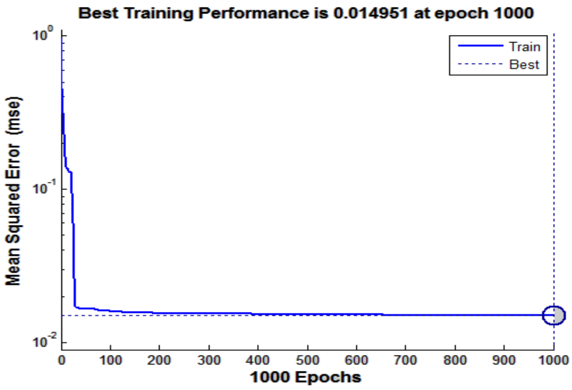


Figure 4. The performance analysis of the cascaded neural network during the classification phase

4. CONCLUSION

As of late there has been an extensive enthusiasm for distinguishing the best list of capabilities qualities for IDS classifiers. With the developing number of interruptions revealed there is cause for making precise IDSs with low rates of false positives. Information mining based IDSs have shown higher exactness, to novel sorts of interruption and powerful conduct. Moreover, it has been noticed that interruption discovery must stay aware of the sheer size, speed and flow which current systems are relied upon to work on. In this study we have presented a novel data mining algorithm for IDs on multiple benchmark datasets. The learning model proposes an associative cascaded learning network with several attributes taken in association through a water filing algorithm. The performance analysis of the study gives an effective way of using such hefty process in relatively lower computational time bounds. The proposed algorithm greatly automates the data mining process of IDS with no-human intervention at both the training and testing phases. There is still a huge room for improvement with this framework to increase the accuracy to the best it can possibly offer with the parallel computing environment.

REFERENCES

[1] United States of America. US Government Accountability Office**.,** "Report on Cyber Security - Actions needed to Address Challenges facing Federal Systems", *Washington: GAO-15-573T,* 2015.
[2] Morgan, L., "List of Cyber Attacks and Data Breaches in 2014", *IT Governance,* 2014.
[3] Watson, M., "JP Morgan suffers data breach affecting 76 million customers", *IT Governance,* 2015.
[4] “Report and response regarding Leakage of Customers personal Information.” *Last accessed on 17 February 2015,* 2014.
[5] Tobak, S., “3 Revelations from the Sony Hack", *Fox Business, Last accessed on 29 January 2015,* 2014.
[6] Peterson, A., The Washington Post, “Why it‟s so hard to calculate the cost of the Sony Pictures hack.”, 2014.
[7] Trend Micro Incorporated, Simply Security. “The Reality of the Sony Pictures Breach”, 2014.
[8] Heady, R., Luger, G.F., Maccabe, A., and Servilla, M., “The architecture of a Network Level Intrusion Detection System”, *Department of Computer Science, College of Engineering, University of New Mexico,* pp. 1-17, 1990
[9] Bace, R., and Mell, P., “NIST Special Publication on Intrusion Detection Systems”, *Booz-Allen and Hamilton Inc, Mclean VA, 2001,* pp. 5-22, 2001.
[10] Kemmerer, R. A., and Vigna, G., “Intrusion Detection: A brief History and Overview”, *Computer, [supplement to security and privacy magazine],* pp. 27-30, 2002.
[11] Allen, J., Christie, A., Fithen, W., McHugh, J., and Pickel, J., “State of the practice of intrusion detection technologies”, vol. *CMU/SEI-99-TR-028, Canergie-Mellon Univ Pittsburgh PA Software Engineering Institute,* pp. 3-23, 2000.
[12] Masud, M., Khan, L., and Thuraisingham, B., *“Data mining tools for malware detection”, Boca Raton, FL: CRC Press,* pp. 15- 38, 2012.
[13] Menal, D., Sumeet, G., "Detection of Rogue Access Point in WLAN using Hopfield Neural Network", *International Journal of Electrical and Computer Engineering (IJECE),* Vol. 7, No.2, pp. 1060-1070, 2017.
[14] Xianfeng, Z., Zheng, F., "Virtual Instrument of Harmonics Detection Based on Neural Network Adaptive Filters", *TELKOMNIKA,* Vol. 13, No. 2, pp. 556 - 562, 2015.
[15] *Azriyenni, A., Mustafa, M.W., Sukma, D.Y,. Dame, M.E.,* "Backpropagation Neural Network Modeling for Fault Location in Transmission Line 150 kV", *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)* Vol. 2, No. 1, pp. 1-12, 2014.